PRESIDENCY UNIVERSITY

Presidency University Act, 2013 of the Karnataka Act No. 41 of 2013 | Established under Section 2(f) of UGC Act, 1956
Approved by AICTE, New Delhi

PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

Program Regulations and Curriculum 2025-2029

BACHELOR OF TECHNOLOGY (B.Tech.) in INFORMATION SCIENCE AND TECHNOLOGY - IST

based on Choice Based Credit System (CBCS) and Outcome Based Education (OBE)

Regulation Number: PU/AC-26.9/SoCSE6/IST/2025-2029

Resolution No. 9 of the 26th Meeting of the Academic Council held on 25th July 2025, and ratified by the Board of Management in its 27th Meeting held on 28th July 2025.

July 2025

Table of Contents

PART A – PROGRAM REGULATIONS

1. Vision & Mission of the University and the School / Department

1.1 Vision of the University

To be a Value-driven Global University, excelling beyond peers and creating professionals of integrity and character, having concern and care for society.

1.2 Mission of the University

- Commit to be an innovative and inclusive institution by seeking excellence in teaching, research and knowledge-transfer.
- Pursue Research and Development and its dissemination to the community, at large.
- Create, sustain and apply learning in an interdisciplinary environment with consideration for ethical, ecological and economic aspects of nation building.
- Provide knowledge-based technological support and services to the industry in its growth and development.
- To impart globally-applicable skill-sets to students through flexible course offerings and support industry's requirement and inculcate a spirit of new-venture creation.

1.3 Vision of Presidency School of Computer Science and Engineering

To be a value based, practice-driven School of Computer Science and Engineering, committed to developing globally-competent Engineers, dedicated to developing cutting-edge technology, towards enhancing Quality of Life.

1.4 Mission of Presidency School of Computer Science and Engineering

- Cultivate a practice-driven environment, with computing-based pedagogy, integrating theory and practice.
- Attract and nurture world-class faculty to excel in Teaching and Research, in the realm of Computing Sciences.
- Establish state-of-the-art computing facilities, for effective Teaching and Learning experiences.
- Promote Interdisciplinary Studies to nurture talent for global impact.
- Instill Entrepreneurial and Leadership Skills to address Social, Environmental and Community-needs.

2. Preamble to the Program Regulations and Curriculum

This is the subset of Academic Regulations and it is to be followed as a requirement for the award of B.Tech degree.

The Curriculum is designed to take into the factors listed in the Choice Based Credit System (CBCS) with focus on Social Project Based Learning, Industrial Training, and Internship to enable the students to become eligible and fully equipped for employment in industries, choose higher studies or entrepreneurship.

In exercise of the powers conferred by and in discharge of duties assigned under the relevant provision(s) of the Act, Statutes and Academic Regulations, 2025 of the University, the Academic Council hereby makes the following Regulations.

3. Short Title and Applicability

a. These Regulations shall be called the Bachelor of Technology Degree Program Regulations and Curriculum 2025-2029.

b. These Regulations are subject to, and pursuant to the Academic Regulations 2025.
c. These Regulations shall be applicable to the ongoing Bachelor of Technology Degree Programs of the 2025-2029 batch, and to all other Bachelor of Technology Degree Programs which may be introduced in future.
d. These Regulations shall supersede all the earlier Bachelor of Technology Degree Program Regulations and Curriculum, along with all the amendments thereto.
e. These Regulations shall come into force from the Academic Year 2025-2026.

4. Definitions

In these Regulations, unless the context otherwise requires:

a. *"Academic Calendar" means the schedule of academic and miscellaneous events as approved by the Vice Chancellor;*
b. *"Academic Council" means the Academic Council of the University;*
c. *"Academic Regulations" means the Academic Regulations, of the University;*
d. *"Academic Term" means a Semester or Summer Term;*
e. *"Act" means the Presidency University Act, 2013;*
f. *"AICTE" means All India Council for Technical Education;*
g. *"Basket" means a group of courses bundled together based on the nature/type of the course;*
h. *"BOE" means the Board of Examinations of the University;*
i. *"BOG" means the Board of Governors of the University;*
j. *"BOM" means the Board of Management of the University;*
k. *"BOS" means the Board of Studies of a particular Department/Program of Study of the University;*
l. *"CGPA" means Cumulative Grade Point Average as defined in the Academic Regulations;*
m. *"Clause" means the duly numbered Clause, with Sub-Clauses included, if any, of these Regulations;*
n. *"COE" means the Controller of Examinations of the University;*
o. *"Course In Charge" means the teacher/faculty member responsible for developing and organising the delivery of the Course;*
p. *"Course Instructor" means the teacher/faculty member responsible for teaching and evaluation of a Course;*
q. *"Course" means a specific subject usually identified by its Course-code and Course-title, with specified credits and syllabus/course-description, a set of references, taught by some teacher(s)/course-instructor(s) to a specific class (group of students) during a specific Academic Term;*
r. *"Curriculum Structure" means the Curriculum governing a specific Degree Program offered by the University, and, includes the set of Baskets of Courses along with minimum credit requirements to be earned under each basket for a degree/degree with specialization/minor/honours in addition to the relevant details of the Courses and Course catalogues (which describes the Course content and other important information about the Course). Any specific requirements for a particular program may be brought into the Curriculum structure of the specific program and relevant approvals should be taken from the BOS and Academic Council at that time.*
s. *"DAC" means the Departmental Academic Committee of a concerned Department/Program of Study of the University;*
t. *"Dean" means the Dean / Director of the concerned School;*
u. *"Degree Program" includes all Degree Programs;*

v. "Department" means the Department offering the degree Program(s) / Course(s) / School offering the concerned Degree Programs / other Administrative Offices;

w. "Discipline" means specialization or branch of B.Tech. Degree Program;

x. "HOD" means the Head of the concerned Department;

y. "L-T-P-C" means Lecture-Tutorial-Practical-Credit – refers to the teaching – learning periods and the credit associated;

z. "MOOC" means Massive Open Online Courses;

aa. "MOU" means the Memorandum of Understanding;

bb. "NPTEL" means National Program on Technology Enhanced Learning;

cc. "Parent Department" means the department that offers the Degree Program that a student undergoes;

dd. "Program Head" means the administrative head of a particular Degree Program/s;

ee. "Program Regulations" means the Bachelor of Technology Degree Program Regulations and Curriculum, 2025-2029;

ff. "Program" means the Bachelor of Technology (B.Tech.) Degree Program;

gg. "PSCS" means the Presidency School of Computer Science and Engineering;

hh. "Registrar" means the Registrar of the University;

ii. "School" means a constituent institution of the University established for monitoring, supervising and guiding, teaching, training and research activities in broadly related fields of studies;

jj. "Section" means the duly numbered Section, with Clauses included in that Section, of these Regulations;

kk. "SGPA" means the Semester Grade Point Average as defined in the Academic Regulations, 2021;

ll. "Statutes" means the Statutes of Presidency University;

mm. "Sub-Clause" means the duly numbered Sub-Clause of these Program Regulations;

nn. "Summer Term" means an additional Academic Term conducted during the summer break (typically in June-July) for a duration of about eight (08) calendar weeks, with a minimum of thirty (30) University teaching days;

oo. "SWAYAM" means Study Webs of Active Learning for Young Aspiring Minds.

pp. "UGC" means University Grant Commission;

qq. "University" means Presidency University, Bengaluru; and

rr. "Vice Chancellor" means the Vice Chancellor of the University.

5. Program Description

The Bachelor of Technology Degree Program Regulations and Curriculum 2025-2029 are subject to, and, pursuant to the Academic Regulations, 2025. These Program Regulations shall be applicable to the following ongoing Bachelor of Technology (B.Tech.) Degree Programs of 2025-2027 offered by the Presidency School of Computer Science and Engineering (PSCS):

1. Bachelor of Technology in Computer Science and Engineering, abbreviated CSE
2. Bachelor of Technology in Computer Science and Technology (Big Data), abbreviated as CBD
3. Bachelor of Technology in Computer Science and Engineering (Block Chain), abbreviated as CBC
4. Bachelor of Technology in Computer Science and Technology (Dev Ops), abbreviated as CDV
5. Bachelor of Technology in Computer Science and Engineering (Cyber Security), abbreviated as CCS
6. Bachelor of Technology in Computer Science and Engineering (Internet of Things), abbreviated as CIT

7. Bachelor of Technology in Computer Science and Engineering (Data Science), abbreviated as CSD
8. Bachelor of Technology in Information Science and Technology, abbreviated as IST
9. Bachelor of Technology in Information Science and Engineering (Artificial Intelligence and Robotics), abbreviated as ISE
10. Bachelor of Technology in Computer Science and Engineering (Artificial Intelligence and Machine Learning) abbreviated as CAI
11. Bachelor of Technology in Artificial Intelligence and Data Science
12. Bachelor of Technology in Robotics and Artificial Intelligence

5.1 These Program Regulations shall be applicable to other similar programs, which may be introduced in future.

5.2 These Regulations may evolve and get amended or modified or changed through appropriate approvals from the Academic Council, from time to time, and shall be binding on all concerned.

5.3 The effect of periodic amendments or changes in the Program Regulations, on the students admitted in earlier years, shall be dealt with appropriately and carefully, so as to ensure that those students are not subjected to any unfair situation whatsoever, although they are required to conform to these revised Program Regulations, without any undue favour or considerations

6. Minimum and Maximum Duration

6.1 Bachelor of Technology Degree Program is a Four-Year, Full-Time Semester based program. The minimum duration of the B.Tech. Program is four (04) years and each year comprises of two academic Semesters (Odd and Even Semesters) and hence the duration of the B.Tech. program is eight (08) Semesters.

6.2 A student who for whatever reason is not able to complete the Program within the normal period or the minimum duration (number of years) prescribed for the Program, may be allowed a period of two years beyond the normal period to complete the mandatory minimum credits requirement as prescribed by the concerned Program Regulations and Curriculum. In general, the permissible maximum duration (number of years) for completion of Program is 'N' + 2 years, where 'N' stands for the normal or minimum duration (number of years) for completion of the concerned Program as prescribed by the concerned Program Regulations and Curriculum.

6.3 The time taken by the student to improve Grades/CGPA, and in case of temporary withdrawal/re-joining (Refer to Clause **Error! Reference source not found.** of Academic Regulations), shall be counted in the permissible maximum duration for completion of a Program.

6.4 In exceptional circumstances, such as temporary withdrawal for medical exigencies where there is a prolonged hospitalization and/or treatment, as certified through hospital/medical records, women students requiring extended maternity break (certified by registered medical practitioner), and, outstanding sportspersons representing the University/State/India requiring extended time to participate in National/International sports events, a further extension of one (01) year may be granted on the approval of the Academic Council.

6.5 The enrolment of the student who fails to complete the mandatory requirements for the award of the concerned Degree (refer Section 19.**Error! Reference source not found.** of

Academic Regulations) in the prescribed maximum duration (Clauses 18.1 and 18.2 of Academic Regulations), shall stand terminated and no Degree shall be awarded.

7 Programme Educational Objectives (PEO)

After four years of successful completion of the program, the graduates shall be able to:

PEO1. Demonstrate as a Computer Engineering Professional

PEO2. Engage in lifelong learning through research and professional development

PEO3. Serve as a leader in the profession through consultancy, extension activities and/ or entrepreneurship

8 Programme Outcomes (PO) and Programme Specific Outcomes (PSO)

8.1 Programme Outcomes (PO)

On successful completion of the Program, the students shall be able to:

PO1: Engineering Knowledge: Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization to develop to the solution of complex engineering problems.

PO2: Problem Analysis: Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development.

PO3: Design/Development of Solutions: Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required.

PO4: Conduct Investigations of Complex Problems: Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions.

PO5: Engineering Tool Usage: Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems.

PO6: The Engineer and The World: Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment.

PO7: Ethics: Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws.

PO8: Individual and Collaborative Team work: Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.

PO9: Communication: Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences

PO10: Project Management and Finance: Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.

PO11: Life-Long Learning: Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change.

8.2 Program Specific Outcomes (PSOs):

On successful completion of the Program, the students shall be able to:

PSO01: Exhibit a solid grasp of information science fundamentals to explain core concepts, analyze computational challenges using theoretical knowledge, and design and develop reliable, efficient systems and applications.

PSO02: Combine software development practices, programming skills and information science domain knowledge to create practical, real-world applications, preparing them for a range of career opportunities in software development, advanced studies, research, or entrepreneurial ventures

PSO03: Create, deploy, and supervise robust data systems that maintain consistency, integrity, and availability, while performing analysis as well as audits to ensure compliance, detect anomalies, uphold accountability through sound data management, administration, and risk assessment practices.

9 Admission Criteria (as per the concerned Statutory Body)

The University admissions shall be open to all persons irrespective of caste, class, creed, gender or nation. All admissions shall be made on the basis of merit in the qualifying examinations; provided that forty percent of the admissions in all Programs of the University shall be reserved for the students of Karnataka State and admissions shall be made through a Common Entrance Examination conducted by the State Government or its agency and seats shall be allotted as per the merit and reservation policy of the State Government from time to time**.** The admission criteria to the B.Tech. Program is listed in the following Sub-Clauses:

9.1 An applicant who has successfully completed Pre-University course or Senior Secondary School course (+2) or equivalent such as (11+1), 'A' level in Senior School Leaving Certificate Course from a recognized university of India or outside or from Senior Secondary Board or equivalent, constituted or recognized by the Union or by the State Government of that Country for the purpose of issue of qualifying certificate on successful completion of the course, may apply for and be admitted into the Program.

9.2 Provided further, the applicant must have taken Physics and Mathematics as compulsory subjects in the Pre-University / Higher Secondary / (10+2) / (11+1) examination, along with either Chemistry / Biology / Electronics / Computer Science / Biotechnology subject, and, the applicant must have obtained a minimum of 45% of the total marks (40% in case of candidates belonging to the Reserved Category as classified by the Government of Karnataka) in these subjects taken together.

9.3 The applicant must have appeared for Joint Entrance Examinations (JEE) Main / JEE (Advanced) / Karnataka CET / COMED-K, or any other State-level Engineering Entrance Examinations.

9.4 Reservation for the SC / ST and other backward classes shall be made in accordance with the directives issued by the Government of Karnataka from time to time.

9.5 Admissions are offered to Foreign Nationals and Indians living abroad in accordance with the rules applicable for such admission, issued from time to time, by the Government of India.
9.6 Candidates must fulfil the medical standards required for admission as prescribed by the University.
9.7 If, at any time after admission, it is found that a candidate had not in fact fulfilled all the requirements stipulated in the offer of admission, in any form whatsoever, including possible misinformation and any other falsification, the Registrar shall report the matter to the Board of Management (BOM), recommending revoking the admission of the candidate.
9.8 The decision of the BOM regarding the admissions is final and binding.

10 Lateral Entry / Transfer Students requirements

10.1 Lateral Entry

The University admits students directly to the second year (3rd Semester) of the B.Tech. Degree program as per the provisions and/or regulations of the Government of Karnataka pertaining to the "Lateral Entry" scheme announced by the Government from time to time. Further, the general conditions and rules governing the provision of Lateral Entry to the B.Tech. Program of the University are listed in the following Sub-Clauses:

10.1.1 Admission to 2nd year (3rd Semester) of the B.Tech. Degree program shall be open to the candidates who are holders of a 3-year Diploma in Engineering (or equivalent qualification as recognized by the University), who have secured not less than forty-five percentage (45%) marks in the final year examination (5th and 6th Semesters of the Diploma Program) in the appropriate branch of Engineering. Provided that, in case of SC / ST and OBC candidates from Karnataka the minimum marks for eligibility shall be forty percent (40%).

10.1.2 Provided further that, candidates seeking Lateral Entry may be required to complete specified bridge Courses as prescribed by the University. Such bridge Courses, if any, shall not be included in the CGPA computations.

10.1.3 All the existing Regulations and Policies of the University shall be binding on all the students admitted to the Program through the provision of Lateral Entry.

10.1.4 The Course requirements prescribed for the 1st Year of the B.Tech. Program shall be waived for the student(s) admitted through Lateral Entry and the duration of the B.Tech. Program for such students is three (03) years, commencing from the 3rd Semester (commencement of the 2nd Year) of the B.Tech. Program and culminating with the 8th Semester (end of the 4th Year) of the B.Tech. Program.

10.1.5 Provided that, if a Lateral Entry student misses any mandatory program specific courses that are typically offered in the 1st year (1st or 2nd semesters), then those courses must be cleared by the students as soon as possible, preferably during the Summer Term.

10.1.6 The existing Program Regulations of the concerned Program to which the student is admitted through the provision of Lateral Entry shall be binding on the student with effect from

the 3rd Semester of the Program. i.e., the Program Structure and Curriculum from the 3rd to 8th Semesters of the Program concerned shall be binding on the student admitted through Lateral Entry. Further, any revisions / amendments made to the Program Regulations thereafter, shall be binding on all the students of the concerned Program.

10.1.7 All the Courses (and the corresponding number of Credits) prescribed for the 1st Year of the concerned B.Tech. Program shall be waived for the student(s) admitted to the concerned B.Tech Program through Lateral Entry. Further, the *Minimum Credit Requirements* for the award of the B.Tech. Degree in the concerned Program shall be prescribed / calculated as follows:

The ***Minimum Credit Requirements*** for the award of the Bachelor of Technology (B.Tech.) Degree prescribed by the concerned Bachelor of Technology Degree Program Regulations and Curriculum, 2025-2029, minus the number of Credits prescribed / accepted by the Equivalence Committee for the 1st Year (1st and 2nd Semesters) of the B.Tech. Program.

For instance, if the *Minimum Credit Requirements* for the award of the Bachelor of Technology (B.Tech.) Degree as prescribed by the Regulations for B.Tech. (Information Science and Technology) is "N" Credits, and, if the total credits prescribed in the 1st Year (total credits of the 1st and 2nd Semesters) of the Program concerned is "M" Credits, then the *Minimum Credit Requirements* for the award of the B.Tech. in Information Science and Technology for a student who joins the Program through the provision of the Lateral Entry, shall be "N – M" Credits.

10.1.8 Further, no other waiver except the Courses prescribed for the 1st year of the B.Tech. Program of the University shall be permissible for students joining the B.Tech. Program through the provision of Lateral Entry.

10.2 Transfer of student(s) from another recognized University to the 2nd year (3rd Semester) of the B.Tech. Program of the University

A student who has completed the 1st Year (i.e., passed in all the Courses / Subjects prescribed for the 1st Year) of the B.Tech. / B.E. / B.S., Four-Year Degree Program from another recognized University, may be permitted to transfer to the 2nd Year (3rd Semester) of the B.Tech. Program of the University as per the rules and guidelines prescribed in the following Sub-Clauses:

10.2.1 The concerned student fulfils the criteria specified in Sub-Clauses 2.3.1, 2.3.2 and 2.3.3.

10.2.2 The student shall submit the Application for Transfer along with a non-refundable Application Fee (as prescribed by the University from time to time) to the University no later than July 10 of the concerned year for admission to the 2nd Year (3rd Semester) B.Tech. Program commencing on August 1 on the year concerned.

10.2.3 The student shall submit copies of the respective Marks Cards / Grade Sheets / Certificates along with the Application for Transfer.

10.2.4 The transfer may be provided on the condition that the Courses and Credits completed

by the concerned student in the 1st Year of the B.Tech. / B.E. / B.S. Four Degree Program from the concerned University, are declared equivalent and acceptable by the Equivalence Committee constituted by the Vice Chancellor for this purpose. Further, the Equivalence Committee may also prescribe the Courses and Credits the concerned students shall have to mandatorily complete, if admitted to the 2nd Year of the B.Tech. Program of the University.

10.2.5 The Branch / Discipline allotted to the student concerned shall be the decision of the University and binding on the student.

11 Change of Branch / Discipline / Specialization

A student admitted to a particular Branch of the B.Tech. Program will normally continue studying in that Branch till the completion of the program. However, the University reserves the right to provide the option for a change of Branch, or not to provide the option for a change of Branch, at the end of 1st Year of the B.Tech. Program to eligible students in accordance with the following rules and guidelines: framed by the University from time to time.

11.1 Normally, only those students, who have passed all the Courses prescribed for the 1st Year of the B.Tech. Program and obtained a CGPA of not less than 6.50 at the end of the 2nd Semester, shall be eligible for consideration for a change of Branch.

11.2 Change of Branch, if provided, shall be made effective from the commencement of the 3rd Semester of the B.Tech. Program. There shall be no provision for change of Branch thereafter under any circumstances whatsoever.

11.3 The student provided with the change of Branch shall fully adhere to and comply with the Program Regulations of the concerned Branch of the B.Tech. Program, the Fee Policy pertaining to that Branch of the B.Tech. Program, and, all other rules pertaining to the changed Branch existing at the time.

11.4 Change of Branch once made shall be final and binding on the student. No student shall be permitted, under any circumstances, to refuse the change of Branch offered.

11.5 The eligible student may be allowed a change in Branch, strictly in order of *inter se* merit, subject to the conditions given below:

11.5.1 The actual number of students in the 3rd Semester in any particular Branch to which the transfer is to be made, should not exceed the intake fixed by the University for the concerned Branch;

11.5.2 The actual number of students in any Branch from which transfer is being sought does not fall below 75% of the total intake fixed by the University for the concerned Branch.

The process of change of Branch shall be completed within the first five days of Registration for the 3rd Semester of the B.Tech. Program.

12 Specific Regulations regarding Assessment and Evaluation (including the Assessment Details of NTCC Courses, Weightages of Continuous Assessment and End Term Examination for various Course Categories)

12.1 The academic performance evaluation of a student in a Course shall be according to the University Letter Grading System based on the class performance distribution in the Course.

12.2 Academic performance evaluation of every registered student in every Course registered by the student is carried out through various components of Assessments spread across the Semester. The nature of components of Continuous Assessments and the weightage given to each component of Continuous Assessments (refer Clause 12.5 of Academic Regulations) shall be clearly defined in the Course Plan for every Course, and approved by the DAC.

12.3 Format of the End-Term examination shall be specified in the Course Plan.

12.4 Grading is the process of rewarding the students for their overall performance in each Course. The University follows the system of Relative Grading with statistical approach to classify the students based on the relative performance of the students registered in the concerned Course except in the following cases:

- Non-Teaching Credit Courses (NTCC)
- Courses with a class strength less than 30

Absolute grading method may be adopted, where necessary with prior approval of concerned DAC.

Grading shall be done at the end of the Academic Term by considering the aggregate performance of the student in all components of Assessments prescribed for the Course. Letter Grades (Clause **Error! Reference source not found.** of Academic Regulations) shall be awarded to a student based on her/his overall performance relative to the class performance distribution in the concerned Course. These Letter Grades not only indicate a qualitative assessment of the student's performance but also carry a quantitative (numeric) equivalent called the Grade Point.

12.5 Assessment Components and Weightage

13 Table 1: Assessment Components and Weightage for different category of Courses

Nature of Course and Structure	aluation Component	Weightage
Lecture-based Course L component in the L-T-P Structure is predominant (more than 1) amples: 3-0-0; 3-0-2; 2-1-0; 2-0-2, 2-0-4 etc.)	ntinuous Assessments	50%
	nd Term Examination	50%
Lab/Practice-based Course P component in the L-T-P Structure is predominant amples: 0-0-4; 1-0-4; 1-0-2; etc.)	ntinuous Assessments	75%
	nd Term Examination	25%

Skill based Courses like Industry Internship, Capstone project, Research Dissertation, Integrative Studio, Interdisciplinary Project, Summer / Short Internship, Social Engagement / Field Projects, Portfolio, and such similar Non-Teaching Credit Courses, where the pedagogy does not lend itself to a typical L-T-P structure	idelines for the assessment components for the various types of Courses, with recommended weightages, shall be specified in the concerned Program Regulations and Curriculum / Course Plans, as applicable.

The exact weightages of Evaluation Components shall be clearly specified in the concerned PRC and respective Course Plan.

Normally, for Practice/Skill based Courses, without a defined credit structure (L–T–P) [NTCC], but with assigned Credits (as defined in Clause 5.2 of the Academic Regulations), the method of evaluation shall be based only on Continuous Assessments. The various components of Continuous Assessments, the distribution of weightage among such components, and the method of evaluation/assessment, shall be as decided and indicated in the Course Plan/PRC. The same shall be approved by the respective DAC.

13.1 **Minimum Performance Criteria:**

13.1.1 **Theory only Course and Lab/Practice Embedded Theory Course**

A student shall satisfy the following minimum performance criteria to be eligible to earn the credits towards the concerned Course:

a. A student must obtain a minimum of 30% of the total marks/weightage assigned to the End Term Examinations in the concerned Course.

b. The student must obtain a minimum of 40% of the AGGREGATE of the marks/weightage of the components of Continuous Assessments, Mid Term Examinations and End Term Examinations in the concerned Course.

13.1.2 **Lab/Practice only Course and Project Based Courses**

The student must obtain a minimum of 40% of the AGGREGATE of the marks/weightage of all assessment components in the concerned Course.

13.1.3 A student who fails to meet the minimum performance criteria listed above in a Course shall be declared as "Fail" and given "F" Grade in the concerned Course. For theory Courses, the student shall have to re-appear in the "Make-Up Examinations" as scheduled by the University in any subsequent semester, or, re-appear in the End Term Examinations of the same Course when it is scheduled at the end of the following Semester or Summer Term, if offered. The marks obtained in the Continuous Assessments (other than the End Term Examination) shall be carried forward and be included in computing the final grade, if the student secures the minimum requirements (as per Sub-Clause 13.1.1 and 13.1.2 of Academic Regulations) in the "Make-Up Examinations" of the concerned Course. Further, the student has an option to re-register for the Course and clear the same in the summer term/ subsequent semester if he/she wishes to do so, provided the Course is offered.

13 Additional clarifications - Rules and Guidelines for Transfer of Credits from MOOC, etc. – Note: These are covered in Academic Regulations

The University allows students to acquire credits from other Indian or foreign institutions and/or Massive Open Online Course (MOOC) platforms, subject to prior approval. These credits may be transferred and counted toward fulfilling the minimum credit requirements for the award of a degree. The process of transfer of credits is governed by the following rules and guidelines:

14.1 The transfer of credits shall be examined and recommended by the Equivalence Committee (Refer **Error! Reference source not found.** of Academic Regulations) and approved by the Dean - Academics.

14.2 Students may earn credits from other Indian or foreign Universities/Institutions with which the University has an MOU, and that MOU shall have specific provisions, rules and guidelines for transfer of credits. These transferred credits shall be counted towards the minimum credit requirements for the award of the degree.

14.3 Students may earn credits by registering for Online Courses offered by *Study Web of Active Learning by Young and Aspiring Minds* (SWAYAM) and *National Program on Technology Enhanced Learning* (NPTEL), or other such recognized Bodies/ Universities/Institutions as approved by the concerned BOS and Academic Council from time to time. The concerned School/Parent Department shall publish/include the approved list of Courses and the rules and guidelines governing such transfer of credits of the concerned Program from time to time. The Rules and Guidelines for the transfer of credits specifically from the Online Courses conducted by SWAYAM/ NPTEL/ other approved MOOCs are as stated in the following Sub-Clauses:

14.3.1 A student may complete SWAYAM/NPTEL/other approved MOOCs as mentioned in Clause 14.3 (as per Academic Regulations) and transfer equivalent credits to partially or fully complete the mandatory credit requirements of Discipline Elective Courses and/or the mandatory credit requirements of Open Elective Courses as prescribed in the concerned Curriculum Structure. However, it is the sole responsibility of the student to complete the mandatory credit requirements of the Discipline Elective Courses and the Open Elective Courses as prescribed by the Curriculum Structure of the concerned Program.

14.3.2 SWAYAM/NPTEL/ other approved MOOCs as mentioned in Clause 14.3 (as per Academic Regulations) shall be approved by the concerned Board of Studies and placed (as Annexures) in the concerned PRC.

14.3.3 Parent Departments may release a list of SWAYAM/NPTEL/other approved MOOCs for Pre-Registration as per schedule in the Academic Calendar or through University Notification to this effect.

14.3.4 Students may Pre-Register for the SWAYAM/NPTEL/other approved MOOCs in the respective Departments and register for the same Courses as per the schedule announced by respective Online Course Offering body/institute/ university.

14.3.5 A student shall request for transfer of credits only from such approved Courses as mentioned in Sub-Clause 14.3.2 above.

14.3.6 SWAYAM/NPTEL/other approved MOOCs Courses are considered for transfer of credits only if the concerned student has successfully completed the SWAYAM/NPTEL/other approved MOOCs and obtained a certificate of successful/satisfactory completion.

14.3.7 A student who has successfully completed the approved SWAYAM/NPTEL/ other approved MOOCs and wants to avail the provision of transfer of equivalent credits, must submit the original

Certificate of Completion, or such similar authorized documents to the HOD concerned, with a written request for the transfer of the equivalent credits. On verification of the Certificates/Documents and approval by the HOD concerned, the Course(s) and equivalent Credits shall forwarded to the COE for processing of results of the concerned Academic Term.

14.3.8 The credit equivalence of the SWAYAM/NPTEL/other approved MOOCs are based on Course durations and/or as recommended by the Course offering body/institute/university. The Credit Equivalence mapped to SWAYAM/ NPTEL approved Courses based on Course durations for transfer of credits is summarised in Table shown below. The Grade will be calculated from the marks received by the Absolute Grading Table **Error! Reference source not found.** in the Academic Regulations

Table 2: Durations and Credit Equivalence for Transfer of Credits from SWAYAM-NPTEL/ other approved MOOC Courses		
Sl. No.	**Course Duration**	**Credit Equivalence**
1	4 Weeks	1 Credit
2	8 Weeks	2 Credits
3	12 Weeks	3 Credits

14.3.9 The maximum permissible number of credits that a student may request for credit transfer from MOOCs shall not exceed 20% of the mandatory minimum credit requirements specified by the concerned Program Regulations and Curriculum for the award of the concerned Degree.

14.3.10 The University shall not reimburse any fees/expense; a student may incur for the SWAYAM/NPTEL/other approved MOOCs.

14.4 The maximum number of credits that can be transferred by a student shall be limited to forty percent (40%) of the mandatory minimum credit requirements specified by the concerned Program Regulations and Curriculum for the award of the concerned Degree. However, the grades obtained in the Courses transferred from other Institutions/MOOCs, as mentioned in this Section (13.**Error! Reference source not found.**), shall not be included in the calculation of the CGPA.

14.5 Mandatory Non-Credit Course Completion Requirements: All mandatory non-credit courses shall be satisfactorily completed by the student as part of the degree requirements. These courses will be evaluated and awarded letter grades based on the following criteria:

- S (Satisfactorily Completed): Awarded when the student successfully completes all prescribed course requirements.
- NC (Not Completed): Awarded when the student fails to meet the prescribed course requirements.

A student receiving an NC grade must reappear for and complete the course in accordance with the guidelines prescribed by the University.

In the case of non-taught and non-credited mandatory courses—where students are advised to undertake learning through MOOC platforms—there shall be a clearly defined Course Catalogue and a corresponding Course Plan. The Course Plan shall outline the assessment components, which will form the basis for evaluation.

PART B: PROGRAM STRUCTURE

14. Structure / Component with Credit Requirements Course Baskets & Minimum Basket wise Credit Requirements

The B.Tech. (Information Science and Technology) Program Structure (2025-2029) totalling 160 credits. Table 3 summarizes the type of baskets, number of courses under each basket and the associated credits that are mandatorily required for the completion of the Degree.

Table 3: B.Tech. (Information Science and Technology) 2025-2029: Summary of Mandatory Courses and Minimum Credit Contribution from various Baskets

Sl. No.	Baskets	Credit Contribution
1	Humanities and Social Sciences including Management Courses (HSMC)	10
2	Basic Science Courses (BSC)	24
3	Engineering Science Courses (ESC)	22
4	Professional Core Courses (PCC)	64
5	Professional Elective Courses (PEC)	18
6	Open Elective Courses (OEC)	06
7	Project Work (PRW)	16
8	Mandatory Courses (MAC) *	0
	Total Credits	**160 (Minimum)**

* Please refer to Table 3.6, (where the number '6' corresponds to the serial number of the Mandatory course basket.)

In the entire Program, the practical and skill based course component contribute to an extent of approximately 58% out of the total credits of 160 for B.Tech. (Information Science and Technology) program of four years' duration.

15. Minimum Total Credit Requirements of Award of Degree

As per the AICTE guidelines, a minimum of 160 credits is required for the award of a B.Tech. degree.

16. Other Specific Requirements for Award of Degree, if any, as prescribed by the Statutory Bodies,

16.1 The award of the Degree shall be recommended by the Board of Examinations and approved by the Academic Council and Board of Management of the University**.**

16.2 A student shall be declared to be eligible for the award of the concerned Degree if she/he:

a. Fulfilled the Minimum Credit Requirements and the Minimum Credits requirements under various baskets;

b. Secure a minimum CGPA of 4.50 in the concerned Program at the end of the Semester/Academic Term in which she/he completes all the requirements for the award of the Degree as specified in Sub-Clause a of Academic Regulations**;**

c. No dues to the University, Departments, Hostels, Library, and any other such Centers/ Departments of the University; and

d. No disciplinary action is pending against her/him.

17. Curriculum Structure – Basket Wise Course List (not Semester Wise)
List of Courses Tabled – aligned to the Program Structure
(Course Code, Course Name, Credit Structure (LTPC), Contact Hours, Course Basket, Type of Skills etc., as applicable).

Table 3.1 : List of Humanities and Social Sciences including Management Courses (HSMC)									
S. No	Course Code	Course Name	L	T	P	C	Contact Hours	Type of Skills	Pre-requisites
1	ENG1900	English for Technical Communication	2	0	0	2	2	S	-
2	DES1146	Introduction to Design Thinking	1	0	0	1	1	F	-
3	ENG2501	Advanced English	2	0	0	2	2	S	-
4	FIN1002	Essentials of Finance	3	0	0	3	3	S	-
5	APT4005	Aptitude for Employability	0	0	2	1	2	S/EM	-
6	PPS3018	Preparedness for Interview	0	0	2	1	2	S	-
Total No. of Credits						**10**			

Table 3.2 : List of Basic Science Courses (BSC)									
S. No	Course Code	Course Name	L	T	P	C	Contact Hours	Type of Skills	Pre-requisites
1	MAT2301	Calculus and Differential Equations	3	1	0	4	4	F	-
2	PHY2501	Optoelectronics and Quantum Physics	3	0	0	3	3	F	-
3	PHY2504	Optoelectronics and Quantum Physics Lab	0	0	2	1	2	F	-
4	MAT2402	Probability and Statistics	3	1	0	4	4	F	-
5	CHE2501	Chemistry of Smart Materials	3	0	0	3	3	S	-
6	CHE2502	Chemistry of Smart Materials Lab	0	0	2	1	2	S	-
7	MAT2303	Linear Algebra and Vector Calculus	3	1	0	4	4	EM	-
8	MAT2404	Discrete Mathematics	3	1	0	4	4	F	-
Total No. of Credits						**24**			

Table 3.3 : List of Engineering Science Courses (ESC)									
S. No	Course Code	Course Name	L	T	P	C	Contact Hours	Type of Skills	Pre-requisites
1	MEC1006	Engineering Graphics	2	0	0	2	2	S	-
2	CSE1500	Computational Thinking using Python	2	0	2	3	4	S	-
3	ECE2022	Digital Design	2	0	0	2	2	F/S	-
4	ECE2052	Digital Design Lab	0	0	2	1	2	F/S	-
5	CIV1200	Foundations of Integrated Engineering	2	0	0	2	2	S	-

6	EEE1200	Basics of Electrical and Electronics Engineering	3	0	0	3	3	F/S	-
7	EEE1250	Basics of Electrical and Electronics Engineering Lab	0	0	2	1	2	F/S	-
8	ECE1511	Design Workshop	1	0	2	2	3	S/EM	-
9	CSE2264	Essentials of AI	3	0	0	3	3	S/EM	-
10	CSE2265	Essentials of AI Lab	0	0	2	1	2	S/EM	-
11	CSE2274	Competitive Programming and Problem Solving	0	0	4	2	4	S/EM	-
Total No. of Credits						**22**			

Table 3.4 : List of Professional Core Courses (PCC)

S. No	Course Code	Course Name	L	T	P	C	Contact Hours	Type of Skills	Pre-requisites
1	CSE2200	Problem Solving using C	2	0	0	2	2	S	-
2	CSE2201	Problem Solving using C Lab	0	0	4	2	4	S	-
3	CSE2251	Data Communication and Computer Networks	3	0	0	3	3	S	-
4	CSE2252	Data Communication and Computer Networks Lab	0	0	2	1	2	S	-
5	CSE2253	Data Structures	3	0	0	3	3	S	-
6	CSE2254	Data Structures Lab	0	0	2	1	2	S	-
7	CSE2255	Object Oriented Programming Using Java	3	0	0	3	3	S/EM	-
8	CSE2256	Object Oriented Programming Using Java Lab	0	0	2	1	2	S/EM	-
9	CSE2257	Computer Organization and Architecture	3	0	0	3	3	S	-
10	CSE2258	Web Technologies	3	0	0	3	3	S/EM	-
11	CSE2259	Web Technologies Lab	0	0	2	1	2	S/EM	-
12	CSE2260	Database Management Systems	3	0	0	3	3	S	-
13	CSE2261	Database Management Systems Lab	0	0	2	1	2	S	-
14	CSE2500	Data Analytics	3	0	0	3	2	S/EM	MAT2402
15	CSE2501	Data Analytics Lab	0	0	2	1	2	S/EM	MAT2402
16	CSE2262	Analysis of Algorithms	3	1	0	4	4	S	-
17	CSE2263	Analysis of Algorithms Lab	0	0	2	1	2	S	-
18	CSE2266	Theory of Computation	3	0	0	3	3	S	-
19	CSE2502	Cryptography and Network Security	3	0	0	3	3	S	CSE2251

20	CCS2504	Ethical Hacking	2	0	0	2	2	S/EM	CSE2251
21	CCS2505	Ethical Hacking Lab	0	0	4	2	4	S/EM	CSE2251
22	CSE2269	Operating Systems	3	0	0	3	3	S	-
23	CSE2270	Operating Systems Lab	0	0	2	1	2	S/EM	-
24	CSD2001	Applied Data Science	3	0	0	3	3	S/EM	-
25	CSD2002	Applied Data Science Lab	0	0	2	1	2	S/EM	-
26	CSE2271	Software Design and Development	3	0	0	3	3	S	-
27	CSE2272	Cloud Computing	2	0	0	2	2	S/EM	-
28	CSE2273	Cloud Computing Lab	0	0	2	1	2	S/EM	-
29	CSE2505	Mobile Application Development	2	0	0	2	2	S/EM	CSE2255
30	CSE2506	Mobile Application Development Lab	0	0	4	2	4	S/EM	CSE2255
Total No. of Credits						**64**			

Table 3.5 : List of course in Project Work basket (PRW)

S. No	Course Code	Course Name	L	T	P	C	Contact Hours	Type of Skills	Pre-requisites
1	CSE7300	Capstone Project	0	0	0	10	-	S/ EM/ EN	-
2	CSE7000	Internship	0	0	0	02	-	S/ EM/ EN	-
3	CSE7100	Mini Project	0	0	0	04	-	S/ EM/ EN	-
Total No. of Credits						**16**			

	Table 3.6 : List of Mandatory Courses (MAC)					
S.No	**Course Code**	**Course Name**	**L**	**T**	**P**	**C**
1	PPS1025	Industry Readiness Program – I	0	0	2	0
2	CHE7601	Environmental Studies **	-	-	-	0
3	LAW7601	Indian Constitution	0	0	0	0
4	PPS1026	Industry Readiness Program – II	0	0	2	0
5	CIV7601	Universal Human Values and Ethics **	-	-	-	0
6	APT4002	Introduction to Aptitude	0	0	2	0
7	APT4004	Aptitude Training - Intermediate	0	0	2	0
8	APT4006	Logical and Critical Thinking	0	0	2	0
	Total No. of Credits					**0**

18. Practical / Skill based Courses – Internships / Thesis / Dissertation / Capstone Project Work / Portfolio / Mini project

Practical / Skill based Courses like internship, project work, capstone project, research project / dissertation, and such similar courses, where the pedagogy does not lend itself to a typical L-T-P-C

Structure as defined in Clause 5.1 of the Academic Regulations, 2021, are simply assigned the number of Credits based on the quantum of work / effort required to fulfill the learning objectives and outcomes prescribed for the concerned Courses. Such courses are referred to as Non-Teaching Credit Courses (NTCC). These Courses are designed to provide students with hands-on experience and skills essential for their professional development. These courses aim to equip students with abilities in problem identification, root cause analysis, problem-solving, innovation, and design thinking through industry exposure and project-based learning. The expected outcomes are first level proficiency in problem solving and design thinking skills to better equip B.Tech. graduates for their professional careers. The method of evaluation and grading for the Practical / Skill based Courses shall be prescribed and approved by the concerned Departmental Academic Committee (refer Annexure A of the Academic Regulations, 2021). The same shall be prescribed in the Course Handout.

18.1 Internship / In-plant Training / Skill-based Program / International Immersion (IM) / Recognition of Prior Learning (RPL)

A student may undergo an Internship / In-plant Training / Skill-based Program / IM / RPL for a period minimum 04 weeks in an industry / company, government bodies, academic / research institution or recognized online platforms offering certified skill-based programs during the semester break between 4^{th} and 5^{th} semesters, subject to the following conditions:

18.1.1 The Internship / In-plant Training / Skill-based Program / IM / RPL shall be conducted in accordance with the Internship Policy prescribed by the University from time to time.

18.1.2 Internship: student shall undergo internship, either in industry / company, academic / research organizations, government bodies, or international institutions. The objective is to provide practical exposure, industry insights, and real-world experience relevant to the student's field of study.

18.1.3 In-plant Training: student shall undergo training / industrial exposure program aimed at providing with practical insights into real-world working environments. The training may be conducted by industries / companies on-campus or through student visits to industries / companies, government bodies / institutions, or technical organizations.

18.1.4 Skill-based Program: student shall undergo a certified skill-based program of 30 hours / 04 weeks. Skill-based program should cater to Skill-Enhancement, Practical Focus, and Career orientation, Complementary to Curriculum or Industry relevant.

18.1.5 International Immersion (IM): student shall undergo IM aimed at providing global exposure through collaborations with foreign universities, industries, or research institutions. The International Immersion may include industry visits, expert interactions, and cultural exchange activities, enhancing students' international outlook, communication skills, and professional readiness.

18.1.6 Recognition of Prior Learning (RPL): student who shall undergo any formally recognize relevant prior work experience, internships, or project-based learning that meet the internship learning outcomes, thereby allowing students to earn internship credit without repeating equivalent practical training.

18.1.7 The number of Internships available for the concerned Academic Term. Further, the available number of Internships / In-plant training / Skill-based Program / IM / RPL shall be awarded to the students by the University on the basis of merit using the CGPA secured by the student and as per the selection criteria. Provided further, the student fulfils the criteria, as applicable, specified by the industry / company, government bodies, academic / research or through certified courses.

18.1.8 A student may opt for Internship / In-plant Training / IM / RPL in an industry / company, government bodies / academic / research institution, international bodies of her / his choice, subject to the condition that the concerned student takes the responsibility to arrange the Internship / In-plant Training / IM / RPL on her / his own. Provided further, that the industry / company, government bodies, academic / research institution national or international offering such Internship / Training confirms to the University that the Internship shall be conducted in accordance with the Program Regulations and Internship Policy of the University / Rubrics.

18.1.9 A student undergoing RPL must submit an application with supporting documents such as experience letters, project reports, employer feedback, certifications, a self-reflection report etc. Application must be submitted before the commencement of the internship semester.

18.1.10 A student selected for an Internship / In-plant Training / Skill-based Program / IM / RPL in an industry / company, government bodies, academic / research institution shall adhere to all the rules and guidelines prescribed in the Internship Policy of the University.

The performance will be assessed and grades awarded as per the university's academic grading policy. It may be included as a non-GPA course (Grade-based Satisfactorily Completed / Not Completed). The student must secure a minimum of 50% of the total marks to be declared as having successfully completed the course.

18.2 Mini Project Work

A student may opt to do a Mini Project Work for a period of 08-10 weeks in an Industry / Company or academic / research institution or the University Department(s) as an equivalence of Internship during the Semester Break between 4^{th} and 5^{th} Semesters or 6^{th} and 7^{th} Semesters or during the 5^{th} / 6^{th} / 7^{th} Semester as applicable, subject to the following conditions:

18.2.1 The Mini Project Work shall be approved by the concerned HOD and be carried out under the guidance of a faculty member.

18.2.2 The student may do the mini project work in an Industry / Company or academic / research institution of her / his choice subject to the above mentioned condition (Sub-Clause 18.2.1). Provided further, that the Industry / Company or academic / research institution offering such mini project work confirms to the University that the mini project work will be conducted in accordance with the Program Regulations and requirements of the University.

18.3 Capstone Project

A student may undergo a Capstone Project for a period of 12-14 weeks in an industry / company or academic / research institution in the 7^{th} / 8^{th} Semester as applicable, subject to the following conditions:

18.3.1 The Capstone Project shall be in conducted in accordance with the Capstone Project Policy prescribed by the University from time to time.

18.3.2 The selection criteria (minimum CGPA, pass in all Courses as on date, and any other qualifying criteria) as applicable / stipulated by the concerned Industry / Company or academic / research institution for award of the Capstone Project to a student;

18.3.3 The number of Capstone Project available for the concerned Academic Term. Further, the available number of Capstone Project shall be awarded to the students by the University on the basis of merit using the CGPA secured by the student. Provided further, the student fulfils the criteria, as applicable, specified by the Industry / Company or academic / research institution providing the Capstone Project, as stated in Sub-Clause 18.3.2 above.

18.3.4 A student may opt for Capstone Project in an Industry / Company or academic / research institution of her / his choice, subject to the condition that the concerned student takes the responsibility to arrange the Capstone Project on her / his own. Provided further, that the Industry / Company or academic / research institution offering such Capstone Project confirms to the University that the Capstone Project shall be conducted in accordance with the Program Regulations and Capstone Project Policy of the University.

18.3.5 A student selected for a Capstone Project in an industry / company or academic / research institution shall adhere to all the rules and guidelines prescribed in the Capstone Project Policy of the University.

18.4 Research Project / Dissertation

A student may opt to do a Research Project / Dissertation for a period of 12-14 weeks in an Industry / Company or academic / research institution or the University Department(s) as an equivalence of Capstone Project, subject to the following conditions:

18.4.1 The Research Project / Dissertation shall be approved by the concerned HOD and be carried out under the guidance of a faculty member.

The student may do the Research Project / Dissertation in an Industry / Company or academic / research institution of her / his choice subject to the above mentioned condition (Sub-Clause 18.4.1). Provided further, that the Industry / Company or academic / research institution offering such Research Project / Dissertation confirms to the University that the Research Project / Dissertation work will be conducted in accordance with the Program Regulations and requirements of the University.

19. List of Elective Courses under various Specialisations / Stream Basket

Table 3.7: Professional Electives Courses/Specialization Tracks – Minimum of 12 credits is to be earned by the student in a particular track and overall 18 credits.

Track 1 – Artificial Intelligence and Machine Learning

Sl. No.	Course Code	Course Name	L	T	P	C	Contact Hours	Type of Skills	Pre-requisites
1	IST2501	Optimization Methods in Machine Learning	3	0	0	3	3	S	CSE2264
2	CSE3426	Front End Full Stack Development *	2	0	2	3	4	S/EM	CSE2258
3	CSE3427	Java Full Stack Development *	2	0	2	3	4	S/EM	CSE2258
4	CSE3428	.Net Full Stack Development *	2	0	2	3	4	S/EM	CSE2258
5	CAI3411	Generative AI	2	0	2	3	4	S/EM	CSE2264
6	CAI3427	Language Models for Text	2	0	2	3	4	S/	CSE2

		Mining +						EM	264
7	CSD3406	Business Intelligence and Analytics	3	0	0	3	3	S/ EM	CSE2 260
8	CAI3428	Practical Deep Learning with TensorFlow +	2	0	2	3	4	S/ EM	CSE2 264
9	CIT2504	AI and Deep Learning for IoT	3	0	0	3	3	S/ EM	CSE2 264
10	CAI3429	Deep Learning Techniques for Computer Vision +	2	0	2	3	4	S/ EM	CSE2 264
Track 2 – Big Data and Data Science									
Sl. No.	Course Code	Course Name	L	T	P	C	Contact Hours	Type of Skills	Pre-requisites
1	CBD2000	Introduction to Big Data	3	0	0	3	3	F/ S	-
2	CSE3426	Front End Full Stack Development *	2	0	2	3	4	S/ EM	CSE2 258
3	CSE3427	Java Full Stack Development *	2	0	2	3	4	S/ EM	CSE2 258
4	CSE3428	.Net Full Stack Development *	2	0	2	3	4	S/ EM	CSE2 258
5	IST3408	Data Handling and Visualization	2	0	2	3	4	S/ EM	-
6	CAI3427	Language Models for Text Mining +	2	0	2	3	4	S/ EM	CSE2 264
7	CDV3408	Predictive Analytics for System Monitoring & Performance Optimization	3	0	0	3	3	S/ EM	-
8	CSD3411	Cybersecurity and Data Privacy	3	0	0	3	3	S/ EM	CSE2 251
9	CAI3428	Practical Deep Learning with TensorFlow +	2	0	2	3	4	S/ EM	CSE2 264
10	CBD3403	Cloud Storage & Data Management	3	0	0	3	3	S/ EM	CBD2 000
11	CAI3429	Deep Learning Techniques for Computer Vision +	2	0	2	3	4	S/ EM	CSE2 264
Track 3 – Block Chain and Cyber Security									
Sl. No.	Course Code	Course Name	L	T	P	C	Contact Hours	Type of Skills	Pre-requisites
1	CCS2503	Cyber Security	3	0	0	3	3	S/ EM	CSE2 251
2	CCS2506	Intrusion Detection and Prevention System	3	0	0	3	3	S/ EM	CSE2 251
3	CSE3426	Front End Full Stack Development *	2	0	2	3	4	S/ EM	CSE2 258
4	CSE3427	Java Full Stack Development *	2	0	2	3	4	S/ EM	CSE2 258
5	CSE3428	.Net Full Stack Development *	2	0	2	3	4	S/ EM	CSE2 258
6	CCS3400	Digital and Mobile Forensics	2	0	2	3	4	S/ EM	CSE2 251

7	CAI3427	Language Models for Text Mining +	2	0	2	3	4	S/ EM	CSE2 017
8	CCS3405	Vulnerability Assessment and Penetration Testing	3	0	0	3	3	S/ EM	CSE2 269
9	CBC1700	Foundations of Blockchain Technology	3	0	0	3	3	S	-
10	CAI3428	Practical Deep Learning with TensorFlow +	2	0	2	3	4	S/ EM	CSE2 264
11	CCS2509	Malware Analysis	3	0	0	3	3	S/ EM	CSE2 251
12	CBC3400	Cryptography and Security in Blockchain	3	0	0	3	3	S/ EM	-
13	CAI3429	Deep Learning Techniques for Computer Vision +	2	0	2	3	4	S/ EM	CSE2 264

Track 4 – Information Science & Technology

Sl. No.	Course Code	Course Name	L	T	P	C	Contact Hours	Type of Skills	Pre-requisites
1	IST3406	Human Computer Interaction	3	0	0	3	3	S/EM	-
2	ISE2502	Information Retrieval	3	0	0	3	3	S/EM	-
3	CSE3426	Front End Full Stack Development *	2	0	2	3	4	S/ EM	CSE2 258
4	CSE3427	Java Full Stack Development *	2	0	2	3	4	S/ EM	CSE2 258
5	CSE3428	.Net Full Stack Development *	2	0	2	3	4	S/ EM	CSE2 258
6	IST3401	UI UX Design	2	0	2	3	4	S/ EM	-
7	CAI3427	Language Models for Text Mining +	2	0	2	3	4	S/ EM	CSE2 264
8	IST3402	Search Engine Optimization	3	0	0	3	3	S/ EM	-
9	IST3403	Service Oriented Architecture	3	0	0	3	3	S/ EM	-
10	CAI3428	Practical Deep Learning with TensorFlow +	2	0	2	3	4	S/ EM	CSE2 264
11	IST3404	Information System Audit	3	0	0	3	3	S/ EM	-
12	IST3407	Infrastructure Management	3	0	0	3	3	S/ EM	-
13	CAI3429	Deep Learning Techniques for Computer Vision +	2	0	2	3	4	S/ EM	CSE2 264

Total # of Credits to be earned from DE = 18

*** Mandatory for Students selected for Tech Mahindra and Capgemini**
+Mandatory for Students Selected for Samsung Innovation Campus

20. List of Open Electives to be offered by the School / Department (Separately for ODD and EVEN Semesters.

Table 3.8: Open Elective Courses Baskets: Minimum Credits to be earned from this Basket is 06											
Sl. No.	**Course Code**	**Course Name**	**L**	**T**	**P**	**C**	**Type of Skill/ Focus**	**Course Caters to**	**Prerequisites/ Corequisites**	**Antirequisites**	**Future Courses that need this as a Prerequisite**
Chemistry Basket											
1	CHE1003	Fundamentals of Sensors	3	0	0	3	S	ES	-	-	-
2	CHE1004	Smart materials for IOT	3	0	0	3	S	ES	-	-	-
3	CHE1005	Computational Chemistry	2	0	0	2	S	ES	-	-	-
4	CHE1006	Introduction to Nano technology	3	0	0	3	S	ES	-	-	-
5	CHE1007	Biodegradable electronics	2	0	0	2	S	ES	-	-	-
6	CHE1008	Energy and Sustainability	2	0	0	2	S	ES	-	-	-
7	CHE1009	3D printing with Polymers	2	0	0	2	S	ES	-	-	-
8	CHE1010	Bioinformatics and Healthcare IT	2	0	0	2	S	ES	-	-	-
9	CHE1011	Chemical and Petrochemical catalysts	3	0	0	3	S	ES	-	-	-
10	CHE1012	Introduction to Composite materials	2	0	0	2	S	ES	-	-	-
11	CHE1013	Chemistry for Engineers	3	0	0	3	S	ES	-	-	-
12	CHE1014	Surface and Coatings technology	3	0	0	3	S	ES	-	-	-
13	CHE1015	Waste to Fuels	2	0	0	2	S	ES	-	-	-
14	CHE1016	Forensic Science	3	0	0	3	S	ES	-	-	-
Civil Engineering Basket											
1	CIV1001	Disaster mitigation and management	3	0	0	3	S	-	-	-	-
2	CIV1002	Environment Science and Disaster Management	3	0	0	3	FC	-	-	-	-
3	CIV2001	Sustainability Concepts in Engineering	3	0	0	3	S	-	-	-	-
4	CIV2002	Occupational Health and Safety	3	0	0	3	S	-	-	-	-
5	CIV2003	Sustainable Materials and Green Buildings	3	0	0	3	EM	-	-	-	-
6	CIV2004	Integrated Project Management	3	0	0	3	EN	-	-	-	-
7	CIV2005	Environmental Impact Assessment	3	0	0	3	EN	-	-	-	-
8	CIV2006	Infrastructure Systems for Smart Cities	3	0	0	3	EN	-	-	-	-
9	CIV2044	Geospatial Applications for Engineers	2	0	2	3	EM	-	-	-	-
10	CIV2045	Environmental Meteorology	3	0	0	3	S	-	-	-	-
11	CIV3046	Project Problem Based Learning	3	0	0	3	S	-	-	-	-
12	CIV3059	Sustainability for Professional Practice	3	0	0	3	EN	-	-	-	-
Commerce Basket											
1	COM2001	Introduction to Human Resource Management	2	0	0	2	F	HP/GS	-	-	-
2	COM2002	Finance for Non Finance	2	0	0	2	S	-	-	-	-
3	COM2003	Contemporary Management	2	0	0	2	F	-	-	-	-
4	COM2004	Introduction to Banking	2	0	0	2	F	-	-	-	-
5	COM2005	Introduction to Insurance	2	0	0	2	F	-	-	-	-
6	COM2006	Fundamentals of Management	2	0	0	2	F	-	-	-	-
7	COM2007	Basics of Accounting	3	0	0	3	F	-	-	-	-
Computer Science Basket (not to be offered for Computer Science and Engineering students)											
1	CSE2002	Programming in Java	2	0	2	3	S/EM	-	-	-	-
2	CSE2003	Social Network Analytics	3	0	0	3	S	GS	-	-	-

3	CSE2004	Python Application Programming	2	0	2	3	S/ EM	-	-	-	-
4	CSE2005	Web design fundamentals	2	0	2	3	S/ EM/EN	-	-	-	-
Design Basket											
1	DES1001	Sketching and Painting	0	0	2	1	S	-	-	-	-
2	DES1002	Innovation and Creativity	2	0	0	2	F	-	-	-	-
3	DES1121	Introduction to UX design	1	0	2	2	S	-	-	-	-
4	DES1122	Introduction to Jewellery Making	1	0	2	2	S	-	-	-	-
5	DES1124	Spatial Stories	1	0	2	2	S	-	-	-	-
6	DES1125	Polymer Clay	1	0	2	2	S	-	-	-	-
7	DES2001	Design Thinking	3	0	0	3	S	-	-	-	-
8	DES1003	Servicability of Fashion Products	1	0	2	2	F	ES	-	-	-
9	DES1004	Choices in Virtual Fashion	1	0	2	2	F	ES, GS, HP	-	-	-
10	DES1005	Fashion Lifestyle and Product Diversity	1	0	2	2	F	ES, GS, HP	-	-	-
11	DES1006	Colour in Everyday Life	1	0	2	2	F	ES	-	-	-
12	DES2080	Art of Design Language	3	0	0	3	S	-	-	-	-
13	DES2081	Brand Building in Design	3	0	0	3	S	-	-	-	-
14	DES2085	Web Design Techniques	3	0	0	3	S	-	-	-	-
15	DES2089	3D Modeling for Professionals	1	0	4	3	S	-	-	-	-
16	DES2090	Creative Thinking for Professionals	3	0	0	3	S	-	-	-	-
17	DES2091	Idea Formulation	3	0	0	3	S	-	-	-	-
Electrical and Electronics Basket											
1	EEE1002	IoT based Smart Building Technology	3	0	0	3	S	-	-	-	-
2	EEE1003	Basic Circuit Analysis	3	0	0	3	S	-	-	-	-
3	EEE1004	Fundamentals of Industrial Automation	3	0	0	3	S	-	-	-	-
4	EEE1005	Electric Vehicles & Battery Technology	3	0	0	3	S	-	-	-	-
5	EEE1006	Smart Sensors for Engineering Applications	3	0	0	3	S	-	-	-	-
Electronics and Communication Basket											
1	ECE1003	Fundamentals of Electronics	3	0	0	3	F	-	-	-	-
2	ECE1004	Microprocessor based systems	3	0	0	3	F	-	-	-	-
3	ECE3089	Artificial Neural Networks	3	0	0	3	S	-	-	-	-
4	ECE3097	Smart Electronics in Agriculture	3	0	0	3	F/EM	-	-	-	-
5	ECE3098	Environment Monitoring Systems	3	0	0	3	F/EM	-	-	-	-
6	ECE3102	Consumer Electronics	3	0	0	3	F/EM	-	-	-	-
7	ECE3103	Product Design of Electronic Equipment	3	0	0	3	S/F/ EM / EN	-	-	-	-
8	ECE3106	Introduction to Data Analytics	3	0	0	3	F/EM	-	-	-	-
9	ECE3107	Machine Vision for Robotics	3	0	0	3	F/EM	-	-	-	-
English Basket											
1	ENG1008	Indian Literature	2	0	0	2	-	GS/ HP	-	-	-
2	ENG1009	Reading Advertisement	3	0	0	3	S	-	-	-	-
3	ENG1010	Verbal Aptitude for Placement	2	0	2	3	S	-	-	-	-
4	ENG1011	English for Career Development	3	0	0	3	S	-	-	-	-
5	ENG1012	Gender and Society in India	2	0	0	2	-	GS/ HP	-	-	-
6	ENG1013	Indian English Drama	3	0	0	3	-	-	-	-	-
7	ENG1014	Logic and Art of Negotiation	2	0	2	3	-	-	-	-	-
8	ENG1015	Professional Communication Skills for Engineers	1	0	0	1	-	-	-	-	-
DSA Basket											
1	DSA2001	Spirituality for Health	2	0	0	2	F	HP	-	-	-

2	DSA2002	Yoga for Health	2	0	0	2	S	HP	-	-	-
3	DSA2003	Stress Management and Well Being	2	0	0	2	F	-	-	-	-
Kannada Basket											
1	KAN1001	Kali Kannada	1	0	0	1	S	-	-	-	-
2	KAN1003	Kannada Kaipidi	3	0	0	3	S	-	-	-	-
3	KAN2001	Thili Kannada	1	0	0	1	S	-	-	-	-
4	KAN2003	Pradharshana Kale	1	0	2	2	S	-	-	-	-
5	KAN2004	Sahithya Vimarshe	2	0	0	2	S	-	-	-	-
6	KAN2005	Anuvadha Kala Sahithya	3	0	0	3	S	-	-	-	-
7	KAN2006	Vichara Manthana	3	0	0	3	S	-	-	-	-
8	KAN2007	Katha Sahithya Sampada	3	0	0	3	S	-	-	-	-
9	KAN2008	Ranga Pradarshana Kala	3	0	0	3	S	-	-	-	-
Foreign Language Basket											
1	FRL1004	Introduction of French Language	2	0	0	2	S	S	-	-	-
2	FRL1005	Fundamentals of French	2	0	0	2	S	S	-	-	-
3	FRL1009	Mandarin Chinese for Beginners	3	0	0	3	S	S	-	-	-
Law Basket											
1	LAW1001	Introduction to Sociology	2	0	0	0	2	F	HP	-	-
2	LAW2001	Indian Heritage and Culture	2	0	0	0	2	F	HP/GS	-	-
3	LAW2002	Introdcution to Law of Succession	2	0	0	0	2	F	HP/GS	-	-
4	LAW2003	Introduction to Company Law	2	0	0	0	2	F	HP	-	-
5	LAW2004	Introduction to Contracts	2	0	0	2	F	HP	-	-	-
6	LAW2005	Introduction to Copy Rights Law	2	0	0	2	F	HP	-	-	-
7	LAW2006	Introduction to Criminal Law	2	0	0	2	F	HP	-	-	-
8	LAW2007	Introduction to Insurance Law	2	0	0	2	F	HP	-	-	-
9	LAW2008	Introduction to Labour Law	2	0	0	2	F	HP	-	-	-
10	LAW2009	Introduction to Law of Marriages	2	0	0	2	F	HP/GS	-	-	-
11	LAW2010	Introduction to Patent Law	2	0	0	2	F	HP	-	-	-
12	LAW2011	Introduction to Personal Income Tax	2	0	0	2	F	HP	-	-	-
13	LAW2012	Introduction to Real Estate Law	2	0	0	2	F	HP	-	-	-
14	LAW2013	Introduction to Trademark Law	2	0	0	2	F	HP	-	-	-
15	LAW2014	Introduction to Competition Law	3	0	0	3	F	HP	-	-	-
16	LAW2015	Cyber Law	3	0	0	3	F	HP	-	-	-
17	LAW2016	Law on Sexual Harrassment	2	0	0	2	F	HP/GS	-	-	-
18	LAW2017	Media Laws and Ethics	2	0	0	2	F	HP/GS	-	-	-
Mathematics Basket											
1	MAT2008	Mathematical Reasoning	3	0	0	3	S	-	-	-	-
2	MAT2014	Advanced Business Mathematics	3	0	0	3	S	-	-	-	-
3	MAT2041	Functions of Complex Variables	3	0	0	3	S	-	-	-	-
4	MAT2042	Probability and Random Processes	3	0	0	3	S	-	-	-	-
5	MAT2043	Elements of Number Theory	3	0	0	3	S	-	-	-	-
6	MAT2044	Mathematical Modelling and Applications	3	0	0	3	S	-	-	-	-
Mechanical Basket											
1	MEC1001	Fundamentals of Automobile Engineering	3	0	0	3	F	-	-	-	-
2	MEC1002	Introduction to Matlab and Simulink	3	0	0	3	S/EM	-	-	-	-
3	MEC1003	Engineering Drawing	1	0	4	3	S	-	-	-	-
4	MEC2001	Renewable Energy Systems	3	0	0	3	F	ES	-	-	-
5	MEC2002	Operations Research & Management	3	0	0	3	F	-	-	-	-
6	MEC2003	Supply Chain Management	3	0	0	3	S/ EM/ EN	-	-	-	-

7	MEC2004	Six Sigma for Professionals	3	0	0	3	S/EM	-	-	MEC 2008	-
8	MEC2005	Fundamentals of Aerospace Engineering	3	0	0	3	F	-	-	-	-
9	MEC2006	Safety Engineering	3	0	0	3	S/EM	ES	-	-	-
10	MEC2007	Additive Manufacturing	3	0	0	3	F/EM	-	-	-	-
11	MEC3069	Engineering Optimisation	3	0	0	3	S/EM	-	-	-	-
12	MEC3070	Electronics Waste Management	3	0	0	3	F/S	ES	-	-	-
13	MEC3071	Hybrid Electric Vehicle Design	3	0	0	3	S/EM	ES	-	-	-
14	MEC3072	Thermal Management of Electronic Appliances	3	0	0	3	S/EM	-	-	-	-
15	MEC3200	Sustainable Technologies and Practices	3	0	0	3	S/EM	-	-	-	-
16	MEC3201	Industry 4.0	3	0	0	3	S/EM	-	-	-	-
Petroleum Basket											
1	PET1011	Energy Industry Dynamics	3	0	0	3	FC	ES	-	NIL	-
2	PET1012	Energy Sustainability Practices	3	0	0	3	FC	ES	-	NIL	-
Physics Basket											
1	PHY1003	Mechanics and Physics of Materials	3	0	0	3	FC / SD				
2	PHY1004	Astronomy	3	0	0	3	FC				
3	PHY1005	Game Physics	2	0	2	3	FC / SD				
4	PHY1006	Statistical Mechanics	2	0	0	2	FC				
5	PHY1007	Physics of Nanomaterials	3	0	0	3	FC				
6	PHY1008	Adventures in nanoworld	2	0	0	2	FC				
7	PHY2001	Medical Physics	2	0	0	2	FC	ES			
8	PHY2002	Sensor Physics	1	0	2	2	FC / SD				
9	PHY2003	Computational Physics	1	0	2	2	FC				
10	PHY2004	Laser Physics	3	0	0	3	FC	ES			
11	PHY2005	Science and Technology of Energy	3	0	0	3	FC	ES			
12	PHY2009	Essentials of Physics	2	0	0	2	FC				
Management Basket- I											
1	MGT2007	Digital Entrepreneurship	3	0	0	3	S/EM/EN	-	-	-	-
2	MGT2015	Engineering Economics	3	0	0	3	S	-	-	-	-
3	MGT2023	People Management	3	0	0	3	S/EM/ EN	HP	-	-	-
Management Basket- II											
1	MGT1001	Introduction to Psychology	3	0	0	3	F	HP	-	-	-
2	MGT1002	Business Intelligence	3	0	0	3	EN	-	-	-	-
3	MGT1003	NGO Management	3	0	0	3	S	-	-	-	-
4	MGT1004	Essentials of Leadership	3	0	0	3	EM/ EN	GS/ HP	-	-	-
5	MGT1005	Cross Cultural Communication	3	0	0	3	S/EM/ EN	HP	-	-	-
6	MGT2001	Business Analytics	3	0	0	3	S/ EM/EN	-	-	-	-
7	MGT2002	Organizational Behaviour	3	0	0	3	F	HP	-	-	-
8	MGT2003	Competitive Intelligence	3	0	0	3	S	-	-	-	-
9	MGT2004	Development of Enterprises	3	0	0	3	S/EM/EN	-	-	-	-
10	MGT2005	Economics and Cost Estimation	3	0	0	3	S/EM	-	-	-	-
11	MGT2006	Decision Making Under Uncertainty	3	0	0	3	S	-	-	-	-
12	MGT2008	Econometrics for Managers	3	0	0	3	S	-	-	-	-
13	MGT2009	Management Consulting	3	0	0	3	S/EM/EN	-	-	-	-
14	MGT2010	Managing People and Performance	3	0	0	3	S/EM/EN	HP/GS	-	-	-
15	MGT2011	Personal Finance	3	0	0	3	F	-	-	-	-

16	MGT2012	E Business for Management	3	0	0	3	S/EM	-	-	-	-
17	MGT2013	Project Management	3	0	0	3	EN / EM	GS/HP/ ES	-	-	-
18	MGT2014	Project Finance	3	0	0	3	EN / EM	HP	-	-	-
19	MGT2016	Business of Entertainment	3	0	0	3	EM/ EN	-	-	-	-
20	MGT2017	Principles of Management	3	0	0	3	S/EM/ EN	-	-	-	-
21	MGT2018	Professional and Business Ethics	3	0	0	3	S/EM/ EN	HP	-	-	-
22	MGT2019	Sales Techniques	3	0	0	3	S/EM/ EN	HP	-	-	-
23	MGT2020	Marketing for Engineers	3	0	0	3	S/EM/ EN	HP	-	-	-
24	MGT2021	Finance for Engineers	3	0	0	3	S/EM/ EN	HP	-	-	-
25	MGT2022	Customer Relationship Management	3	0	0	3	S/EM/ EN	HP	-	-	-
Media Studies Basket											
1	BAJ3050	Corporate Filmmaking and Film Business	0	0	4	2	EM	HP	-	-	-
2	BAJ3051	Digital Photography	2	0	2	3	EM	HP	-	-	-
3	BAJ3055	Introduction to News Anchoring and News Management	0	0	2	1	EM	-	-	-	-

21. List of MOOC (NPTEL) Courses

21.1 Presidency University students are given the opportunity to study abroad in International Universities through a selection process coordinated by the Office of International Affairs (OIA). Such selected students need to complete their credits for the semester that they are abroad in the following way:

21.1.1 The student needs to study and complete School Core and Program Core Courses in offline mode only.

21.1.2 Massive Open Online Course (MOOC) courses maybe given for Open Elective and Discipline Elective Courses. These courses need to be approved by the concerned BOS and Academic Council from time to time.

21.1.3 SWAYAM/NPTEL/ other approved MOOCs shall be approved by the concerned Board of Studies and placed in the concerned PRC.

21.1.4 Student shall register for these courses in the ERP of Presidency University.

21.1.5 For these MOOC courses faculty coordinators are identified. These faculty should have undergone similar MOOC courses and therefore should be familiar with the mode of class conduction, types of assessments and evaluation procedures.

21.1.6 Study materials shall be provided to the students as video lectures shared by the MOOCs Coordinator(s), or the students may access the approved MOOCs Portal directly. The mode of class conduction is determined by the MOOCs coordinator(s) as detailed in the Course Catalogue and Course Plan.

21.1.7 The question paper shall be prepared by the MOOCs coordinator(s).

21.1.8 Students write the exams in online mode. These exams are scheduled and conducted by the School.

21.1.9 Results are evaluated by School and given to the Office of the Controller of Examinations (CoE).

21.1.10 The details of the duration, credits and evaluation are given below:

Sl#	Duration	Credits
1.	12 weeks	3
2.	8 weeks	2

3	4 weeks	1

21.2 MOOC – Discipline Elective Courses for B.Tech. Computer Science and Information Technology Program

Table 3.9 : MOOC Discipline Elective Courses

Sl.No	Course Code	Course Name	Credits	L-T-P-C
1	CSE3111	Artificial Intelligence: Search Methods for Problem Solving	3	3-0-0-3
2	CSE3112	Privacy and Security in Online Social Media	3	3-0-0-3
3	CSE3113	Computational Complexity	3	3-0-0-3
4	CSE3114	Deep Learning for Computer Vision	3	3-0-0-3
5	CSE3115	Learning Analytics Tools	3	3-0-0-3
6	CSE505	The Joy of Computing Using Python	3	3-0-0-3
7	CSE3119	Coding Skills in Python	3	3-0-0-3
8	CSE3121	Parallel Computer Architecture	3	3-0-0-3
9	CSE3124	Games and Information	3	3-0-0-3
10	CSE3140	Introduction to Industry 4.0 and Industrial Internet of Things	3	3-0-0-3
11	CSE3142	Affective Computing	3	3-0-0-3
12	CSE3196	Foundations of Cyber Physical Systems	3	3-0-0-3
13	CSE3197	Getting Started with Competitive Programming	3	3-0-0-3
14	CSE3198	GPU Architectures and Programming	3	3-0-0-3
15	CSE3199	Artificial Intelligence: Knowledge Representation and Reasoning	3	3-0-0-3
16	CSE3200	Programming in Modern C++	3	3-0-0-3
17	CSE3201	Circuit Complexity Theory	3	3-0-0-3
18	CSE3202	Basics of Computational Complexity	3	3-0-0-3
19	CSE3212	Introduction to Computer and Network Performance Analysis using Queuing	1	1-0-0-1
20	CSE3213	C Programming and Assembly Language	1	1-0-0-1
21	CSE3214	Python for Data Science	1	1-0-0-1
22	CSE3215	Software Conceptual Design	1	1-0-0-1
23	CSE3117	Industrial Digital Transformation	3	3-0-0-3
24	CSE3118	Blockchain for Decision Makers	3	3-0-0-3
25	CSE3349	Technology for Lawyers	3	3-0-0-3
26	CSE3430	Deep Learning for Natural Language Processing	3	3-0-0-3
27	CSE3431	Machine Learning for Engineering and Science Applications	3	3-0-0-3
28	CSE3432	Algorithms in Computational Biology and Sequence Analysis	3	3-0-0-3
29	CSE3433	Introduction to Large Language Models (LLMs)	3	3-0-0-3
30	CSE3434	Quantum Algorithms and Cryptography	3	3-0-0-3

31	CAI3430	Responsible & Safe AI Systems	3	3-0-0-3
32	CCS3416	Practical Cyber Security for Cyber Security Practitioners	3	3-0-0-3
33	IST3409	Design & Implementation of Human-Computer Interfaces	3	3-0-0-3

21.3 MOOC - Open Elective Courses for B. Tech. Computer Science and Information Technology Program

Table 3.10: MOOC Open Elective Courses
Courses duration is 4 weeks (01 credit)/ 8 weeks (02 credits)/ 12 weeks (03 credits)

Sl. No.	Course Code	Course Name	Total Credits	L-T-P-C
1	BBA2022	Supply Chain digitization	3	3-0-0-3
2	BBA2021	E Business	3	3-0-0-3
3	BBB2016	Business Analytics for Management Decisions	3	3-0-0-3
4	BBB2015	Artificial Intelligence for Investments	3	3-0-0-3
5	MEC3001	Design and Development of Product	1	1-0-0-1
6	ENG3004	Perspectives of Neurolinguistics	1	1-0-0-1
7	PPS4009	Working in Contemporary Teams	1	1-0-0-1
8	MGT3001	Data Analysis and Decision Making	3	3-0-0-3
9	MEC3001	Design and Development of Product	1	1-0-0-1
10	EEE3105	Microsensors and Nanosensors	3	3-0-0-3
11	CIV3065	Drone Systems and Control	3	3-0-0-3
12	ECE3183	Neural Networks for Signal Processing - I	3	3-0-0-3
13	CIVXXXX	Disaster Management	3	3-0-0-3

Note :

** MEC3001 is offered to the students who had 1 credit shortage because of implementation of CBCS system during their 1st year.*

*** ENG3004 is offered to the students who had 2 credits shortage along with the MEC3001 because of implementation of CBCS System during their 1st year.*

**** PPS4009 is offered to only International students in place of Interview Preparedness course of their batch mates.*

22. Recommended Semester Wise Course Structure / Flow including the Programme / Discipline Elective Paths / Options

Semester Wise Course Grids/ Tables: First year - CYCLE 1

Sl. No.	Course Code	Course Name	L	T	P	C	Type of Skill/ Focus	Course Caters to	Basket
Semester 1 - Physics Cycle						**19**			
1	MAT2301	Calculus and Differential Equations	3	1	0	4	F		BSC
2	PHY2501	Optoelectronics and Quantum Physics	3	0	0	3	F		BSC
3	MEC1006	Engineering Graphics	2	0	0	2	S		ESC
4	ENG1900	English for Technical Communication	2	0	0	2	S		HSMC
5	CSE1500	Computational Thinking using Python	2	0	2	3	S		ESC

6	ECE2022	Digital Design	2	0	0	2	F/S		ESC
7	DES1146	Introduction to Design Thinking	1	0	0	1	F		HSMC
8	PHY2504	Optoelectronics and Quantum Physics Lab	0	0	2	1	F		BSC
9	PPS1025	Industry Readiness Program – I	0	0	2	0	S		MAC
10	ECE2052	Digital Design Lab	0	0	2	1	F/S		ESC
Semester 2 – Chemistry Cycle						**22**			
1	MAT2402	Probability and Statistics	3	1	0	4	F		BSC
2	CHE7601	Environmental Studies	0	0	0	0	F		MAC
3	CIV1200	Foundations of Integrated Engineering	2	0	0	2	S		ESC
4	CHE2501	Chemistry of Smart Materials	3	0	0	3	S		BSC
5	CSE2200	Problem Solving using C	2	0	0	2	S		PCC
6	ENG2501	Advanced English	2	0	0	2	S		HSMC
7	EEE1200	Basics of Electrical and Electronics Engineering	3	0	0	3	F/S		ESC
8	LAW7601	Indian Constitution	0	0	0	0	F		MAC
9	CSE2201	Problem Solving using C Lab	0	0	4	2	S		PCC
10	CHE2502	Chemistry of Smart Materials Lab	0	0	2	1	S		BSC
11	PPS1026	Industry Readiness Program – II	0	0	2	0	S		MAC
12	EEE1250	Basics of Electrical and Electronics Engineering Lab	0	0	2	1	F/S		ESC
13	ECE1511	Design Workshop	1	0	2	2	S/EM		ESC

First year - CYCLE 2

Sl. No.	Course Code	Course Name	L	T	P	C	Type of Skill/ Focus	Course Caters to	
Semester 1 - Chemistry Cycle						**22**			
1	MAT2402	Probability and Statistics	3	1	0	4	F		BSC
2	CHE7601	Environmental Studies	0	0	0	0	F		MAC
3	CIV1200	Foundations of Integrated Engineering	2	0	0	2	S		ESC
4	CHE2501	Chemistry of Smart Materials	3	0	0	3	S		BSC
5	CSE2200	Problem Solving using C	2	0	0	2	S		PCC
6	ENG2501	Advanced English	2	0	0	2	S		HSMC
7	EEE1200	Basics of Electrical and Electronics Engineering	3	0	0	3	F/S		ESC
8	LAW7601	Indian Constitution	0	0	0	0	F		MAC
9	CSE2201	Problem Solving using C Lab	0	0	4	2	S		PCC
10	CHE2502	Chemistry of Smart Materials Lab	0	0	2	1	S		BSC
11	PPS1026	Industry Readiness Program – II	0	0	2	0	S		MAC
12	EEE1250	Basics of Electrical and Electronics Engineering Lab	0	0	2	1	F/S		ESC
13	ECE1511	Design Workshop	1	0	2	2	S/EM		ESC
Semester 2 – Physics Cycle						**19**			
1	MAT2301	Calculus and Differential Equations	3	1	0	4	F		BSC
2	PHY2501	Optoelectronics and Quantum Physics	3	0	0	3	F		BSC
3	MEC1006	Engineering Graphics	2	0	0	2	S		ESC

4	ENG1900	English for Technical Communication	2	0	0	2	S		HSMC
5	CSE1500	Computational Thinking using Python	2	0	2	3	S		ESC
6	ECE2022	Digital Design	2	0	0	2	F/S		ESC
7	DES1146	Introduction to Design Thinking	1	0	0	1	F		HSMC
8	PHY2504	Optoelectronics and Quantum Physics Lab	0	0	2	1	F		BSC
9	PPS1025	Industry Readiness Program – I	0	0	2	0	S		MAC
10	ECE2052	Digital Design Lab	0	0	2	1	F/S		ESC

Sl. No.	Course Code	Course Name	L	T	P	C	Type of Skill/ Focus	Course Caters to	Basket
Semester 3						**23**			
1	MAT2303	Linear Algebra and Vector Calculus	3	1	0	4	EM		BSC
2	CSE2251	Data Communication and Computer Networks	3	0	0	3	S		PCC
3	CSE2252	Data Communication and Computer Networks Lab	0	0	2	1	S		PCC
4	CSE2253	Data Structures	3	0	0	3	S		PCC
5	CSE2254	Data Structures Lab	0	0	2	1	S		PCC
6	CSE2255	Object Oriented Programming Using Java	3	0	0	3	S/EM		PCC
7	CSE2256	Object Oriented Programming Using Java Lab	0	0	2	1	S/EM		PCC
8	CSE2257	Computer Organization and Architecture	3	0	0	3	S		PCC
9	CSE2258	Web Technologies	3	0	0	3	S/EM		PCC
10	CSE2259	Web Technologies Lab	0	0	2	1	S/EM		PCC
11	CIV7601	Universal Human Values and Ethics	0	0	0	0	F		MAC
12	APT4002	Introduction to Aptitude	0	0	2	0	AT		MAC
Semester 4						**24**			
1	MAT2404	Discrete Mathematics	3	1	0	4	F		BSC
2	CSE2260	Database Management Systems	3	0	0	3	S		PCC
3	CSE2261	Database Management Systems Lab	0	0	2	1	S		PCC
4	CSE2500	Data Analytics	3	0	0	3	S/EM		PCC
5	CSE2501	Data Analytics Lab	0	0	2	1	S/EM		PCC
6	CSE2262	Analysis of Algorithms	3	1	0	4	S		PCC
7	CSE2263	Analysis of Algorithms Lab	0	0	2	1	S		PCC
8	CSE2264	Essentials of AI	3	0	0	3	S/EM		ESC
9	CSE2265	Essentials of AI Lab	0	0	2	1	S/EM		ESC
10	FIN1002	Essentials of Finance	3	0	0	3	S		HSMC

11	APT4004	Aptitude Training - Intermediate	0	0	2	0	AT		MAC
Semester 5						**23**			
1	CSE2266	Theory of Computation	3	0	0	3	S		PCC
2	CSE2502	Cryptography and Network Security	3	0	0	3	S		PCC
3	CCS2504	Ethical Hacking	2	0	0	2	S/EM		PCC
4	CCS2505	Ethical Hacking Lab	0	0	4	2	S/EM		PCC
5	CSE2269	Operating Systems	3	0	0	3	S		PCC
6	CSE2270	Operating Systems Lab	0	0	2	1	S/EM		PCC
7	CSD2001	Applied Data Science	3	0	0	3			PCC
8	CSD2002	Applied Data Science Lab	0	0	2	1			PCC
9	CSEXXXX	Professional Elective – I	3	0	0	3	S/EM		PEC
10	CSE7000	Internship	0	0	0	2	S/EM		PRW
11	APT4006	Logical and Critical Thinking	0	0	2	0	S/EM		MAC
Semester 6						**22**			
1	CSE2271	Software Design and Development	3	0	0	3	S		PCC
2	CSE2272	Cloud Computing	2	0	0	2	S/EM		PCC
3	CSE2273	Cloud Computing Lab	0	0	2	1	S/EM		PCC
4	CSE2505	Mobile Application Development	2	0	0	2	S/EM		PCC
5	CSE2506	Mobile Application Development Lab	0	0	4	2	S/EM		PCC
6	CSE2274	Competitive Programming and Problem Solving	0	0	4	2	S/EM		ESC
7	CSEXXXX	Professional Elective – II	2	0	2	3	S		PEC
8	CSEXXXX	Professional Elective – III	2	0	2	3	S		PEC
9	XXXXXXX	Open Elective – I	3	0	0	3	S		OEC
10	APT4005	Aptitude for Employability	0	0	2	1	S		HSMC

Semester 7						**17**		
1	CSEXXXX	Professional Elective – IV	2	0	2	3	S/EM	PEC
2	CSEXXXX	Professional Elective – V	3	0	0	3	S/EM	PEC
3	CSEXXXX	Professional Elective – VI	3	0	0	3	S/EM	PEC
4	XXXXXXX	Open Elective – II	3	0	0	3	S/EM	OEC
5	CSE7100	Mini Project	-	-	-	4	S/EM	PRW
6	PPS3018	Preparedness for Interview	0	0	2	1	S	HSMC
Semester 8						**10**		
1	CSE7300	Capstone Project	-	-	-	10	S/EM	PRW

Course Catalogues
BSC:

Course Code: PHY2501	**Course Title:** Applied Physics for Computer Engineers **Type of Course: 1] School Core**	L-T-P-C	3	0	0	3
Version No.	1.0					
Course Pre-requisites	**Class 11 & 12 Physics**					
Anti-requisites	**NIL**					
Course Description	The purpose of this course is to enable the students to understand the fundamentals, working and applications of optoelectronic devices and to develop the basic abilities to appreciate the applications of advanced microscopy and quantum computers. The course develops the critical thinking and analytical skills.					
Course Out Comes	On successful completion of the course the students shall be able to: CO1: To understand the concepts of electrical conducting properties of metal, semiconductor and superconductivity. CO2: To understand the principles of quantum mechanics. CO3: Discuss the quantum concepts used in quantum computers. CO4: Explain the applications of lasers and optical fibers in various technological fields.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of "Applied Physics for Computer Science Cluster "and to attain the basic knowledge related to quantum mechanics and computation.					
Course Content:						
Module 1	**Electrical Conductivity Of Solids And Semiconducting Devices**	Assignment	Data collection on efficiency of solar cells.			**No. of Classes : 12**
Topics: Classification of materials based on bandgap, Fermi energy and Fermi level, Fermi level in semiconductors, Law of mass action, Electrical conductivity of a semiconductor, Hall effect, Superconductivity, p-n junctions, Zener diode, Solar cells, I-V characteristics, and LEDs						
Module 2	**Quantum Mechanics**	Assignment				**No. of Classes : 12**
Topics: Introduction, de-Broglie hypothesis, Heisenberg's uncertainty principle- statement and physical significance. Wave function-properties and physical significance. Schrodindger's time independent wave equation, Probability density and normalization of wave function. Wave Function in Ket Notation: Matrix form of wave function, Identity operator, Determination of I|0> and I|1>, Pauli Matrices and its operations on 0 and 1 states, Mention of Conjugate and Transpose, Unitary Matrix U, Examples: 2x2 Matrices and their multiplication (Inner Product), Probability, Orthogonality						
Module 3	**Quantum Computing**	Term paper	Seminar on quantum computers.			**No. of classes : 11**

Topics: Introduction to quantum computing, Moore's law & its end, Differences between classical and quantum computing, Concept of Qubit and its properties, . Representation of qubit by Bloch sphere, Quantum Gates: Single Qubit Gates: Quantum Not Gate, Pauli Z Gate, Hadamard Gate, Phase Gate (or S Gate), T Gate. Multiple Qubit Gates: Controlled gate - CNOT Gate, (Discussion for 4 different input states). Representation of Swap gate, Controlled - Z gate, Toffoli gate. Problems.				
Module 4	**Lasers And Optical Fibers**	Term paper	Case study on medical applications of Lasers.	**No. of classes :10**
Topics: Interactions of radiations with matter, expression for energy density of a system under thermal equilibrium in terms of Einstein's coefficients, conditions for LASER action using Einstein's coefficients, Characteristics of laser, conditions and requisites of laser, Principle of optical fibers, Numerical aperture and acceptance angle (Qualitative), Attenuation, Applications: Point to point communication with block diagram, application of optical fibers in endoscopy.				
Targeted Application & Tools that can be used: 1. Areas of application are optoelectronics industry, Solar panel technologies, quantum computing software, electronic devices using transistors and diodes, memory devices, endoscopy, SQUIDS in MRI, Advanced material characterizations using SEM and STM. 2. Origin, excel and Mat lab soft wares for programming and data analysis.				
Project work/Assignment: Mention the Type of Project /Assignment proposed for this course				
Assessment Type • Midterm exam • Assignment (review of digital/ e-resource from PU link given in references section - mandatory to submit screen shot accessing digital resource.) • Quiz • End Term Exam • Self-Learning 1. Prepare a comprehensive report on non-conventional energy resources in Karnataka and their pros and cons. 2. Write a report on importance of quantum entanglement in supercomputers.				
Text Book 1. Engineering Physics by Avadhanalu, Revised edition, S. Chand Publications,2024. 2. Quantum Computation and Quantum Information, Michael A. Nielsen & Isaac L. Chuang, Cambridge Universities Press, 2010 Edition				
References: 1. Elementary Solid state Physics: Principles and Applications by M.A. Omar, 1st Edition, Pearson Publications, 2002. 2. Principles of Quantum Mechanics by R Shankar, 2nd edition, springer Publications, 2011. 3. Optoelectronics: An Introduction by John Wilson and John Hawkes, 3rd edition, Pearson Publications, 2017. 4. Engineering Physics by Gaur and Gupta, Dhanpat Rai Publications, 2012. 5. Introduction to Quantum Mechanics, David J Griffiths, Cambridge University Press, 2019				
E-Resourses:				

1. https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=553045&site=ehost-live
2. https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=833068&site=ehost-live
3. https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=323988&site=ehost-live
4. https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=1530910&site=ehost-live
5. https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=486032&site=ehost-live

Topics relevant to "SKILL DEVELOPMENT": Fundamentals of materials, Lasers and optical fibers.
for Skill Development through Participative Learning Techniques. This is attained through the Assignment/ Presentation as mentioned in the assessment component in course handout.

Course Code: **PHY2504**	**Course Title:** Applied Physics Lab for Computer Engineers **Type of Course: 1] School Core**	L-T-P-C	0	0	2	1
Version No.	1.0					
Course Pre-requisites	**Class 11 & 12 Physics**					
Anti-requisites	**NIL**					
Course Description	The laboratory provides an opportunity to validate the concepts taught and enhances the ability to use the concepts for technological applications. The laboratory tasks aim to develop following skills: **An attitude of enquiry, confidence and ability to tackle new problems, ability to interpret events and results, observe and measure physical phenomena, select suitable equipment, instrument and materials, locate faults in systems.**					
Course Out Comes	On successful completion of the course the students shall be able to: CO1: To understand electrical and optical properties of materials CO2: Interpret the results of various experiments to verify the concepts used in optoelectronics and advanced devices.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of " Applied Physics for Computer Science Cluster "and attain **Skill Development** through **Experiential Learning** techniques					

List of Laboratory Tasks:
Experiment No. 1: Experimental errors and uncertainty using excel
Level 1: Calculation of accuracy and precision of a given data
Level 2: propagation of errors in addition, subtraction, multiplication and division.

Experiment N0 2: To determine the wavelength of semiconductor diode Laser and to estimate the particle size of lycopodium powder using diffraction.
Level 1: Determination of Wavelength of Laser
Level 2: Finding the particle size of lycopodium powder.

Experiment No. 3: To determine the proportionality of Hall Voltage, magnetic flux density and the polarity of Charge carrier.
Level 1: To determine the proportionality of Hall Voltage and magnetic flux density
Level 2: To determine the polarity of Charge carrier.

Experiment No. 4: To study the I-V characteristics of a given zener diode in forward and reverse bias conditions.

Level 1: To study I –V characteristics of the given Zener diode in reverse bias and to determine break down voltage.
Level 2: To study I –V characteristics of the given Zener diode in forward bias and to determine knee voltage and forward resistance.

Experiment No. 5: To study input and output characteristics of a given Transistor.
Level 1: To determine the input resistance of a given transistor.
Level 2: To determine current transfer characteristics and transistor parameters of a given transistor.

Experiment No. 6: Determination of Fermi energy and Fermi temperature of a given metal and bimetallic wire.

Level 1: Determination of Fermi energy and Fermi temperature of given metal wire.
Level 2: Determination of Fermi energy and Fermi temperature of given bimetallic wire.

Experiment No. 7: To study the I-V characteristics and I-R characteristics of a solar cell as a function of the irradiance.
Level 1: To study the I-V characteristics
Level 2: I-R characteristics of a solar cell as a function of the irradiance.

Experiment No. 8: Calculate the numerical aperture and study the losses that occur in optical fiber cable. .
Level 1: Calculate the numerical aperture.
Level 2: study the losses that occur in optical fiber cable.

Experiment No. 9: Plotting I-V characteristics in forward and reverse bias for LEDs and Determination of knee voltage.
Level 1: Plotting I-V characteristics in forward and reverse bias for LEDs
Level 2: Determination of knee voltage.

Experiment No. 10: Determination of Stefan's constant and verification of Stefan-Boltzmann Law.
Level 1: Determination of Stefan's constant
Level 2: Verification of Stefan-Boltzmann Law.

Experiment No. 11: : Dielectric constant
Level 1: Determination of Dielectric constant of given material

Level 2: compare the obtain results with other materials

Experiment No. 12: determine the wavelength of monochromatic light, such as sodium light, using Newton's rings.
Level 1: Determination of wavelength
Level 2: determine the radius of curvature of the Plano-convex lens.

Targeted Application & Tools that can be used:

1. Areas of application are optoelectronics industry, Solar panel technologies, quantum computing software, electronic devices using transistors and diodes, memory devices, endoscopy, SQUIDS in MRI, Advanced material characterizations using SEM and STM.
2. Origin, excel and Mat lab soft wares for programming and data analysis.

Project work/Assignment: Mention the Type of Project /Assignment proposed for this course

Assessment Type

- Midterm exam
- Assignment (review of digital/ e-resource from PU link given in references section - mandatory to submit screen shot accessing digital resource.)
- Quiz
- End Term Exam
- Self-Learning

1. Prepare a comprehensive report on non-conventional energy resources in Karnataka and their pros and cons.
2. Write a report on importance of quantum entanglement in supercomputers.

Course Code: MAT2402	**Course Title: Probability and Statistics** **Type of Course:1] School Core**	**L-T- P- C**	**3**	**0**	**0**	**3**
Version No.	1.0					
Course Pre-requisites	MAT2301					
Anti-requisites	**NIL**					
Course Description	The course introduces the concepts of probability theory and statistical analysis, covering how to collect, organize, interpret, and draw inferences from data using mathematical models to understand randomness and uncertainty, with applications across various fields like science, engineering, economics, and social sciences.					
Course Objective	The objective of the course is to equip students with the foundational knowledge of probability theory and statistical methods, enabling them to collect, analyze, interpret data, and make informed decisions based on the likelihood of events occurring in various situations, often applied across different fields like science, engineering, and business.					
Course Out Comes	On successful completion of the course the students shall be able to: CO1 - be able to compute conditional probabilities directly and using Bayes' theorem, and check for independence of events. CO2 - be able to set up and work with discrete & continuous random variables; in particular, to understand the Bernoulli, binomial, geometric, Poisson distributions, uniform, normal, and exponential distributions. CO3 - Identifying different types of data relationships (linear, polynomial, exponential, logarithmic). CO4 - be able to use specific significance tests, including z-test, t-test (one- and two-sample), and chi-squared test					
Course Content:						
Module 1	**Basic Probability**					**(6 Classes)**
Probability of an Event, multiplication rule, combinations, permutations, Addition Law, Multiplication Law, Conditional Probability, Bayes's Theorem and Problems.						
Module 2	**Random Variables and Bivariate Distributions**	**Assignment**				**(15 Classes)**
Random Variables (discrete and continuous), Probability Mass/Density Functions, Mathematical Expectations, discrete probability distributions - Binomial distribution, Poisson distribution, geometric distribution, Continuous uniform distribution - exponential distribution, normal distribution, gamma distribution. Bivariate distributions and their properties, distribution of sums and quotients, conditional densities, Bayes' rule.						
Module 3	**Curve Fitting & Statistical Methods**					**(13 Classes)**
Curve Fitting (Straight Line ($y = a + bx$), Parabola ($y = a + bx + cx^2$), Exponential Curves ($y = ae^{bx}$, $y = ab^x$ and $y = ax^b$) Measures of Central tendency, Moments, skewness and Kurtosis, Correlation - Karl Pearson's coefficient of correlation and rank correlation (with & Without repetition, Multiple Correlation - Problems. Regression analysis - lines of regression, Multiple regression - Problems.						
Module 4	**Joint Probability Distribution and Sampling Theory**	**Assignment**				**(15 Classes)**

Joint Probability distribution for two discrete random variables, expectation and covariance.
Random sampling, sampling distributions, Standard Error, Type I & Type II errors, Testing of Hypothesis, Test of significance - Large sample test for single proportion, difference of proportions, single mean, difference of means, and difference of standard deviations, Test for single mean, difference of means and correlation coefficients, test for ratio of variances - Chi-square test for goodness of fit and independence of attributes.

Targeted Application & Tools that can be used:
The contents of this course has direct applications in most of the core engineering courses for problem formulations, Problem Solution and system Design.
Tools Used: R software (Open Source)

Assignment:

Select any one simple differential equation pertaining to the respective branch of engineering, identify the ɛndent and independent variable – Obtain the solution and compare the solution sets by varying the values of the ɛndent variable.

Text Book

1. Ronald .E. Walpole, Raymond. H. Myers, Sharon. L Myers, and Keying E. Ye, "Probability and Statistics for Engineers and Scientists", Pearson Education, Delhi-9th edition, 2012.
2. B. S. Grewal (2017), Higher Engineering Mathematics by, 44th Edition, Khanna Publishers.

References:

1. Miller and Freund, Probability and Statistics for Engineers, Pearson Education Ltd.
2. Erwin Kreyzig, Advanced Engineering Mathematics, John Wiley and sons, Inc.10th Edition.
3. Douglas C. Montgomery & George Runger, Applied Statistics and Probability for Engineers, , Wiley Publications

E-resources/ Web links:

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=ECO95_30102024_10427
https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=ECO95_30102024_100198
https://nptel.ac.in/courses/109104124
https://nptel.ac.in/courses/111106051
https://nptel.ac.in/courses/111102137
https://www.math.hkust.edu.hk/~maqian/ma006_0607F.html
https://www.scu.edu.au/study-at-scu/units/math1005/2022/
Presidency University's Knimbus library URL is: presiuniv.knimbus.com

Topics relevant to SKILL DEVELOPMENT: The course focuses on the concepts of Vector calculus and Linear Algebra with reference to specific engineering problems. The course is of both conceptual and analytical type in nature through Problem Solving**.** This is attained through the assessment component mentioned in the course handout.

Course Code: MAT2404	**Course Title: Discrete Mathematics Type of Course:1] School Core**	**L-T- P- C**	**3**	**1**	**0**	**4**
Version No.		1.0				
Course Pre-requisites		MAT2302				
Anti-requisites		**NIL**				
Course Description		The course explores the study of mathematical structures that are fundamentally discrete (not continuous), focusing on concepts like set theory, logic, graph theory, combinatorics, and number theory, with applications primarily in computer science fields like algorithms, software development, and cryptography; it covers topics such as propositional logic, proof techniques, relations, functions, counting principles, and basic graph algorithms, providing a foundation for analyzing discrete problems and structures within computer science.				

Course Objective		The main objective of the course is that students should learn a particular set of mathematical facts and how to apply them. It teaches students how to think logically and mathematically through five important themes: mathematical reasoning, combinatorial analysis, discrete structures, algorithmic thinking, and applications and modeling. A successful discrete mathematics course should carefully blend and balance all five themes.	
Course Outcomes		On successful completion of the course the students shall be able to: CO1 - Explain logical sentences through predicates, quantifiers and logical connectives. CO2 - Deploy the counting techniques to tackle combinatorial problems CO3 - Comprehend the basic principles of set theory and different types of relations. CO4 - Apply different types of structures of trees for developing programming skills	
Course Content:			
Module 1	**Fundamentals of Logic**		**(10 Classes)**
Basic Connectives and Truth Tables, Propositional Logic, Applications of Propositional Logic, Propositional Equivalences, Predicates and Quantifiers, Nested Quantifiers, Rules of Inference, Introduction to Proofs, Proof Methods and Strategy.			
Module 2	**Principle of Counting**	**Assignment**	**(15 Classes)**
The Well Ordering Principle – Mathematical Induction The Basics of Counting, Permutations and Combinations, Binomial Coefficients and Identities, Generalized Permutations and Combinations, Generating Permutations and Combinations Advanced Principle Counting: The Principle of Inclusion and Exclusion, Generalizations of the Principle, Derangements – Nothing is in its Right Place, Rook Polynomials.			
Module 3	**Relations and Functions**		**(10 Classes)**
Cartesian Products and Relations, Functions, One-to-One, Onto Functions. The Pigeon-hole Principle, Function Composition and Inverse Functions. Relations, Properties of Relations, Computer Recognition – Zero-One Matrices and Directed Graphs, Partial Orders, Lattice, Hasse Diagrams, Equivalence Relations and Partitions.			
Module 4	**Recurrence Relations and Generating Functions**		**(10 Classes)**
Homogeneous and inhomogeneous recurrences and their solutions - solving recurrences using generating functions - Repertoire method - Perturbation method - Convolutions - simple manipulations and tricks.			
Module 5	**Graph Theory & Algorithms on Networks**	**Assignment**	**(15 Classes)**
Definitions and basic results - Representation of a graph by a matrix and adjacency list - Trees - Cycles - Properties - Paths and connectedness - Sub graphs - Graph Isomorphism - Operations on graphs - Vertex and edge cuts - Vertex and edge connectivity, Euler and Hamilton Paths, Shortest-Paths. Tree - Definitions, Properties, and Examples, Routed Trees, Binary search tree, Decision tree, spanning tree: BFS, DFS. Algorithms on Networks - Shortest path algorithm- Dijikstra's algorithm, Minimal spanning tree- Kruskal algorithm and Prim's algorithm.			
Targeted Application & Tools that can be used: Discrete mathematics provides the mathematical foundations for many computer science courses including data structures, algorithms, database theory, automata theory, formal languages, compiler theory, computer security, and operating systems.			
Assignment:			
Assignment 1: Logic Equivalences and Predicate calculus. **Assignment 2: Equivalence Relations and Lattices** **Assignment 3: Recurrence Relations**			
Text Book			

1. Kenneth H. Rosen, “Discrete Mathematics and its Applications”, McGraw-Hill,s 8th Edition,2019. 2. Harary – Graph Theory, Addison-Wesley Publishing Company.
References: 1. Arthur Gill, "Applied Algebra for Computer Science", Prentice Hall. 2. K.D. Joshi, "Discrete Mathematics", Wiley Eastern Ltd. 3. Ralph. P. Grimaldi., “Discrete and Combinatorial Mathematics: An Applied Introduction”, 4th Edition, Pearson Education Asia. **E-resources/ Web links:** https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&ue_id=EBSCO95_30102024_54588 https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&ue_id=EBSCO95_30102024_375 https://www.math.hkust.edu.hk/~maqian/ma006_0607F.html https://www.scu.edu.au/study-at-scu/units/math1005/2022/
Topics relevant to SKILL DEVELOPMENT: The course focuses on the concepts of calculus and differential equation with reference to specific engineering problems. The course is of both conceptual and analytical type in nature through Problem solving**.** This is attained through the assessment component mentioned in course handout.

Course Code: **CHE2501**	**Course Title: Chemistry of Smart Materials (CSE and Allied)** **Type of Course: Theory only-Basic sciences course**	**L-T- P- C**	**3**	0	0	3
Version No.	1.0					
Course Pre-requisites	Fundamental knowledge of organic and inorganic chemistry					
Anti-requisites	**NIL**					
Course Description	The objective of the course is to introduce the students to concepts and applications of chemistry of smart materials. The course also aims to enhance the knowledge of smart materials associated with memory system, display devices, , sensors, energy devices and environment. It will also cultivate an ability to identify chemistry in each of smart engineered materials and interpret solutions for the challenges connected to memory, display, energy, smart, green and sustainable technologies. It targets to strengthen the fundamental concepts behind chemistry of smart materials and then builds an interface with their industrial applications. **This course is designed to cater to Environment and Sustainability**					
Course Objective	The objective of the course is ‘SKILL DEVELOPMENT’ of the student by using Participative learning techniques.					
Course Outcomes	On successful completion of this course the students shall be able to: 1) Relate the knowledge of chemistry to computational approaches to identify materials functionalities and properties 2) Recognize and interpret solutions for the challenges connected to memory, display, smart, green and sustainable technologies. 3) Explain the quality parameters of engineering materials associated with					

	environment and sensors. 4) Interpret the knowledge of sustainable chemistry for E- waste management. 5) Analyse the importance of various electrochemical sources in energy systems.

Course Content:				
Module 1	**mputational Chemistry**	Assignment	ata Collection and analysis	**09 classes**
Topics: Fundamental particles of atom – their mass, charge and location – atomic number and mass number, Stabilizing interactions: Bonded and non-bonded interactions. Chemistry of weak interactions – van der Waals force and hydrogen bonding, Density functional theory. 3D co-ordinate generation for small molecules, geometry optimization by Molview. Chemical Databases: Chemoinformatics, MSDS ***Self- learning topics:*** Scope, cost and efficiency of computational modeling.				
Module 2	**terials for Memory and Display Systems**	Assignment	ata Collection and analysis	**09 Classes**
Topics: Memory Systems : Introduction, classification of electronic memory devices- Transistor, capacitor, charge -transfer and Resistor, types of materials - organic, polymeric and hybrid materials, and applications, manufacturing of semiconductor chips. **Display Systems:** photo and electroactive materials , materials for display -Principle, Properties and applications: Liquid crystals for LCD-Liquid crystals display, Basics of LED: OLED-organic light emitting diode and light emitting electrochemical cells. ***Self- learning topics:*** Green computing: Biocomposite based memory devices				
Module 3	**nomaterials based Smart Sensors and Devices**	Assignment	ata Collection and analysis	**09 Classes**
Topics: Nanomaterials- Introduction, classification based on dimensionality, quantum confinement. Size dependent properties, Synthesis, Properties of CNT and Graphene and their application as **Materials for data analysis and packaging** -RFID and IONT. **Sensors:** Introduction, types, Principle and applications- electrochemical sensor: nanomaterials for sensing applications - Glucose, VOC sensing. **Self-learning topics:** Fullerene, biomolecules in sensing, Strain sensors				
Module 4	**Sustainable Materials and Development**	Quiz/Seminar	Data Collection and analysis	**09 Classes**
Topics: E waste: Introduction, E waste Hazards, E- waste management, Recovery of precious metal- Cu by Hydrometallurgy. **Green Chemistry**: Fundamentals and 12 principles with examples, Carbon footprint and sequestration **Sustainable Chemistry:** -Introduction to Biomaterials- PLA , polymers in bio-compatible and bio-degradable materials - Poly(3-hydroxybutyrate-co-3-hydroxyvalerate) PHBV, synthesis and applications in drug delivery. ***Self-learning topics:*** circular economy- case studies.				
Module 5	**Energy Science**	Quiz/Seminar	Data Collection and analysis	**09 Clas**
Topics: Battery technology: Fundamentals of electrochemistry, Introduction to electrochemical				

storage devices: battery (Lithium-ion battery- $LiMnO_2$, $LiCoO_2$, metal air batteries- LiO_2) and supercapacitors-Introduction, Principle, Types - EDLC, pseudo and asymmetric capacitor.

Photovoltaics: Solar cells - Construction and working principle; types- Inorganic, Organic and quantum dot sensitized (QDSSC's).

Self -learning topics: Battery technology for e-mobility, Green hydrogen

Targeted Application & Tools that can be used:

Application areas are Data storage and analysis, logistics, Biomedicine, Energy, Environment and sustainability

Tools: Molview, chemdraw, excel etc

Project work/Assignment:

Assessment Type

- Midterm exam
- Assignment (review of digital/ e-resource from PU link given in references section - mandatory to submit screenshot accessing the digital resource.)
- Quiz/Student Seminar
- End Term Exam
- Self-learning

Text Book

1. Wiley, "Engineering Chemistry", Wiley.
2. G.A. Ozin and A.C. Arsenault, Nanochemistry: A chemical approach to nanomaterials, Royal Society of Chemistry, 2009

Reference Books

1. Functional and smart materials, Chander Prakash, Sunpreet Singh, J. Paulo Davim, 2020, CRC Press,
ISBN: 978-036-727-510-5.
2. E-waste recycling and management: present scenarios and environmental issues, Khan, Anish, and Abdullah M. Asiri. 2019, Springer, Vol. 33. ISBN: 978-3-030-14186-8.
Essentials of computational chemistry: theories and models, Christopher J Cramer, 2013, John Wiley & Sons. ISBN: 978-0-470-09182-1.
3. Energy storage and conversion devices: Supercapacitors, batteries and hydroelectric cells, Anurag Gaur,
A. L. Sharma, Anil Arya. 2021, CRC press, 1st edition, ISBN: 978-1-003-14176-1.
4. Fundamentals of analytical chemistry: An introduction, Douglas A. Skooget etal., 2004 Thomson Asia pte
Ltd., 8th, ISBN: 978-0-495-55828-6
5. Functional and smart materials, Chander Prakash, Sunpreet Singh, J. Paulo Davim, 2020, CRC Press, ISBN: 978-036-727-510-5.
6. Electrical and electronic devices, circuits and materials: Technological challenges and solutions. Tripathi,
S. L., Alvi, P. A., & Subramaniam, U, 2021, John Wiley & Sons, ISBN: 978-0367564261.
7. F. Jensen, Introduction to Computational Chemistry, 3rd edition, Wiley, 2017.

E resources

1. https://presiuniv.knimbus.com/user#/searchresult?searchId=computational%20chemistry&_t=1738054970142
2. https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_48504
3. https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_147967

4. https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_130301 5. https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_87297 6. https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_67006 7. https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_137261 **8.** https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_86712
Skill Sets All topics in theory component are relevant to Environment and Sustainability.

Course Code: CHE2502	**Course Title: Chemistry of Smart Materials Lab (CSE and Allied)** **Type of Course: Laboratory course- Basic science course**	**-P- C**		**0**		**1**
Version No.	1.0					
Course Pre-requisites	Before undertaking this Chemistry of Smart Materials Lab course, students are expected to possess foundational knowledge of chemistry, including an understanding of acids and bases, metals and metal ions, oxidizing and reducing agents, various types of instrumental analysis, and the proper use of laboratory glassware. Additionally, students should be familiar with handling chemicals and glassware safely and adhering to essential laboratory safety precautions.					
Anti-requisites	**NIL**					
Course Description	The laboratory course aims to develop experimental skills and apply fundamental chemical principles to address chemistry-related problems in engineering. The experiments are carefully designed to complement the theoretical concepts covered in lectures, providing hands-on experience to deepen understanding and reinforce learning. This course is designed to cater to Environment and Sustainability.					
Course Objective	The objective of the course is to **familiarize the learners with the concepts of "Chemistry of Smart Materials Lab" and attain SKILL DEVELOPMENT through EXPERIENTIAL LEARNING techniques**.					
Course Outcomes (COs)	**On successful completion of the course, students shall be able to:** **CO1:** recognize the basic techniques and instrumentation used in chemistry laboratories for quantitative analysis. **CO2:** estimate the presence of acids and metal ions in domestic and industrial waste using laboratory techniques. **CO3:** review the experimental results and demonstrate improved experimental skills through hands-on laboratory experience. **CO4:** classify laboratory techniques such as experimental setups for synthesis, purification, recovery and analysis.					
Course Content:	**Total 30 sessions**					
Experiment 1	Experimental	Data Collection	Analysis and			

			nterpretation
Determination of strength of strong acid in battery electrolyte using conductometric sensors.			
Experiment 2	Experimental	Data Collection	Analysis and Interpretation
Estimation of iron from e-waste using Electrochemical sensors.			
Experiment 3	Experimental	Data Collection	Analysis and Interpretation
Determination of pKa of organic acid of battery electrolyte using pH sensor.			
Experiment 4	Experimental	Data Collection	Analysis and Interpretation
Estimation of copper from PCBs by using colorimeter (Optical Sensor).			

ESC:

Course Code: MEC1006	**Course Title:** Engineering Graphics **Type of Course:** School Core & Theory Only	**L- P- C**	2	0	2
Version No.	1.2				
Course Pre-requisites	**NIL**				
Anti-requisites	**NIL**				
Course Description	**The course is designed with the objective of giving an overview of engineering graphics. It is introductory in nature and acquaints the students with the techniques used to create engineering drawings. The course emphasizes on projection of points, lines, planes and solids and isometric projections.**				
Course Objective	**The objective of the course is to familiarize the learners with the concepts of "Engineering Graphics" and attain SKILL DEVELOPMENT through Problem solving methodologies.**				
Course Outcomes	**On successful completion of this course the students shall be able to:** 1. **Demonstrate competency of Engineering Graphics as per BIS conventions and standards.** 2. **Comprehend the theory of projection for drawing projections of Points, Lines and Planes under different conditions.** 3. **Prepare multiview orthographic projections of Solids by visualizing them in different positions.** 4. **Prepare pictorial drawings using the principles of isometric projections to visualize objects in three dimensions.**				
Course Content:					
Module 1	Introduction to Drawing	Assignment	Standard technical drawing		**02 Sessions**
Topics: **Introduction, drawing instruments and their uses, relevant BIS conventions and standards, Lettering, Line conventions, dimensioning, Selection of drawing sheet size and scale.**					

				[02 Hours: Comprehension Level]
Module 2	Orthographic projections of	Assignment	Projection methods Analysis	**10 Sessions**
	Points, Straight Lines and Plane Surfaces			
Topics: **Introduction, Definitions – Elements of projection and methods of projection, Planes of projection, reference line and conventions adopted. First angle and third angle projections. Projection of Points in all 4 quadrants.** **Projections of Straight Lines (located in first quadrant/first angle projection only): True and apparent lengths, true and apparent Inclinations to reference planes. (No application problems). Projection of Plane surfaces (First angle projection): Regular plane surfaces – triangle, square, rectangle, pentagon, hexagon and circle – in different positions inclined to both the planes using change of position method only.** **[10 Hours: Application Level]**				
Module 3	Orthographic Projections of Solids	Assignment	Multi-view drawing Analysis	**10 Sessions**
Topics: **Introduction, Projection of right regular prisms, pyramids, cone, hexahedron and tetrahedron in different positions (Problems resting on HP only and First angle projection).** **[10 Hours: Application Level]**				
Module 4	Isometric Projections of Solids (Using isometric scale only)	Assignment	Spatial Visualization	**8 Sessions**
Topics: **Introduction, Isometric scale, Isometric projections of right regular prisms, cylinders, pyramids, cones and their frustums, spheres and hemispheres, hexahedron (cube), and combination of 2 solids, conversion of orthographic view to isometric projection of simple objects.** **[8 Hours: Application Level]**				
Text Book: **1.N. D. Bhatt, "Engineering Drawing: Plane and Solid Geometry," Charotar Publishing House Pvt. Ltd.**				
References: 1. **K.R. Gopalakrishna, "Engineering Graphics", Subhash Publishers, Bangalore.** 2. **D. M. Kulkarni, A. P. Rastogi, A. K. Sarkar, "Engineering Graphics with AutoCAD," Prentice Hall.** 3. **D. A. Jolhe, "Engineering Drawing with Introduction to AutoCAD," Tata McGraw Hill.** **Web resources:** **https://nptel.ac.in/courses/112103019**				

Topics relevant to "SKILL DEVELOPMENT": Projection in first and third angle for **SKILL DEVELOPMENT** through **Problem Solving methodologies**. This is attained through the assessment component mentioned in the course handout.

<table>
<tr><td>Course Code: ECE2022</td><td colspan="3">Course Title: Digital Design
Type of Course: Program Core</td><td>L- T-P- C</td><td>2</td><td>0</td><td>0</td><td>2</td></tr>
<tr><td>Version No.</td><td colspan="8">1.0</td></tr>
<tr><td>Course Pre-requisites</td><td colspan="8">[1] Elements of Electronics/Electrical Engineering, 2] Basic concepts of number representation, Boolean Algebra</td></tr>
<tr><td>Anti-requisites</td><td colspan="8">NIL</td></tr>
<tr><td>Course Description</td><td colspan="8">The purpose of this course is to enable the students to appreciate the fundamentals of digital logic circuits and Boolean algebra focusing on both combinational and sequential logic circuits. The course emphasizes on minimization techniques for making canonical and low-cost digital circuit implementations. This course deals with analysis and design of digital electronic circuits. The course also creates a foundation for future courses which includes Computer Architecture, Microprocessors, Microcontrollers, and Embedded Systems etc.
The course enhances the Design, Implementation and Programming abilities through laboratory tasks. The associated laboratory provides an opportunity to verify the theoretical knowledge.</td></tr>
<tr><td>Course Objective</td><td colspan="8">The objective of the course is to familiarize the learners with the concepts of Digital Design and attain the SKILL DEVELOPMENT through EXPERIENTIAL LEARNING.</td></tr>
<tr><td>Course Outcomes</td><td colspan="8">On successful completion of this course the students shall be able to:
i.Describe the concepts of number systems, Boolean algebra and logic gates.
ii.Apply minimization techniques to simplify Boolean expressions.
iii.Demonstrate the Combinational circuits for a given logic
iv.Demonstrate the Sequential and programmable logic circuits</td></tr>
<tr><td>Course Content:</td><td colspan="8"></td></tr>
<tr><td>Module 1</td><td colspan="2">Fundamentals of Number systems- Boolean algebra and digital logic</td><td>Application Assignment</td><td colspan="2">Data Analysis task</td><td colspan="2">06 classes</td></tr>
<tr><td colspan="9">Topics:
Review of Number systems and logic gates, Number base conversions, Overview of Boolean functions and simplifications, two, three, four variable K-Maps- Don't care conditions- Both SOP and POS- Universal Gates (NAND & NOR) Implementations. Introduction to HDL.</td></tr>
<tr><td>Module 2</td><td colspan="2">Boolean function simplification</td><td>Application Assignment</td><td colspan="2">Data Analysis task</td><td colspan="2">08 Classes</td></tr>
<tr><td colspan="9">Topics:
Introduction to Combinational circuits, Analysis, Design procedure, Binary Adder and Subtractor, Magnitude comparator, Parity generator and checker, Multiplexers-Demultiplexers, Decoders, Encoders and Priority Encoders, HDL Models of combinational circuits.</td></tr>
<tr><td>Module 3</td><td colspan="2">Combinational Logic circuits:</td><td>Application Assignment</td><td colspan="2">Programming Task & Data Analysis task</td><td colspan="2">08 Classes</td></tr>
<tr><td colspan="9">Topics:
Introduction to sequential circuits, Storage elements: latches and flip flops, Characteristic tables and equations, excitation table, Analysis of clocked sequential circuits, Mealy & Moore Models of finite state machines - Registers & Counters. HDL Models of Sequential circuits.</td></tr>
<tr><td colspan="9">Targeted Application & Tools that can be used:
Digital electronics is the foundation of all modern electronic devices such as cellular phones, MP3 players, laptop computers, digital cameras, high definition televisions, Home Automation, Communication in systems in industries Professionally Used Software: HDL/VHDL/Verilog HDL/ OOPS</td></tr>
<tr><td colspan="9">Text Book(s):</td></tr>
</table>

1. Mano, M. Morris and Ciletti Michael D., *"Digital Design"*, Pearson Education, 6th edition

2. Thomas L. Floyd "DIGITAL LOGIC DESIGN" , Pearson Education, fourth edition.

Reference(s):
Reference Book(s):

R1. Jain, R. P., *"Modern Digital Electronics",* McGraw Hill Education (India), 4th Edition

R2. Roth, Charles H., Jr and Kinney Larry L., *"Fundamentals of logic Design"*, Cengage Learning, 7th Edition

Online Resources (e-books, notes, ppts, video lectures etc.): Book Free Download (studymaterialz.in)

1. eBook1: Mano, M. Morris and Ciletti Michael D., *"Digital Design"*, Pearson Education.
2. {[PDF] Digital Design By M. Morris Mano, Michael D Ciletti Book Free Download }
3. eBook2:Floyd "DIGITAL LOGIC DESIGN" fourth edition- ePub, eBook- [PDF] DIGITAL LOGIC DESIGN FOURTH EDITION FLOYD | abri.engenderhealth.org.
4. NPTEL Course- NPTEL :: Electrical Engineering - NOC:Digital Electronic Circuits
5. Digital Logic Design PPT Slide 1 (iare.ac.in)
6. Lab Tutorial: Multisim Tutorial for Digital Circuits - Bing video

CircuitVerse - Digital Circuit Simulator online

Learn Logisim➠ Beginners Tutorial | Easy Explanation! - Bing video

Digital Design 5: LOGISIM Tutorial & Demo

7. https://presiuniv.knimbus.com/user#/home

E-content:

1. Z. Xin-Li and W. Hong-Ying, "The Application of Digital Electronics in Networking Communication," 2016 Eighth International Conference on Measuring Technology and Mechatronics Automation (ICMTMA), 2016, pp. 684-687, doi: 10.1109/ICMTMA.2016.168.
2. An encoding technique for design and optimization of combinational logic circuit DipayanBhadra;Tanvir Ahmed Tarique;Sultan Uddin Ahmed;Md. Shahjahan;KazuyukiMurase2010 13th International Conference on Computer and Information Technology (ICCIT)
3. A. Matrosova and V. Provkin, "Applying Incompletely Specified Boolean Functions for Patch Circuit Generation," 2021 IEEE East-West Design & Test Symposium (EWDTS), 2021, pp. 1-4, doi: 10.1109/EWDTS52692.2021.9581029.
4. A. Matrosova, V. Provkin and E. Nikolaeva, "Masking Internal Node Faults and Trojan Circuits in Logical Circuits," *2019 IEEE East-West Design & Test Symposium (EWDTS)*, 2019, pp. 1-4, doi: 10.1109/EWDTS.2019.8884434.

Topics relevant to "SKILL DEVELOPMENT": Adders, Multiplexers, Decoders / Encoders; Flip-Flops, Counters and Registers for Skill Development through Experiential Learning techniques. This is attained through assessment component mentioned in course handout.

Course Code: ECE2052	Course Title: Digital Design Lab Type of Course: Theory &Integrated Laboratory	L- T-P- C	0	0	2	1
Version No.	1.0					
Course Pre-requisites	1]Elements of Electronics/Electrical Engineering, 2] Basic concepts of number representation, Boolean Algebra					
Anti-requisites	NIL					
Course Description	The purpose of this course is to enable the students to appreciate the fundamentals of digital logic circuits and Boolean algebra focusing on both combinational and sequential logic circuits. The course emphasizes on minimization techniques for making canonical and low-cost digital circuit implementations. This course deals with analysis and design of digital electronic circuits. The course also creates a foundation for future courses which includes					

	Computer Architecture, Microprocessors, Microcontrollers, and Embedded Systems etc. The course enhances the Design, Implementation and Programming abilities through laboratory tasks. The associated laboratory provides an opportunity to verify the theoretical knowledge.
Course Objective	The objective of the course is to familiarize the learners with the concepts of Digital Design and attain the SKILL DEVELOPMENT through EXPERIENTIAL LEARNING.
Course Outcomes	On successful completion of this course the students shall be able to: i. Implement various universal gates and Boolean functions circuits using logic gates. ii. Implement various combinational and sequential logic circuits using logic gates.
Course Content:	

List of Laboratory Tasks:

Experiment N0 1: Verify the Logic Gates truth table
Level 1: By using Digital Logic Trainer kit
Level 2: By using Analog devices like RPS, Volt meter, Resistors and ICs

Experiment No. 2: Verify the Boolean Function and Rules
Level 1: By using Digital Logic Trainer kit
Level 2: By using Analog devices like RPS, Volt meter, Resistors and ICs

Experiment No. 3: Design and Implementations of HA/FA
Level 1: By using basic logic gates and Trainer Kit
Level 2: By using Universal logic gates and Trainer Kit

Experiment No. 4: Design and Implementations of HS/FS
Level 1: By using basic logic gates and Trainer Kit
Level 2: By using Universal logic gates and Trainer Kit

Experiment No. 5: Design and Implementations of combinational logic circuit for specifications
Level 1: Specifications given in the form of Truth table
Level 2: Specification should be extracted from the given scenario

Experiment No. 6: Study of Flip flops

Experiment No. 7: Design and Implementations of sequential logic circuit for specifications
Level 1: Specifications given in the form of Truth table
Level 2: Specification should be extracted from the given scenario

Experiment No.8: HDL coding for basic combinational logic circuits
Level 1: Gate level Modeling
Level 2: Behavioral Modeling

Experiment No.9: HDL coding for basic sequential logic circuit
Level 1: Gate level Modeling
Level 2: Behavioral Modeling

Targeted Application & Tools that can be used:

Digital electronics is the foundation of all modern electronic devices such as cellular phones, MP3 players, laptop computers, digital cameras, high definition televisions, Home Automation, Communication in systems in industries

Professionally Used Software: HDL/VHDL/Verilog HDL/ OOPS

Text Book(s):

1. Mano, M. Morris and Ciletti Michael D., *"Digital Design"*, Pearson Education, 6th edition

2. Thomas L. Floyd "DIGITAL LOGIC DESIGN" , Pearson Education, fourth edition.

Reference(s):
Reference Book(s):

R1. Jain, R. P., *"Modern Digital Electronics",* McGraw Hill Education (India), 4th Edition

R2. Roth, Charles H., Jr and Kinney Larry L., *"Fundamentals of logic Design"*, Cengage Learning, 7th Edition

Online Resources (e-books, notes, ppts, video lectures etc.): Book Free Download (studymaterialz.in)

1. eBook1: Mano, M. Morris and Ciletti Michael D., *"Digital Design"*, Pearson Education.
2. {[PDF] Digital Design By M. Morris Mano, Michael D Ciletti Book Free Download }
3. eBook2:Floyd "DIGITAL LOGIC DESIGN" fourth edition- ePub, eBook- [PDF] DIGITAL LOGIC DESIGN FOURTH EDITION FLOYD | abri.engenderhealth.org.
4. NPTEL Course- NPTEL :: Electrical Engineering - NOC:Digital Electronic Circuits
5. Digital Logic Design PPT Slide 1 (iare.ac.in)
6. Lab Tutorial: Multisim Tutorial for Digital Circuits - Bing video

CircuitVerse - Digital Circuit Simulator online

Learn Logisim➠ Beginners Tutorial | Easy Explanation! - Bing video

Digital Design 5: LOGISIM Tutorial & Demo

7. https://presiuniv.knimbus.com/user#/home

E-content:

1. Z. Xin-Li and W. Hong-Ying, "The Application of Digital Electronics in Networking Communication," 2016 Eighth International Conference on Measuring Technology and Mechatronics Automation (ICMTMA), 2016, pp. 684-687, doi: 10.1109/ICMTMA.2016.168.
2. An encoding technique for design and optimization of combinational logic circuit DipayanBhadra;Tanvir Ahmed Tarique;Sultan Uddin Ahmed;Md. Shahjahan;KazuyukiMurase2010 13th International Conference on Computer and Information Technology (ICCIT)
3. A. Matrosova and V. Provkin, "Applying Incompletely Specified Boolean Functions for Patch Circuit Generation," 2021 IEEE East-West Design & Test Symposium (EWDTS), 2021, pp. 1-4, doi: 10.1109/EWDTS52692.2021.9581029.
4. A. Matrosova, V. Provkin and E. Nikolaeva, "Masking Internal Node Faults and Trojan Circuits in Logical Circuits," *2019 IEEE East-West Design & Test Symposium (EWDTS)*, 2019, pp. 1-4, doi: 10.1109/EWDTS.2019.8884434.

Topics relevant to "SKILL DEVELOPMENT": Adders, Multiplexers, Decoders / Encoders; Flip-Flops, Counters and Registers for Skill Development through Experiential Learning techniques. This is attained through assessment component mentioned in course handout.

Course Code: EEE1200	**Course Title: Basics of Electrical and Electronics Engineering.** **Type of Course: Professional Core - Theory**	**L-T-P-C**	3	0	0	3
Version No.	2.0					
Course Pre-requisites	**NIL**					
Anti-requisites	**NIL**					
Course Description	This is a fundamental Course which is designed to know the use of basics of electrical and electronics engineering principles occurs in various fields of Engineering. The course emphasises on the characteristics and applications of electrical and electronic devices. The course also emphasizes on the working, analysis and design of electrical circuits using both active & passive components. Additionally, this course creates a foundation for the future courses such as Electrical machines, power system, power electronics Linear Integrated Circuits, Analog Communication and Digital Communication etc.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of Basics of Electrical and Electronics Engineering and attain **Skill Development** through **Participative Learning** techniques.					

Course Outcomes	**On successful completion of this course the students shall be able to:** 1. **Apply** basic laws of Electrical Engineering to compute voltage, currents and other parameters in the circuits. 2. **Discuss** various fundamental parameters appearing in the characteristics of semiconductor devices and their applications. 3. **Summarize** the operations of different biasing configurations of BJTs and amplifiers. 4. **Discuss** the performance characteristics and applications of various electrical Machines.			
Course Content:				
Module 1	**Introduction to Electrical Circuits**	Assignment/ Quiz	Numerical solving Task	**10 Sessions**
DC Circuits: Concept of Circuit and Network, Types of elements, Network Reduction Techniques- Series and parallel connections of resistive networks, Star–to-Delta Transformations, Mesh Analysis, Nodal Analysis, Numerical examples. **AC Circuits:** Fundamentals of single phase circuits - Series RL, RC and R-L-C Circuits, Concept of active power, reactive power and Power factor, Numerical examples. Introduction to three phase system and relation between line and phase values in Star & Delta connection, Numerical examples.				
Module 2	**Semiconductor and Diode applications**	Assignment/ Quiz	Memory Recall based Quizzes	**10 Sessions**
Mass Action Law, Charge densities in a semiconductor, Types of SC, Junction diodes -Ideal and practical behaviour, Modelling the Diode Forward Characteristic, and Diode applications like rectifiers, Clipping and clamping circuits. Zener diode, characteristics and its applications like voltage regulator.				
Module 3	**Transistors and its Applications**	Assignment/ Quiz	Memory Recall-based Quizzes	**10 Sessions**
Transistor characteristics, Current components, BJT Configurations (CB, CC, CE configurations) and their current gains. Operating point, Biasing & stabilization techniques: Fixed Bias, Voltage divider bias and its stability factor and load line analysis. Single and multistage amplifier, Darlington pair. JFET (Construction, principal of Operation and Volt –Ampere characteristics). Pinch- off voltage, Comparison of BJT and FET. MOSFET (Construction, principal of Operation and symbol), MOSFET characteristics in Enhancement and Depletion modes.				
Module 4	**Fundamentals of Electrical Machines**	Assignment/ Quiz	Numerical solving Task	**10 Sessions**
Electrical Machines: Single phase transformers: principle of operation and EMF equation, Numerical examples. DC Motor: principle of operation, Back EMF, torque equation, Numerical examples. AC Motor: Principle operation of Induction Motors and its Applications. **Special Machines:** Introduction to special electrical machines and its applications.				
Targeted Application & Tools that can be used: **Targeted Applications:** Application Area includes all electrical and electronic circuits (power supply unit, regulator unit, embedded devices, hardware electronics etc.). The students will be able to join a profession which involves basics to high level of electronic circuit design. Professionally Used Software: Multisim/ P Spice Besides these software tools hardware equipment such as Multimeters, Function Generators, Power Supplies, Oscilloscopes etc., can be used to perform component/circuit testing and analysis..				
Project Work/ Assignment:				
1. Article review: At the end, of course an article topic will be given to an individual or a group of students. They need to refer the library resources and write a report on their understanding about the assigned article in appropriate format. **2. Presentation:** There will be a group presentation, where the students will be given a topic. They will have to explain/demonstrate the working and discuss the applications for the same. **3. Case Study**: - At the end of the course students will be given a 'real-world' application based circuits like Power Amplifier, Signal/Function Generator etc. as a case study. Students will be submitting a report which will include Circuit Diagrams, Design, Working Mechanism and Results etc. in appropriate format				
Text Book(s): 1. Kothari D. P. & Nagrath I. J., "Basic Electrical and Electronics Engineering", Tata McGraw-Hill 2. Education 3. Theraja B.L. and Theraja A.K., "A Textbook of Electrical Technology: Basic Electrical Engineering" in S.I. System of Units, 23rd ed., New Delhi: S. Chand, 2002. 4. A.P.Malvino, Electronic Principles,7thEdition, Tata McGraw Hill,2007 5. J. Millman, C. C. Halkias and C. D. Parikh, "Millman's Integrated Electronics", McGraw Hill Education, 2nd Edition.				

6. Basics of Electrical & Electronics Laboratory Manual.
Reference Book (s): 1. John Hiley, Keith Brown and Ian McKenzie Smith, "HUGHES Electrical and Electronic Technology", 10th Edition (Indian Edition published by Dorling Kindersley), Pearson,2011 2. Samarajit Ghosh, "Fundamentals of Electrical and Electronics Engineering", 2nd Edition, Prentice Hall India, 2007. 3. K Uma Rao, A Jaya Lakshmi, "Basic Electrical engineering" IK International publishing house Pvt. Ltd 4. R. L. Boylestad and L. Nashelsky, "Electronic Devices and Circuit Theory", Pearson Education India 7th Edition. 5. A K. Maini, V. Agrawal, "Electronic Devices & Circuits", Wiley, 2nd Edition 6. A.S Sedra, K. C. Smith, "Microelectronic Circuits", Oxford University Press, 6th Edition **Online Resources (e-books, notes, ppts, video lectures etc.):** 1. https://presidencyuniversity.linways.com 2. https://www.digimat.in/nptel/courses/video/108105112/L01 "Fundamentals of Electrical Engineering-Basic Concepts, Examples" 3. Seminar Topic: https://nptel.ac.in/courses/108/105/108105153/ "Electrical Measurements" 4. Video lectures on "Electronic Devices" by Prof. Dr. A. N. Chandorkar, IIT Bombay http://www.satishkashyap.com/2013/03/video-lectures-on-electron-devices-by.html 5. Video lectures on "Analog Electronics" by Prof. S.C. Dutta Roy, IIT Delhi https://nptel.ac.in/courses/108/102/108102095/ 6. Video lectures on "Diodes", by Prof. Chitralekha Mahanta, IIT Guwahati, https://nptel.ac.in/courses/117/103/117103063/ **E-content:** 1. "Introduction to Electrical Machines https://nptel.ac.in/courses/108/102/108102146/" M. -Y. Kao, H. Kam and C. Hu, "Deep-Learning-Assisted Physics-Driven MOSFET Current Voltage Modeling," in IEEE Electron Device Letters, vol. 43, no. 6, pp. 974-977, June 2022, doi: 10.1109/LED.2022.3168243 https://ieeexplore-ieee-org-resiuniv.knimbus.com/document/9758727 2. F. Bonet, O. Aviñó-Salvadó, M. Vellvehi, X. Jordà, P. Godignon and X. Perpiñà, "Carrier Concentration Analysis in 1.2 kV SiC Schottky Diodes Under Current Crowding," in IEEE Electron Device Letters, vol. 43, no. 6, pp. 938-941, June 2022, doi: 10.1109/LED.2022.3171112. https://ieeexplore-ieeeorg-presiuniv.knimbus.com/document/9764749 3. M. Chanda, S. Jain, S. De and C. K. Sarkar, "Implementation of Subthreshold Adiabatic Logic for Ultralow-Power Application," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. .23, no. 12, pp. 2782-2790, Dec. 2015. https://ieeexplore.ieee.org/document/7018053 4. R. Raut and O. Ghasemi, "A power efficient wide band trans-impedance amplifier in submicron CMOS integrated circuit technology," 2008 Joint 6th International IEEE Northeast Workshop on Circuits and Systems and TAISA Conference, 2008, pp. 113-116, doi: 0.1109/NEWCAS.2008.4606334. https://ieeexplore.ieee.org/document/4606334
Topics relevant to "SKILL DEVELOPMENT": Performing suitable experiments to compute the electric circuit parameters, performance operation of machines, and operation of semiconductor devices for **Skill Development** through **Participative Learning techniques.** This is attained through assessment component mentioned in course plan.

Course Code EEE1250	**Course Title: Basics of Electrical and Electronics Engineering Laboratory** **Type of Course: Professional Core - Laboratory**	**L-T- P- C**	0	0	2	1
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	NIL					
Course Description	This fundamental laboratory provides an opportunity to validate the concepts taught in the basics of electrical and electronics engineering and enhances the ability to visualize real system performance, using both hardware and simulation tools.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of Basics of Electrical and Electronics Engineering and attain Skill Development through Experiential Learning techniques.					

Basic skill sets required for the laboratory:	
	The students shall be able to develop: 1. **An attitude of enquiry.** 2. **Confidence and ability to tackle new problems.** 3. **Ability to interpret events and results.** 4. **Ability to work as a leader and as a member of team.** 5. Assess errors and eliminate them. 6. **Observe and measure physical phenomenon.** 7. **Write Reports.** 8. **Select suitable equipment, instrument and materials.** 9. Locate faults in systems. 10. Manipulative skills for setting and handling equipment. 11. **The ability to follow standard test procedures.** 12. **An awareness of the need to observe safety precautions.** 13. To judge magnitudes without actual measurement.
Course Out Comes	**On successful completion of the course the students shall be able to:** 1. Apply basic laws of Electrical Engineering to compute voltage, currents, and other parameters in the circuits. 2. Demonstrate the working of electrical machines to observe performance characteristics. 3. Demonstrate the working of electronic circuits to obtain the V-I Characteristics of various semiconductor devices. 4. Sketch the characteristics and waveforms relevant to standard electrical and electronic circuits
Course Content:	
	List of Laboratory Tasks: **Experiment No 1:** Verification of KVL and KCL for a given DC circuit. **Level 1:** Study and Verify KVL and KCL for the given electrical Circuit. **Level 2:** For the same circuit considered in level 1, perform the simulation using NI LabVIEW/Multisim/MATLAB. **Experiment No 2:** Analyse AC series circuits – RL, RC and RLC . **Level 1:** Conduct an experiment to perform and verify the impedance, current and power of Series RL and RC circuits **Level 2:** Conduct an experiment to perform and verify the impedance and current of RLC series circuits. **Experiment No 3:** Calculation of power and power factor of the given AC Circuit. **Level 1:** Conduct an experiment to measure the power and power factor for given resistive load. **Level 2:** Conduct an experiment to measure the power and power factor for given inductive load. **Experiment No 4:** Perform the experiments on given Transformer. **Level 1:** Verify the EMF equation of a transformer and compute the voltage transformation ratio. **Level 2:** Study the effect of load on the secondary side of the transformer and verify the EMF equation under load conditions. **Experiment No 5:** Load test on DC shunt motor **Level 1:** Conduct load test on DC shunt motor and find its efficiency at different loads **Level 2:** Conduct load test on DC shunt motor and plot the performance characteristics. **Experiment 6:** Study of PN-Junction Diode Characteristics in Forward and Reverse Bias Conditions. **Level 1:** Carry out an experiment to plot VI Characteristics and hence find the cut-in voltage on forward characteristics for the Silicon P-N Junction diode. **Level 2:** Carry out an experiment to plot VI Characteristics of Zener diode and hence find the zener voltage on reverse characteristics for the Silicon P-N Junction zener diode.

	Experiment No. 7: To observe the output waveform of half wave and full wave rectifier circuit and compute ripple factor and efficiency **Level 1:** Identify the components required for a rectifier circuit, rig up the circuit, and sketch the output waveforms without filter. **Level 2:** Rig up the rectifier circuit with RC filter, observe the output waveforms, determine the efficiency and ripple factor. **Experiment 8:** To construct clipping and clamping circuits for different reference voltages and to verify the responses. **Level 1:**Identify the components required for building a Clipper / Clamper circuit. Rig up the circuit according to the circuit diagram given and sketch the output waveform. **Level 2:** Given a sinusoidal input of 10 V p-p, implement a positive / negative clipper with output clipped at 2 V. **Experiment 9:** To calculate various parameters of emitter follower circuit using BJT **Level 1:** Identify the components required to implement an emitter follower circuit. Rig up the circuit and observe the variations in output waveform with respect to the variations in input waveform. **Level 2:** Determine the values of Z_{in} input impedance and Z_{out} output impedance for Emitter Follower. **Experiment 10:** To Implement RC Coupled amplifier using a BJT and sketch the frequency response. **Level 1**: Identify the components required to implement an RC coupled amplifier circuit. Rig up the circuit and sketch the frequency response. **Level 2**: From the frequency response curve determine the value of the mid band gain and the bandwidth.

Targeted Application & Tools that can be used:

Targeted Applications: Application Area includes all electrical and electronic circuits (power supply unit, regulator unit, embedded devices, hardware electronics etc.). The students will be able to join a profession which involves basics to high level of electronic circuit design.

Professionally Used Software: Multisim/ P Spice

Besides these software tools hardware equipment such as Multimeters, Function Generators, Power Supplies, Oscilloscopes etc., can be used to perform component/circuit testing and analysis.

Course Material

1. Basics of Electrical and Electronics Engineering Laboratory Manual, Presidency University, Bengaluru.

Text Book:

2. Kothari D. P. & Nagrath I. J., "Basic Electrical and Electronics Engineering", Tata McGraw-Hill

Reference Books:

3. John Hiley, Keith Brown and Ian McKenzie Smith, "HUGHES Electrical and Electronic Technology", 10th Edition (Indian Edition published by Dorling Kindersley), Pearson,2011
4. Samarajit Ghosh, "Fundamentals of Electrical and Electronics Engineering", 2nd Edition, Prentice Hall India, 2007.
5. K Uma Rao, A Jaya Lakshmi, "Basic Electrical engineering" IK International publishing house Pvt. Ltd
6. R. L. Boylestad and L. Nashelsky, "Electronic Devices and Circuit Theory", Pearson Education India 7th Edition.
7. A K. Maini, V. Agrawal, "Electronic Devices & Circuits", Wiley, 2nd Edition
8. A.S Sedra, K. C. Smith, "Microelectronic Circuits", Oxford University Press, 6th Edition

Online Learning Resources:

9. https://presidencyuniversity.linways.com
10. https://www.digimat.in/nptel/courses/video/108105112/L01 "Fundamentals of Electrical Engineering-Basic Concepts, Examples"
11. Video lectures on "Diodes", by Prof. Chitralekha Mahanta, IIT Guwahati, https://nptel.ac.in/courses/117/103/117103063/

Topics relevant to "SKILL DEVELOPMENT": All the experiments which are listed are for **Skill Development** through **Experiential Learning Techniques**. This is attained through the assessment component mentioned in course handout.

Course Code: ECE1511	**Course Title: Design Workshop**	**L- T-P- C**	1	0	2	2
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	NIL					
Course Description	This course is designed to provide an in-depth understanding of Arduino, microcontrollers Raspberry pi and their application in various real time projects involving sensors. Throughout the course, students will learn the fundamentals of Arduino and Raspberry Pi programming and gain hands-on experience with a wide range of sensors. Students will explore how to connect and interface sensors with Arduino and Raspberry Pi boards, read sensor data, and use it to control various output devices This course is suitable for beginners who are interested in exploring the world of electronics and developing practical applications using Arduino, Raspberry Pi and sensors.					
Course Objective	The objective of the course is **Employability Skills** of student by using **PARTICIPATIVE LEARNING** techniques.					
Course Outcomes	**On successful completion of the course the students shall be able to** 1. Explain the main features of the Arduino & the Raspberry Pi prototype board. 2. Demonstrate the hardware interfacing of the peripherals to Arduino and Raspberry Pi system. 3. Understand the types of sensors and its functions 4. Demonstrate the functioning of live projects carried out using Arduino and Raspberry Pi system.					
Course Content:						

Module 1	**Basic concepts of Microcontrollers**	Hands-on	Interfacing Task and Analysis	**3 Sessions**

Topics:
Introduction to Arduino, ESP and Node MCU Pin configuration and architecture, Device and platform features, Concept of digital and analog ports, Familiarizing with Arduino Interfacing Board, API's , Introduction to Embedded C and Arduino platform, Arduino Datatypes and variables, Arduino i/o Functions, Arduino Communications, Arduino IDE, Various Cloud Platforms.

Module 2	**Sensory Devices**	Hands-on	Interfacing Task and Analysis	**3 Sessions**

Arduino Sensors: Humidity Sensor, Temperature Sensor, Water Detector / Sensor, PIR Sensor, Ultrasonic Sensor, Connecting Switches and actuators, sensor interface with Arduino.
Introduction to 3D Printer: 3D Printer technology and its working Principles, Applications.
Introduction to online Simulators: Working with AutoCAD/Fusion 360 Simulator.

Module 3	**Introduction to Micro python**	Hands-on	Interfacing Task and Analysis	**4 Sessions**

Topics:
Introduction to MicroPython, Comparison with other programming languages, Setting up the MicroPython development environment, Basics of MicroPython syntax and structure.

Module 4	Working with Raspberry-pi	Hands-on	Interfacing Task and Analysis	5 Sessions

Introduction to raspberry pi boards, pin-diagram, different types of raspberry pi boards and its application, LED and switch control. Mastering Modules, Setup Raspberry - PuTTY SSH,VNC Viewer to interface with more complicated sensors and actuators. Various Libraries and its functions.

Lab: Name of the Experiments:

1. **Introduction Lab 1:**
   Level 1: Overview on Arduino based Micro-controller, and sensors.
   Level 2: Interfacing of Arduino and ESP boards with sensors and other components.
2. **Lab 2: Smart Plant Monitoring**
   Level 1- Push button-controlled LED.
   Level 2- Automatic Irrigation and monitoring System using Arduino
3. **Lab 3: Robotics with Arduino**.
   Level 1- Servo Motor control using Arduino
   Level 2: DC Motor Control Using Arduino for Robotics.
4. **Lab 4: Environmental pollution using ESP.**
   Level 1 - IoT based air Pollution Monitoring System.
   Level 2- IoT Based water pollution system
5. **Introduction Lab for raspberry pi:**
   Level 1: Overview on Different Raspberry Pi Boards, and sensors.
   Level 2: Configuring the Raspberry Pi and Interfacing with sensors and other components.
6. **Lab 7:** Raspberry Pi based Object Detection using TensorFlow and OpenCV.
7. **Lab 8:** Speech Recognition on Raspberry Pi for Voice Controlled Home Automation.
8. **Lab 9:** Design the website using HTML and CSS, and host the website on Raspberry Pi.
9. **Introduction Lab for 3D printing:**
   Overview of 3D printing. Design of 3D structure using the CAD. Understand the steps of fabrication of simple rectangular box using 3D printer.
10. **Lab 10:** Design and print of Hollow Cylindrical structure using 3D CAD and 3D printer.
11. **Lab 11** Demonstration of Jetson nano board and its capability. **(OPTIONAL)**
12. **Lab 12:** Revision
13. **Lab 13:** Revision
14. **Lab 14:** Mini Project
15. **Lab 15:** Mini Project Evaluation.

Topics: Types of Arduino boards, Thonny Python, Python IDLE, **sensors, 3D Printer**

Targeted Application & Tools that can be used:

Application Area:

Home Automation, Environmental Monitoring, Agriculture and Farming, Industrial Automation, Internet of Things (IoT), Robotics, Wearable Devices, Security Systems, Education and Learning. These are just a few examples of the many application areas where Arduino, Raspberry Pi and sensors can be applied. The flexibility and affordability of Arduino, and Raspberry Pi combined with the wide range of sensors available, allow for endless possibilities in creating innovative projects.

Professionally Used Software: Students can use open SOURCE Software's Arduino IDE and Tincker CAD, Thonny Python, Python IDLE etc.

Project work/Assignment:

1. Projects: At the end of the course students will be completing the project work on solving many real time issues.

2. Book/Article review: At the end of each module a book reference or an article topic will be given to an individual or a group of students. They need to refer the library resources and write a report on their understanding about the assigned article in appropriate format. Presidency University Library Link .

3. Presentation: There will be a presentation from interdisciplinary students group, where the students will be given a project on they have to demonstrate the working and discuss the applications for the same

Textbook(s):

1. **Monk Simon "Programming Arduino: Getting Started with Sketches", Mc Graw Hill Publications Second Edition**
2. **Monk Simon "Raspberry Pi Cookbook: Software and Hardware Problems and Solutions", Publisher(s): O'Reilly Media, Inc. ISBN: 9781098130923 fourth Edition.**

References

Reference Book(s)

1. **Neerparaj Rai "Arduino Projects for Engineers" BPB publishers,first edition, 2016.**
2. Ryan Turner "Arduino Programming " Nelly B.L. International Consulting Ltd. first edition,2019.
3. **Charles Bell Micro Python for the Internet of Things: A Beginner's Guide to Programming with Python on Microcontrollers" by" Edition 1, 2017, ISBN 978-1-4842-3123-4**
4. Stewart Watkiss "Learn Electronics with Raspberry Pi " Apress Berkeley, CA . second edition,2020. ISBN978-1-4842-6348-8
5. Jo Prusa, "Basic of 3D printing", Prusa Research, 3rd edition.
6. Volker Ziemann, "**A Hands-On Course in Sensors Using the Arduino and Raspberry Pi (Series in Sensors)**", CRC Press, 1st Edition. 2018.

Online Resources (e-books, notes, ppts, video lectures etc.):

1. Arduino trending Projects < https://www. https://projecthub.arduino.cc/>
2. Introduction to Arduino < https://onlinecourses.swayam2.ac.in/aic20_sp04/preview>
3. Case studies on Wearable technology< https://www.hticiitm.org/wearables>
4. Raspberry-pi Projects < https://magpi.raspberrypi.com/articles/category/tutorials/>
5. Introduction to internet of things< https://nptel.ac.in/courses/106105166>

E-content:

1. Cattle Health Monitoring System Using Arduino and IOT (April 2021 | IJIRT | Volume 7 Issue 11 | ISSN: 2349-6002)

2. M H Hemanth Kumar, Ravi Pratap Singh, Nishu Sharma, Pragya Singh" IOT BASED SMART SECURITY SYSTEM USING ARDUINO" 2021 JETIR August 2021, Volume 8, Issue 8.

3. R. Maheswar, P. Jayarajan, S. Vimalraj, G. Sivagnanam, V. Sivasankaran and I. S. Amiri, "Energy Efficient Real Time Environmental Monitoring System Using Buffer Management Protocol," 2018, pp. 1-5, doi: 10.1109/ICCCNT.2018.8494144. **https://ieeexplore.ieee.org/document/8494144**.

4. Yaser S Shaheen,Hussam., " Arduino Mega Based Smart Traffic Control System ," December 2021 Asian Journal of Advanced Research and Reports 15(12): 43-52, 2021(15(12): 43-52, 2021):15(12): 43-52, 2021.

5. Basil, Eliza Sawant, S.D. "IoT based traffic light control system using Raspberry Pi " DOI 10.1109/ICECDS.2017.8389604

6. Supriya S, 2Dr. Aravinda " Green leaf disease detection and identification using Raspberry Pi https://www.irjet.net/archives/V9/i8/IRJET-V9I847.
7. Dr. E.N. Ganesh., "Health Monitoring System using Raspberry Pi and IOT" DOI : http://dx.doi.org/10.13005/ojcst12.01.03

Topics relevant to development of "SKILL": System design for achieving Sustainable Development Goals.

Course Code: **CSE2274**	**Course Title:** Competitive Programming and Problem Solving **Type of Course: Program Core**	L-T-P-C	0	0	4	2
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	**NIL**					
Course Description	The **Competitive Programming and Problem Solving** course equips students with efficient problem-solving skills for coding competitions and real-world challenges. Starting with brute-force solutions, students learn to optimize time and space complexity using advanced techniques like dynamic programming, greedy algorithms, and backtracking. Hands-on practice on platforms like CodeChef and Codeforces helps tackle problems involving number theory, data structures, and algorithmic paradigms. By understanding CP constraints and fostering a strategic mindset, students gain the confidence to excel in competitions, technical interviews, and practical applications.					
Course Out Comes	On successful completion of the course the students shall be able to: CO1 : Understanding the issues of online platforms and Competitive Programming (CP) and developing brute force coding for commonly asked CP problems. CO2 : Analyzing the space and time complexity of brute force solutions and designing efficient solutions. CO3 : Evaluating the applicability of suitable algorithmic approaches to solve relevant CP problems. CO4: Creating efficient solutions of CP problems using the learnt algorithmic approaches.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Competitive Programming and Problem Solving** and attain **Skill Development** through **Experiential Learning** techniques.					

Module 1: Introduction to Competitive Programming
Overview of Efficient Coding for Problem Solving and CP: Introduction to competitive programming (CP); revisit of complexity analysis; introduction to online platforms such as codechef, codeforces etc and online submission; constraints during CP, online testing process and common errors such as TLE; use of STL

Module 2: Number Theory for Problem-Solving
Use of Number Theory for problem-solving: reducing time/space complexity of brute force[Text Wrapping Break]coding solution of Sieve Method, Inverse Module, Euclidian Method of factorization; efficient coding[Text Wrapping Break]for Permutation Combination; XORing based and pattern-based solutions.

Module 3: Optimizing Time & Space Using Sequential Storage
Coding for Optimizing time and Space using Sequential Storage: two pointer approach;[Text Wrapping Break]problem-solving using arrays and strings such as rotation on sorted arrays, duplicate removal, string[Text Wrapping Break]matching algorithms; Kadane's algo, stacks, priority-queues and hashing based efficient coding;[Text Wrapping Break]median based problems and alternate solutions.

Module 4: Non-Linear Data Structures
Applying Non-Linear Data Structures for real-life problems: design of efficient solutions for[Text Wrapping Break]problems such as finding loops in a linked list, memory efficient DLL, block reversal in LL; problem[Text Wrapping Break]solving using trees and binary trees, Catalan numbers, applications of graphs, spanning tree and path[Text Wrapping Break]algos for CP problems with reduced time/space complexity.

Module 5: Problem Solving using Advanced Topics

CP Problem Solving using Advanced Topics: concept of disjoint sets and their efficient[Text Wrapping Break]representation, algorithmic approaches such as Greedy, Backtracking, Dynamic Programming and[Text Wrapping Break]applying them for CP problems using bottom-up dynamic programming.

List of Laboratory Tasks:

1. You are given the finishing times of 'N' runners in a marathon. Write a program to find the runner who finished in the third position. **Focus:** Basic data structures (arrays), sorting algorithms (e.g., insertion sort, selection sort), and basic input/output.
2. In the same marathon, you are given the finishing times of 'N' runners and their bib numbers. Write a program to efficiently find the top 10 runners and their corresponding bib numbers. **Focus:** Efficient sorting algorithms (e.g., merge sort, quick sort), data structures like priority queues, and optimizing for large datasets.
3. A library maintains a list of books with their unique IDs. Write a program to check if a given book ID is present in the library. Focus: Searching algorithms (linear search), basic data structures (arrays or lists).
4. The library wants to implement a system to quickly find books by their titles. Suggest an efficient data structure (e.g., a hash table or a trie) and explain how to implement it to achieve fast book lookups. Focus: Understanding the trade-offs between different data structures, choosing the most appropriate data structure for a specific problem, and implementing efficient search operations.
5. An online store sells products with different prices. Write a program to calculate the total cost of a given list of products. **Focus:** Basic arithmetic operations, working with arrays or lists to store product prices.
6. The online store offers discounts based on the total purchase amount. Design an algorithm to efficiently calculate the final cost of an order, considering different discount rules (e.g., percentage discounts, fixed amount discounts, tiered discounts). Focus: Algorithmic design, conditional statements, handling complex scenarios with multiple rules, and potentially using dynamic programming techniques for optimization.
7. You are given two integers, 'a' and 'm'. Calculate 'a' raised to the power 'm' modulo a large prime number 'p'. **Focus:** Basic modular arithmetic operations (modular exponentiation), understanding the modulo operator.
8. In a secure communication system, you need to efficiently compute the modular exponentiation for very large values of 'm'. Implement and analyze the efficiency of the binary exponentiation algorithm for this task. **Focus:** Efficient algorithms for modular exponentiation (binary exponentiation), time complexity analysis, and understanding the importance of efficient algorithms in cryptography.
9. You have a deck of 'N' cards. Calculate the total number of possible hands of size 'K' that can be drawn from the deck. **Focus:** Basic combinatorics (combinations), factorial calculations.
10. In a card game, you need to calculate the probability of drawing certain combinations of cards (e.g., a pair, a three-of-a-kind) from a shuffled deck. Design an efficient algorithm to calculate these probabilities. **Focus:** Advanced combinatorics (permutations and combinations with repetitions), probability calculations, and optimizing calculations to avoid overflows.
11. You are given a network of devices represented as a graph. Determine if there is a path between two given devices in the network. **Focus:** Graph traversal algorithms (depth-first search or breadth-first search).
12. In a secure network, you need to detect and isolate compromised devices. Design an algorithm that efficiently identifies devices that exhibit anomalous behavior (e.g., unusual traffic patterns) using XOR-based techniques for data comparison and pattern matching. **Focus:** Applying XOR operations for data comparison and pattern recognition, understanding the properties of XOR (e.g., commutative, associative), and designing algorithms for network anomaly detection.
13. You are given an array representing the speeds of cars on a highway. Find the minimum time required for all cars to pass a certain point. **Focus:** Basic array traversal, finding the minimum element in an array.
14. In a more realistic scenario, cars have different lengths. Implement a two-pointer approach to simulate the movement of cars and determine the minimum time for all cars to pass a given point. **Focus:** Two-pointer technique, simulating real-world scenarios with arrays, optimizing time complexity.
15. Given a string, find the number of occurrences of a specific substring within the string. **Focus:** Basic string manipulation, string matching (brute-force approach).

16. Implement the KMP (Knuth-Morris-Pratt) string matching algorithm to efficiently find all occurrences of a given pattern within a large text document. **Focus:** Advanced string matching algorithms, understanding the concept of the "next" array in KMP, optimizing for large input sizes.
17. An online auction platform receives bids for different items. Implement a data structure (e.g., a priority queue) to efficiently track the highest bid for each item. **Focus:** Priority queues, insertion and extraction operations on priority queues, basic implementation of a priority queue using an array or a suitable library.
18. The auction platform needs to handle a large number of bids concurrently. Design and implement a system that efficiently processes bids, updates the highest bid for each item, and handles potential race conditions. **Focus:** Concurrent data structures and algorithms, thread safety, handling race conditions, optimizing for high-throughput scenarios.
19. A social network can be represented as a graph where users are nodes, and connections between users are edges. Write an algorithm to find if two given users are connected in the network. **Focus:** Graph traversal algorithms (depth-first search or breadth-first search), basic graph representation (adjacency list or adjacency matrix).
20. In a large social network, efficiently finding the shortest path between two users is crucial. Implement Dijkstra's algorithm to find the shortest paths between users in the network, considering edge weights (e.g., representing the strength of connections). **Focus:** Shortest path algorithms (Dijkstra's algorithm), graph algorithms with weighted edges, optimizing for large graphs.
21. A file system can be modeled as a tree structure. Implement a function to traverse the file system and print the names of all files and directories. **Focus:** Tree traversal algorithms (depth-first search or breadth-first search), basic tree representation (using nodes and pointers).
22. Design and implement a file system that supports efficient operations like creating directories, deleting files, and finding files based on their names or paths. Consider using a combination of tree structures and hash tables for efficient indexing and searching. **Focus:** Designing and implementing file system structures, using multiple data structures together, optimizing for common file system operations.
23. An online shopping cart can be represented as a tree, where each node represents an item or a category of items. Write an algorithm to calculate the total price of all items in the shopping cart. **Focus:** Tree traversal, calculating sums within a tree structure.
24. Implement a system that allows customers to apply discounts and coupons to their shopping carts. Consider using a combination of trees and other data structures (e.g., hash tables) to efficiently apply discounts and calculate the final price. **Focus:** Applying discounts and promotions to tree-like structures, efficient implementation of discount rules, optimizing for complex pricing scenarios.
25. In a social network, users can form groups. Given a list of friendships, determine if all users in a specific group are connected (directly or indirectly) through friendships. **Focus:** Disjoint set union (DSU) data structure, basic connectivity checks.
26. Design an efficient algorithm to find the minimum number of new friendships needed to connect all users in the social network into a single, connected component. **Focus:** Applying DSU for finding connected components, greedy algorithms, optimization for minimizing connections.
27. A treasure hunt involves a series of clues leading to the final treasure. Given a list of possible paths and their associated costs, find the cheapest path to reach the treasure. **Focus:** Greedy algorithms (e.g., Dijkstra's algorithm for shortest paths), basic graph representation.
28. In a more complex treasure hunt, there are time constraints associated with each path. Design an algorithm to find the fastest path to the treasure while considering both path costs and time constraints.
    Focus: Combining greedy approaches with other techniques (e.g., priority queues), handling multiple constraints, optimizing for time-critical scenarios.
29. In a simplified chess game with only rooks, determine the minimum number of moves required for a rook to reach a specific target square on an empty board. **Focus:** Breadth-first search (BFS) on a graph (the chessboard), basic graph traversal.
30. In a more realistic chess game with multiple pieces and obstacles, implement a minimax algorithm with alpha-beta pruning to determine the best move for a player. **Focus:** Game tree search, minimax algorithm, optimization techniques like alpha-beta pruning, handling complex game states.

Targeted Application & Tools that can be used:

1. C or C++ Compiler (g++): The standard compiler for CP. Familiarize students with compilation flags (e.g., -O2 for optimization).
2. IDE (Integrated Development Environment): Code:: Blocks, Visual Studio, CLion, or similar IDEs. These provide debugging capabilities, code completion, and other helpful features.
3. Online Judges (CodeChef, Codeforces, LeetCode, HackerRank): Essential for practicing and submitting solutions.
4. Debugger (gdb): Crucial for understanding code execution and finding bugs. Origin, excel and Mat lab soft wares for programming and data analysis.
5. Number Theory Libraries: Some libraries provide pre-built functions for number theory operations (though often it's better to implement them yourself for learning).
6. Wolfram Alpha: A useful tool for verifying number theory calculations and exploring concepts.
7. **String Libraries:** Familiarize students with the string manipulation functions available in C++.
8. **Graph Visualization Tools:** Tools like Graphviz can be helpful for visualizing graphs and understanding graph algorithms.
9. **DP Debugging Techniques:** Practice debugging DP solutions, as they can be complex. Visualizing the DP table can be helpful.

Text Books:

1. **Guide to Competitive Programming: Learning and Improving Algorithms Through Contests" (3rd Edition),** ***Antti Laaksonen, springer, 2024***
2. **"Data Structures and Algorithms in Java: A Project-Based Approach" –** ***Dan S. Myers, Cambridge University Press***

Reference Books:

1. **Data Structures and Algorithmic Thinking with Python/C++/Java",** ***Narasimha Karumanchi, 5th Edition, Career Monk, 2017.***
2. **Introduction to Algorithms, Thomas H. Cormen (Author), Charles E. Leiserson (Author), Ronald L. Rivest , fourth edition April 2022**

Web Resources

1. https://nptel.ac.in/courses/106106231
2.

Project work/Assignment: Mention the Type of Project /Assignment proposed for this course

Assessment Type

- Midterm exam
- Assignment (review of digital/ e-resource from PU link given in references section - mandatory to submit screen shot accessing digital resource.)
- Quiz
- End Term Exam
- Self-Learning

Course Code: CIV1200	**Course Title:** Foundations of Integrated Engineering **Type of Course:** Theory Only	- T-P- C	2	0	0	2
Version No.	1.0					
Course Pre-requisites	**NIL**					
Anti-requisites	**NIL**					
Course Description	This interdisciplinary course introduces first-year engineering students to foundational principles and practices across key engineering domains, emphasizing real-world problem-solving, sustainability, and ethical innovation. Students explore how civil, mechanical, electrical, and IT systems intersect with emerging technologies like IoT, AI, and geomatics to address global challenges. Through case studies, learners gain deeper understanding of smart infrastructure, prototyping mechanical/electronic systems, and securing IT solutions. Topics include bioinformatics for environmental monitoring, GIS-enabled urban					

	planning, renewable energy integration, and cybersecurity fundamentals. The course cultivates a holistic understanding of engineering's role in sustainable development, safety, and ethical decision-making, preparing students to contribute meaningfully to multidisciplinary projects in a technology-driven world.			
Course Objective	The objective of the course is skill development of student by using Participative Learning techniques.			
Course Outcomes	On successful completion of this course the students shall be able to: 1] Recall key principles of Agile, DevOps, and bioinformatics used in interdisciplinary engineering contexts. 2] Explain the role of GIS, LiDAR, and sustainable materials in designing smart infrastructure and disaster management systems. 3] Describe core components of mechanical systems and their real-world applications. 4] Describe the functionality of IoT-enabled wearable devices, embedded systems, and renewable energy integration in smart grids. 5] List foundational IT concepts such as cloud computing architectures, cybersecurity threats, and blockchain applications.			
Course Content:				
Module 1	**Foundations of Engineering Practice**	Assignment	Case studies	**6 Sessions**
Real-world problem-solving using data logic and practical applications, Collaboration and Innovation through multi-domain project, Engineering Ethics & Environmental Impact Emerging Fields: Automation, and Introduction to bioinformatics and its application Sustainability & Safety: Circular economy principles, carbon footprint analysis.				
Module 2	**Civil Engineering & Geomatics**	Assignment	Article Review	**6 Sessions**
Smart Infrastructure & Geomatics: GIS mapping, LiDAR, drone surveys for urban planning, Geospatial data analysis for disaster management. Sustainable Construction: 3D-printed structures, self-healing concrete, Digital twins for infrastructure monitoring. Green Innovations: Net-zero energy buildings, rainwater harvesting systems.				
Module 3	**Mechanical Engineering in Action**	Assignment & Quiz	Data Collection	**6 Sessions**
Advanced Manufacturing: Collaborative robots (cobots), additive manufacturing and 3D printing, Reverse engineering and prototyping. Energy Systems: Solar/wind energy harvesting, piezoelectric applications. Biomechanics: Prosthetics design, ergonomic product lifecycle.				
Module 4	**Electrical & Electronics Engineering**	Assignment & Quiz	Data Collection and visualization	**6 Sessions**
Smart Devices & Systems: Embedded systems, Wearable technology, Edge computing and hardware platforms Energy Innovations: EV charging infrastructure, wireless power transfer, Smart grid integration with renewables.				
Module 5	**Fundamentals of IT**	Assignment & Quiz	Case studies	**6 Sessions**
Core IT Topics: Networking basics, Cloud computing Cybersecurity & Data: Encryption, phishing prevention, zero-trust models, Database management. Emerging Tech: Blockchain for supply chains, AI/ML basics, IoT integration with cloud platforms				

Targeted Application & Tools that can be used:

Application Areas include Interdisciplinary problem-solving, Smart city planning, disaster management, Robotics prototyping, renewable energy systems, Wearable health tech, smart grids, Secure cloud systems.

Tools: 3D Printers, Autocad, Tinkercad, ArcGIS / QGIS, Arduino/Raspberry Pi

Text Book:

1. William Oakes & Les Leone, "Engineering Your Future: An Introduction to Engineering", Oxford University Press, 9th Edition, 2021
2. Barry F. Kavanagh, “Introduction to Geomatics”, Pearson, 5th Edition, 2021
3. Ian Gibson, David Rosen, & Brent Stucker, “Additive Manufacturing Technologies”, Springer, 3rd Edition, 2021
4. Sudip Misra, "The Internet of Things: Enabling Technologies, Protocols, and Use Cases", Wiley, 2nd Edition, 2022
5. James Kurose & Keith Ross, “Computer Networking: A Top-Down Approach”, Pearson, 8th Edition, 2020

References

1. Supratim Choudhuri, “Bioinformatics for Beginners: Genes, Genomes, and Molecular Evolution", Academic Press, 1st Edition, 2023,
2. Robert McGinn, "The Ethical Engineer: Contemporary Concepts and Cases", Princeton University Press, 1st Edition, 2020
3. Charles J. Kibert, “Sustainable Construction: Green Building Design and Delivery”, Wiley, 5th Edition, 2022
4. Anthony M. Townsend, "Smart Cities: Big Data, Civic Hackers, and the Quest for a New Utopia", W.W. Norton & Company, 1st Edition, 2020
5. David Buchla, "Renewable Energy Systems: A Smart Energy Systems Approach", Pearson, 2nd Edition, 2023
6. Charles Platt, "Make: Electronics: Learning Through Discovery", Make Community, 3rd Edition, 2021
7. Charles J. Brooks, Christopher Grow, & Philip Craig, “Cybersecurity Essentials", Wiley, 2nd Edition, 2021

Web-resources:

1. Post-parametric Automation in Design and Construction
https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=1155197&site=ehost-live
2. Smart Cities : Introducing Digital Innovation to Cities
https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=1993146&site=ehost-live
3. Innovation Energy: Trends and Perspectives or Challenges of Energy Innovation
https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=2323766&site=ehost-live
4. Additive Manufacturing: Opportunities, Challenges, Implications
https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=1134464&site=ehost-live

Course Code: MAT2301	**Course Title: Calculus and Differential Equations Type of Course:BSC**		**L-T- P- C**	**3**	**1**	**0**	**4**
Version No.		1.0				
Course Pre-requisites		Basic Concepts of Limits, Differentiation, Integration (PU level)				
Anti-requisites		**NIL**				
Course Description		Calculus and differential equations are used ubiquitously throughout mathematics, statistics and operations research. In this course, students can be able to build upon the foundations of calculus established to greatly enhance their repertoire of theory and practice in these areas. The application of calculus and differential equations in the description and modelling of real-world problems will also be considered. This unit will extend the problem-solving skills, range of knowledge and use of techniques in differential and integral calculus. The course focuses on the concepts of Calculus and Differential Equations with reference to specific engineering problems. The course is of both conceptual and analytical type in nature.				
Course Objective		The goal of the course Calculus and Differential Equations is to facilitate the students with a concrete foundation of differential calculus and to solve the first and higher-order ordinary differential equations enabling them to acquire the knowledge of these mathematical tools.				
Course Out Comes		On successful completion of the course the students shall be able to: 1) Apply the knowledge of calculus to solve problems related to polar curves and its applications in determining the bentness of a curve. 2) Apply the principles of integral calculus to evaluate integrals. 3) Learn the notion of partial differentiation to calculate rate of change of multivariate functions and solve problems related to composite functions and Jacobian. 4) Solve first-order linear/nonlinear ordinary differential equations analytically using standard methods.				
Course Content:						
Module 1	**Differential Calculus**					**(10 Classes)**
Polar Coordinates, polar curves, angle between radius vector and the tangent, angle between two curves, pedal equations, curvature and radius of curvature. Limit, continuity and partial derivatives, directional derivatives, total derivative; Tangent plane and normal line; Maxima, minima and saddle points; Method of Lagrange multipliers; Gradient, curl and divergence.						
Module 2	**Integral Calculus**		**Assignment**			**(10 Classes)**
Evolutes and involutes; Evaluation of definite and improper integrals; Beta and Gamma functions and their properties; Applications of definite integrals to evaluate surface areas and volumes of revolutions. Rolle's theorem, Mean value theorems, Taylor's and Maclaurin theorems with remainders; Indeterminate forms and L'Hospital's rule; Maxima and minima.						
Module 3	**Multivariable Calculus**					**(10 lectures)**
Multiple Integration: Double integrals (Cartesian), change of order of integration in double integrals, Change of variables (Cartesian to polar), Applications: areas and volumes, Center of mass and Gravity (constant and variable densities); Triple integrals (Cartesian), orthogonal curvilinear coordinates, Simple applications involving cubes, sphere and rectangular parallelepipeds; Scalar line integrals, vector line integrals, scalar surface integrals, vector surface integrals, Theorems of Green, Gauss and Stokes.						
Module 4	**Differential Equations**		**Assignment**			**(15 lectures)**
Definition, types of Differential Equations, Applications, Variable Separable, Homogeneous, Exact, linear and Bernoulli's equations, Euler's equations, Equations not of first degree: equations solvable for p, equations solvable for y, equations solvable for x and Clairaut's type. Linear differential equations of second and higher order with constant coefficients - Non-Homogeneous term of the						

type Q(x) = e^{ax}, Sin ax, Cosax, $e^{ax}v(x)$, $x^n v(x)$ - Method of variation of parameters.

Targeted Application & Tools that can be used:

Differential calculus is used extensively in science and engineering. It can solve problems related to motion, velocity, acceleration, angles of incline or curve on a surface, etc.

Differential Equations are used to model the behavior of electromagnetic fields, including in the design of antennas, microwave ovens, and other devices. Biology: PDEs are used to model biological processes, such as the spread of diseases and the development of biological tissues.

Tools Used: Python.

Assignment:

1. Select any one simple differential equation pertaining to the respective branch of engineering, identify the dependent and independent variable – Obtain the solution and compare the solution sets by varying the values of the dependent variable.

Text Book

1. Erwin Kreyzig, Advanced Engineering Mathematics, John Wiley and sons, Inc.10th Edition
2. B. S. Grewal (2017), Higher Engineering Mathematics by, 44th Edition, Khanna Publishers.

References:

1. Victor Henner, Tatyana Belozerova, Mickhail Khenner, Ordinary and Partial Differential Equations, CRC Press, Edition, 2013.
2. Walter Ledermann, Multiple integrals, Springer, 1st edition
3. G.B. Thomas and R.L. Finney, Calculus and Analytic geometry, 9th Edition, Pearson, Reprint, 2002.
4. Veerarajan T., Engineering Mathematics for first year, Tata McGraw-Hill, New Delhi, 2008.

E-resources/ Web links:

1. https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_103205
2. https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_106839
3. https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_61605
4. https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_134719
5. https://www.math.hkust.edu.hk/~maqian/ma006_0607F.html
6. https://www.scu.edu.au/study-at-scu/units/math1005/2022/

Topics relevant to SKILL DEVELOPMENT: The course focuses on the concepts of calculus and differential equation with reference to specific engineering problems. The course is of both conceptual and analytical type in nature through Problem solving**.** This is attained through the assessment component mentioned in course handout.

	Points, Straight Lines and Plane Surfaces			
Topics: **Introduction, Definitions – Elements of projection and methods of projection, Planes of projection, reference line and conventions adopted. First angle and third angle projections. Projection of Points in all 4 quadrants.** **Projections of Straight Lines (located in first quadrant/first angle projection only): True and apparent lengths, true and apparent Inclinations to reference planes. (No application problems). Projection of Plane surfaces (First angle projection): Regular plane surfaces – triangle, square, rectangle, pentagon, hexagon and circle – in different positions inclined to both the planes using change of position method only.** **[10 Hours: Application Level]**				
Module 3	Orthographic Projections of Solids	Assignment	Multi-view drawing Analysis	**10 Sessions**
Topics: **Introduction, Projection of right regular prisms, pyramids, cone, hexahedron and tetrahedron in different positions (Problems resting on HP only and First angle projection).** **[10 Hours: Application Level]**				
Module 4	Isometric Projections of Solids (Using isometric scale only)	Assignment	Spatial Visualization	**8 Sessions**
Topics: **Introduction, Isometric scale, Isometric projections of right regular prisms, cylinders, pyramids, cones and their frustums, spheres and hemispheres, hexahedron (cube), and combination of 2 solids, conversion of orthographic view to isometric projection of simple objects.** **[8 Hours: Application Level]**				
Text Book: **1.N. D. Bhatt, "Engineering Drawing: Plane and Solid Geometry," Charotar Publishing House Pvt. Ltd.**				
References: **1. K.R. Gopalakrishna, "Engineering Graphics", Subhash Publishers, Bangalore.** **2. D. M. Kulkarni, A. P. Rastogi, A. K. Sarkar, "Engineering Graphics with AutoCAD," Prentice Hall.** **3. D. A. Jolhe, "Engineering Drawing with Introduction to AutoCAD," Tata McGraw Hill.** **Web resources:** **https://nptel.ac.in/courses/112103019**				
Topics relevant to "SKILL DEVELOPMENT": Projection in first and third angle for **SKILL DEVELOPMENT** through **Problem Solving methodologies**. This is attained through the assessment component mentioned in the course handout.				

ENG1900	**English for Technical Communication**	**L- T- P- C**	2	0	0	2
Version No.	1.0					
Course Pre-requisites						
Anti-requisites	**NIL**					
Course Description	This course enhances the technical communication skills of BTech students, focusing on clarity, precision, and conciseness in academic and professional settings. Students will learn to differentiate between general and technical communication, analyze technical content, develop structured writing skills, and deliver effective presentations. Through interactive activities such as TED Talk analyses, report writing, and presentation practice, the course provides hands-on experience for real-world applications. By the end, students will be equipped to communicate complex technical information effectively in various professional contexts.					
Course Outcomes	On successful completion of the course the students shall be able to: 1. Differentiate between general and technical communication. 2. Explain key reading comprehension techniques to enhance understanding of technical texts. 3. Write clear, concise, and well-structured technical reports and documents. 4. Deliver technical presentations and implement peer feedback for continuous improvement. 5. Explain ethical practices in digital communication for professional use.					

Course Content: Theory

Module 1	**Technical communication**	**Quiz**	**Listening**	**9 Hours**

Introduction to Communication

Technical vs. General Communication

Characteristics of technical communication

Importance of clarity, precision, and objectivity

Activity:

- Watching TED Talks/videos to identify differences in technical and general vocabulary

Module 2	Technical Reading	Assignment	Reading	12 Hours

Reading Comprehension

Note making & Notetaking

Content Analysis

Activity:

- Reading technical articles and answering comprehension questions
- Note making techniques

Module 3	Technical Writing	Assignment	Writing	12hours

Paragraph Writing

Structure of a paragraph (topic sentence, supporting details, coherence)

Report Writing

Structure of technical and project reports (Introduction, Methods, Results, Discussion)

Activity:

- Writing a structured paragraph on a technical topic
- Writing project reports

Module 4	Professional Presentation	Presentation	Speaking	12Hours

Introduction to Presentation Skills

Preparing a Presentation

- Structuring content (Introduction, Body, Conclusion)
- Designing effective slides (Text. visual aids, readability, and impact)

Delivering a Presentation

- Engagement techniques, Storytelling, narration, pitching ideas handling Q&A
- Conviction, commitment, generating interest through enthusiasm

Demonstration & Practice

- Giving presentations on topics based on their academic interest
- Evaluating and providing peer feedback

Activity:

- Analyze a real-world engineering issue and present solutions using a structured approach.

Targeted Application & Tools that can be used: YouTube, Instagram, Quill Bot, Grammarly, & Padlet.

References:

Text books:

1. Gupta, R.C. *Technical Communication.* 2nd ed., Cambridge University Press, 2021.
2. Lannon, John M., and Laura J. Gurak. *Technical Communication.* 15th ed., Pearson, 2022.

Reference Books:

1. Gerson, Sharon J., and Steven M. Gerson. Technical Communication: Process and Product. 9th ed., Pearson, 2020.
2. Lannon, John M., and Laura J. Gurak. Technical Communication. 15th ed., Pearson, 2022.
3. Markel, Mike, and Stuart A. Selber. Technical Communication. 13th ed., Bedford/St. Martin's, 2020.

Web Resources:

1. https://owl.purdue.edu/owl/subject_specific_writing/technical_writing.
2. https://journals.ieeeauthorcenter.ieee.org/.
3. https://www.stc.org/.
4. https://ocw.mit.edu/.https://www.ted.com/talks.
   https://owl.purdue.edu/owl/subject_specific_writing/professional_technical_writing/digital_writing.html

Topics Relevant to "employability": Teamwork and Collaboration, Critical Thinking and Problem- Solving

Topics Relevant to "Human Values and Professional Ethics": Critical reasoning, Inclusivity and Fairness

Course Code: **PPS1025**	**Course Title: Industry Readiness Program – I (Audited Course)** Type of Course: Practical Only Course	L- T - P- C	0	0	2	0

Version No.	1.0
Course Pre-requisites	• Students are expected to understand Basic English. • Students should have desire and enthusiasm to involve, participate and learn.
Anti-requisites	NIL
Course Description	This course is designed to enable students to set SMART goals, form professional & personal ethics for success and learn various email writing techniques. The course will benefit learners in presenting themselves effectively through various activities and learning methodologies.
Course Objective	The objective of the course is to **familiarize the learners with the concepts of "Employability for Young Professionals" and attain SKILL DEVELOPMENT through PARTICIPATIVE LEARNING techniques.**
Course Out Comes	**On successful completion of this course the students shall be able to:** CO 1 **Define** their career goals CO 2 **Practice ethical habits** for better career success CO3 **Demonstrate** effective email writing techniques
Course Content	

Module 1	**Goal Setting & Grooming**	Classroom activities	**10 Hours**
Topics: SMART Goals, formal grooming through self-introduction activity **Activity:** Real world scenarios			

Module 2	**Habit Formation**	Role plays	**10 Hours**
Topics: Professional and Personal ethics for success and activity-based practice **Activity:** Students to present 2 min video on building professional ethics			
Module 3	**Email Etiquettes**	Individual and group presentation	**10 Hours**
Topics: Types of prompts to generate effective or desired results for email etiquettes **Activity:** Individual student presenting various search prompts			
Faculty: L&D			
Targeted Application & Tools that can be used: 1. TED Talks 2. You Tube Links 3. Activities			
Assignment proposed for this course Assignment 1: SMART Goal Assignment 2: AI tools for prompt search			
Continuous Individual Assessment Module 1: Presentation Module 2: Activity based assessment Module 3: Class assessment			

The topics related to skill development:

Students acquire knowledge on SMART goals, implement grooming standards, practice ethical behavior in class and campus, acquire hands-on experience to use AI tools to get search prompts for desired email etiquettes.

Course Code: CHE7601	**Environmental Studies** **Type of Course: MOOC course**	**- P- C**				
		ntact hours				
Course Pre-requisites	NIL					
Anti-requisites	**NIL**					
Course Description	This course is designed to improve the learners' SKILL DEVELOPMENT by using PATICIPATIVE LEARNING techniques. This course aims to familiarize students with fundamental environmental concepts and their relevance to business operations, preparing them to address forthcoming sustainability challenges. It is designed to equip students with the knowledge and skills needed to make decisions that account for environmental consequences, fostering environmentally sensitive and responsible future managers. **This course is designed to cater to Environment and Sustainability**					
Course Objective	The objective of the course is **'SKILL DEVELOPMENT'** of the student by using **'PARTICIPATIVE LEARNING'** techniques					
Course Outcomes	On successful completion of this course the students shall be able to: 1. Describe the issues related to natural resources, ecosystems and biodiversity 2. Identify environmental hazards affecting air, water and soil quality 3. Recognize the importance of healthy environment and finding the sustainable methods to protect the environment 4. Convert skills to address immediate environmental concerns through changes in environmental processes, policies, and decisions					
Course Content:						
Module 1	**Understanding Environment, Natural Resources, and Sustainability**					
Topics: Classification of natural resources, issues related to Population growth and their overutilization, and strategies for their conservation. Water, air, soil, mineral, energy and food source. Effect of human activities on natural resources. Concept of sustainability- Sustainable Development Goals (SDGs)- targets and indicators, challenges and strategies for SDGs; Sustainable practices in managing resources, including deforestation, water conservation, Desalination – types, energy security, and food security issues, Life Cycle thinking and Circular Economy.						
Module 2	**Ecosystems, Biodiversity, and Sustainable Practices**					
Topics: **Ecosystems and ecosystem services:** Various natural ecosystems, Major ecosystem types in India and their basic characteristics; forests, wetlands, grasslands, agriculture, coastal and marine; Ecosystem services- classification and their significance. The importance of biodiversity, Types of biodiversity, Biodiversity and Climate Change, the threats it faces, hotspots, and the methods used for its conservation. Strategies for in situ and ex situ conservation, mega diverse nation.						
Module 3	**Environmental Pollution, Waste Management, and Sustainable Development**					
Topics: Types of pollution- Chemical, - Biological, Biomedical, noise, air, water, soil, thermal, radioactive and marine pollution,						

and their impacts on society. Urbanization and Urban environmental problems; effects, and mitigation. Causes of pollution, such as global climate change, ozone layer depletion, the greenhouse effect, and acid rain, with a particular focus on pollution episodes in India. Importance of adopting cleaner technologies; Solid waste management;

Sustainable Materials and Technologies: Biodegradable and compostable materials, Recycled and reclaimed materials (E-waste management), Sustainable manufacturing processes.

Module 4	**Social Issues, Legislation, and Practical Applications**			

Topics:

Overview of key environmental legislation and the judiciary's role in environmental protection, including the Water (Prevention and Control of Pollution) Act of 1974, the Environment (Protection) Act of 1986, and the Air (Prevention and Control of Pollution) Act of 1981. Hazardous waste Rule 1989, Biomedical Waste handling 1998, Fly Ash Rule 1999, Municipal Solid Waste Rule 2000, Battery Rules 2001, E- Waste Rules 2011, Plastic waste management Rules 2016, Construction Demolition waste Rules 2016 National Biodiversity Action Plan (NBAP)

Major International Environmental Agreements: Convention on Biological Diversity (CBD), The Biological Diversity (Amendment) Act, 2023, United Nations Framework Convention on Climate Change (UNFCCC); Kyoto Protocol; Paris Agreement.

Major International organisations and initiatives: United Nations Environment Programme (UNEP), United Nations Educational, Scientific and Cultural Organization (UNESCO), Intergovernmental Panel on Climate Change (IPCC).

Targeted Application & Tools that can be used:

Application areas are Energy, Environment and sustainability

Tools: Online Tools – NPTEL and Swayam.

Project work/Assignment:

Assessment Type

- Online exams (MCQs) will be conducted by the department of Chemistry

Online Link*:

1) Lecure by Dr. Samik Chowdhury, Dr. Sudha Goel, NPTEL course: Environmental Science, https://nptel.ac.in/courses/109105203, 2024.

2) Lecture by Dr. Padmavati, Dr Narendran Thiruthy, NPTEL Course: Biodiversity Protection, Farmers and Breeders Rights, https://nptel.ac.in/courses/129105008, 2024.

* Other source links are available in below Resources link.

Text Book

3. G. Tyler Miller and Scott Spoolman (2020), Living in the Environment, 20th Edition, Cengage Learning, USA

4. Poonia, M.P. Environmental Studies (3rd ed.), Khanna Book Publishing Co.

5. Bharucha, E. Textbook of Environmental Studies (3rd ed.) Orient Blackswan Private Ltd.

6. Dave, D., & Katewa, S. S. Text Book of Environmental Studies. Cengage Learning India Pvt Ltd.

7. Rajagopalan, R. Environmental studies: from crisis to cure (4th ed.). Oxford University Press.

8. Basu, M., & Xavier Savarimuthu, S. J. Fundamentals of environmental studies. Cambridge University Press.

9. Roy, M. G. Sustainable Development: Environment, Energy and Water Resources. Ane Books.

10. Pritwani, K. Sustainability of business in the context of environmental management. CRC Press.

11. Wright, R.T. & Boorse, D.F. Environmental Science: Toward A Sustainable Future (13th ed,). Pearson.

Reference Books

1. Varghese, Anita, Oommen, Meera Anna, Paul, Mridula Mary, Nath, Snehlata (Editors) (2022), Conservation through Sustainable Use: Lessons from India. Routledge.

2. William P. Cunningham and Mary Ann Cunningham (2020), Principles of Environmental Science: Inquiry & Applications, 9th Edition, McGraw-Hill Education, USA.

3. Richard A. Marcantonio, Marc Lame (2022). Environmental Management: Concepts and Practical Skills. Cambridge University Press.

4. Manahan, S.E. (2022). Environmental Chemistry (11th ed.). CRC Press. https://doi.org/10.1201/9781003096238

5. Theodore, M. K. and Theodore, Louis (2021) Introduction to Environmental Management, 2nd Edition. CRC Press

Resources:

1. https://nptel.ac.in/courses/109105203
2. https://archive.nptel.ac.in/courses/120/108/120108004/
3. https://nptel.ac.in/courses/127105018
4. https://onlinecourses.nptel.ac.in/noc23_lw06/preview
5. https://onlinecourses.swayam2.ac.in/ini25_bt02/preview
6. https://archive.nptel.ac.in/courses/120/108/120108002/
7. https://onlinecourses.swayam2.ac.in/ini25_bt02/preview
8. https://nptel.ac.in/courses/102104088
9. https://nptel.ac.in/courses/124107165
10. https://nptel.ac.in/courses/109106200
11. https://archive.nptel.ac.in/content/storage2/courses/120108004/module1/lecture1.pdf
12. https://onlinecourses.swayam2.ac.in/nou25_ge19/preview
13. https://onlinecourses.swayam2.ac.in/ini25_hs01/preview
14. http://kcl.digimat.in/nptel/courses/video/105105184/L32.html
15. https://nptel.ac.in/courses/105105169

Topics relevant to Skill Development:

An attitude of enquiry.

Write reports

The topics related to Environment and Sustainability :

All topics in theory component are relevant to Environment and Sustainability.

periment 5	perimental	ta Collection	Analysis and Interpretation
nductometric estimation of mixture of acids in Recycling process of E- waste.			
periment 6	perimental	ta Collection	Analysis and Interpretation
termination of viscosity coefficient of a given organic liquid using Ostwald's Viscometer (viscoelastic property).			
periment 7	perimental	ta Collection	Analysis and Interpretation
Recovery of valuable metals (copper) from e- waste by Iodometric titration.			
periment 8	perimental	ta Collection	Analysis and Interpretation
imation of iron in electronic devices using Std. Potassium permanganate solution.			
periment 9	perimental	ta Collection	Analysis
me photometric estimation of sodium (Battery Recycling- Optical Sensor).			
periment 10	perimental	ta Collection	Analysis
nthesis of conducting polyaniline for gas sensor applications (Demonstration experiment).			
periment 11	perimental	ta Collection	Analysis
een synthesis of nanomaterials (Demonstration experiment).			
periment 12	perimental	ta Collection	Analysis
Recovery of valuable metals from e- waste by eletroless method (Demonstration experiment).			
y 8 experiments will be conducted out of 12			

ntinuous Internal Assessment:

- Midterm exam
- Experimental Evaluation
- Viva-voce
- Endterm exam

xt Book

1. Lab manual for Engineering chemistry by B. Ramadevi and P. Aparna, S. Chand Publications, New Delhi (2022)
2. Vogel's text book of practical organic chemistry 5th edition
3. Inorganic Quantitative analysis by A.I. Vogel, ELBS Publications.
4. College Practical Chemistry by V.K. Ahluwalia, Narosa Publications Ltd. New Delhi

ferences

1. Engineering Chemistry Laboratory Manual (English, Paperback, Dr Manoj Kumar Solanki), Edu-creation Publishing

E-resources:

1. https://books-library.net/files/download-pdf-ebooks.org-kupd-679.pdf

Video Links:

1. https://www.youtube.com/watch?v=gd1YQr-74sw
2. https://www.youtube.com/watch?v=wVJ8WQax0rQ
3. https://www.youtube.com/watch?v=aWwEGCNtKwk
4. https://www.youtube.com/watch?v=JhBs_8DrPYo
5. https://www.youtube.com/watch?v=5bFAx2b_6A8
6. https://www.youtube.com/watch?v=_IVVZnAFfrM
7. https://www.youtube.com/watch?v=BBhuXOh9vOM
8. https://www.youtube.com/watch?v=j-nW3Jhc794

e topics related to Skill Development

All the experiments are relevant to Skill Development through Experiential Learning Techniques. This is attained through assessment component mentioned in course handout.

Course Code: CSE2200	**Course Title: Program Solving Using C** **Type of Course: Theory**		**L- T-P-C**	2	0	0	2
Version No.	1.0					
Course Pre-requisites	**NIL**					
Anti-requisites	**NIL**					
Course Description	The course is designed to provide complete knowledge of C language. Students will be able to develop logics which will help them to create programs and applications in C. ACAlso by learning the basic programming constructs they can easily switch over to any other language in future.					
Course Object	The objective of the course is to familiarize the learners with the concepts of Problem Solving Using C and attain Employability through Problem Solving Methodologies.					
Course Outcomes	**On successful completion of this course the students shall be able to:** 1. Write algorithms and to draw flowcharts for solving problems 2. Demonstrate knowledge and develop simple applications in C programming constructs 3. Develop and implement applications using arrays and strings 4. Decompose a problem into functions and develop modular reusable code 5. Solve applications in C using structures and Union 6. Design applications using Sequential and Random Access File Processing.					
Course Content:						
Module 1	Introduction to C Language	Quiz	Problem Solving	**6 Sessions**		
Topics: Introduction to Programming – Algorithms – Pseudo Code - Flow Chart – Compilation – Execution – Preprocessor Directives (#define, #include, #undef) - Overview of C – Constants, Variables and Data types – Operators and Expressions – Managing Input and Output Operations – Decision Making and Branching - Decision Making and Looping.						
Module 2	Introduction to Arrays and Strings	Quiz	Problem Solving	**6 Sessions**		
Topics: **Arrays:** Introduction – One Dimensional Array – Initialization of One Dimensional Arrays – Example Programs – Sorting (Bubble Sort, Selection Sort) – Searching (Linear Search) - Two Dimensional Arrays – Initialization of Two Dimensional Arrays. Example Programs – Matrix operations. **Strings:** Introduction – Declaring and Initializing String Variables – Reading Strings from Terminal – Writing String to Screen – String Handling Functions.						
Module 3	Functions and Pointers	Quiz	Problem Solving	**6 Sessions**		
Topics: **Functions:** Introduction – Need for User-defined functions – Elements of User-Defined Functions: declaration, definition and function call–Categories of Functions – Recursion. **Pointers:** Introduction – Declaring Pointer Variables – Initialization of Variables – Pointer Operators – Pointer Arithmetic – Arrays and Pointers – Parameter Passing: Pass by Value, Pass by Reference.						

Module 4	Structures and Union	Quiz	Problem Solving	**6 Sessions**
Topics: **Structures:** Introduction – Defining a Structure – Declaring Structure Variable – Accessing Structure Members – Array of Structures – Arrays within Structures – **Union:** Introduction – Defining and Declaring Union – Difference Between Union and Structure.				
Module 5	File handling	Case Study	Problem Solving	**6 Sessions**
Topics: Files: Defining and Opening a File – Closing a File – Input / Output Operations on File – Random Access Files				
Text Book(s): **1. E. Balaguruswamy, "Programming in ANSI C", 8th Edition, 2019, McGraw Hill Education, ISBN: 978-93-5316- 513-0.**				
Reference Book(s): **1. Yashwant Kanetkar, Let us C, 17th Edition, BPB Publications, 2020.** **2. ReemaThareja, "Programming in C", Oxford University Press, Second Edition, 2016.** **3. Kernighan, B.W and Ritchie,D.M, "The C Programming language", Second Edition, Pearson Education, 2015** **4. Schildt Herbert, "C: The Complete Reference", Tata McGraw Hill Education, 4th Edition, 2014.** **5. Stephen G. Kochan, "Programming in C", Addison-Wesley Professional, 4th Edition, 2014.**				
Web Links and Video Lectures: **1. https://nptel.ac.in/courses/106/105/106105171/** **2. https://archive.nptel.ac.in/courses/106/104/106104128/**				

Course Code: CSE2201	Course Title: Program Solving Using C Lab Type of Course: Lab	L- T-P-C	0	0	4	2
Version No.	1.0					
Course Pre-requisites	**NIL**					
Anti-requisites	**NIL**					
Course Description	The course is designed to provide complete knowledge of C language. Students will be able to develop logics which will help them to create programs and applications in C. ACAlso by learning the basic programming constructs they can easily switch over to any other language in future.					
Course Object	The objective of the course is to familiarize the learners with the concepts of Problem Solving Using C and attain Employability through Problem Solving Methodologies.					
Course Outcomes	**On successful completion of this course the students shall be able to:** 1. Write algorithms and to draw flowcharts for solving problems 2. Demonstrate knowledge and develop simple applications in C programming constructs 3. Develop and implement applications using arrays and strings 4. Decompose a problem into functions and develop modular reusable code 5. Solve applications in C using structures and Union 6. Design applications using Sequential and Random Access File Processing.					
Course Content:						

List of Practicals:

Lab Sheet 1: 10 Sessions

Program 1: Sum of Two Numbers
Program 2: Find the Greatest of Three Numbers
Program 3: Check Even or Odd using Conditional Operator
Program 4: Print Multiplication Table using Loop
Program 5: Count Digits in a Number using While Loop
Program 6: Demonstration of Preprocessor Directives
Program 7: Simple Calculator using Switch Case

Lab Sheet 2: 10 Sessions

Program 1: Check Whether a Number is Positive, Negative or Zero
Program 2: Find the Sum of First N Natural Numbers
Program 3: Check Whether a Number is Prime or Not
Program 4: Find Factorial of a Number
Program 5: Reverse a Number
Program 6: Simple Number Guessing Game

Lab Sheet 3: 10 Sessions

Program 1: Linear Search in a One-Dimensional Array
Program 2: Bubble Sort on an Integer Array
Program 3: Matrix Addition (2D Arrays)
Program 4: Count Vowels in a String
Program 6: Selection Sort on an Array

Lab Sheet 4: 10 Sessions

Program 1: Sum of Two Numbers Using User-Defined Function
Program 2: Factorial Using Recursion
Program 3: Swap Two Numbers Using Call by Value (No Swap)
Program 4: Swap Two Numbers Using Call by Reference (With Swap)
Program 5: Pointer Basics - Access and Modify Variable via Pointer
Program 6: Accessing Array Elements Using Pointers

Lab Sheet 5: 10 Sessions
Program 1: Basic Structure Usage
Program 2: Input and Display Array of Structures
Program 3: Array Inside Structure (Student Marks)
Program 4: Structure with Nested Structures (Date of Birth)
Program 5: Union Example and Member Access

Lab Sheet 6: 10 Sessions
Program 1: Write to a File (Text Mode)
Program 2: Read from a File (Text Mode)
Program 3: Append Data to a File
Program 4: Count Characters, Words and Lines in a File
Program 5: Write and Read Structure to/from a Binary File
Program 6: Random Access in File (Update a Record)

Text Book(s):

1. E. Balaguruswamy, "Programming in ANSI C", 8th Edition, 2019, McGraw Hill Education, ISBN: 978-93-5316- 513-0.

Reference Book(s):

6. Yashwant Kanetkar, Let us C, 17th Edition, BPB Publications, 2020.
7. ReemaThareja, "Programming in C", Oxford University Press, Second Edition, 2016.
8. Kernighan, B.W and Ritchie,D.M, "The C Programming language", Second Edition, Pearson Education, 2015
9. Schildt Herbert, "C: The Complete Reference", Tata McGraw Hill Education, 4th Edition, 2014.
10. Stephen G. Kochan, "Programming in C", Addison-Wesley Professional, 4th Edition, 2014.

Web Links and Video Lectures:

1. https://nptel.ac.in/courses/106/105/106105171/
2. https://archive.nptel.ac.in/courses/106/104/106104128/

ENG2501	**Advanced English**	**T- P- C**	2	0	0	2
Version No.	2.0					
Course Pre-requisites	ENG1900 - English for Technical Communication					
Anti-requisites	**NIL**					
Course Description	This course is designed to equip students to enhance their communication abilities in Listening, Speaking, Reading, and Writing. The curriculum covers interpersonal communication principles, the art of speech writing and delivery (including impromptu speaking), strategic approaches to critical reading, the identification of logical fallacies, and persuasive writing. Furthermore, the course will introduce students to the potential of AI tools and the techniques of prompt engineering to elevate their communication skills in the digital age. Upon course completion, students will be well-prepared to communicate effectively and critically in both academic and professional environments.					
Course Out Come	On successful completion of the course the students shall be able to: 1. Recognize the elements of interpersonal and cross-cultural communication to address communication challenges effectively. 2. Demonstrate the ability to deliver structured and impromptu speeches using effective speaking techniques. 3. Interpret textual and visual materials using critical reading strategies to evaluate arguments, logic, and persuasion. 4. Produce persuasive and analytical essays using effective argumentation techniques and structured writing strategies.					

Course Content:

Module 1	**undations of Effective Communication**	**se Studies/ Role play**	**oss-Cultural Competency**	**12 Classes**

Topics:

- Fundamentals of Interpersonal Communication
- Verbal, Non-verbal, and Paraverbal communication.
- Cultural dimensions theory (Hofstede's Cultural Dimensions).
- Active Listening Techniques
- Common Errors in Communication

Activities:

- Instagram/YouTube Vocabulary Activity
- Charades with a Twist/Tone and Emotion Experiment/Mixed Messages Challenge/Role Reversal Conversations/Observation Exercise

Module 2	**astering Speech Delivery**	**M**	**blic Speaking Confidence**	**12 Classes**

Topics:

- Introduction to Prompt Engineering
- Speech Preparation and Organization
- Techniques for Effective Impromptu Speaking
- Practice Speech Delivery

Activities:

- Speech Writing
- Impromptu Speech

Module 3	**itical Reading and Logical Analysis**	**orksheet**	**itical Thinking and Analysis**	**12 Classes**

Topics:

- Critical Reading Strategies: Contextualizing, Figurative Language, Evaluating Logic of an Argument, Recognizing Emotional Manipulation, Analysing Visuals
- Recognizing Logical Fallacies: Slippery Slope, False Dilemma, Post Hoc, Hasty Generalization, Ad Hominem, Straw Man, Bandwagon, No True Scotsman, Red Herring, Appeal to Authority, Sunk Cost, Appeal to ignorance

Activities:

- Critical Reading Worksheet/Identifying Bias in News Articles

Module 4	**riting Effective Arguments**	**signment**	**ear and Coherent Writing**	**Classes**

Topics**:**

- Understanding Critical Writing
- Building Arguments (Pathos, Ethos, Logos)
- Techniques for Persuasion

Activities:

- Causes or Effects/Appeal Mash-Up/Debates on Controversial Topics
- Opinion Writing

Targeted Application & Tools that can be used: Quizziz, Chatgpt, Gemini, Youtube, Instagram, Quillbot, Grammarly, Padlet

References

1. Adler, R. B., Rodman, G., & DuPré, A. (2019). *Understanding human communication (14th ed.)*. Oxford University Press.
2. Moore, B. N., & Parker, R. (2020). *Critical thinking* (13th ed.). McGraw-Hill Education.
3. Hamilton, C. (2020). *Communicating for success* (2nd ed.). Routledge.
4. Ting-Toomey, S., & Dorjee, T. (2018). Intercultural competence: A model for teaching and assessing cross-cultural communication. *Journal of Intercultural Communication, 47*(2), 213–229. https://doi.org/10.1016/j.jicc.2018.03.004
5. https://www.ted.com/

Topics Relevant to "employability": Teamwork and Collaboration, Critical Thinking and Problem-Solving

Topics Relevant to "Human Values and Professional Ethics": Critical reasoning, Inclusivity and Fairness

Course Code: LAW7601	**Course Title: Indian Constitution** **Type of Course: Theory**	**L-T-P-C**	0	0	0	0
Version No.						
Course Prerequisites						
Anti-requisites	NIL					
Course Description	The purpose of this course is to introduce the students to the theory, concepts and practice of Constitution of India which is the law of the land. Further, the course aims at acquainting the students with basic approaches and methodologies to analyse and decide on the ethical dilemma in the field of engineering. The course is both conceptual and analytical. The course develops critical thinking skills by augmenting the student's ability to comprehend the conceptual and legal framework of Constitution of India. Ethics and values are very beautifully weaved into the tapestry of the Indian Constitution. Therefore, the course provides an introduction to the essential, theoretical basis of engineering ethics and its application through a range of industry relevant topics such as engineers' responsibility for safety and risks, responsibility of employers, rights of engineers etc.					
Course Objective	1. To introduce the students to the conceptual framework of Constitution of India and engineering ethics. 2. To enhance the practical knowledge on responsibility of engineering professionals as citizens of India. 3. To acquaint the student with the relevant contemporary issues surrounding constitutional values and professional ethics. 4. To orient the students about the ethical concepts and frameworks enabling them to identify the codes and moral values relevant to the professional world.					
Course Outcomes	**On successful completion of this course the students shall be able:** 1. To understand foundational Indian constitutional law concepts and values. 2. To identify the different pillars of democracy and their functions. 3. To analyse the role of the engineers' responsibility in ensuring safety of the society and the employer.					

Module 1	**Introduction to the Indian Constitution**	Knowledge	Quiz	**5 Classes**
Course Content:				

Meaning of Constitution, Constitutional Law and Constitutionalism, India before and after adoption of Constitution, Preamble, Salient Features, Concept and Relevance of Fundamental Rights, Fundamental Duties and Directive Principles of State Policy in brief.				
Module 2	**Pillars of Democracy: Legislature Executive and Judiciary**	Knowledge	Short Essay	**5 Classes**
Federalism, Union and State Executive, Parliament and State Legislature, Union and State Judiciary, Amendment of the Constitution				
Module 3	**Engineering Ethics**	Analysis	Presentation on conceptual understanding and problem based scenarios	**5 Classes**
Scope & Aims of Engineering & Professional Ethics, Code of Ethics as defined in the website of Institution of Engineers (India), Profession, Professionalism, and Professional Responsibility, Conflicts of Interest, Engineering Standards, the impediments to Responsibility, IPRs (Intellectual Property Rights), Necessity of responsible experimentation ,Case Studies on Challenger, Chernobyl, and Boeing.				
Project work/Assignment: Quiz on Fundamental Rights, Short Essay on Judicial Activism in India, Problem based assignments of engineering ethics.				
Resources: 1. M.P. Jain, Indian Constitutional Law, 8th Edition, Lexis Nexis, 2022. 2. M.W.Martin and R. Schinzinger, Ethics in Engineering, 4thEdition, McGraw Hill Education, 2015.				

Course Code: **PPS1026**	**Course Title: Industry Readiness Program – II (Audited Course)** Type of Course: Practical Only Course	L- T - P- C	0	0	2	0
Version No.	1.0					
Course Pre-requisites	• Students are expected to understand Basic English. • Students should have desire and enthusiasm to involve, participate and learn.					
Anti-requisites	NIL					
Course Description	This course is designed to enable students learn styles of communication, team building and use empathy in leadership. The course will benefit learners in preparing themselves effectively through various activities and learning methodologies.					
Course Objective	The objective of the course is to **familiarize the learners with the concepts of "Industry Readiness for Young Professionals" and attain SKILL DEVELOPMENT through PARTICIPATIVE LEARNING techniques.**					
Course Out Comes	**On successful completion of this course the students shall be able to:** CO 1 **Apply** different communication skills for success in workplace CO 2 **Practice team building skills** for career success CO3 **Demonstrate** ethical leadership skills in workplace					

Course Content			
Module 1	**Effective Communication**	Classroom activities	**10 Hours**
Topics: Practice effective communication skills (Verbal, Non-verbal, Written and Visual) **Activity:** Use social media prompts to prepare self-introduction videos			
Module 2	**Team Building**	Group Activity	**10 Hours**
Topics: Skills of an effective team player **Activity:** Student group activity to build class networking			
Module 3	**Leadership**	Case study	**10 Hours**
Topics: Types of leadership, using empathy in leadership **Activity:** Individual presentation by students on corporate leaders.			
Faculty : L&D			
Targeted Application & Tools that can be used: 4. TED Talks 5. You Tube Links 6. Activities			
Assignment proposed for this course Assignment 1: One minute reel Assignment 2: Team building assignment			
Continuous Individual Assessment Module 1: L-S-R-W class assessment Module 2: Team Presentation Module 3: Individual Assessment			

The topics related to skill development:

Students acquire knowledge on effective communication skills, team building skills and how to prepare themselves to be leaders in workplace using empathy and implement various skill sets during the course of their time in the university.

Course Code: MAT2303	**Course Title: Linear Algebra & Vector Calculus** **Type of Course:1] School Core**	**L-T- P- C**	**3**	**1**	**0**	**4**
Version No.	1.0					
Course Pre-requisites	Basic Concepts of Limits, Differentiation, Integration, Matrices (PU Level)					
Anti-requisites	**NIL**					
Course Description	This course explores the fundamental concepts of vectors, matrices, and their operations within the context of calculus, including vector differentiation and integration, while applying these tools to solve problems related to linear systems, transformations, and geometric interpretations in higher dimensions, often with applications in fields like physics, engineering, and computer graphics; key topics include vector algebra, matrix operations, determinants, eigenvalues, eigenvectors, gradients, divergence, curl, line integrals, surface integrals, and the fundamental theorems of vector calculus like Green's Theorem, Stokes' Theorem, and the Divergence Theorem.					
Course Objective	The course is intended to develop computational proficiency involving procedures in Matrices, Linear Algebra and Vector Calculus which are useful to all engineering disciplines. This course is to equip students with the ability to understand and manipulate vectors in multidimensional space, apply matrix operations to solve systems of linear equations, and utilize concepts like gradients, divergence, and curl to analyze physical phenomena, all while developing a strong foundation for applying these tools in various scientific and engineering fields like physics, mechanics, and computer graphics.					
Course Out Comes	On successful completion of the course the students shall be able to: CO1 - Use matrix methods and certain techniques to solve the system of linear equations and to find eigen values, eigen vectors of a matrix to check whether it is diagonalizable. CO2 - Understand the abstract notions of vector space and dimensionality of it. CO3 - find the matrix representation of a linear transformation given bases of the relevant vector spaces. CO4 - Learn different notions of vector and scalar fields with their properties. Understanding the major theorems (Green's, Stokes', Gauss') and some applications of these theorems.					
Course Content:						
Module 1	**Systems of Linear Equations**			6. ***Classes)***		
Systems of Linear Equations, Matrices and Elementary Row Operations, Echelon forms, Matrix operations, invertible matrices, Determinants and their properties, Cramer's Rule, LU-decomposition, Applications of Systems of Linear Equations.						

Module 2	**Vector Space**	**Assignment**	***(9Classes)***

Linear Combinations and Linear Independence, Vectors in $\mathbb{R}^n$, Linear Combinations, Linear Independence Vector Spaces, Definition of a Vector Space, Subspaces, Basis and Dimension, Coordinates and Change of Basis, Orthogonal bases and orthogonal projections.

Module 3	**Linear Transformations**		***(15 lectures)***

Linear Transformations, Algebra of transformations, The Null Space and Range, Isomorphisms, Matrix Representation of Linear Transformations, Similarity Eigenvalues and Eigenvectors, Eigen values and Eigen vectors, Diagonalization.

Inner Product Spaces, The Dot Product on R^n and Inner Product Spaces, Orthonormal Bases, Orthogonal Complements, Application: Least Squares Approximation, Diagonalization of Symmetric Matrices, Application: Quadratic Forms.

Singular Value Decomposition: Singular values, computing singular value decomposition, and Introduction to principal component analysis.

Module 4	**Vector Calculus**	**Assignment**	***(15 lectures)***

Vector & Scalar Functions and Fields, Derivatives, Curve, Arc length, Curvature & Torsion, Gradient of Scalar Field, Directional Derivative, Divergence of a Vector Field, Curl of a Vector Field, Physical interpretation, solenoidal and irrotational vector fields. Problems.

Line Integrals, Path Independence of Line Integrals, Green`s Theorem in the plane, Surface Integrals, Divergence Theorem of Gauss, Stokes`s Theorem.

Targeted Application & Tools that can be used:

Solve systems of linear equations using various methods including Gaussian and Gauss Jordan elimination and inverse matrices.

Perform matrix algebra, invertibility, and the transpose and understand vector algebra in R^n.

Determine relationship between coefficient matrix invertibility and solutions to a system of linear equations and the inverse matrices.

Find eigenvalues and eigenvectors and use them in applications.

Find the dimension of spaces such as those associated with matrices and linear transformations.

Understand real vector spaces and subspaces and apply their properties.

Compute inner products in a real vector space and compute angle and orthogonality in inner product spaces.

Create orthogonal and orthonormal bases: Gram-Schmidt process and use bases and orthonormal bases to solve application problems.

Prove basic results in linear algebra using appropriate proof-writing techniques such as linear independence of vectors; properties of subspaces; linearity, injectivity and subjectivity of functions; and properties of eigenvectors and eigenvalues.

Assignment:

paration of power-point slides, which include videos, animations, pictures, graphics for better understanding applications of Linear Algebra and Vector Calculus to engineering applications – The faculty will allocate pters/ parts of chapters to groups of students so that the entire syllabus of Linear Algebra and Vector Calculus vered.

Text Book

1. Gilbert Strang, Linear Algebra and its applications, Wellesley-Cambridge Press,U.S.; 6th edition.
2. B. S. Grewal (2017), Higher Engineering Mathematics by, 44th Edition, Khanna Publishers.

References:

1. Introduction to Linear Algebra with Application, Jim Defranza, Daniel Gagliardi, Tata McGraw-Hill
2. Elementary Linear Algebra, Applications version, Anton and Rorres, Wiley India Edition.
3. Advanced Engineering Mathematics, Erwin Kreysig, Wiley Publication.
4. Elementary Linear Algebra, Ron Larson, Cengage Learning .
5. Linear Algebra and its Applications, David C. Lay, Pearson Education.

E-resources/ Web links:

1. https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_9607
2. https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_143156
3. https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=CUSTOM_PACKAGE_EBSCO_29052023_270975
4. https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_94555
5. https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_243864
6. https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_224531
7. NPTEL Video Lectures Matrices and Linear Algebra:
8. https://nptel.ac.in/courses/111106051/
9. NPTEL Video Lectures Differential Equations:
10. https://nptel.ac.in/courses/111106100/
11. NPTEL Vector Calculus:
12. https://nptel.ac.in/courses/111/105/111105122/
13. https://www.math.hkust.edu.hk/~maqian/ma006_0607F.html
14. https://www.scu.edu.au/study-at-scu/units/math1005/2022/

Topics relevant to SKILL DEVELOPMENT: The course focuses on the concepts of Vector calculus and Linear Algebra with reference to specific engineering problems. The course is of both conceptual and analytical type in nature through Problem Solving**.** This is attained through the assessment component mentioned in the course handout.

Course Code: CSE2253	**Course Title:** Data Structures **Type of Course: Theory**	**L-T- P- C**	3	0	0	3
Version No.	1.0					
Course Pre-requisites						
Anti-requisites	NIL					
Course Description	This course introduces the fundamental concepts of data structures and to emphasize the importance of choosing an appropriate data structure and technique for program development. This course has theory and lab component which emphasizes on understanding the implementation and applications of data structures using Java programming language. With a good knowledge in the fundamental concepts of data structures and practical experience in implementing them, the student can be an effective designer, developer for new software applications.					
Course Objective	The objective of the course is SKILL DEVELOPMENT of student by using EXPERIENTIAL LEARNING techniques					
Course Out Comes	**On successful completion of the course the students shall be able to:** **CO1:** Describe the concept of basic data structure, stacks, queues, and arrays and their operations. **[Understand]** **CO2:** Utilize linked lists for real-time scenarios. **[Apply]** **CO3:** Apply an appropriate non-linear data structure for a given scenario. **[Apply]** **CO4:** Demonstrate different searching and sorting techniques. **[Apply]**					
Course Content:						
Module 1	**Introduction to Data Structure and Linear Data Structure – Stacks and Queues**	Assignment	Program activity			**9 Hours**
Introduction – Introduction to Data Structures, Types and concept of Arrays**.** **Stack -** Concepts and representation, Stack operations, stack implementation using array and Applications of Stack**.** **Queues -** Representation of queue, Queue Operations, Queue implementation using array, Types of Queue and Applications of Queue**.**						
Module 2	**Linear Data Structure- Linked List**	Assignment	Program activity			**12 Hours**
Topics: Linked List - Singly Linked List, Operation on linear list using singly linked storage structures, Circular List, Applications of Linked list. **Recursion** - Recursive Definition and Processes.						
Module 3	**Non-linear Data Structures - Trees**	Assignment	Program activity			**12 Hours**

Topics: Trees - Introduction to Trees, Binary tree: Terminology and Properties, Use of Doubly Linked List, Binary tree traversals: Pre-Order traversal, In-Order traversal, Post - Order traversal, Binary Serach Tree, AVL Trees - Red Black Tree, Expression Tree , Heaps.

Module 4	**Non-linear Data Structures - Graphs and Hashing**	Assignment	Program activity	**6 Hours**

Topics: Graphs: Basic Concept of Graph Theory and its Properties, Representation of Graphs . ADT, Elementary graph operations, Minimum Cost spanning trees, Shortest path and Transitive closure.
Hashing: Introduction, Static Hashing, Dynamic Hashing

Module 5	**Searching & Sorting**	Assignment	Program activity	**6 Hours**

Topic: Sorting & Searching - Sequential and Binary Search, Sorting – Selection and Insertion sort, Quick sort, Merge Sort, Bubble sort.

List of Laboratory Tasks:

Lab sheet -1
Level 1: Prompt the user, read input and print messages.Programs using class, methods and objects
Level 2: Programming Exercises on fundamental Data structure - Arrays based on Scenario.

Lab sheet -2
Level 1: Programming Exercises on Stack and its operations
Level 2: Programming Exercises on Stack and its operations with condition

Lab sheet -3
Level 1: Programming on Stack application infix to postfix Conversion
Level 2: -

Lab sheet -4
Level 1: Programming on Stack application – Evaluation of postfix

Lab sheet -5
Level 1: Programming Exercises on Queues and its operations with conditions
Level 2: -

Lab sheet -6
Level 1: Programming Exercises on Linked list and its operations.
Level 2: Programming Exercises on Linked list and its operations with various positions

Lab sheet -7
Level 1: Programming Exercises on Circular Linked list and its operations.
Level 2: Programming Exercises on Circular Linked list and its operations with various positions

Lab sheet -8
Level 1: Programming Exercises on factorial of a number
Level 2: Programming the tower of Hanoi using recursion

Lab sheet -9
Level 1: -

Level 2: Programming the tower of Hanoi using recursion

Lab sheet -10

Level 1: Programming Exercise on Doubly linked list and its operations

Level 2: -

Lab sheet -11

Level 1: Program to Construct Binary Search Tree and Graph

Level 2: Program to traverse the Binary Search Tree in three ways)in-order, pre-order and post-order(and implement BFS and DFS

Lab sheet -12

Level 1: Program to Implement the Linear Search & Binary Search

Level 2: Program to Estimate the Time complexity of Linear Search

Lab sheet -13

Level 1: Program to Implement and Estimate the Time complexity of Selection Sort

Level 2: Program to Implement and Estimate the Time complexity of Insertion Sort

Lab sheet -14 (Beyond syllabus activity)

Level 1: Program to Construct AVL Tree

Level 2:

Lab sheet -15 (Beyond syllabus activity)

Level 1: Program to Construct RED BLACK Tree

Targeted Application & Tools that can be used

Use of PowerPoint software for lecture slides and use of Modern IDE like VS Code and Eclipse for lab programs to execute.

Project work/Assignment:

Assignment: Students should complete the lab programs by end of each practical session and module wise assignments before the deadline.

Text Book

T1 Ellis Horowitz and Sartaj Sahni, Fundamentals of Data Structures in C, 2nd Edition, Universities Press, reprint 2018.

T2 Seymour Lipschutz, Data Structures Schaum's Outlines, Revised 1st Ed, McGraw Hill, 2014.

References

R1 Data structures and program design in C by Robert Kruse, Tondo C L, Bruce Leung, Pearson education publishers, 2017.

R2 Programming and Data Structure by Jackulin C Salini etal., Ane books publishers, 2019.

Web resources:

1. For theory: https://onlinecourses.nptel.ac.in/noc20_cs85/preview
2. https://puniversity.informaticsglobal.com/login

Topics relevant to development of "Skill Development":

Linked list and stacks

Topics relevant to development of "Environment and sustainability: Queues

Course Code: CSE2258	**Course Title: Web Technologies Type of Course: Program core** Theory	**L-T- P- C**	3	0	0	3

Version No.		1.0
Course Pre-requisites		NIL
Anti-requisites		**NIL**
Course Description		This course highlights the comprehensive introduction to scripting languages that are used for creating web-based applications. The associated laboratory provides an opportunity to implement the concepts and enhance critical thinking and analytical skills.
Course Objective		**The objective of the course is to familiarize the learners with the concepts of Web Technology and attain Skill Development through Experiential Learning techniques.**
Course Outcomes		**On successful completion of this course the students shall be able to:** **CO1:** Implement web-based application using client-side scripting languages. **(Apply)** **CO2**: Apply various constructs to enhance the appearance of a website. **(Apply)** **CO3:** Apply server-side scripting languages to develop a web page linked to a database. **(Apply)**
Course Content:		

Module 1	**Introduction to XHTML**	Quizzes and Assignments	Quizzes on various features of XHTML, simple applications		**20 Sessions**
	Basics: Web, WWW, Web browsers, Web servers, Internet. **XHTML:** Origins and Evolution of HTML and XHTML: Basic Syntax, Standard XHTML Document Structure, Basic Text Markup, Images, Hypertext Links, Lists, Tables, Forms, Frames, Syntactic Differences between HTML and XHTML, Demonstration of applications using XHTML for Responsive web pages.				
Module 2	**Advanced CSS**	Quizzes and assignments	Comprehension based Quizzes and assignments; Application of CSS in designing webpages		**20 Sessions**

	Advanced CSS: Layout, Normal Flow, Positioning Elements, Floating Elements, Constructing Multicolumn Layouts, Approaches to CSS Layout, Responsive Design, CSS Frameworks

XML: Basics, Demonstration of applications using XML with XSLT.

Module 3	**PHP – Application Level**	Quizzes and assignments	Application of PHP in web designing	**20 Sessions**

PHP: Introduction to server-side Development with PHP, Arrays, Superglobal Arrays, $GET and $ POST, $_SERVER Array, $_Files Array, Reading/Writing Files, PHP Classes and Objects, Object Oriented Design, Working with Databases, SQL, Database APIs, Managing a MySQL Database. Accessing MySQL in PHP, Applications.

List of Laboratory Tasks:

Experiment No. 1: Demonstration of XHTML features

Level 1: Demonstration of various XHTML Tags (Level 1)
Level 2: Design and develop static web pages for an online Book store (Level 2).

Experiment No. 2: Application of CSS in web designing

Level 1: Design a document using XHTML and CSS to create a catalog of items for online electronic shopping.

Level 2: Create and save XML document for students' information and display the same using cascaded style sheet.

Experiment No. 3: Application of PHP in web designing.

Level 1: Write a PHP program to read the personal information of a person such as first name, last name, age, permanent address, and pin code entered by the user into a table created in MySQL. Read the same information from the database and display it on the front end.

Level 2: Using PHP develop a web page that accepts book information such as ISBN number, title, authors, edition, and publisher and store information submitted through the web page in MySQL database.

Experiment No. 4: Building a website.

Build a website for organizing an International Conference. The conference website must be able to collect the author's details and upload a file.

Targeted Application & Tools that can be used: **Xampp web server to be used to demonstrate PHP.**

Project work/Assignment:

Assignments are given after completion of each module which the student need to submit within the stipulated deadline.

Textbook(s):

1] Robert. W. Sebesta, "*Programming the World Wide Web"*, Pearson Education, 9th Edition, 2016.

2]Paul Deitel, Harvey Deitel, Abbey Deital,"Internet & World Wide Web How to Program", Fifth Edition, Pearson Education, 2021.

3]CSS Notes for Professionals, ebook available at https://books.goalkicker.com/CSSBook/ (Retrieved on Jan. 20, 2022)

4]Deitel, Deitel, Goldberg,"*Internet & World Wide Web How to Program*", Fifth Edition, Pearson Education, 2021.

Reference Book(s):

R1. Randy Connolly, Ricardo Hoar,"Fundamentals of Web Development", Pearson Education India, 1st. Edition.2016.

R2. Jeffrey C. Jackson,"Web Technologies: A Computer Science Perspective", Pearson Education, 1st Edition,2016.

Additional web-based resources

W1. W3schools.com
W2. Developer.mozilla.org/en-US/docs/Learn
W3. docs.microsoft.com
W4. informit.com/articles/ The Relationship Between Web 2.0 and Social Networking
https://presiuniv.knimbus.com/user#/home

Topics related to development of "FOUNDATION":

1. Web, WWW, Web browsers, Web servers, Internet.
2. CSS, PHP.
3. Designing the website for healthcare.

The objective of the course is to familiarize the learners with the concepts of Web Technology and attain Skill Development through Experiential Learning techniques.

Course Code: CSE1500	**Course Title: Computational Thinking Using Python** **Type of Course: Lab Integrated**	**L- T- P- C**	2	0	2	3
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	NIL					
Course Description	This course introduces students to the essential skills of **computational thinking** and their practical application through the **Python programming language**. By combining problem-solving strategies with coding, students will learn to decompose complex challenges, identify patterns, abstract general principles, and design algorithms to build functional programs					
Course Objective	The objective of the course is to familiarize the learners with the concepts of Computational Thinking and use the Computational Thinking Principles to solve the computational Problems using Python Language					

Course Outcomes	Upon successful completion of this course, students will be able to:
	• Explain and apply the core principles of **computational thinking**:
	  ○ Decomposition
	  ○ Pattern Recognition
	  ○ Abstraction
	  ○ Algorithm Design
	• Use **Python** to implement solutions to real-world problems.
	• Write and debug Python code using functions, loops and conditions
	• Design simple programs and algorithms to automate repetitive or complex tasks.
	• Collaborate effectively and communicate problem-solving approaches using pseudocode and Python.					

Course Content:						
Module 1	**Pillars of Computational Thinking**	Comprehension			**9 Sessions**	
What is computational thinking? Why is it important? Pillars of computational thinking: decomposition; pattern recognition; data representation and abstraction; algorithms Applying computational thinking to case studies						
Module 2	**Algorithm Design & Problem-Solving Strategies**	Application			**9 Sessions**	
Introduction to Algorithms, Introduction to Problem Solving techniques: Brute Force, Divide and conquer, Common algorithms: find-max, linear search, binary search and other simple Algorithms						
Module 3	**Applied Computational Thinking using Python**	Application			**12 Sessions**	
Introduction to Python, Data representation: variables, lists, Conditionals, Loops and Iteration Basic Example programs to illustrate the programming constructs						
Targeted Application & Tools that can be used: **Google Colab, Python**						
Text Book 1. "Computational Thinking for the Modern Problem Solver" – David D. Riley & Kenny A. Hunt						

2. "Mastering Python 3 Programming: Ultimate Guide to Learn Python Coding Fundamentals and Real-World Applications" Subburaj Ramaswamy, BPB publications

References

1. **Sweigart, Al.**
Automate the Boring Stuff with Python: Practical Programming for Total Beginners.
No Starch Press, 2015.
https://automatetheboringstuff.com

Severance, Charles.
Python for Everybody: Exploring Data Using Python 3.
CreateSpace Independent Publishing, 2016.
https://www.py4e.com

Wing, Jeannette M.
"Computational Thinking." *Communications of the ACM*, vol. 49, no. 3, 2006, pp. 33–35.
https://doi.org/10.1145/1118178.1118215

Downey, Allen B.
Think Python: How to Think Like a Computer Scientist.
Green Tea Press, 2015.
http://greenteapress.com/wp/think-python-2e/

.

E-Resources

https://edu.google.com/resources/programs/exploring-computational-thinking

Topics relevant to "SKILL DEVELOPMENT": Decomposition, Abstraction, Pattern recognition, Data Representation ,Algorithms

Cuours e Code: CSE2251	**Course Title:** Data Communication and Computer Networks **Type of Course:** **Theory**		**L- T-P- C**	3	**0**	0	**3**
Version No.	1.0						
Course Pre-requisites	NIL						
Anti-requisites	**NIL**						
Course Description	The objective of this course is to provide knowledge in data communications and computer networks, its organization and its implementation, and gain practical experience in the installation, monitoring, and troubleshooting of LAN systems. The associated laboratory is designed to implement and simulate various networks using Cisco packet tracer, NS2. All the lab exercises will focus on the fundamentals of creating multiple networks, topologies and analyzing the network traffics.						
Course Objective	The objective of the course is to familiarize the learners with the concepts of Data Communications and Computer Networks and attain Employability through Problem Solving Methodologies.						
Course Outcomes	On successful completion of the course, the students shall be able to: 1] Ilustrate the Basic Concepts Of Data Communication and Computer Networks. 2] Analyze the functionalities of the Data Link Layer. 3] Apply the Knowledge of IP Addressing and Routing Mechanisms in Computer Networks. 4] Demonstrate the working principles of the Transport layer and Application Layer.						
Course Content:							
Module 1	Introduction and Physical Layer- CO1	Assignment	Problem Solving	**7 Sessions**			
Introduction to Computer Networks and Data communications, Network Components – Topologies, Transmission Media –Reference Models -OSI Model – TCP/IP Suite. Physical Layer -Analog and Digital Signals – Digital and Analog Signals – Transmission - Multiplexing and Spread Spectrum.							
Module 2	Reference Models and Data Link Layer – CO2	Assignment	Problem Solving	**7 Sessions**			
Data Link Layer - Error Detection and Correction – Parity, LRC, CRC, Hamming Code, Flow Control and Error Control, Stop and Wait, ARQ, Sliding Window, Multiple Access Protocols, CSMA/CD,CSMA/CA, IEEE 802.3, IEEE 802.11 Ethernet.							
Module 3	Network Layer –CO3	Assignment	Problem Solving	**10 Sessions**			
Network Layer Services - Network Layer Services, Switching Techniques, IP Addressing methods- IPv4 IPV6 – Subnetting. Routing, - Distance Vector Routing – RIP-BGP-Link State Routing –OSPF-Multi cast Routing-MOSPF- DVMRP – Broad Cast Routing. EVPN-							

VXLAN, VPLS, ELAN.				
Module 4	Transport and Application Layer -CO3	Assignment	Problem Solving	**10 Sessions**

Transport Layers - Connection management – Flow control – Retransmission, UDP, TCP, congestion control, – Congestion avoidance (DECbit, RED)
The Application Layer: Domain Name System (DNS), Domain Name Space, SSH, FTP, Electronic Mail (SMTP, POP3, IMAP, MIME) – HTTP – – SNMP, Web Services, Virtual Networking.

Targeted Application & Tools that can be used: Cisco Packet Tracer, Wireshark, and NS2.

Case Study/Assignment: Choose and analyze a network from any organization/Assignment proposed for this course in CO1-CO4

Problem Solving: Choose and appropriate devices and implement various

network concepts.

Programming: Simulation of any network using NS2.

Text Book(s):

1. 1. Behrouz A. Forouzan, “Data Communications and Networking 5E”, 5 th Edition,

Tata McGraw-Hill, 2017.

2. Andrew S Tanenbaum, Nick Feamster & David J Wetherall, “Computer

Networks” Sixth Edition, Pearson Publication, 2022

Reference(s):

1.References

1. “Computer Networking: A Top-Down Approach”, Eighth Edition, James F. Kurose, Keith W. Ross, Pearson publication, 2021.
2. William Stallings, Data and Computer Communication, 8th Edition, Pearson Education, 2007.
3. Larry L. Peterson and Bruce S. Davie: Computer Networks – A Systems Approach, 4th Edition, Elsevier, 2007.

E- Resources:

1. https://archive.nptel.ac.in/courses/106/105/106105183/
2. http://www.nptelvideos.com/course.php?id=393
3. https://www.youtube.com/watch?v=3DZLItfbqtQ
4. https://www.youtube.com/watch?v=_fIdQ4yfsfM
5. https://www.digimat.in/keyword/106.html
6. https://puniversity.informaticsglobal.com/login

Course Code: CSE2257	**Course Title:** Computer Organization and Architecture **Type of Course:** PCC, **Theory only**	L- C	T-	P-	3	0	0	3
Version No.	2.0							
Course Pre-requisites	NIL							
Anti-requisites	NIL							
Course Description	This course introduces the core principles of computer architecture and organization from basic to intermediate level. This theory based course emphasizes on understanding the interaction between computer hardware and software. It equips the students with the intuition behind assembly-level instruction set architectures. It helps the students to interpret the operational concepts of computer technology as well as performance enhancement.							
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Computer Organization and Architecture** and attain **Skill Development** through **Participative Learning** techniques.							
Course Outcomes	On successful completion of the course the students shall be able to:] Describe the basic components of a computer and their interconnections. [Remember] 2] Explain Instruction Set Architecture and Memory Unit[Understand] 3] Apply appropriate techniques to carry out selected arithmetic operations [Apply] 4] Explain the organization of memory and processor sub-system [Understand]							
Course Content:								

Module 1	Basic Structure ofcomputers	Assignment	Data Analysis task		**12 Sessions**
	Topics: Computer Types, Functional Units, Basic Operational concepts, Bus Structures, Computer systems RISC & CISC,Performance – Processor Clock, Basic Performance Equation, Clock Rate, Performance Measurement. Arithmetic Operations on Signed numbers. Instructions and Instruction Sequencing, Instruction formats, Memory Instructions.				
Module 2	Instruction Set Architecture and Memory Unit	Assignment	Analysis, Data Collection		**12 Sessions**
	Topics: **Instruction Set Architecture:** Addressing Modes, Stacks and Subroutines. **Memory System:** Memory Location and Addresses, Memory Operations, Semiconductor RAM Memories, Internal Organization of Memory chips, Cache memory mapping Techniques.				
Module 3	Arithmetic and Input/output Design	Case Study	Data analysis task		**10 Sessions**
	Topics: **Arithmetic:** Carry lookahead Adder, Signed-Operand Multiplication, Integer Division, and Floating point operations. Input/output Design: Accessing I/O Devices, I/O communication, Interrupt Hardware, Direct Memory Access, Buses, Interface Circuits				
Module 4	BPU and Pipelining	Assignment	Analysis, Data Collection		**11 Sessions**

	Topics: **Basic Processing Unit:** Fundamental Concepts - Single Bus organization, Control sequence, Execution of aComplete Instruction, Multiple Bus Organization. **Pipelining:** Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, Hazards.
	Targeted Application & Tools that can be used: Targeted employment sector is processor manufacturing and memory chip fabrication vendors like Intel, AMD, Motorola, NVidia, Samsung, Micron Technology, western Digital etc. Targeted job profiles include Memory circuit design and verification engineers, Physical system design engineer, System programmer, Fabrication engineer etc. **Tools:** Virtual Lab, IIT KGP Tejas – Java Based Architectural Simulator, IIT Delhi
	Project work/Assignment:
	Each batch of students (self-selected batch mates – up to 4 in a batch) will be allocated case studies/assignments
	Textbook(s): **1.** Carl Hamacher, Zvonko Vranesic, Safwat Zaky, "Computer Organization", Sixth Edition, McGraw-HillHigher Education, 2023 reprint. 2. William Stallings, "Computer Organization & Architecture – Designing for Performance", 11th Edition, Pearson Education Inc., 2019.
	References 1. David A. Patterson & John L. Hennessy, "Computer Organization and Design MIPS Edition- The Hardware/Software Interface", 6th Edition, Morgan Kaufmann, Elsevier Publications, November 2020. Web References: 1. NPTEL Course on "Computer architecture and organization" IIT Kharagpur By Prof. Indranil Sengupta, Prof. Kamalika Datta. https://nptel.ac.in/courses/106105163 2. NPTEL Course on "Computer Organization", IIT Madras By Prof. S. Raman. https://nptel.ac.in/courses/106106092 3. https://puniversity.informaticsglobal.com:2229/login.aspx
	Topics relevant to "SKILL DEVELOPMENT": Generation of Computers, CISC and RISC processors, Bus Arbitration, Collaboration and Data collection for Term assignments and Case Studies for Skill Development through Participative Learning techniques. This is attained through assessment component mentioned in course handout.

Course Code: CSE2254	**Course Title:** Data Structures Lab **Type of Course: Lab**		**L-T- P- C**	0	0	2	1
Version No.	1.0						
Course Pre-requisites	NIL						
Anti-requisites	NIL						
Course Description	This course introduces the fundamental concepts of data structures and to emphasize the importance of choosing an appropriate data structure and technique for program development. This course has theory and lab component which emphasizes on understanding the implementation and applications of data structures using Java programming language. With a good knowledge in the fundamental concepts of data structures and practical experience in implementing them, the student can be an effective designer, developer for new software applications.						
Course Objective	The objective of the course is SKILL DEVELOPMENT of student by using EXPERIENTIAL LEARNING techniques						
Course Out Comes	**On successful completion of the course the students shall be able to:** **CO1:** Describe the concept of basic data structure, stacks, queues, and arrays and their operations. **[Understand]** **CO2:** Utilize linked lists for real-time scenarios. **[Apply]** **CO3:** Apply an appropriate non-linear data structure for a given scenario. **[Apply]** **CO4:** Demonstrate different searching and sorting techniques. **[Apply]**						
Course Content:							
Module 1	**Introduction to Data Structure and Linear Data Structure – Stacks and Queues**	Assignment	Program activity	**9 Hours**			
Introduction – Introduction to Data Structures, Types and concept of Arrays**.** **Stack -** Concepts and representation, Stack operations, stack implementation using array and Applications of Stack**.** **Queues -** Representation of queue, Queue Operations, Queue implementation using array, Types of Queue and Applications of Queue**.**							
Module 2	**Linear Data Structure- Linked List**	Assignment	Program activity	**12 Hours**			
Topics: Linked List - Singly Linked List, Operation on linear list using singly linked storage structures, Circular List, Applications of Linked list. **Recursion** - Recursive Definition and Processes.							
Module 3	**Non-linear Data Structures - Trees**	Assignment	Program activity	**12 Hours**			
Topics: Trees - Introduction to Trees, Binary tree: Terminology and Properties, Use of Doubly Linked List, Binary tree traversals: Pre-Order traversal, In-Order traversal, Post - Order traversal, Binary Serach Tree, AVL Trees - Red Black Tree, Expression Tree , Heaps.							
Module 4	**Non-linear Data Structures - Graphs and Hashing**	Assignment	Program activity	**6 Hours**			

Topics: Graphs: Basic Concept of Graph Theory and its Properties, Representation of Graphs . ADT, Elementary graph operations, Minimum Cost spanning trees, Shortest path and Transitive closure.
Hashing: Introduction, Static Hashing, Dynamic Hashing

Module 5	**Searching & Sorting**	Assignment	Program activity	**6 Hours**

Topic: Sorting & Searching - Sequential and Binary Search, Sorting – Selection and Insertion sort, Quick sort, Merge Sort, Bubble sort.

List of Laboratory Tasks:

Lab sheet -1
Level 1: Prompt the user, read input and print messages.Programs using class, methods and objects
Level 2: Programming Exercises on fundamental Data structure - Arrays based on Scenario.

Lab sheet -2
Level 1: Programming Exercises on Stack and its operations
Level 2: Programming Exercises on Stack and its operations with condition

Lab sheet -3
Level 1: Programming on Stack application infix to postfix Conversion
Level 2: -

Lab sheet -4
Level 1: Programming on Stack application – Evaluation of postfix

Lab sheet -5
Level 1: Programming Exercises on Queues and its operations with conditions
Level 2: -

Lab sheet -6
Level 1: Programming Exercises on Linked list and its operations.
Level 2: Programming Exercises on Linked list and its operations with various positions

Lab sheet -7
Level 1: Programming Exercises on Circular Linked list and its operations.
Level 2: Programming Exercises on Circular Linked list and its operations with various positions

Lab sheet -8
Level 1: Programming Exercises on factorial of a number
Level 2: Programming the tower of Hanoi using recursion

Lab sheet -9
Level 1: -
Level 2: Programming the tower of Hanoi using recursion

Lab sheet -10
Level 1: Programming Exercise on Doubly linked list and its operations
Level 2: -

Lab sheet -11
Level 1: Program to Construct Binary Search Tree and Graph
Level 2: Program to traverse the Binary Search Tree in three ways)in-order, pre-order and post-order(and implement BFS and DFS

Lab sheet -12
Level 1: Program to Implement the Linear Search & Binary Search

Level 2: Program to Estimate the Time complexity of Linear Search

Lab sheet -13

Level 1: Program to Implement and Estimate the Time complexity of Selection Sort

Level 2: Program to Implement and Estimate the Time complexity of Insertion Sort

Lab sheet -14 (Beyond syllabus activity)

Level 1: Program to Construct AVL Tree

Level 2:

Lab sheet -15 (Beyond syllabus activity)

Level 1: Program to Construct RED BLACK Tree

Targeted Application & Tools that can be used

Use of PowerPoint software for lecture slides and use of Modern IDE like VS Code and Eclipse for lab programs to execute.

Project work/Assignment:

Assignment: Students should complete the lab programs by end of each practical session and module wise assignments before the deadline.

Text Book

T1 Ellis Horowitz and Sartaj Sahni, Fundamentals of Data Structures in C, 2nd Edition, Universities Press, reprint 2018.

T2 Seymour Lipschutz, Data Structures Schaum's Outlines, Revised 1st Ed, McGraw Hill, 2014.

References

R1 Data structures and program design in C by Robert Kruse, Tondo C L, Bruce Leung, Pearson education publishers, 2017.

R2 Programming and Data Structure by Jackulin C Salini etal., Ane books publishers, 2019.

Web resources:

3. For theory: https://onlinecourses.nptel.ac.in/noc20_cs85/preview
4. https://puniversity.informaticsglobal.com/login

Topics relevant to development of "Skill Development":

Linked list and stacks

Topics relevant to development of "Environment and sustainability: Queues

Course Code: CSE2259	**Course Title: Web Technologies Lab** **Type of Course: Lab**		**L-T- P-C**	0	0	2	1

Version No.		1.0
Cours **e Pre- requisites**		**NIL**
Anti-requisites		**NIL**
Course Description		This course highlights the comprehensive introduction to scripting languages that are used for creating web-based applications. The associated laboratory provides an opportunity to implement the concepts and enhance critical thinking and analytical skills.
Course Objective		**The objective of the course is to familiarize the learners with the concepts of Web Technology and attain Skill Development through Experiential Learning techniques.**
Course Outcomes		**On successful completion of this course the students shall be able to:** **CO1:** Implement web-based application using client-side scripting languages. **(Apply)** **CO2**: Apply various constructs to enhance the appearance of a website. **(Apply)** **CO3:** Apply server-side scripting languages to develop a web page linked to a database. **(Apply)**
Course Content:		

Module 1	**Introduction to XHTML Features**	Quizzes and Assignments	Quizzes on various features of XHTML, simple applications		**8 Sessions**
	Standard XHTML Document Structure, Basic Text Markup such as headings, paragraphs, lists, tables, forms, and semantic tags.				
Module 2	**CSS Styling**	Quizzes and assignments	Comprehension based Quizzes and assignments; Application of CSS in designing webpages		**10 Sessions**
	Apply CSS3 to style HTML elements, including layout techniques, color schemes, typography, and responsive design principles.				

XML: Basics, Demonstration of applications using XML with XSLT.

Module 3	**PHP – Application Level**	Quizzes and assignments	Application of PHP in web designing	**12 Sessions**

PHP: Introduction to server-side Development with PHP, Arrays, Superglobal Arrays, $GET and $ POST, $_SERVER Array, $_Files Array, Reading/Writing Files, PHP Classes and Objects, Object Oriented Design, Working with Databases, SQL, Database APIs, Managing a MySQL Database. Accessing MySQL in PHP, Applications.

List of Laboratory Tasks:

Experiment No. 1: Demonstration of XHTML features

Level 1: Demonstration of various XHTML Tags (Level 1)
Level 2: Design and develop static web pages for an online Book store (Level 2).

Experiment No. 2: Application of CSS in web designing

Level 1: Design a document using XHTML and CSS to create a catalog of items for online electronic shopping.

Level 2: Create and save XML document for students' information and display the same using cascaded style sheet.

Experiment No. 3: Application of PHP in web designing.

Level 1: Write a PHP program to read the personal information of a person such as first name, last name, age, permanent address, and pin code entered by the user into a table created in MySQL. Read the same information from the database and display it on the front end.

Level 2: Using PHP develop a web page that accepts book information such as ISBN number, title, authors, edition, and publisher and store information submitted through the web page in MySQL database.

Experiment No. 4: Building a website.

Build a website for organizing an International Conference. The conference website must be able to collect the author's details and upload a file.

Targeted Application & Tools that can be used: **Xampp web server to be used to demonstrate PHP.**

Project work/Assignment:

Assignments are given after completion of each module which the student need to submit within the stipulated deadline.

Textbook(s):

1] Robert. W. Sebesta, "*Programming the World Wide Web"*, Pearson Education, 9th Edition, 2016.
2]Paul Deitel, Harvey Deitel, Abbey Deital,"Internet & World Wide Web How to Program", Fifth Edition, Pearson Education, 2021.

3]CSS Notes for Professionals, ebook available at https://books.goalkicker.com/CSSBook/ (Retrieved on Jan.

20, 2022)

4]Deitel, Deitel, Goldberg,"*Internet & World Wide Web How to Program*", Fifth Edition, Pearson Education, 2021.

Reference Book(s):

R1. Randy Connolly, Ricardo Hoar,"Fundamentals of Web Development", Pearson Education India, 1st. Edition.2016.

R2. Jeffrey C. Jackson,"Web Technologies: A Computer Science Perspective", Pearson Education, 1st Edition,2016.

Additional web-based resources

W1. W3schools.com
W2. Developer.mozilla.org/en-US/docs/Learn
W3. docs.microsoft.com
W4. informit.com/articles/ The Relationship Between Web 2.0 and Social Networking
https://presiuniv.knimbus.com/user#/home

Topics related to development of "FOUNDATION":

4. Web, WWW, Web browsers, Web servers, Internet.
5. CSS, PHP.
6. Designing the website for healthcare.

Course Code: **CSE2252**	**Course Title:** Data Communication and Computer Networks Lab **Type of Course: LAB**	**L- T-P- C**	0	**0**	2	1
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	**NIL**					
Course Description	This lab-based course provides hands-on experience in the principles and practices of data communications and computer networking. It is designed to complement theoretical concepts covered in the associated lecture course. Through a series of structured experiments and practical exercises, students will gain proficiency in configuring, analyzing, and troubleshooting computer networks. Key topics include network topology design, IP addressing and subnetting, Ethernet and LAN technologies, routing and switching, TCP/IP protocol suite, and basic network security measures. Students will work with industry-standard tools and equipment, including routers, switches, protocol analyzers, and network simulation software such as Cisco Packet Tracer or Wireshark.					
Course Objective	The objective of this lab course is to provide students with practical, hands-on experience in the configuration, operation, and troubleshooting of data communication systems and computer networks. Through guided experiments and real-world scenarios, students will reinforce theoretical knowledge, develop essential technical skills, and gain a deeper understanding of networking concepts, protocols, and devices used in modern communication systems.					
Course Outcomes	On successful completion of the course, the students shall be able to: 1. Design and configure basic network topologies using routers, switches, and end devices to meet specified requirements. Analyze and troubleshoot network connectivity and performance issues using tools such as Wireshark and network simulators. Demonstrate understanding of key networking protocols (e.g., TCP/IP, ARP, ICMP, DHCP) through practical implementation and observation. Apply IP addressing and subnetting techniques to efficiently allocate and manage network resources in various networking scenarios.					
Course Content:						
Module 1,2,3,4	Physical Layer, Network Layer, Transport Laye	Lab Assignment	Problem Solving	**24 Sessions**		

List of Laboratory Tasks:

Lab sheet -1, M-1, 3 [2 Hours]
Experiment No 1:
Level 1: Study of basic network commands and network configuration commands.

Lab sheet -2, M-1[2 Hours]
Experiment No 1:
Level 1: Identify and explore Network devices, models and cables. Introduction to Cisco packet tracer.

Experiment No. 2:
Level 2 – Create various network topologies using a cisco packet tracer.

Lab sheet -3, M-2,3 [2 Hours]
Experiment No. 1:
Level 2 - Basic Configuration of switch/router using Cisco packet tracer.
Experiment No. 2:
Level 2 -Configure the privilege level password and user authentication in the switch/router.

Lab sheet – 4, M-3 [2 Hours]
Experiment No. 1:
Level 2 - Configure the DHCP server and wireless router and check the connectivity

Lab sheet – 5, M-3 [2 Hours]
Experiment No. 1:
Level 2 - Configure the static routing in the Cisco packet tracer.
Experiment No. 2:
Level 2 - Configure the dynamic routing protocol in the Cisco packet tracer.

Lab sheet – 6, M-4 [2 Hours]
Experiment No. 1: Configuration of DNS Server with Recursive & Integrative approach in Cisco packet tracer.
Lab sheet – 7, M-4 [2 Hours]
Experiment No. 1:
Configure the telnet protocol in the router using the Cisco packet tracer.
Lab sheet – 8, M-4[2 Hours]
Experiment No. 1:
Level1- Introduction to NS2 and basic TCL program.
Lab sheet – 9, M-4 [2 Hours]
Experiment No. 1:
Level 1: Simulate three node Point to point network using UDP in NS2.
Experiment No. 2:
Simulate transmission of Ping message using NS2.
Lab sheet – 10, M-4[2 Hours]
Experiment No. 1:
Simulate Ethernet LAN using N-node in NS2.
Experiment No. 2:
Simulate Ethernet LAN using N-node using multiple traffic in NS2
Lab sheet –11, M-3,4 [2 Hours]
Experiment No. 1:
Level 1- Introduction to Wire Shark.
Experiment No. 2:
Level 2- Demonstration of packet analysis using wire shark.

Lab sheet –12, M-1,2,3 [2 Hours]
Experiment No. 1:
Level 2- Demonstration of switch and router configuration using real devices

Targeted Application & Tools that can be used: Cisco Packet Tracer, Wireshark, and NS2.
Case Study/Assignment: Choose and analyze a network from any organization/Assignment

proposed for this course in CO1-CO4
Problem Solving: Choose and appropriate devices and implement various network concepts.
Programming: Simulation of any network using NS2.

Text Book(s):

1. 1. Behrouz A. Forouzan, “Data Communications and Networking 5E”, 5 th Edition, Tata McGraw-Hill, 2017.
2. Andrew S Tanenbaum, Nick Feamster & David J Wetherall, “Computer Networks” Sixth Edition, Pearson Publication, 2022

Reference(s):

1.References

1. “Computer Networking: A Top-Down Approach”, Eighth Edition, James F. Kurose, Keith W. Ross, Pearson publication, 2021.
2. William Stallings, Data and Computer Communication, 8th Edition, Pearson Education, 2007.
3. Larry L. Peterson and Bruce S. Davie: Computer Networks – A Systems Approach, 4th Edition, Elsevier, 2007.

E- Resources:

1. https://archive.nptel.ac.in/courses/106/105/106105183/
2. http://www.nptelvideos.com/course.php?id=393
3. https://www.youtube.com/watch?v=3DZLItfbqtQ
4. https://www.youtube.com/watch?v=_fIdQ4yfsfM
5. https://www.digimat.in/keyword/106.html
6. https://puniversity.informaticsglobal.com/login

Course Code: CSE2260	**Course Title: Database Management Systems** **Type of Course: Theory**	**L-T-P-C**	3	0	0	3
Version No.						
Course Pre-requisites	NIL					
Anti-requisites	NIL					
Course Description	This course introduces the foundational principles of database management systems, including data models, schemas, and architectures. This course provides a solid foundation on the relational model of data and the use of relational algebra. It develops skills in SQL for data definition, manipulation, and control, enabling students to construct and execute complex queries. The course also introduces the concept of object oriented and object relational databases and modern database technologies like **NoSQL.** The also course allows the students to gain insights into data storage structures and indexing strategies for optimizing query performance.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of Database Management Systems and attain **Employability** through **Problem Solving** Methodologies.					
Course OutComes	On successful completion of the course the students shall be able to: 1. Describe the fundamental elements of relational database management systems. [Understand] 2. Examine databases using SQL query processing and Optimization. [Apply] 3. Design simple database systems applying the normalization constraints and demonstrate the database transaction processing, recovery, and security. [Apply] 4. Interpret the concept of advanced databases and its applications. [Apply]					

Course Content:

Module 1	**Introduction to Database Modelling and Relational Algebra(Understand)**	Assignment	**Problem Solving**	**10 Sessions**

Topics:
Introduction to Database: Schema, Instance, 3-shema architecture, physical and logical data independence, Data isolation problem in traditional file system, advantages of database over traditional file systems. Entity Relationship (ER) Model, ER Model to Relational Model, Examples on ER model.
Relational Algebra with selection, projection, rename, set operations, Cartesian product, joins (inner and outer joins), and division operator. Examples on Relational Algebra Operations.

Module 2	**Fundamentals of SQL and Query Optimization (Apply)**	Assignment	**Programming**	**11 Sessions**

Topics:
SQL Database Querying, DDL, DML, Constraints, Operators, Set Operators, Aggregate Functions, Joins, Views, Procedures, Functions and Triggers.
Database programming issues and techniques: Embedded SQL, Dynamic SQL; SQL / PSM and NoSQL.
Query Optimization: Purpose, transformation of relational expressions, estimating cost and statistics ofexpression, choosing evaluation plans, linear and bushy plans, dynamic programming algorithms.

Module 3	**Relational Database Design & Transaction Management (Apply)**	Assignment	**Problem Solving**	**12 Sessions**

Topics:
Relational database design: Problems in schema design, redundancy and anomalies, Normal Forms based on Primary Keys-(1NF,2NF, 3NF), Boyce-Codd Normal Form, Multi valued Dependency (Fourth Normal Form), JoinDependencies (Fifth Normal Form), lossy and lossless decompositions, Database De-normalization.
Transaction Management: The ACID Properties; Transactions and Schedules; Concurrent Execution of Transactions; Lock- Based Concurrency Control; Performance of locking; Transaction support in SQL; Introduction to crash recovery; 2PL, Serializability and Recoverability; Lock Management; The write-ahead log protocol; Check pointing; Recovering from a System Crash; Media Recovery; Other approaches and interaction with concurrency control.

Module 4	**Advanced DBMS Topics (Apply)**	**Assignment**	**Case Study**	**12 Sessions**

Topics:
Advanced topics: Object oriented database management systems, Deductive database management systems, Spatial database management systems, Temporal database management systems,Constraint database management systems.
New database applications and architectures such as Data warehousing, Multimedia, Mobility, NoSQL, NativeXML databases (NXD), Document-oriented databases, Statistical databases.

Targeted Application & Tools that can be used:
Application Area: Relational database systems for Business, Scientific and Engineering Applications.Tools/Simulator used: MySQL DB for student practice.
Also demonstration of ORACLE DB on object-relational database creation and JDBC connection.

1. Problem Solving: Constructing ER-Diagrams for a given real time requirements, Normalizing the databases, querying the databases using relational algebra.
2. Programming: Implementation of any given scenario using MySQL.

Text Books:
T1. Elmasri R and Navathe S B, “Fundamentals of Database System”, Pearson Publication, 7th Edition, 2018.
T2. RamaKrishna & Gehrke, “Database Management Systems” 3rd Edition, 2018, McGraw-Hill Education.
T3. W. Lemahieu, S. vanden Broucke and B. Baesens, “Principles of Database Management: Practical Guide to Storing, Managing and Analyzing Big and Small Data”, Cambridge University Press, 2018.

References
R1 Avi Silberschatz, Henry F. Korth, S. Sudarshan, “Database System Concepts”, McGraw-Hill ,7th Edition, 2019.
R2 M. Kleppmann, “Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems”, O’Reilly, 2017.

Topics relevant to development of “FOUNDATION SKILLS”: S - Skill Development: Relational database design using ER- Relational mapping, Implementation of given database scenario using MYSQLDB.
Topics relevant to development of Employability: Develop, test and implement computer databases, creating sophisticated, interactive and secure database applications
Topics relevant to “HUMAN VALUES &PROFESSIONAL ETHICS”: Nil

Course Code: CSE2261	Course Title: Database Management Systems Lab Type of Course: Lab	L-T-P-C	0	0	2	1
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	NIL					
Course Description	The Database Management Systems (DBMS) Laboratory is designed to provide students with hands-on experience in database design, implementation, and management using SQL and database management tools such as MySQL. The lab complements theoretical concepts learned in database courses by allowing students to practice database creation, querying, and optimization techniques. The DBMS Lab enables students to develop industry-relevant skills in database management, preparing them for careers in **software development, data engineering, and database administration**.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of DatabaseManagement Systems and attain **Employability** through **Problem Solving** Methodologies.					
Course OutComes	On successful completion of the course the students shall be able to: 5. Demonstrate the database concepts, practice, and SQL queries. [Apply] 6. Design and implement database schemas while applying normalization techniques to optimize structure. [Apply]] 7. Develop and implement stored procedures, triggers, and views for automation and efficiency. [Apply] 8. To Design and build database applications for real world problems. [Apply]					

Course Content:

List of Laboratory Tasks:
Create Employee, Student, Banking and Library databases and populate them with required data. Do the following experiments of different lab sheets on those databases.

Labsheet-1 [3 Practical Sessions]
Experiment No 1: [1 Session]
1. To study and implement the different language of Structured Query Language.
Level 1: Perform operations using Data Definition Language and Data Manipulation Language commands including different variants of SELECT on Student DB.
Level 2: Identify the given requirements; valid attributes and data types and Perform DDL and DML operations on a given scenario. [Banking Databases]
Experiment No. 2: [2 Sessions]
2. To study and implement the concept of integrity constraints in SQL.
Level 1: Create tables on Banking database using PRIMARY KEY, NOT NULL, UNIQUE, FOREIGN KEY and demonstrate the working of relational, logical, pattern matching, BETWEEN, IS NULL, IN and NOT IN Special Operators on Student Database.
Level 2: Enforce different types of data and referential integrity constraints. Then try queries with special operators based on the student database. [Banking Database].

Labsheet-2 [3 Practical Sessions]
Experiment No. 3: [1 Session]
3. Implement complex queries in SQL.
Level 1: Implement the conjugate of GROUP BY, ORDER BY and aggregate functions on Banking Database. Level 2: Implement MySQL DB queries on library database using appropriate clauses and aggregate functions. Also order the data either in ascending and descending order using corresponding clause. [Library databases].

Experiment No. 4: [2 Session]
4. To study and implement different types of Set and Join Operations [2 Slots]
Level 1: Demonstrate different types of Set Operations (UNION, UNION ALL, INTERSECT, MINUS) and Join Operations (INNER JOINs, OUTER JOINS, CROSS JOIN, NATURAL JOIN) on two or more tables of Airline Database. Level 2: Use Set and Join operations to retrieve the data from two or more relations(tables) as per the given

scenario. [Airline Database]

Labsheet-3 [2 Practical Sessions]
Experiment No. 5: [2 sessions]
5. To study and implement Views, and Procedures in MySQL DB.
Level 1: Implement MySQL Views, and Procedures in ORACLE DB on Employee database.
Level 2: Analyze the requirement and construct views, and Procedures on Mini Project Domain. [Banking Database]

Labsheet-4 [2 Practical Sessions]
Experiment No. 6: [2 Sessions]
6. To study and implement Functions, and Triggers in MySQL DB.
Level 1: Implement Oracle Functions and Triggers in Oracle on Employee database.
Level 2: Analyze the requirement and construct Functions and Triggers. [Supply chain Database]

Labsheet-5 [2 Practical Sessions]
Experiment No. 7: [2 Sessions]
7. To study and implement Functions, and Triggers in MySQL DB.
Level 1: Implement Oracle Functions and Triggers in Oracle on Employee database.
Level 2: Determine the requirement and construct Functions and Triggers. [Supply chain Database]

Labsheet-6 [4 Practical Sessions]
Experiment No. 8: [2 Sessions]
8. To implement the concept of forms and reports.
Level 1: Implement the concept of forms and reports.
Level 2: Examine the schema relationship.

Experiment No. 9: [2 Sessions]
9. Create the database using the given schema. (Flight Management)
Level 1: Implement a relational database based on the provided schema for the Flight Management system, including the creation of tables, relationships, and constraints.
Level 2: Demonstrate schema relationships by defining primary and foreign keys to ensure data integrity within the Flight Management database.

Labsheet-7 [4 Practical Sessions]
Experiment No. 10: [2 Sessions]
10. Create the database using the given schema. (Company database)
Level 1: Implement the database schema by defining tables, relationships, and constraints according to the given Company Database schema.
Level 2: Demonstrate the schema's relationships and data integrity by creating and linking tables as per the specified requirements.
Experiment No. 11: [2 Sessions]
11. Create the database using the given schema. (Student Library)
Level 1: Implement forms and reports based on the provided Student Library database schema, ensuring effective data entry and reporting mechanisms.
Level 2: Demonstrate the schema relationships within the Student Library database, demonstrating how these relationships influence the creation and functionality of forms and reports.

Labsheet-8 [1 Sessions]
12. Design a mini project based on the databases such as Inventory Management System, University Management System, Hospital Management System, etc.
Level 1: Implement the real time database.
Level 2: Analyze the working of database in real time.

Targeted Application & Tools that can be used:
Application Area: Relational database systems for Business, Scientific and Engineering Applications.Tools/Simulator used: MySQL DB for student practice.
Also demonstration of ORACLE DB on object-relational database creation and JDBC connection.

Percentage of changes in this version: 50% of changes from earlier version. New topics are highlighted initalic.

3. Problem Solving: Constructing ER-Diagrams for a given real time requirements, Normalizing the databases, querying the databases using relational algebra.
4. Programming: Implementation of any given scenario using MySQL.

Text Books:

T1. Elmasri R and Navathe S B, “Fundamentals of Database System”, Pearson Publication, 7th Edition, 2018.
T2. RamaKrishna & Gehrke, “Database Management Systems” 3rd Edition, 2018, McGraw-Hill Education.
T3. W. Lemahieu, S. vanden Broucke and B. Baesens, “Principles of Database Management: Practical Guide to Storing, Managing and Analyzing Big and Small Data”, Cambridge University Press, 2018.

References

R1 Avi Silberschatz, Henry F. Korth, S. Sudarshan, “Database System Concepts”, McGraw-Hill ,7th Edition, 2019.
R2 M. Kleppmann, “Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems”, O’Reilly, 2017.

Topics relevant to development of “FOUNDATION SKILLS”: S - Skill Development: Relational database designusing ER- Relational mapping, Implementation of given database scenario using MYSQLDB.

Topics relevant to development of Employability: Develop, test and implement computer databases, creatingsophisticated, interactive and secure database applications

Topics relevant to “HUMAN VALUES &PROFESSIONAL ETHICS”: Nil

Course Code: CSE2262	**Course Title: Analysis of Algorithms** **Type of Course: Theory**		**L- T-P- C**	3	1	0	4
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	Nil					
Course Description	This course introduces techniques for the design and analysis of efficient algorithms and methods of applications. This course discusses the classic approaches for algorithm design such as Divide and Conquer, Dynamic Programming, Greedy method. This course also describes other basic strategies searching solution space. The core concepts of analyzing algorithms and classifying them into various complexity classes is covered in the end.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Analysis of Algorithms** and attain **Skill Development** through **Problem Solving** Methodologies.					
Course Out Comes	On successful completion of the course the students shall be able to: 1. Compute efficiency of a given algorithm.[Apply] 2. Apply divide and conquer technique for searching and sorting Problems.[Apply] 3. Apply the Dynamic Programming technique for a given problem. [Apply] 4. Apply greedy technique for solving a Problem.[Apply] 5. Demonstrate Back tracking technique and limitations of Algorithms.[Apply]					
Course Content:						
Module 1	**Introduction**	Assignment	Simulation/Data Analysis	**10 Sessions**		
Introduction, Asymptotic Notations and its properties, Best case, worst case and average case- Sequential search, Sorting; Mathematical analysis for Recursive and Non-recursive algorithms: Substitution method and Master's Theorem.						
Module 2	**Divide-and-conquer**	Assignment	Simulation/Data Analysis	**08 Sessions**		
Introduction. Insertion Sort; Merge sort, Quick sort, Binary search.						
Module 3	**Dynamic programming**	Term paper/Assignment	Simulation/Data Analysis	**10 Sessions**		
Introduction with examples, Principles of Memoization, 0-1 Knapsack Problem, Bellman-Ford algorithm, Floyd-Warshall's Algorithms. Chain Matrix Multiplication.						
Module 4	**Greedy technique**	Term paper/Assignment	Simulation/Data Analysis	**09 Sessions**		
Introduction, Fractional Knapsack Problem, Minimal Spanning Tree: Prim's Algorithm and Kruskal's Algorithm, Single-source Shortest Path: Dijkstra's Algorithm						
Module 5	**Complexity Classes**	Term paper/Assignment	Simulation/Data Analysis	**08 Sessions**		
Complexity Classes- P,NP- NP Hard and NP Complete - Boolean Satisfiability Problem (SAT). Branch and Bound: Knapsack problem; Backtracking, - N-Queens problem.						

Text Book

1. Anany Levitin, *"Introduction to the Design and Analysis of Algorithms*", 3rd edition, Pearson Education, 2018.
2. Thomas H.Cormen, Charles E.Leiserson, Ronald L. Rivest and Clifford Stein, *"Introduction to Algorithms*", 4th edition, MIT Press, 2022.

References

1. J. Kleinberg and E. Tardos, "*Algorithm Design*", Addison-Wesley, 2005.
2. Tim Roughgarden, *"Algorithms Illuminated"* (books 1 through 3), "Operating Systems Design and Implementation", Soundlikeyourself Publishing, 2017-2019.
3. AV Aho, J Hopcroft, JD Ullman, *"The Design and Analysis of Algorithms"*, Addison-Wesley, 1974.
4. Donald E. Knuth, "*The Art of Computer Programming",* Volumes 1and 3 Pearson.

Web-Resources

1. NPTEL: https://onlinecourses.nptel.ac.in/noc19_cs47/preview
2. Coursera: Analysis of Algorithms by Princeton University
3. Algorithms Specialization in Coursera by Stanford University(Group of 4 courses).
4. Algorithms Coding Contest Links maintained by Prof Gerth Stølting Brodal of Aarhus University

Topics relevant to "SKILL DEVELOPMENT": knapsack, prim's, kruskal's algorithm, quick sort, binary search for **Skill Development** through **Problem Solving methodologies**. This is attained through assessment component mentioned in course handout.

Course Code: CSE2263	**Course Title:** **Analysis of Algorithms Lab** **Type of Course: Lab**	**L- T-P- C**	0	0	2	1
Version No.	1					
Course Pre-requisites	**NIL**					
Anti-requisites	NIL					
Course Description	This course introduces techniques for the design and analysis of efficient algorithms and methods of applications. This course discusses the classic approaches for algorithm design such as Divide and Conquer, Dynamic Programming, Greedy method. This course also describes other basic strategies searching solution space. The core concepts of analyzing algorithms and classifying them into various complexity classes is covered in the end.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Analysis of Algorithms** and attain **Skill Development** through **Experiential Learning** Methodologies.					
Course Out Comes	**On successful completion of the course the students shall be able to:** 1. Compute efficiency of a given algorithm. [Applying] 2. Apply divide and conquer technique for searching and sorting Problems. [Applying] 3. Apply the Dynamic Programming technique for a given problem. [Applying] 4. Apply greedy technique for solving a Problem. [Applying] 5. Demonstrate Back tracking technique and limitations of Algorithms. [Applying]					
Course Content						
Module 1	**Introduction**					**3 Sessions**
Measuring running time of an algorithm, Compare running time of algorithms, Implement sorting algorithms such as bubble sort, selection sort						
Module 2	**Divide-and-conquer**					**3 Sessions**
Compare searching algorithms: Linear Search, Binary Search; Compare Sorting algorithms: Insertion Sort, Merge Sort, QuickSort.						
Module 3	**Dynamic programming**					**3 Sessions**
Introduction and memorization: Factorial; Coin Change Problem ; Floyd-Warshall's Algorithm.						
Module 4	**Greedy technique**					**3 Sessions**
Fractional Knapsack Problem; Minimal Spanning Tree Algorithms-Prim's Algorithm, Kruskal's algorithm						
Module 5	**Complexity Classes**					**3 Sessions**
Branch and Bound: Knapsack problem; Backtracking, - N-Queens problem.						
List of Laboratory Tasks: **1. Measuring running time of an algorithm** **Objective: To experimentally determine the running time of basic algorithms for input size n=10, 100, 1000, etc. by taking difference of starting time and ending time.** **2. Compare running time of algorithms** **Objective: To execute two algorithms to solve the same problem, and to comparatively evaluate the better algorithm for large values of N.** **3. Implement sorting algorithms such as bubble sort, selection sort** **Objective: To implement comparison based sorting strategies.** **4. Compare searching algorithms** **Objective: To implement two searching strategies and compare their performance.** **5. Compare Sorting algorithms** **Objective: To implement searching strategies that follow top down design approach(Insertion sort, merge sort).**						

6. Quick Sort
Objective: To demonstrate Quick sort and its variants, and their impact on running time.
7. Dynamic Programming
Objective: To demonstrate Dynamic Programming approach with the help of Factorial algorithm.
8. Coin Change Problem
Objective: To implement an efficient algorithm for the Coin Change problem.
9. Floyd-Warshall's Algorithm
Objective: To demonstrate how dynamic programming is used with the help of Floyd-Warshall's algorithm.
10. Fractional Knapsack Problem
Objective: To demonstrate how greedy method can be used to solve the Fractional Knapsack Problem.
11. Minimal Spanning Tree Algorithm
Objective: To implement greedy strategy to solve the Minimal Spanning Tree problem using Prim's Algorithm.
12. Kruskal's Minimal Spanning Tree Algorithm
Objective: To implement greedy strategies to solve the Minimal Spanning Tree problem using Kruskal's Algorithm.
13. Knapsack Problem
Objective: To implement Knapsack problem using branch and bound technique.
14. N-Queen's Problem
Objective: To demonstrate backtracking method with the help of N-Queen's problem.
15. Case Study
Objective: To demonstrate how various techniques can be used to solve the same problem with the help of Knapsack problem.

Targeted Application & Tools that can be used

PyTorch/Jupyter Notebook – For Python programming

Text Book

T1 Anany Levitin, *"Introduction to the Design and Analysis of Algorithms*", 3rd edition, Pearson Education, 2018.

T2 Thomas H.Cormen, Charles E.Leiserson, Ronald L. Rivest and Clifford Stein, *"Introduction to Algorithms*", 4th edition, MIT Press, 2022.

References

R1. J. Kleinberg and E. Tardos, "*Algorithm Design*", Addison-Wesley, 2005.
R2. Tim Roughgarden, *"Algorithms Illuminated"* (books 1 through 3), "Operating Systems Design and Implementation", Soundlikeyourself Publishing, 2017-2019.
R3. AV Aho, J Hopcroft, JD Ullman, *"The Design and Analysis of Algorithms"*, Addison-Wesley, 1974.
R4. Donald E. Knuth, "*The Art of Computer Programming",* Volumes 1and 3 Pearson.

Web **Based Resources and E-books:**
W1. NPTEL: https://onlinecourses.nptel.ac.in/noc19_cs47/preview
W2. Coursera: Analysis of Algorithms by Princeton University
W3. Algorithms Specialization in Coursera by Stanford University(Group of 4 courses).
W4. Algorithms Coding Contest Links maintained by Prof Gerth Stølting Brodal of Aarhus University

Topics relevant to "EMPLOYABILITY SKILLS": The lab experiments and assessments enable the student to acquire Skill Development through Experiential Learning techniques

Course Code: CSE2264	**Course Title:** Essentials of AI **Type of Course:** Theory	**L- T-P- C**	3	0	0	3
Version No.	2.0					
Course Pre-requisites	NIL					
Anti-requisites	**NIL**					
Course Description	This course is a comprehensive introductory course designed to equip learners with the fundamental Python programming skills necessary to work with artificial intelligence (AI) technologies. This course is aimed at individuals who are new to AI but have a basic understanding of programming concepts. It combines Python programming fundamentals with hands-on experience in implementing AI techniques such as machine learning, neural networks, and natural language processing.					
Course Objective	The objective of the course is to Understand Python Programming Fundamentals, Manipulate and Process Data with Python, Implement Machine Learning Algorithms and Build and Train Neural Networks for AI Applications.					
Course Outcomes	On successful completion of the course the students shall be able to: CO 1: Apply Python Programming to AI Projects CO 2: Build and Train Machine Learning Models CO 3: Develop Deep Learning Models with Neural Networks CO 4: Deploy AI Solutions and Understand Ethical Implications					
Course Content:						
Module 1	Introduction to Python Programming for AI	Assignment	Implementation	**10 Sessions**		
Topics: Python Basics: Variables, Data Types, Operators, and Control Flow Functions, Loops, and Conditionals statements, Data Structures: Lists, Tuples, Dictionaries, Sets ,Introduction to Libraries: NumPy and Pandas for data manipulation, Basic Input/Output and File Handling Introduction to Python for AI: Libraries and Frameworks Overview						
Module 2	Data Processing, Visualization	Assignment	Implementation	10 Sessions		
Topics: cleaning and preprocessing with Pandas,Handling missing data, outliers, and duplicates, Data transformation (Normalization, Encoding), Introduction to Matplotlib and Seaborn for Data Visualization, Exploratory Data Analysis (EDA), Visualizing datasets to understand patterns and relationships.						
Module 3	Introduction to Machine Learning	Mini - Project	Implementation	10 Sessions		
Topics: What is Machine Learning? Types of ML algorithms Supervised Learning: Regression, Classification, Unsupervised Learning: Clustering, Key ML Algorithms: Linear Regression, Decision Trees, K-Means ,Introduction to Scikit-learn library Model evaluation (Accuracy, Precision, Recall, Confusion Matrix)						
Module 4	Neural Networks and Deep Learning	Quiz	Implementation	10 Sessions		

Topics:
Introduction to Neural Networks and Deep Learning, Perceptron Model and Backpropagation Deep Neural Networks and Activation Functions, Introduction to TensorFlow and Keras, Building and Training Neural Networks for Image and Text Classification, Overview of Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs)

Targeted Application & Tools that can be used:

Applications:

- **Data Preprocessing**: Clean and manipulate data from various sources such as CSV, Excel, SQL databases, and APIs.
- **Exploratory Data Analysis (EDA)**: Gain insights into datasets by identifying trends, patterns, and outliers.
- **Predictive Modeling**: Build models for classification (e.g., spam detection) and regression (e.g., house price prediction).
- **Clustering**: Group data into clusters for unsupervised learning tasks (e.g., customer segmentation).
- **Model Evaluation**: Assess model performance using appropriate metrics such as accuracy, precision, recall, and F1-score.

Tools:

1. **Pandas**: For data manipulation and cleaning (e.g., handling missing values, merging datasets).
2. **NumPy**: For numerical operations and working with arrays and matrices.
3. **Matplotlib**: For creating static, animated, and interactive visualizations.
4. **Seaborn**: For advanced data visualizations (e.g., heatmaps, pair plots).
5. **Plotly**: For creating interactive visualizations, especially useful for large datasets.
6. **Scikit-learn**: The go-to library for implementing machine learning algorithms (e.g., linear regression, decision trees, k-means clustering).
7. **XGBoost**: For advanced gradient boosting models, particularly for large-scale machine learning tasks.
8. **TensorFlow** (for deep learning in Module 4): A powerful open-source library for building machine learning and deep learning models.
9. **Keras**: High-level neural network API, built on top of TensorFlow, to easily create deep learning models.

NLTK: The Natural Language Toolkit for various text processing tasks like tokenization, stemming, and part-of-speech tagging.
spaCy: A fast NLP library for advanced NLP tasks such as named entity recognition and dependency parsing.
Transformers (by Hugging Face): A powerful library for using pre-trained Transformer-based models like BERT, GPT, and others for advanced NLP tasks.

Text Book(s):

T1: **Essentials of Python for Artificial Intelligence and Machine Learning** by Pramod Gupta and Anupam Bagchi

Reference(s):

- "Artificial Intelligence with Python" – Prateek Joshi
- "Python Machine Learning" – Sebastian Raschka & Vahid Mirjalili
- "Hands-On Artificial Intelligence with Python" – Teet Straus
- "Deep Learning for Coders with Fastai and PyTorch" – Jeremy Howard & Sylvain Gugger

urse Code: CSE1700	**Course Title: Essentials of Artificial Intelligence** **ype of Course: Engineering Sciences Core - Theory**	-P-C	3			3
rsion No.						
urse Pre-requisites	CSE1508 – Data Structures					
ti-requisites						
urse Description	This course introduces the student to the basics of artificial intelligence. In this course, the student first learns the various search methods for problem-solving, followed by knowledge-based logic representations. After that, the student will learn about uncertainty in AI, as well as approaches to solve such challenges such as Naïve Bayes Classifier and Hidden Markov Models. Topics: Uninformed search, Heuristic search, Local search, Adversarial search, Constraint satisfaction, logic, First Order Resolution, Probability, Naïve Bayes Classifier, and Hidden Markov Model (HMM).					
urse Objectives	The objective of the course is EMPLOYBILITY of student by using EXPERIENTIAL LEARNING techniques.					
urse OutComes	On successful completion of this course the students shall be able to: 1. Explain different methods of searching, proving, and analysis in AI [Understand] 2. Implement various graphical and adversarial search algorithms. [Apply] 3. Prove, by resolution, different situations using First Order Logic [Apply] 4. Solve sequence labeling problems using HMM [Apply]					

Course Content:

dule 1	arch Methods for Problem-Solving	oblem-Solving Tests	TEL Assignments	No. of Sessions: 13
Introduction – State Space Search; General Formulation of Search Problems; Data Structures used in Searching. **Uninformed Search Algorithms** – Breadth First Search, Depth First Search, Uniform Cost Search, Generalized Uniform Cost Search (a.k.a Dijkstra's Single-Source Shortest Path), Iterative Deepening Depth-First Search, Time and Space Complexity Analysis of Uninformed Search Algorithms. **Heuristic Search Algorithms** – Heuristics and Admissibility, Greedy Best-First Search, A* Search and weighted A* Search.				
dule 2	vanced Search Methods	oblem-Solving Tests	TEL Assignments	No. of Sessions: 12
Local Search – Local Search, Hill Climbing, Genetic Algorithms, Gradient Descent. **Adversarial Search** – Minimax Search, Alpha-Beta Pruning, Ideal Ordering. **Constraint Satisfaction** – Constraint Satisfaction Problems Definitions and Examples – Map Colouring, N Queens, Cryptarithmetic, Generalized CSP; Back-tracking Heuristics; Arc Consistency and Path Consistency				
dule 3	owledge-Based Logic Representation	tomated Theorem Proving using FOL Resolution	TEL Assignments	No. of Sessions: 10
Propositional Logic – Syntax and Semantics of Propositional Logic. Logical connectives. Inference Rules. Conjunctive and Disjunctive Normal Forms. **First Order Logic** – Syntax and Semantics of Propositional Logic. Logical connectives. Inference Rules. Conjunctive and Disjunctive Normal Forms. **Resolution** – Resolution Principle. Propositional and First Order Resolution. Applications for solving				

story problems using Resolution				
dule 4	**certainty in AI**	**presenting problems as HMM**	**TEL Assignments**	**No. of Sessions: 06**

Probability – Probability Definitions. Conditional Probability. Bayes Theorem. Naïve Bayes Classifier. Using Naïve Bayes Classifier for Supervised Learning. **Hidden Markov Models** – Definition of HMM. Sequence Labeling and Markov Assumption. Sub-Problems in HMM and their solutions – Forward Probability and Viterbi Algorithm. Applications of Sequence Labeling in Natural Language Processing (Eg. Part-of-Speech Tagging). **Introduction to Deep Learning** – Artificial Neurons, Activation Functions, Multilayer Perceptron.

Targeted Application & Tools that can be used:

1. Implementation of a shortest-path finder using different search algorithms.
2. Implementation of a sequence labeler using Viterbi Algorithm.

ject work/Assignment: Mention the Type of Project /Assignment proposed for this course

1. Group project on one of the topics mentioned above (Eg. Adversarial search).

xtbook(s):

1. Stuart Russel and Peter Norvig. *Artificial Intelligence: A Modern Approach*. 4th Edition. Pearson Education. 2022.
2. Lavika Goel. *Artificial Intelligence: Concepts and Applications*. 1st Edition. Wiley. 2021.
3. Elaine Rich, Kevin Knight and Shivashankar B Nair. *Artificial Intelligence*. 4th Edition. MedTech Science Press. 2024.

ferences:

1. Deepak Khemani. *A First Course in Artificial Intelligence*. 1st Edition. 6th Reprint, 2018.
2. Munesh Chandra Trivedi. *A Classical Approach to Artificial Intelligence*. 2nd Edition. Khanna Publishers. 2018.
3. George Luger. *Artificial Intelligence: Structures and Strategies for Complex Problem Solving*. 6th Edition. Pearson Education. 2021.

eblinks

1. NPTEL Courses: Mausam (IIT Delhi), "An Introduction to Artificial Intelligence" Link: https://nptel.ac.in/courses/106102220.
2. Shyamanta M. Hazarika (IIT Guwahati), "Fundamentals of Artificial Intelligence". Link: https://nptel.ac.in/courses/112103280. Useful for the full course.
3. Deepak Khemani (IIT Madras), "Artificial Intelligence: Search Methods for Problem-Solving". Link: https://nptel.ac.in/courses/106106226. Useful for Module 1 and 2
4. Deepak Khemani (IIT Madras), "Artificial Intelligence: Knowledge Representation and Reasoning". Link: https://nptel.ac.in/courses/106106140. Useful for Module 3.
5. Deepak Khemani (IIT Madras), "AI: Constraint Satisfaction". Link: https://nptel.ac.in/courses/106106158. Useful for Module 2.

Course Code: CSE2265	**Course Title:** Essentials of AI Lab **Type of Course:** Lab	**L-T-P-C**	0	0	2	1
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	NIL					
Course Description	This course introduces the student to the basics of artificial intelligence. In this course, the student first learns the various search methods for problem-solving, followed by knowledge-based logic representations. After that, the student will learn about uncertainty in AI, as well as approaches to solve such challenges such as Naïve Bayes Classifier and Hidden Markov Models. Topics: Uninformed search, Heuristic search, Local search, Adversarial search, Constraint satisfaction, logic, First Order Resolution, Probability, Naïve Bayes Classifier, and Hidden Markov Model (HMM).					
Course Objectives	The objective of the course is EMPLOYBILITY of student by using EXPERIENTIAL LEARNING techniques.					
Course OutComes	On successful completion of this course the students shall be able to: 1. Explain different methods of searching, proving, and analysis in AI [Understand] 2. Implement various graphical and adversarial search algorithms. [Apply] 3. Prove, by resolution, different situations using First Order Logic [Apply] 4. Solve sequence labeling problems using HMM [Apply]					

Course Content: **No. of Sessions: 30 (60 hours)**

NOTE: Each experiment will be run across **TWO** lab sessions. In the first lab session, the students will have to do a preliminary experiment (Eg. Implement an uninformed search algorithm like BFS). In the second lab session, they will have to test their solution using a given input which is read from a file.

Experiment No. 1: File Handling
Level 1: Read text files using Python
Level 2: Parse text files using Python

Experiment No. 2: Implementation of Graph Representations
Level 1: Implement graph representations by taking input from the console
Level 2: Implement graph representations by taking input from files.

Experiment No. 3 & 4: Implementation of Uninformed Search Algorithms
Level 1: Implement uninformed search algorithms – BFS and DFS – on unweighted graphs.
Level 2: Implement uninformed search algorithms – Uniform Cost Search and Dijkstra's SSSP – on weighted graphs

Experiment No. 5: Implementation of Heuristic Search Algorithms
Level 1: Calculate the upper-bounds of admissible heuristics using Dijkstra's SSSP.
Level 2: Implement Greedy Best-First Search and A* Search Algorithms.

Experiment No. 6 & 7: Implementation of Adversarial Search
Level 1: Implement a Game Tree
Level 2: Perform Alpha-Beta Pruning and Ideal Ordering

Experiment No. 8 & 9: Implementation of a CSP Solver
Level 1: Implement a CSP solver to solve a cryptarithmetic problem
Level 2: Implement a CSP solver for map colouring

Experiment No. 10: Using Python Packages for CSP
Level 1: Implement a CSP solver for Sudoku
Level 2: Implement a CSP solver for Addoku

Experiment No. 11: Implement a Family Tree Parser
Level 1: Perform logic programming using logpy.
Level 2: Implement a family tree parser

Experiment No. 12 & 13: Implement a Decision Maker
Level 1: Implement a Minesweeper solver
Level 2: Implement a Battleship solver

Experiment No. 14 & 15: Hidden Markov Model
Level 1: Implement a generic HMM
Level 2: Build a PoS Tagger using a HMM with the Brown Corpus and the Universal Dependencies Tagset.

Targeted Application & Tools that can be used:

1. Google Colab
2. Python IDEs like PyCharm

Project work/Assignment: Mention the Type of Project /Assignment proposed for this course

The course is a lab-based course with all the assessments centrally evaluated. Every experiment consists of **two sessions**. The first session involves exploring a solution to the problem. The second session involves solving a particular problem.

Textbook(s):

1. Stuart Russel and Peter Norvig. *Artificial Intelligence: A Modern Approach*. 4th Edition. Pearson Education. 2022.
2. Prateek Joshi and Alberto Artasanchez. *Artificial Intelligence with Python*. 2nd Edition. Packt. 2020.

References:

1. Deepak Khemani. *A First Course in Artificial Intelligence*. 1st Edition. 6th Reprint, 2018.
2. Munesh Chandra Trivedi. *A Classical Approach to Artificial Intelligence*. 2nd Edition. Khanna Publishers. 2018.

Course Code: **CSE2500**	**Course Title: Data Analytics** **Type of Course: Theory**		**L-T-P- C**	2	0	0	2
Version No.	**1.0**					
Course Pre-requisites	**MAT1003**					
Anti-requisites	**NIL**					
Course Description	**Fundamentals of Data Analytics is designed for inspecting, cleansing, transforming, and modeling data with the goal of discovering useful information, and supports in decision-making. The course begins by covering Data extraction, pre-processing, and transformation. It delivers the basic statistics and taught in an intuitive way to analysis the data. This course will help the students to apply the knowledge on data analysis to a wide range of applications.**					
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Fundamentals of Data Analytics** and attain **SKILL DEVELOPMENT** through **PROBLEM SOLVING** Methodologies.					
Course Out Comes	**On successful completion of this course, the students shall be able to:** **CO1**:Describe different types of data and variables. **CO2**: Explain data using appropriate statistical methods. **CO3**: Demonstrate the collection, processing and analysis of data for any given application and illustrate various charts using visualization methods. **CO4**: Apply the Data Analysis techniques by R Programming					
Course Content:						
Module 1	**Introduction to Data Analysis- CO1**	Assignment	Data Collection, data analysis, Programming	**06 classes**		
Topics: Introducing Data, overview of data analysis: Data in the Real World, Data vs. Information, The Many "Vs" of Data, Structured Data and Unstructured Data, Types of Data, Data Analysis Defined, Types of Variables, Central Tendency of Data, Scales of Data, Sources of Data. Data preparation. **R Studio:** Base R-R Studio IDE-Introduction to R Projects and R Markdown. Basic R: R as a calculator-Scripts and Comments-R Variables. Data I/O: Working Directories-Importing Data Exporting Data-More ways to save-Data I/O in Base R.						
Module 2	**Data Analysis and Visualization-CO2**	Case studies	Programming	**10 classes**		
Topics: Data Summarization**:** One Quantitative and Categorical Variable. Data Classes: One Dimensional Data Classes-Data Frames and Matrices-Lists. Data Cleaning: Dealing with Missing Data-Strings and Recoding Variables. Manipulating Data in R: Reshaping Data-Merging Datasets. Data Visualizations: Plotting with ggplot2- Plotting with Base R						
Module 3	**Statistical Analysis -CO3**	Case studies	R programming	**7 classes**		
Topics: Proportion tests-Chi squared test-Fisher exact test-Correlation-T test-Wilcoxon Rank sum tests-Wilcoxon signed rank test- one-way ANOVA test- Kruskal Wallis test						
Module 4	Predictive Analysis-CO4	Case studies	Programming	**7 classes**		

Topics: Linear least-squares – implementation – the goodness of fit – testing a linear model – weighted resampling. Regression using Stats models – multiple regression – nonlinear relationships – logistic regression – estimating parameters – accuracy. Time series analysis – moving averages – missing values – serial correlation – autocorrelation. Introduction to survival analysis

Targeted Application & Tools that can be used:

Application Area are Decision making in business, health care, financial sector, Medical diagnosis etc.

Text Books

1. Glenn J. Myatt and Wayne P. Johnson, "Making Sense of Data I: A Practical Guide to Exploratory Data Analysis and Data Mining Paperback", Import, 22 July 2014.
2. Introduction to statistics and Data analytics, Christian H, Michael S, Springer,2016
3. Introduction to R- Robert Parker, John Mushcelli and Andrew Jaffe, Johns Hopkins University, 2020 (E-resource)
4. Introduction to Time Series and Forecasting (Springer Texts in Statistics), Peter Brockwell, Richard A. Davis, Springer, 2016.

References

1. Making Sense of Data I: A Practical Guide to Exploratory Data Analysis and Data Mining Paperback, Glenn J. Myatt and Wayne P. Johnson, Import, 22 July 2014.
2. The R Software-Fundamentals of Programming and Statistical Analysis -Pierre Lafaye de Micheaux, Remy Drouilhet, Benoit Liquet, Springer 2013.

Online resources:
http://www.modernstatisticswithr.com/solutions.html#solutionsch3
https://johnmuschelli.com/intro_to_r/
https://users.phhp.ufl.edu/rlp176/Courses/PHC6089/R_notes/

Topics relevant to development of "FOUNDATION SKILLS":

1. Statistical Concepts for data, visualization techniques.
2. Data collection for project based assignments.
3. Inferential Statistics (T test, Z test)
4. Probability Calculation

for Skill Development through Problem Solving methodologies. This is attained through assessment component mentioned in course handout.

Course Code: CSE2501	**Course Title: Data Analytics Lab** **Type of Course: Lab**	**L-T-P- C**	0	0	2	1
Version No.	**1.0**					
Course Pre-requisites	**MAT1003**					
Anti-requisites	**NIL**					
Course Description	**Fundamentals of Data Analytics is designed for inspecting, cleansing, transforming, and modeling data with the goal of discovering useful information, and supports in decision-making. The course begins by covering Data extraction, pre-processing, and transformation. It delivers the basic statistics and taught in an intuitive way to analysis the data. This course will help the students to apply the knowledge on data analysis to a wide range of applications.**					
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Fundamentals of Data Analytics** and attain **SKILL DEVELOPMENT** through **PROBLEM SOLVING** Methodologies.					
Course Out Comes	**On successful completion of this course, the students shall be able to:** **CO1**:Describe different types of data and variables. **CO2**: Explain data using appropriate statistical methods. **CO3**: Demonstrate the collection, processing and analysis of data for any given application and illustrate various charts using visualization methods. **CO4**: Apply the Data Analysis techniques by R Programming					
Course Content:						
Module 1	**Introduction to Data Analysis-**	Assignment	Programming		**09 classes**	

List of Laboratory Tasks:

Experiment No. 1: Introduction to R and RStudio

Level 1: Getting Started with R and RStudio

- Installing R and RStudio.
- Basic R syntax and commands.

Level 2: Working with RStudio

- Understanding the RStudio interface.
- Creating and managing R scripts.

Experiment No. 2: Basic Data Handling in R

Level 1: Data Types and Structures in R

- Vectors, matrices, and data frames.
- Lists and factors.

Level 2: Data Import and Export

- Reading data from CSV, Excel, and text files.
- Exporting data to different formats.

Level 3: Exploring Datasets

- Using functions like head(), summary(), and str().

Experiment No. 3: Basic Data structure in R

Level 1: a. Demonstrate a program to join columns and rows in a data frame using cbind() and rbind() in R.

b. Implement different data structures in R (Vectors, Lists, Data Frames)

Level 2: R AS CALCULATOR APPLICATION a. Using with and without R objects on console

a. Using mathematical functions on console
b. Write an R script, to create R objects for the calculator application

Module 2	**Data Analysis and Visualization -CO2**	Assignment	Programming	**13 classes**

Experiment No. 1: Data Cleaning and Preprocessing

Level 1: Handling Missing Data in R
- Identifying missing values.
- Imputing missing values using mean, median, or other methods.

Level 2: Data Transformation in R
- Standardizing and normalizing data.
- Log-transformations and scaling.

Experiment No. 2: Exploratory Data Analysis (EDA) with R

Level 1: Descriptive Statistics
- Calculating mean, median, and standard deviation.
- Visualizing data using histograms, box plots, and scatter plots.

Experiment No. 3: Data Visualization with ggplot2
Level 1: Demonstrate various graphs that can be made and altered using the ggplot2 package.
Level 2: Create 500 random temperature readings for six cities over a season and then plot the generated data using ggplot2 packages in R

Module 3	**Statistical Analysis -CO3**	Assignment	programming	**10 classes**

Experiment No. 1: Perform Tests of Hypotheses hypothesis test (parametric)

Level 1: How to perform tests of hypotheses about the mean when the variance is known. How to compute the p-value. Explore the connection between the critical region, the test statistic, and the p-value.
Level 2: A teacher claims that people who work for only five hours per week will score significantly lower than people who work for ten hours per week on a quantitative abilities test. He brings twenty people and randomly assigned them to one or two groups. In one group he has participants who work for ten hours and in another group, he has participants who work for five hours. He conducts the test for all participants. Scores on the test range from one to ten with higher scores representing better performance. Test if there is any significant difference between those who work for five hours per week versus those who work for ten hours per week based on the test performance.

Experiment No 2: Hypothesis – Non-Parametric Test
Level 1: A car manufacturing company like to find the sales of three types of cars produced by them in three regions and is given. Test if there is an association between the regions and types of cars purchased.
Experiment No 3: Correlation and Covariance

Level 1: Using the iris data set in R
a. Find the correlation matrix.
b. Plot the correlation plot on dataset and visualize giving an overview of relationships among data on iris data.
c. Analysis of covariance: variance (ANOVA), if data have categorical variables on iris data.

Level 2 : Ramesh is doing a statistics paper in his post-graduation course. He met his friend Amal who is a textile engineer. Ramesh, who is doing his internship at ABC Researchers, is interested in a question. He poses this question to Amal and tries to find if he can answer. The question is as follows: The data regarding sales of soft- drinks and sales of cotton clothes in a place during the last 12 months are given. Find if there

is any association between sales of soft drinks and sales of cotton clothes. Also explain the reason if there is any relationship.

Module 4	Predictive Analysis -CO4	Assignment	Programming	**10 classes**

Experiment No 1: Regression Model

Level 1: Import data from web storage (http://www.ats.ucla.edu/stat/data/binary.csv). Name the dataset and now do Logistic Regression to find out the relation between variables that are affecting the admission of a student in an institute based on his or her GRE score, GPA obtained, and rank of the student. Also check the model is fit or not. Require (foreign), require (MASS).

Level 2: Demonstrate multiple regressions, if data have a continuous Independent variable. Apply on the above dataset

Experiment No. 2: Time Series Analysis in R
Level 1: Demonstrate Timeseries analysis using Time Series Data Library at http://robjhyndman.com/TSDL/.

Targeted Application & Tools that can be used:
Application Area are Decision making in business, health care, financial sector, Medical diagnosis etc.

Text Books

1. Glenn J. Myatt and Wayne P. Johnson, "Making Sense of Data I: A Practical Guide to Exploratory Data Analysis and Data Mining Paperback", Import, 22 July 2014.
2. Introduction to statistics and Data analytics, Christian H, Michael S, Springer,2016
3. Introduction to R- Robert Parker, John Mushcelli and Andrew Jaffe, Johns Hopkins University, 2020 (E-resource)
4. Introduction to Time Series and Forecasting (Springer Texts in Statistics), Peter Brockwell, Richard A. Davis, Springer, 2016.

References

1. Making Sense of Data I: A Practical Guide to Exploratory Data Analysis and Data Mining Paperback, Glenn J. Myatt and Wayne P. Johnson, Import, 22 July 2014.
2. The R Software-Fundamentals of Programming and Statistical Analysis -Pierre Lafaye de Micheaux, Remy Drouilhet, Benoit Liquet, Springer 2013.

Online resources:
http://www.modernstatisticswithr.com/solutions.html#solutionsch3
https://johnmuschelli.com/intro_to_r/
https://users.phhp.ufl.edu/rlp176/Courses/PHC6089/R_notes/

Topics relevant to development of "FOUNDATION SKILLS":

2. Statistical Concepts for data, visualization techniques.
3. Data collection for project based assignments.
4. Inferential Statistics (T test, Z test)
5. Probability Calculation

for Skill Development through Problem Solving methodologies. This is attained through assessment component mentioned in course handout.

Course Code: **CSE2503**	**Course Title:** Scalable Application Development using Java **Type of Course:** Lab	**L- T-P- C**	3	**0**	0	**3**
Version No.	2.0					
Course Prerequisites	CSE1006					
Anti-requisites	**NIL**					
Course Description	This course provides a comprehensive guide to designing, developing, and deploying scalable Java applications, covering high-performance architectures, distributed systems, microservices, cloud deployment, and DevOps integration.					
Course Objective	The objective of the course is to Understand Scalability Principles, developing high performance Java applications, design and implement scalable architecture, deploy and manage scalable data.					
Course Outcomes	On successful completion of the course the students shall be able to: CO 1: Design and Develop Scalable Java Applications CO 2: Design and Implement Scalable Architectures and Micro services CO 3: Implement Scalable Data Management Techniques CO 4: Deploy and Monitor Applications in Cloud Environments					
Course Content:						
Module 1	Foundations of Scalable Java Applications	Assignment	Implementation	**10 Sessions**		
Topics: **Introduction to Scalability -** Defining scalability: vertical vs. horizontal scaling, Monolithic vs. Microservices Architecture, Performance bottlenecks and solutions. **Java Performance Optimization -** JVM internals and tuning, Garbage Collection (GC) strategies, Profiling and monitoring Java applications **Concurrency and Multithreading -** Java concurrency model and thread management, Executor framework, ForkJoinPool, Thread safety, locks, and synchronization.						
Module 2	Scalable Architectures and Microservices	Assignment	Implementation	10 Sessions		
Topics: **Microservices and Distributed Systems -** Principles of microservices, Service-to-service, communication (REST, gRPC, Kafka), API Gateway, Service Discovery, and Load Balancing **Design Patterns for Scalability -** Singleton, Factory, and Builder patterns, CQRS (Command Query Responsibility Segregation), Circuit Breaker and Retry patterns (Resilience4j) **Event-Driven Architecture & Asynchronous Processing -** Event Sourcing with Kafka, WebSockets and Reactive Programming, Handling failures in distributed systems.						
Module 3	Scalable Data Management and Caching	Mini - Project	Implementation	10 Sessions		
Topics: **Database Scalability -** SQL vs. NoSQL databases (MySQL, PostgreSQL, MongoDB, Cassandra), Sharding, Replication, and Partitioning strategies, Optimizing queries (Indexing, Query Caching) **Caching Strategies -** In-memory caching with **Redis, Memcached,** Distributed caching techniques, Cache Invalidation and Eviction Policies **Data Streaming & Batch Processing -** Apache Kafka for event-driven data pipelines, Apache Spark for large-scale data processing.						

Module 4	Cloud Deployment and DevOps for Scalability	Quiz	Implementation	10 Sessions

Topics:
Cloud-Native Development - Containerization with **Docker,** Kubernetes for scaling and orchestration, Cloud deployment strategies (AWS, GCP, Azure),
CI/CD & DevOps for Scalable Applications - Building CI/CD pipelines (Jenkins, GitHub Actions), Canary & Blue-Green Deployments, Infrastructure as Code (Terraform, Kubernetes Helm)
Security & Reliability - Securing APIs with **OAuth2, JWT, API Gateway,** Load Balancing & Rate Limiting, Monitoring with **Prometheus, ELK Stack**

Targeted Application & Tools that can be used:

Applications:
The course will focus on building and scaling the following types of applications:

- **E-Commerce Platforms** – Handling high traffic, product catalogs, and real-time inventory.
- **Banking & FinTech Applications** – Secure and high-availability transactions.
- **Streaming & Event-Driven Applications** – Using Kafka for real-time data processing.
- **Social Media & Messaging Platforms** – Scalable messaging and real-time updates.
- **SaaS & Cloud-Native Applications** – Multi-tenant, API-based scalable solutions.

1.

Tools:

Programming & Frameworks

- **Java 17+** – Core programming language for scalable applications.
- **Spring Boot** – Microservices development, REST APIs, and dependency injection.
- **Spring Cloud** – Service discovery, load balancing, and resilience patterns.
- **Quarkus** – Lightweight, high-performance microservices alternative.

Database & Caching

- **SQL Databases:** MySQL, PostgreSQL – Scalable relational data management.
- **NoSQL Databases:** MongoDB, Cassandra – High-availability distributed databases.
- **Redis / Memcached** – In-memory caching for faster response times.

Messaging & Event-Driven Architecture

- **Apache Kafka** – Real-time event streaming and asynchronous communication.
- **RabbitMQ** – Message brokering for decoupling services.
- **gRPC** – High-performance inter-service communication.

Cloud & Deployment

1. **Docker** – Containerization for application portability.
2. **Kubernetes** – Scaling, orchestration, and auto-recovery.
3. **AWS / GCP / Azure** – Cloud deployment and auto-scaling.

CI/CD & DevOps

- **GitHub Actions / Jenkins** – Automated build and deployment pipelines.
- **Terraform** – Infrastructure as Code for cloud provisioning.
- **Helm** – Kubernetes package management for scalable applications.

Text Book(s):

T1: "Designing Data-Intensive Applications" – *By Martin Kleppmann*
T2: "Java Concurrency in Practice" – *By Brian Goetz*
T3: "Spring Microservices in Action" – *By John Carnell*
T4. "Cloud Native Java" – *By Josh Long & Kenny Bastani*

Reference(s):

- **"Designing Data-Intensive Applications"** – *Martin Kleppmann*
- **"Java Performance: The Definitive Guide"** – *Scott Oaks*
- **"Spring Microservices in Action"** – *John Carnell*
- **"Kubernetes Up & Running"** – *Kelsey Hightower, Brendan Burns, Joe Beda*

Course Code: **CSE2504**	**Course Title:** Scalable Application Development using Java Lab **Type of Course:** Lab	**L- T-P- C**	0	**0**	2	1
Version No.	2.0					
Course Prerequisites	CSE1006					
Anti-requisites	**NIL**					
Course Description	This course provides a hands-on, practical approach to building scalable, high-performance applications using Java and related technologies. This course is designed to complement theoretical concepts by offering real-world lab exercises focused on the development of microservices architectures, cloud-native applications, and distributed systems. In this lab-intensive course, students will work on building and deploying scalable applications using Spring Boot, Spring Cloud, Docker, Kubernetes, and Apache Kafka. Students will gain experience in implementing RESTful APIs, asynchronous messaging, data caching, and load balancing to ensure that applications can handle increased traffic and scale efficiently. The course will also cover essential techniques for optimizing performance, including JVM tuning, database optimization, and memory management.					
Course Objective	The primary objectives of the course are to Develop hands-on expertise in building scalable applications using Java and modern frameworks like Spring Boot, Spring Cloud, and Apache Kafka, Implement microservices architectures that enable applications to handle increasing loads efficiently through distributed systems and cloud-native practices, Gain practical experience in optimizing performance by leveraging tools for JVM tuning, database optimization, and memory management to improve application responsiveness and scalability., Work with containerization technologies such as Docker and Kubernetes to deploy Java applications in cloud environments with automated continuous integration/continuous deployment (CI/CD) pipelines, Master service discovery, load balancing, and caching mechanisms to ensure high availability, fault tolerance, and low-latency operations in production-grade applications and Apply event-driven architectures to build scalable and resilient systems using tools like Apache Kafka for real-time data processing and messaging.					
Course Outcomes	On successful completion of the course the students shall be able to: CO1. Implement Performance Optimization Techniques CO2 Design and Build Scalable Microservices CO3Integrate Event-Driven Architectures and Caching. CO4 Deploy and Scale Applications in Cloud Environments.					
Course Content:						

Module 1	Foundations of Scalable Java Applications	Assignment	Implementation	**10 Sessions**

Lab Assignment 1: Setting Up Development Environment

Objective: Set up the Java development environment and configure a Spring Boot project for scalability testing.

Tasks:

- Install **Java 17**, **Maven**, and **IDE** (IntelliJ or Eclipse).
- Set up a basic **Spring Boot project** using **Spring Initializr** with **RESTful endpoints**.
- Verify application functionality by running a **local server** and testing API responses via **Postman** or **curl**.
- Add a basic **Spring Boot Actuator** to monitor application health and performance.

Deliverables:

- Working **Spring Boot application** with basic endpoints.
- Screenshots of successful tests (Postman or curl).

Lab Assignment 2: Performance Optimization with JVM

Objective: Profile and optimize a Java application for better performance.

Tasks:

1. Implement a **simple Java application** that performs a memory-intensive task (e.g., sorting a large dataset).
2. Use **VisualVM** to monitor **JVM memory usage, CPU usage, and garbage collection**.
3. Optimize the application by adjusting **JVM flags** (e.g., **heap size**, **garbage collection strategy**).
4. Measure the impact of optimizations on **execution time and memory usage**.

Deliverables:

5. Profiled and optimized **Java application** with performance comparison charts.
6. Detailed report on **JVM tuning** and optimization strategies.

Lab Assignment 3: Implementing Multi-threading

Objective: Understand Java's concurrency model and implement multi-threading for parallel tasks.

Tasks:

1. Implement a **multi-threaded Java application** that simulates multiple tasks (e.g., processing large files, image processing).
2. Use the **Executor framework** to manage thread pools.
3. Measure the **execution time** and compare the performance of **single-threaded** vs **multi-threaded** approaches.

Deliverables:

4. Source code for **multi-threaded application** with explanations of thread management.
5. Execution time comparison chart.

Module 2	Scalable Architectures and Microservices	Assignment	Implementation	10 Sessions

Lab Assignment 4: Building a Simple Microservice with Spring Boot

Objective: Develop a **basic microservice** with Spring Boot.

Tasks:

1. Create a **Spring Boot microservice** that provides a RESTful API (e.g., a book or customer service).
2. Implement **basic CRUD operations** (Create, Read, Update, Delete).
3. Use **Spring Data JPA** to interact with an **SQL database** (e.g., MySQL).
4. Write unit tests using **JUnit** for API endpoints.

Deliverables:

5. Working **Spring Boot microservice** with API documentation.
6. Source code with unit tests.

Lab Assignment 5: Implementing Service Discovery & Load Balancing

Objective: Enable **service discovery** and **load balancing** using **Spring Cloud Eureka**.

Tasks:

- Set up a **Spring Cloud Eureka server** for service discovery.
- Create two **Spring Boot microservices** that register with the Eureka server.
- Implement **load balancing** with **Spring Cloud Ribbon** by making API calls to different instances of the microservices.
- Test and verify load balancing behavior using **Postman** or **curl**.

Deliverables:

- Spring Cloud **Eureka server** and two **microservices**.
- Load balancing validation and test results.

Lab Assignment 6: Building an Event-Driven System with Kafka

Objective: Implement an **event-driven architecture** using **Apache Kafka** for inter-service communication.

Tasks:

- Set up **Apache Kafka** locally or in Docker.
- Create two Spring Boot applications: one as a **Kafka producer** and the other as a **consumer**.
- Implement asynchronous message communication where the producer sends messages (e.g., order events) and the consumer processes them.
- Add error handling and retry logic using **Spring Kafka**.

Deliverables:

- Kafka producer and consumer applications with **message processing logic**.
- Screenshots or logs showing messages being passed from producer to consumer.

Module 3	Scalable Data Management and Caching	Mini - Project	Implementation	10 Sessions

Lab Assignment 7: Integrating SQL and NoSQL Databases

Objective: Learn to integrate **relational (SQL)** and **non-relational (NoSQL)** databases with Spring Boot applications.

Tasks:

- Integrate a **MySQL** database into a **Spring Boot microservice** and implement CRUD operations.
- Set up a **MongoDB** instance (locally or via Docker) and create a second microservice using **Spring Data MongoDB**.
- Compare the performance and scalability aspects of both databases.

Deliverables:

- Source code for Spring Boot microservices using **MySQL** and **MongoDB**.
- Database performance comparison with benchmarks.

Lab Assignment 8: Implementing Redis Caching

Objective: Improve application performance using **Redis** as an in-memory cache.

Tasks:

1. Integrate **Redis** with your Spring Boot application.
2. Cache frequently accessed data (e.g., product information, user profiles) in Redis.
3. Implement **cache expiration** and **cache invalidation** strategies.
4. Measure the performance improvement by comparing **cache hits vs. misses**.

Deliverables:

5. Redis-integrated Spring Boot application with caching logic.

6. Performance comparison between cached and non-cached operations.

Lab Assignment 9: Data Streaming with Kafka

Objective: Implement a **data streaming pipeline** with Kafka for real-time data processing.

Tasks:

- Build a **data pipeline** where **Kafka producers** stream events (e.g., logs, transactions) to **Kafka brokers**.
- Use **Kafka consumers** to process these events in real-time (e.g., updating a database or triggering a workflow).
- Implement **Kafka Streams** to process data within the Kafka ecosystem.

Deliverables:

- Kafka producer-consumer setup with real-time data flow.

Module 4	Cloud Deployment and DevOps for Scalability	Quiz	Implementation	10 Sessions

Lab Assignment 10: Dockerizing a Spring Boot Application

Objective: Containerize a Spring Boot application using **Docker** for scalability.

Tasks:

- Write a **Dockerfile** for your Spring Boot application.
- Build a **Docker image** and run the application in a Docker container.
- Test the application running in the container and compare performance to local deployment.

Deliverables:

- Dockerized **Spring Boot application** with a functional test.
- Docker image and run command documentation.

Lab Assignment 11: Deploying with Kubernetes

Objective: Deploy a Spring Boot microservice to a **Kubernetes cluster** for horizontal scaling.

Tasks:

- Deploy the **Dockerized Spring Boot application** to a **Kubernetes cluster** (local Minikube or cloud-based).
- Set up **Kubernetes Pods, Services, and Deployments** to scale the microservice.
- Implement **auto-scaling** based on CPU or memory usage.

Deliverables:

- Kubernetes **deployment YAML files** for Spring Boot application.
- Running Kubernetes cluster with auto-scaling behavior.

Lab Assignment 12: Setting Up CI/CD for Scalable Applications

Objective: Automate the deployment process using **CI/CD pipelines**.

Tasks:

- Set up a **Jenkins** or **GitHub Actions** CI/CD pipeline to automate the build, test, and deployment process.
- Implement **co+ntinuous deployment** (CD) for deploying a **Dockerized microservice** to a Kubernetes cluster.
- Test the pipeline by pushing changes to a GitHub repository and triggering the deployment process automatically.

Deliverables:

- **CI/CD pipeline configuration** (Jenkins or GitHub Actions).
- Deployment automation logs and screenshots.

Targeted Application & Tools that can be used:

Applications:

The course will focus on building and scaling the following types of applications:

- **E-Commerce Platforms** – Handling high traffic, product catalogs, and real-time inventory.
- **Banking & FinTech Applications** – Secure and high-availability transactions.
- **Streaming & Event-Driven Applications** – Using Kafka for real-time data processing.
- **Social Media & Messaging Platforms** – Scalable messaging and real-time updates.
- **SaaS & Cloud-Native Applications** – Multi-tenant, API-based scalable solutions.

Tools:

Programming & Frameworks

1. **Java 17+** – Core programming language for scalable applications.
2. **Spring Boot** – Microservices development, REST APIs, and dependency injection.
3. **Spring Cloud** – Service discovery, load balancing, and resilience patterns.
4. **Quarkus** – Lightweight, high-performance microservices alternative.

Database & Caching

- **SQL Databases:** MySQL, PostgreSQL – Scalable relational data management.
- **NoSQL Databases:** MongoDB, Cassandra – High-availability distributed databases.
- **Redis / Memcached** – In-memory caching for faster response times.

Messaging & Event-Driven Architecture

- **Apache Kafka** – Real-time event streaming and asynchronous communication.
- **RabbitMQ** – Message brokering for decoupling services.
- **gRPC** – High-performance inter-service communication.

Cloud & Deployment

- **Docker** – Containerization for application portability.
- **Kubernetes** – Scaling, orchestration, and auto-recovery.
- **AWS / GCP / Azure** – Cloud deployment and auto-scaling.

CI/CD & DevOps

- **GitHub Actions / Jenkins** – Automated build and deployment pipelines.
- **Terraform** – Infrastructure as Code for cloud provisioning.

Helm – Kubernetes package management for scalable applications.

Text Book(s):

T1: "Spring in Action" by Craig Walls
T2: "Java Performance: The Definitive Guide" by Scott Oaks
T3: "Designing Data-Intensive Applications" by Martin Kleppmann
T4. "Spring Microservices in Action" by John Carnell

Reference(s):

1. "Designing Data-Intensive Applications" – Martin Kleppmann
2. "Java Performance: The Definitive Guide" – Scott Oaks
3. "Spring Microservices in Action" – John Carnell
4. "Kubernetes Up & Running" – Kelsey Hightower, Brendan Burns, Joe Beda

Course Code: CSE2266	**Course Title:** Theory of Computation **Type of Course:** Theory Only		**L- T-P- C**	3	**0**	0	**3**
Version No.	2.0					
Course Pre-requisites	NIL					
Anti-requisites	**NIL**					
Course Description	The course deals with introduction of formal languages and the correspondence between language classes and the automata that recognize them. Topics include: Formal definitions of grammars and acceptors, Deterministic and Nondeterministic systems, Grammar ambiguity, finite state and push-down automata; normal forms; Turing machines and its relations with algorithms.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of Theory of Computation as mentioned above and attain Skill Development through Problem Solving Methodologies.					
Course Outcomes	On successful completion of the course the students shall be able to: 1. Describe various components of Automata. (Knowledge) 2. Illustrate Finite Automata for the given Language. (Application) 3. Distinguish between Regular grammar and Context free grammar. (Comprehension) 4. Construct Push down Automata. (Application) 5. Construct Turing machine for a Language. (Application)					
Course Content:						
Module 1	Introduction to automata theory	Assignment	Problems on Strings and Language operations	**6 classes**		
Topics: Introduction to Automata Theory, Applications of Automata Theory, Alphabets, Strings, Languages & operations on languages, Representation of automata, Language recognizers, Finite State Machines (FSM): Deterministic FSM, Regular languages, Designing FSM, Nondeterministic FSMs						
Module 2	Finite Automata	Assignment	Assignment Problems on DFA, NFA's	13 Sessions		
Topics: Basic concepts of Finite automata, DFA- definitions of DFA, Deterministic Accepters Transition Graphs and Languages and DFA's, Regular Languages, NFA- Definition of a Nondeterministic Accepter, Languages and NFA's Why Non- determinism? Equivalence of Deterministic and Nondeterministic Finite Accepters, Reduction of the Number of States in Finite Automata.						
Module 3	Regular Expressions & Context Free Grammar	Assignment	Problems on RE, CFG, PT, PL and Ambiguity	12 Sessions		

Topics:
Formal Definition of a Regular Expression, Languages Associated with Regular Expressions, Languages, Regular Languages (RL) and Non-regular Languages: Closure properties of RLs, to show some languages are not RLs, Closure Properties of Regular Context Free Grammars-Examples of Context-Free Languages, Leftmost and Rightmost Derivations, Derivation Trees, Relation Between Sentential Forms and Derivation Trees, Ambiguity in Grammars and Languages: Ambiguous Grammars, Removing Ambiguity, Chomsky Normal Form, Gribiche Normal Form.

Module 4	Push down Automata	Assignment	Problems on pushdown Automaton	08 Sessions

Topics:
Definition of a Pushdown Automaton, Language Accepted by a Pushdown Automaton, Acceptance by Final State, Acceptance by Empty Stack, From Empty Stack to Final State, From Final State to Empty Stack Equivalence of PDA's and CFG's: From Grammars to Pushdown Automata.

Module 5	Turing Machine	Assignment	Problems on Turning Machine	07 Sessions

Topics:
Definition of a Turing Machine, Turing Machines as Language Accepters, Example Languages to construct Turing
machine, Turing Machines as Transducers, Halting Programming Techniques for Turing Machines

Targeted Application & Tools that can be used:
Targeted Application:
1. Text Processing
2. Compilers
3. Text Editors
4. Robotics Applications
5. Artificial Intelligence

Tools:
1. JFLAP (Java Formal Language and Automata Package) Software simulation tool. It's interactive educational
software written in Java to experiment topics in automata theory.
2. Turing machine Online simulators.

Text Book(s):
1. Peter Linz, "An introduction to Formal Languages and Automata", Jones and Bartlett Publications 6th Ed, 2018.

Reference(s):
1. Aho, Ullman and Hopcroft, "Theory of Computation", Pearson India 3rd Edition 2008.
2. Michael Sipser, "Theory of Computation", Cengage India 3rd Ed, 2014.
E-Resources
NPTEL course – https://onlinecourses.nptel.ac.in/noc21_cs83/preview

Course Code: CSE2502	**Course Title:** Cryptography and Network Security **Type of Course:** Theory		**L- T- P- C**	3	**0**	0	**3**
Version No.	2.0					
Course Pre-requisites	CSE2251					
Anti-requisites	**NIL**					
Course Description	The Course deals with the principles and practice of cryptography and network security, focusing in particular on the security aspects of the web and Internet					
Course Objective	The objective of the course is to familiarize the learners with the concepts of Cryptography and Network Security above and attain Skill Development through Problem Solving methodologies.					
Course Outcomes	On successful completion of this course the students shall be able to: 1. Describe the basic concept of Cryptography 2. Classify different types of Cryptographic Algorithms 3. Solve Mathematical problems required for Cryptography 4. Illustrate Network Security concepts					
Course Content:						
Module 1	Introduction to Cryptography	Assignment	Recognize the techniques	**7 Sessions**		
Introduction to Cryptography, Model of Network Security, OSI Security architecture, Security Attacks: active attacks, passive attacks, services: Authentication, Access Control, Data Confidentiality, Data Integrity, Nonrepudiation, Substitution Ciphers : Play-fair and Hill Cipher, Vigenere cipher, Introduction to Block Cipher and Stream Cipher, Feistel Structure, ECB modes of block cipher.						
Module 2	Symmetric Encryption Algorithm	Assignment	Analysis of solutions	9 Sessions		
Overview of Public Key Cryptography, RSA, Diffie-Helman Key exchange, Man in the middle attack, Cryptographic Hash functions, Secure Hash Algorithm, Message Authentication Codes – HMAC, Digital Signature, Ei-gamal Encryption, Elliptic curve cryptography overview.						
Module 3	Public Key Cryptography	Assignments	Analysis of solutions	9 Sessions		
Overview of Public Key Cryptography, RSA, Diffie-Helman Key exchange, Man in the middle attack, Cryptographic Hash functions, Secure Hash Algorithm, Message Authentication Codes – HMAC, Digital Signature, Ei-gamal Encryption, Elliptic curve cryptography overview						
Module 4	Network Security	Assignment	Analysis of solutions	05 Sessions		
Topics: Network Security fundamentals, Network Security applications: Authentication: Kerberos, PKI, Network Security applications: e-mail security: PGP, MIME, Network Security applications: IP Security: IPSec architecture, Network Security applications: DNS Security.						
Targeted Application & Tools that can be used: Students get the knowledge about cryptography techniques followed, the algorithms used for encryption and decryptions & the techniques for authentication and confidentiality of messages.						
Text Book(s): T1 William Stallings, "Cryptography and Network Security - Principles and Practices", 7th Edition, Pearson publication, ISBN: 978-93-325-8522-5, 2017						

Reference(s):

R1 Bruice Schneier, “Applied Cryptography – Protocols, Algorithms and Source code in C”, Second Edition, Wiley
Publication, ISBN: 978-81-265-1368-0, 2017
R2 Cryptography and Network Security, Express Learning, ITL Education Solution Limited.
R3 e-pg pathshala UGC lecture series
Web references:
https://puniversity.informaticsglobal.com:2229/login.aspx?direct=true&db=nlebk&AN=2233842&site =ehost-live
https://nptel.ac.in/courses/106105031.

Topics relevant to “Skill Development”: Topics relevant to “Skill Development”:

1. Play-fair and Hill Cipher
2. Euclidean and Extended Euclidean Algorithm
3. Secure Hash Algorithm
4. Diffie-Helman Key exchange
5. Totient Function.
6. Fermat’s little theorem

Course Code: CSE2267	**Course Title: Machine Learning Techniques** **Type of Course: Program Core -Theory**	**L-T-P-C**	3	0	0	3
Version No.						
Course Pre-requisites	NIL					
Anti-requisites						
Course Description	Machine Learning algorithms are the key to developing intelligent systems such as Apple's Siri, Google's self-driving cars, and more. This course introduces the core concepts and essential algorithms of various machine learning techniques, including Regression learning, Bayesian learning, Ensemble learning, Perceptron learning (foundations of Neural Networks), Unsupervised learning, Competitive learning, learning from Gaussian Mixture Models, and learning to detect outliers. The course lectures cover both the theoretical foundations and practical algorithms for these learning methods. Lab sessions complement the lectures, enabling students to develop intelligent systems for real-life problems using Python libraries.					
Course Objectives	The primary objective of this course is to enhance the **EMPLOYABILITY** of students by leveraging **EXPERIENTIAL LEARNING** techniques. Students will gain practical skills in applying machine learning to real-world challenges.					
Course Out Comes	On successful completion of this course the students shall be able to: 5. **Apply** advanced supervised machine learning methods for predictive modeling. [APPLY] 6. **Produce** machine learning models with better predictive performance using meta-learning (ensemble) algorithms. [APPLY] 7. **Create** predictive models using Perceptron learning algorithms, understanding their foundational role in neural networks. [APPLY] 8. **Employ** advanced unsupervised learning algorithms for clustering, competitive learning, and outlier detection. [APPLY] 9. **Implement** machine learning-based intelligent models using Python libraries and frameworks. [APPLY]					

Course Content:

Module 1	**pervised Learning**	**ignment**	**dule Tests**	**No. of Sessions: 12**
Overview of Machine Learning (ML): ML workflow, types of ML (supervised, unsupervised, reinforcement), types of features, Feature Engineering (Data Imputation Methods). Regression: Introduction, Simple Linear Regression, Loss Functions (MSE, MAE, RMSE), Polynomial Regression, Logistic Regression, Softmax Regression with Cross-Entropy as cost function. Bayesian Learning: Bayes Theorem, estimating conditional probabilities for categorical and continuous features, Naïve Bayes for supervised learning, Bayesian Belief Networks. Support Vector Machines (SVM): Soft margin and kernel tricks (Polynomial, RBF, Sigmoid). Evaluation Methodologies: Testing Dataset, Train-Validation-Testing split, N-Fold Cross Validation (K-Fold, Stratified K-Fold)..				
Module 2	**emble Learning**	**ignment**	**dule Tests**	**No. of Sessions: 11**
Introduction to Ensemble Learning: Motivation, bias-variance trade-off. Bagging: Using subsets of instances (Bagging, Pasting), using subsets of features (Random Patches, Random Subspaces method). Ensemble Methods: Voting Classifier (Hard and Soft Voting), Random Forest (Algorithm, Feature Importance). Boosting: AdaBoost (Adaptive Boosting), Gradient Boosting Machines (GBM), Extremely Randomized Trees, Stacking (Meta-Learning).				
Module 3	**rceptron Learning &**	**ignment**	**dule Tests**	**No. of**

	ural Networks Foundation			**Sessions: 11**
From Biological to Artificial Neurons: Basic structure of a neuron. Perceptrons: Single-layer Perceptrons, Linear Threshold Units, logical computations with Perceptrons (AND, OR, NOT). Activation Functions: Common activation functions (Sigmoid, Tanh, ReLU, Leaky ReLU, ELU, Softmax) and their properties. Loss Functions: Common loss functions for classification (Binary Cross-Entropy, Categorical Cross-Entropy) and regression (MSE, MAE). Multi-layer Perceptrons (MLP): Architecture, advantages over single Perceptrons. Backpropagation Algorithm: Understanding the concept of backpropagation, Gradient Descent and its variants (Stochastic Gradient Descent, Mini-batch Gradient Descent, Adam, RMSprop).				
Module 4	**supervised Learning & Advanced Topics**	**ignment**	**dule Tests**	**No. of Sessions: 11**
Clustering: Simple K-Means Clustering (Algorithm, Simple and Mini-Batch), updating centroids incrementally. K-Means Enhancements: Finding the optimal number of clusters (Elbow Method, Silhouette Coefficient), drawbacks of K-Means, K-Means++. Hierarchical Clustering: Divisive hierarchical clustering (Bisecting K-Means), clustering using Minimum Spanning Tree (MST). Competitive Learning: Clustering using Kohonen's Self-Organizing Maps (SOM). Density-Based Clustering: DBSCAN (Density-Based Spatial Clustering of Applications with Noise). Probabilistic Clustering: Clustering using Gaussian Mixture Models (GMM) with Expectation-Maximization (EM) algorithm. Outlier Detection Methods: Isolation Forest, Local Outlier Factor (LOF). Association Rule Mining: Introduction, Apriori Algorithm (concepts only). Collaborative Filtering: Introduction (User-based, Item-based).				
Targeted Application & Tools that can be used: 3. Google Colab 4. Python IDEs like PyCharm				
Teaching Methodology **Lectures:** Interactive lectures covering theoretical foundations and algorithmic details. **Lab Sessions:** Hands-on sessions focusing on implementing algorithms using Python libraries (e.g., scikit-learn, NumPy, Pandas, Matplotlib, Seaborn, Keras/TensorFlow for Perceptron). **Problem-Based Learning:** Real-world case studies and problems will be discussed and solved in labs. **Experiential Learning:** Emphasis on practical application through assignments, mini-projects, and a final course project. **Discussions:** Encouraging critical thinking and peer learning through in-class discussions. **Project work/Assignment:** 1. Students can be assigned a mini project to develop a machine learning application for real-life problems in various domains such as health care, business intelligence, environmental modeling, etc.				
Textbook(s): T1. Aurélien Géron. *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow*, Oreilly, 3rd Edition, 2022.				
References: R1. Raschka, S., Liu, Y., & Mirjalili, V. (2022). Machine Learning with PyTorch and Scikit-Learn: Develop machine learning and deep learning models with Python. Packt Publishing Ltd. R2. Alpaydin, E. (2021). Machine Learning, revised and updated edition. MIT Press. R3. Deisenroth, M. P., Faisal, A. A., & Ong, C. S. (2020). Mathematics for machine learning. Cambridge University Press. R4. Andreas C Muller, and Sarah Guido. *Introduction to Machine Learning with Python: A Guide for Data Scientists*, O'Reilly, 1st Edition, 2016. **Weblinks** . W1. NPTEL Courses: https://nptel.ac.in/courses/106106139 (IIT M), https://nptel.ac.in/courses/106105152 (IIT Kgp)				

. Scikit-learn documentation: https://scikit-learn.org/stable/ . TensorFlow documentation: https://www.tensorflow.org/ . Keras documentation: https://keras.io/ . Pandas documentation: https://pandas.pydata.org/ . NumPy documentation: https://numpy.org/ . Kaggle: For datasets and competitions.

Course Code: CSE2268	**Course Title:** Machine Learning Techniques Lab **Type of Course:** Program Core -Laboratory	**L-T-P-C**	0	0	2	1
Version No.	1.0					
Course Pre-requisites	Nil					
Anti-requisites	NIL					
Course Description	Machine Learning algorithms are the key to develop intelligent systems such as Apple's Siri, Google's self-driving cars etc. This course introduces the concepts of the core machine learning techniques such as Regression learning, Bayesian learning, Ensemble learning, Perceptron learning, Unsupervised learning, Competitive learning, learning from Gaussian mixture models and learning to detect outliers. Course lectures covers both the theoretical foundations as well as the essential algorithms for the various learning methods. Lab sessions complement the lectures and enable the students in developing intelligent systems for real life problems.					
Course Objectives	The objective of the course is EMPLOYBILITY of student by using EXPERIENTIAL LEARNING techniques.					
Course Out Comes	On successful completion of this course the students shall be able to: 10. **Apply** advanced supervised machine learning methods for predictive modeling. [Apply] 11. **Produce** machine learning models with better predictive performance using meta learning algorithms [Apply] 12. **Create** predictive models using Perceptron learning algorithms [Apply] 13. **Employ** advanced unsupervised learning algorithms for clustering, competitive learning and outlier detection [Apply] 14. **Implement** machine learning based intelligent models using Python libraries. [Apply]					
Course Content:					**No. of Sessions: 15 (30 hours)**	
Experiment No. 1: File Handling Using Python **Level 1:** Read a CSV file using Python **Level 2:** Read a text file using Python **Experiment No. 2: Methods for handling missing values** **Level 1:** Given a data set from UCI repository, implement the different ways of handling missing values in it using Scikit-learn library of Python **Level 2:** Implement one of these methods using a custom defined function in Python. **Experiment No. 3: Data Visualization**						

Level 1: Perform Exploratory Data Analysis for a given data set by creating Scatter Plot, Pair Plot, Count Plot using Matplotlib and Seaborn
Level 2: Create Heat Maps, WordCloud

Experiment No. 4: Regression learning
Level 1: Given a data set from UCI repository, implement the simple linear regression algorithm and estimate the models parameters and the performance metrics. Plot the learning curves.
Level 2: Implement the polynomial regression algorithm. Compare the learning curves of Polynomial and
Linear Regression.

Experiment No. 5: Logistic Regression
Level 1: Write custom code for generating the logistic/sigmoid plot for a given input
Level 2: Given a data set from UCI repository, implement the Logistic regression algorithm. Estimate the class probabilities for a given test data set. Plot and analyze the decision boundaries.

Experiment No. 6: Bayesian Learning
Level 1: Given a data set from UCI repository, implement a classification model using the Bayesian algorithm.
Level 2: Implement a Naïve Bayes classifier using 5-fold cross-validation

Experiment No. 7: Support Vector Machine (SVM)
Level 1: Given data sets from UCI repository, implement a linear SVM and a non-linear SVM based classification model.
Level 2: Construct kernels with 5-fold cross-validation for SVM.

Experiment No. 8 & 9: Ensemble Learning
Level 1: Implement Ensemble Learning algorithms such as Bagging, Pasting and Out-of Bag Evaluation
Level 2: Random Patches and Random Subspace Method, Adaboost and Gradient Boosting, Stacking.

Experiment No. 10: Perceptron Learning
Level 1: Implement the Perceptron Classifier
Level 2: An Image Classifier Using the Sequential API of Keras

Experiment No. 11 & 12: Unsupervised Learning
Level 1: K-means – simple and mini-batch. Finding the optimal number of clusters using Elbow method and Silhoutte Coefficient . Compare the inertia of both as k increases. Tuning the hyperparameter 'k' using GridSearchCV.
Level 2: Using clustering for Image segmentation and Preprocessing. Kmeans++

Experiment No. 13: Density Based Clustering
Level 1: Implement DBSCAN – clustering using the local density estimation. Perform hard and soft clustering for new instances.
Level 2: Outlier Detection using Isolation Forest and Local Outlier Factor

Experiment No. 14: Association Rule Mining
Level 1: Implement the Apriori Algorithm for Association Rule Mining
Level 2: Implement the Dynamic Itemset Counting Algorithm for Association Rule Mining.

Experiment No. 15: Collaborative Filtering
Level 1: Implement Collaborative Filtering using Item-Based Filtering
Level 2: Implement Collaborative Filtering using User-Based Filtering

Targeted Application & Tools that can be used:

5. Google Colab 6. Python IDEs like PyCharm
Project work/Assignment: Mention the Type of Project /Assignment proposed for this course Students can be assigned a mini project to develop a machine learning application for real-life problems in various domains such as health care, business intelligence, environmental modeling, etc.
Textbook(s): **3.** Aurélien Géron. *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow*, Oreilly, 3rd Edition, 2022.
References: R1. Andreas C Muller, and Sarah Guido. *Introduction to Machine Learning with Python: A Guide for Data Scientists*, O'Reilly, 1st Edition, 2016. **Weblinks** **W1**. NPTEL Courses: https://nptel.ac.in/courses/106106139 (IIT M), https://nptel.ac.in/courses/106105152 (IIT Kgp)

Course Code: CSE2269	**Course Title:** Operating Systems **Type of Course: Program Core and Theory Only**		**L-T- P- C**	3	0	0	3
Version No.	1.0						
Course Pre-requisites	NIL						
Anti-requisites	**NIL**						
Course Description	This course introduces the concepts of Operating System operations, Operating System structure and its design and implementation. It covers the classical Operating Systems internal algorithms such as process scheduling, synchronization, deadlocks detection and recovery and memory management. The course also enhances the problem solving, systems programming ability and case studies.						
Course Object	The objective of the course is to familiarize the learners with the concepts of Operating Systems and attain **Employability** through **Problem Solving** Methodologies.						
Course Out Comes	On successful completion of the course the students shall be able to: 1] Describe the fundamental concepts of operating Systems and case studies. [**Knowledge**] 2] Demonstrate various CPU scheduling algorithms. .[**Application**] 3] Apply various tools to handle synchronization problems.[**Application**] 4] Demonstrate deadlock detection and recovery methods **[Application]** 5] Illustrate various memory management techniques.[**Application**]						
Course Content:							
Module 1	Introduction to Operating System	Assignment	Programming				**9 Hours**
Topics: Introduction to OS , Operating-System Operations, Operating System Services, , System Calls and its types, Operating System Structure, System Program and its types, Linkers and Loaders, Overview of OS design and implementation, Open-source operating system							
Module 2	Process Management	Assignment/Case Study	Programming/Simulation				**11 Hours**
Topics: Process Concept, Operations on Processes, Inter Process Communication, Communication in client-server systems (sockets, RPC, Pipes), Introduction to threads - Multithreading Models, Thread Libraries, Threading Issues, Process Scheduling– Basic concepts, Scheduling Criteria, Scheduling Algorithms: FCFS, SJF, SRTF,							

RR and Priority.				
Module 3	Process Synchronization and Deadlocks	Assignment	Programming	**11 Hours**
Topics: The Critical-Section Problem- Peterson's Solution, Synchronization hardware, Semaphores, Classic Problems of Synchronization with Semaphore Solution- Producer-Consumer Problem, Reader-Writer problems, Dining Philosopher's Problem, . Introduction to Deadlocks, Necessary conditions for deadlock, Resource allocation Graph, Methods for handling deadlock: Deadlock Prevention and Implementation, Deadlock Avoidance and Implementation, Deadlock detection & Recovery from Deadlock.				
Module 4	Memory Management	Assignment	Programming/Simulation	**10 Hours**
Topics: Introduction to Memory Management, Basic hardware-Base and Limit Registers, Memory Management Unit(MMU), Dynamic loading and linking, Swapping, Contiguous and Non-Contiguous Memory Allocation, Segmentation, Paging - Structure of the Page Table – Virtual Memory and Demand Paging – Page Faults and Page Replacement Algorithms, Copy-on-write, Allocation of Frames, Thrashing Introduction to File system management: File System Interface (access methods, directory structures), File system implementation.				
Targeted Application: **Application area is traffic management system, banking system, health care and many more systems where in there are resources and entities that use and manage the resources.** **Software Tools:** Oracle Virtual Box/VMWare Virtualization software [Virtual Machine Managers]. Used to install and work on multiple guest Operating Systems on top of a host OS. Intel Processor identification utility: This software is used to explain about multi-core processors. It helps to identify the specifications of your Intel processor, like no of cores, Chipset information, technologies supported by the processor etc.				
Project work/Assignment				
Demonstrate process concepts in LINUX OS. **Simulation of CPU scheduling algorithms.** **Develop program to demonstrate use of Semaphores in threads.** **Develop program to demonstrate use of deadlock avoidance algorithms.** **Develop program to demonstrate use of page replacement algorithms.** **Simulation of memory allocation strategies [first fit, best fit and worst fit].**				
Text Book Silberschatz A, Galvin P B and Gagne G , "Silberschatz's Operating System Concepts", Paperback, Global Edition Wiley, 2019				
References Silberschatz A, Galvin P B and Gagne G, "Operating System Concepts", 10th edition Wiley, 2018. William Stallings, "Operating Systems",Ninth Edition, By Pearson Paperback ,1 March 2018. Sundaram RMD, Shriram K V, Abhishek S N, B Chella Prabha, " Cracking the Operating System skills", Dreamtech, paperback, 2020 Remzi H. Arpaci-Dusseau Andrea C. Arpaci-dusseau , "Operating Systems: Three Easy Pieces, Amazon digital Services", September 2018. **E-resources/Weblinks** https://www.os-book.com/OS9/ https://pages.cs.wisc.edu/~remzi/OSTEP/ https://codex.cs.yale.edu/avi/os-book/OS10/index.html				

Course Code: CSE2270	**Course Title:** Operating S**ystems Lab** **Type of Course: Lab Only**	**L-T- P- C**	0	0	2	1
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	**NIL**					
Course Description	This laboratory course provides hands-on experience with the core concepts of operating systems through practical assignments, simulations, and case studies. It covers foundational aspects such as system calls, process and thread management, inter-process communication, synchronization, deadlocks, memory management, and file systems. Students will implement and simulate real-time OS components and scheduling algorithms, fostering deeper understanding of OS architecture and design. The lab also introduces modern OS tools, programming interfaces, and the basics of open-source OS environments.					
Course Object	The objective of the course is to familiarize the learners with the concepts of Operating Systems and attain **Employability** through **Problem Solving** Methodologies.					
Course Out Comes	On successful completion of the course the students shall be able to: 1] Demonstrate system-level programming using system calls and OS structures. [**Apply**] 2] Simulate process scheduling and multithreading techniques. [**Apply**] 3] Apply various tools to handle synchronization problems using semaphores and shared memory. [**Apply**] 4] Demonstrate memory management and file system concepts using simulation or scripting. **[Apply]**					
Course Content:						

Targeted Application:
Application area is traffic management system, banking system, health care and many more systems where in there are resources and entities that use and manage the resources.

Software Tools:
Oracle Virtual Box/VMWare Virtualization software [Virtual Machine Managers]. Used to install and work on multiple guest Operating systems on top of a host OS.

Intel Processor identification utility: This software is used to explain about multi-core processors. It helps to identify the specifications of your Intel processor, like no of cores, Chipset information, technologies supported by the processor etc.

List of Laboratory Tasks**:**

Lab sheet -1

L1: Write a program to demonstrate the use of fork() and exec() system calls in process creation.

L2: A system has limited memory and high-priority real-time processes. Design a scheduling algorithm that ensures responsiveness while preventing starvation.

Lab sheet -2

L1: Implement First-Come-First-Serve (FCFS) process scheduling using C or Python.

L2: You are designing a server that handles thousands of client connections. Compare multithreading and multiprocessing for this task and implement a basic server model.

Lab sheet -3

L1: Implement Round Robin Scheduling with a fixed time quantum.

L2: In a banking system, concurrent access to accounts leads to data corruption. Design a synchronization solution to avoid race conditions.

Lab sheet -4

L1: Write a program to create threads using Pthreads or Python's threading module.

L2: You're tasked with building a file access tracker in an OS. Implement a system to log file access patterns and identify frequent accesses.

Lab sheet -5

L1: Demonstrate inter-process communication (IPC) using pipes.

L2: A simulation tool needs to emulate process suspension and resumption. Design and implement such a mechanism using signals or condition variables.

Lab sheet -6

L1: Simulate the Producer-Consumer problem using semaphores.

L2: You're developing a system where sensor devices (producers) generate temperature readings, and data processors (consumers) store and process these readings. To prevent race conditions and ensure buffer safety, implement a synchronization mechanism using semaphores.

Lab sheet -7

L1: Implement Dining Philosophers Problem using threads and synchronization.

L2: In a multi-threaded cafeteria simulation, five philosophers sit around a circular table, each alternating between thinking and eating. To eat, a philosopher must hold two forks (represented by shared resources). Your task is to avoid deadlock and ensure no philosopher starves using thread synchronization techniques.

Lab sheet -8

L1: Write a program to simulate First Fit, Best Fit, and Worst Fit memory allocation strategies.

L2: A system with limited memory blocks needs to allocate memory to processes arriving with various size requests. Your task is to implement three classic memory allocation strategies—First Fit, Best Fit, and Worst Fit—to allocate memory to each process efficiently. Simulate and compare how memory gets allocated in each strateg

Lab sheet -9

L1: Demonstrate paging using a simple page table simulation.

L2: A program has a logical address space divided into pages. The system's memory is divided into equal-sized frames. When a program executes, its pages are loaded into available frames in main memory. Simulate the address translation process using a page table and demonstrate how a logical address is converted to a physical address.

Lab sheet -10

L1: Write a program to simulate page replacement algorithms like FIFO and LRU.

L2: In a virtual memory system, a process accesses pages in a specific order. The memory can only hold a limited number of pages (frames). When a page is needed and the memory is full, a page replacement algorithm is used to decide which page to evict. Simulate and compare FIFO and LRU algorithms for a given page reference string.

Lab sheet -11

L1: Simulate file directory structure (single level/two level).
L2: A university campus computer lab has limited memory space available for each student login session. When students open files or run programs, memory pages are loaded into available memory frames. Due to the limited number of frames, some pages must be replaced when new ones are needed. The lab system uses page replacement algorithms to decide which pages to evict when memory is full..

Lab sheet -12

L1: Write a shell script to demonstrate file handling commands in Linux.
L2: Design a command-line mini shell that can run background and foreground processes and handle basic built-in commands like cd, pwd, exit.

Project work/Assignment

Demonstrate process concepts in LINUX OS.
Simulation of CPU scheduling algorithms.
Develop program to demonstrate use of Semaphores in threads.
Develop program to demonstrate use of deadlock avoidance algorithms.
Develop program to demonstrate use of page replacement algorithms.
Simulation of memory allocation strategies [first fit, best fit and worst fit].

Text Book

Silberschatz A, Galvin P B and Gagne G , “Silberschatz's Operating System Concepts”, Paperback, Global Edition Wiley, 2019

References

Silberschatz A, Galvin P B and Gagne G, “Operating System Concepts”, 10th edition Wiley, 2018.
William Stallings, “Operating Systems”,Ninth Edition, By Pearson Paperback ,1 March 2018.
Sundaram RMD, Shriram K V, Abhishek S N, B Chella Prabha, “ Cracking the Operating System skills”, Dreamtech, paperback, 2020
Remzi H. Arpaci-Dusseau Andrea C. Arpaci-dusseau , “Operating Systems: Three Easy Pieces, Amazon digital Services”, September 2018.

E-resources/Weblinks

https://www.os-book.com/OS9/
https://pages.cs.wisc.edu/~remzi/OSTEP/
https://codex.cs.yale.edu/avi/os-book/OS10/index.html

Course Code: **CSE7000**	**Course Title:** **Internship** **Type of Course:** **PRW**	**L- T-P- C**	-	-	-	2
Version No.	**1.0**					
Course Pre-requisites	**NIL**					
Anti-requisites	**NIL**					
Course Description	Students observe science and technology in action, develop an awareness of the method of scientific experimentation, and often get an opportunity to see, study and operate sophisticated and costly equipment. They also learn about the implementation of the principles of management they have learnt in class, when they observe multidisciplinary teams of experts from engineering, science, economics, operations research, and management deal with techno-economic problems at the micro and macro levels. Finally, it enables them to develop and refine their language, communication and inter-personal skills, both by its very nature, and by the various evaluation components, such as seminar, group discussion, project report preparation, etc. The broad-based core education, strong in mathematics and science and rich in analytical tools, provides the foundation necessary for the student to understand properly the nature of real-life problems.					
Course Objectives	The objective of the course is to familiarize the learners with the concepts of Professional Practice and attain Employability Skills through Experiential Learning techniques.					
Course Outcomes	On successful completion of this course the students shall be able to: 1. Identify the engineering problems related to local, regional, national or global needs. (Understand) 2. Apply appropriate techniques or modern tools for solving the intended problem. (Apply) 3. Design the experiments as per the standards and specifications. (Analyze) 4. Interpret the events and results for meaningful conclusions. (Evaluate)					

F	**Suggested Course Title: LOGICAL & CRITICAL THINKING** **Type of Course:**	**L- T-P-C**	0	0	2	1
Version No.	1.0					
Course Pre-requisites	Students should have the basic concepts of Logical reasoning and Critical thinking, along with its applications in real life problems.					
Anti-requisites	Nil					
Course Description	This is a skill-based training program for the engineering students (Undergraduate). This course is designed to enable the students to enhance their skills in Logical reasoning and Critical thinking.					
Course Objective	The objective of the course is to familiarize the learners with concepts in in Logical reasoning and Critical thinking through problem solving techniques suitable for their career development.					
Course Outcomes	On successful completion of the course the students shall be able to:					
	CO1] **Understand** all the concepts.					
	CO2] **Apply** the concepts in problem solving (Bloom's taxonomy Level 3)					
Course Content:						
Module 1	Logical Thinking	Assignment				**16 Hours**
	Topics:					
	Syllogisms, Cubes and Dices, Mirror and Water images, Paper cutting and Folding, Embedded figures & Completion of figures, Data Interpretation, Data sufficiency					
Module 2	Critical Thinking	Assignment				**14 Hours**
	Topics:					
	Analogy, Symbol and Notations, Statement and assumption, Cause of action, Statement and conclusion, Puzzles					
	Targeted Application & Tools that can be used:					
	Application area: Placement activities and Competitive examinations.					
	Tools: LMS					
Evaluation	**Continuous Evaluation**					
	· Topic wise evaluation					
	· Mid-Term & End Term					
	Text Book					
	1. A new approach to reasoning verbal, non-verbal & analytical by BS Sijwali					

	2. R S Aggarwal
	3. Kiran publications
	References
	1. www.indiabix.com
	2. www.testbook.com
	3. www.youtube.com/c/TheAptitudeGuy/videos
	Topics relevant to Skill Development Logical reasoning and Critical thinking for Skill Development through Problem solving Techniques. This is attained through assessment component mentioned in course handout.

Course Code: CSE2272	**Course Title:** Cloud computing **Type of Course:** Theory	**L- T-P- C**	2	**0**	0	2
Version No.	2.0					
Course Pre-requisites	NIL					
Anti-requisites	**NIL**					
Course Description	This Course is designed to introduce the concepts of Cloud Computing as a new computing paradigm. Cloud Computing has emerged in recent years as a new paradigm for hosting and delivering services over the Internet. The students can explore various Cloud Computing terminology, principles and applications. Understanding different views of the Cloud Computing such as theoretical, technical and commercial aspects. Topics include: Evolution of cloud computing and its services available today, Introduction, Architecture of cloud computing, Infrastructure, platform, software, Types of cloud, Business models, cloud services, Collaborating using cloud services, Virtualization for cloud, Security, Standards and Applications.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of Could computing and Virtualization and attain Employability through Participative Learning techniques.					
Course Outcomes	On successful completion of the course the students shall be able to: · Describe fundamentals of cloud computing, virtualization and cloud computing services. · Discuss high-throughput and data-intensive computing. · Explain security and standards in cloud computing. · Demonstrate the installation and configuration of virtual machine.					
Course Content:						

Module 1	Introduction to Cloud and Virtualization	Assignment	Virtualization	**10 Sessions**
Topics: Introduction to Cloud and Virtualization Cloud Computing at a Glance, Historical Developments, Building Cloud Computing Environments, Computing Platforms and Technologies, Virtualization, Characteristics of Virtualized Environments Taxonomy of Virtualization Techniques, Virtualization and Cloud Computing, Technology Examples, Cloud Computing Architecture, IaaS, PaaS, SaaS, Types of Clouds, Economics of Cloud				
Module 2	High Throughput and Data Intensive Computing	Assignment	Virtualization	**10 Sessions**
Topics: High Throughput and Data Intensive Computing: Task computing, MPI applications, Task based programming, Introduction to DIC, Technologies for DIC, Aneka Map Reduce Programming.				
Module 3	Cloud Security and Standards	Assignment	Virtualization	**9 Sessions**
Topics: Cloud Security and Standards: Cloud Security Challenges, Software-as-a-Service Security, Application standards, Client standards, Infrastructure and Service standards.				
Module 4	Cloud Platforms	Assignment	Virtualization	**9 Sessions**
Cloud Platforms, Advances in cloud: introduction to Amazon Web Services: Introduction to Google App Engine, Introduction to Microsoft Azure. Media Clouds - Security Clouds - Computing Clouds - Mobile Clouds – Federated Clouds – Hybrid Cloud				
Targeted Application & Tools that can be used:				
Text Book(s): 1. John Rittinghouse and James Ransome, “Cloud Computing, Implementation, Management and Security”, CRC Press. 2. Rajkumar Buyya, Christian Vecchiola, and Thamarai Selvi, “Mastering Cloud Computing”, McGraw Hill Education.				

Reference(s):
1. David E.Y. Sarna, "Implementing and Developing Cloud Applications", CRC Press. 2. Anthony T Velte, Toby J Velte, Robert Elsenpeter, "Cloud Computing: A Practical Approach", Tata McGraw-Hill. Web resources: https://presiuniv.knimbus.com/user#/home

Course Code: CSE2273	**Course Title:** Cloud computing Lab **Type of Course :** Lab	**L- T-P- C**	0	**0**	2	**1**
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	**NIL**					
Course Description	This course is designed to give hands-on experience with cloud platforms, services, and deployment models. Students will learn to set up, configure, and manage cloud environments using platforms like AWS, Microsoft Azure, and Google Cloud. The course covers virtualization, containerization, serverless computing, cloud storage, security, and scalability. Through practical assignments, students will develop skills in deploying cloud applications, managing cloud resources, automating cloud workflows, and implementing cost-effective cloud solutions.					
Course Objective	The objective of the course is to Understand Cloud Infrastructure, deploy and manage virtual machines, implement cloud storage, develop and deploy cloud applications, optimize cost and performance.					
Course Outcomes	On successful completion of the course the students shall be able to: CO1· Deploy and Manage Cloud Resources. CO2. Develop and Deploy Cloud-based Applications CO3. Optimize Performance and Cost in the Cloud CO4. Implement Security and Automation in Cloud Environments					
Course Content:						
Module 1	Introduction to Cloud and Virtualization	Assignment	Virtualization			**10 Sessions**

Lab Assignment 1: Setting Up Virtual Machines on Cloud

- Create a **Virtual Machine (VM)** on AWS/Azure/GCP
  Configure OS, storage, and network settings
- Connect to the VM using SSH/RDP
  Install web server (Apache/Nginx) and deploy a static webpage

Lab Assignment 2: Containerization Using Docker

- Install **Docker** on a local or cloud VM
- Create and run a **Docker container**

- Build a **custom Docker image** with a simple Python/Node.js application
- Push the image to **Docker Hub** and deploy it on a new VM

Module 2	High Throughput and Data Intensive Computing	Assignment	Virtualization	**10 Sessions**

Lab Assignment 1: Setting Up a Distributed Computing Environment

- Launch a **Hadoop or Spark cluster** on AWS EMR / Azure HDInsight / Google Dataproc
- Configure **HDFS (Hadoop Distributed File System)** for big data storage
- Run a **basic MapReduce job** on sample data

Lab Assignment 2: Data Preprocessing with Cloud Storage

- Store large datasets in **Amazon S3 / Azure Blob Storage / Google Cloud Storage**
- Use **Apache Spark or Hadoop** to read, clean, and process data
- Convert datasets into **Parquet or Avro formats** for efficient storage

Lab Assignment 3: Batch Processing with Apache Spark

- Load large datasets (e.g., logs, tweets, transaction data) into **Spark DataFrame**
- Perform **ETL (Extract, Transform, Load) operations** on the data
- Use **SparkSQL** for querying large datasets

Lab Assignment 4: Real-Time Data Processing with Spark Streaming

- Set up **Kafka / AWS Kinesis / Google Pub/Sub** for real-time data ingestion
- Process streaming data using **Spark Streaming**
- Perform windowed aggregations and visualize real-time trends

Lab Assignment 5: Cloud-Based Machine Learning with Big Data

- Use **Google BigQuery ML / AWS SageMaker / Azure Machine Learning** for model training
- Train a **linear regression or classification model** on a large dataset
- Deploy the trained model as an API for real-time predictions

Lab Assignment 6: Running Parallel Machine Learning Workloads

- Implement **distributed ML training** using **Spark MLlib or TensorFlow on Cloud TPUs**
- Train models on a large dataset and optimize performance using distributed execution

Lab Assignment 7: Auto-Scaling and Load Balancing for Data Processing

- Deploy a **serverless Spark job** using **AWS Glue / Azure Synapse**
- Implement **auto-scaling for high-throughput jobs**
- Measure performance improvements using cloud monitoring tools

Lab Assignment 8: Cost Optimization for High-Throughput Data Processing

- Analyze **cloud cost reports** for data-intensive workloads
- Optimize cloud storage and compute resources for cost-efficiency
- Compare **on-demand vs. reserved vs. spot instances** for cost savings

Module 3	Cloud Security and Standards	Assignment	Virtualization	**9 Sessions**

Lab Assignment 9: Configuring Identity and Access Management (IAM)

- Set up **IAM roles and policies** in AWS / Azure / Google Cloud
- Create and assign users with **least privilege access**
- Implement **Multi-Factor Authentication (MFA)** for added security
- Audit IAM policies using **AWS IAM Access Analyzer / Azure Security Center**

Lab Assignment 10: Setting Up Single Sign-On (SSO) and Role-Based Access Control (RBAC)

- Configure **AWS Cognito / Azure Active Directory / Google IAM** for authentication
- Implement **Role-Based Access Control (RBAC)** for users and groups
- Integrate **OAuth 2.0 / OpenID Connect (OIDC) / SAML** for secure authentication

Lab Assignment 11: Encrypting Data at Rest and in Transit

- Encrypt **cloud storage (S3, Blob, Cloud Storage)** using **KMS (Key Management Service)**
- Set up **TLS/SSL certificates** for secure web traffic encryption
- Enable **database encryption (AWS RDS, Azure SQL, GCP Cloud SQL)**

Lab Assignment 12: Implementing Compliance & Governance in Cloud

- Enable **GDPR, HIPAA, ISO 27001 compliance tools** in cloud platforms
- Use **AWS Config / Azure Policy / GCP Security Command Center** to enforce compliance
- Conduct **security audits and generate compliance reports**

Lab Assignment 13: Implementing Cloud Monitoring & Threat Detection

- Configure **AWS CloudTrail / Azure Monitor / GCP Operations Suite** for activity logging
- Set up **intrusion detection systems (IDS) & anomaly detection**
- Analyze security logs using **Amazon GuardDuty / Azure Sentinel / Chronicle Security**

Lab Assignment 14: Automating Security Incident Response

- Deploy a **Serverless Lambda / Azure Logic App** to automatically respond to security incidents
- Implement **automated alerts** for suspicious activity
- Test a **denial-of-service (DDoS) simulation** and implement mitigation strategies

Module 4	Cloud Platforms	Assignment	Virtualization	**9 Sessions**

Lab Assignment 15: Getting Started with Cloud Platforms

- Create a **free-tier account** on AWS, Azure, or Google Cloud
- Navigate the **Cloud Console, CLI, and SDKs**
- Explore and configure **dashboard, billing, and IAM settings**

Lab Assignment 16: Launching a Virtual Machine (VM) on Cloud

- Deploy a **VM instance** using **AWS EC2, Azure Virtual Machines, or Google Compute Engine**
  Configure **OS, storage, networking, and security groups**
  Connect to the instance using **SSH (Linux) or RDP (Windows)**

Lab Assignment 17: Cloud Storage and File Management

- Create **Object Storage (AWS S3 / Azure Blob Storage / Google Cloud Storage)**
  Upload, download, and set access permissions for files
  Implement **Lifecycle Policies and Versioning**

Lab Assignment 18: Cloud Database Management

- Deploy a **Relational Database (AWS RDS / Azure SQL Database / Cloud SQL)**
  Connect and query the database using **MySQL/PostgreSQL clients**
  Set up database **backups and automatic scaling**

Lab Assignment 19: Configuring Virtual Networks in Cloud

- Set up a **Virtual Private Cloud (VPC) / Azure Virtual Network / GCP VPC**
  Configure **subnets, firewalls, and security groups**
  Test **network communication between two VMs**

Lab Assignment 20: Deploying a Web Application on Cloud

Deploy a Python/Node.js/Java web app using:

- **AWS Elastic Beanstalk**
- **Azure App Service**

• **Google App Engine** **Connect the app to Cloud Database (RDS, CosmosDB, Firestore)** **Monitor application performance and logs**
Targeted Application & Tools that can be used:
Text Book(s): 1. John Rittinghouse and James Ransome, "Cloud Computing, Implementation, Management and Security", CRC Press. 2. Rajkumar Buyya, Christian Vecchiola, and Thamarai Selvi, "Mastering Cloud Computing", McGraw Hill Education.
Reference(s): 1. David E.Y. Sarna, "Implementing and Developing Cloud Applications", CRC Press. 2. Anthony T Velte, Toby J Velte, Robert Elsenpeter, "Cloud Computing: A Practical Approach", Tata McGraw-Hill. Web resources: https://presiuniv.knimbus.com/user#/home

Course Code: **CSE2506**	**Course Title:** Mobile Application Development Lab **Type of Course:** Lab	**L- T-P- C**	0	**0**	4	2
Version No.	2.0					
Course Pre-requisites	CSE1006					
Anti-requisites	**NIL**					
Course Description	The course provides hands-on experience in designing, developing, and deploying mobile applications for Android and iOS platforms. Students will work with native development frameworks such as Android Studio (Java/Kotlin) and Xcode (Swift), as well as explore cross-platform tools like Flutter or React Native.					
Course Objective	The objective of the course is to develop Native and Cross-Platform Mobile Applications, design Interactive and Responsive User Interfaces, integrate Backend Services and APIs, implement State Management and Performance Optimization, ensure Mobile App Security and Data Protection					
Course Outcomes	On successful completion of the course the students shall be able to: 1. Develop Functional Mobile Applications 2. Design and Implement Interactive UIs 3. Integrate Cloud Services and APIs 4. Integrate Backend Systems and Data Management 5. Deploy, Publish, and Maintain advanced Mobile Application					
Course Content:						

Module 1	Introduction and Architecture of Android	Assignment	Simulation/Data Analysis	10 Sessions
1.a. Design an app to read user inputs using edit text and display the result of arithmetic operations using toast message. 1.b. Create an android app to calculate the current age of yourself, select your DOB using date picker. 2. Design an app to input your personal information. Use an autocomplete text view to select your place of birth.				
Module 2	User Interfaces, Intent and Fragments	Term paper/Assignment	Simulation/Data Analysis	15 Sessions
3. a. Design an app to select elective course using spinner view and on click of the display button, toast your ID and selected elective course. 3. b. Design a restaurant menu app to print the total amount of orders.				
Module 3	Components of Android	Term paper/Assignment	Simulation/Data Analysis	15 Sessions
4. Develop an android app that uses intent to maintain the following scenario. Check the eligibility criteria for voting. Input the Aadhar no., Name & age in the first activity. If the age is above 18, display the voter's detail in the second activity. Else, display, “You are not eligible to vote” in the second Activity. 5. Demonstrate the use of fragment with list of buttons representing various colors, and on click of these buttons, the appropriate color is filled in the next fragment. Create an Android application to input the vitals of a person (temperature, BP). If the vitals are abnormal, give proper notification to the user. 6. Create an android app to for movie ticket booking. Save the user name of the customer using shared preferences. After completion of booking, retrieve the username from the shared preferences and print the ticket details.				
Module 4	Notifications and Data Persistence	Term paper/Assignment	Simulation/Data Analysis	15 Sessions
7. Create an android application to manage the details of students' database using SQLite.Use necessary UI components, which perform the operations such as insertion, modification, removal and view. Presidency University needs an APP for Admission eligibility checking for students, for that you need to take the following information from the Student: registration ID, physics, chemistry and mathematics marks (PCM), fees is allotted as below criteria. PCM (Total marks %) Fee concession 90 above 80 % 70 to 89 60 % Below 69 % no concession On click on the button “Registration” details should be stored in the database using SQLite. Create button DISPLAY ALL (full students list) on click on the button it should display the students list per the fee concession. 8. A company need to design an app that plays soft music automatically in the background. Create an app to achieve this functionality.				

9. Create an android application such that your view object in the Activity can be Animated with fade-in effect. Create an appropriate XML file named fade-in and write the application to perform the property animation.

Module 5	Advance App Development	Term paper/Assignment	Simulation/Data Analysis	15 Sessions

10. Demonstrate how to send SMS and email.
11. Create an android application to transfer a file using WiFi. Create an android application "Where am I" with an Activity that uses the GPS Location provider to find the device's last known location.

Targeted Application & Tools that can be used:

Applications:

- Native Android Applications (Java/Kotlin)
  - Android Mobile Apps built for Android smartphones and tablets using Java or Kotlin programming languages.
  - Target audience: Android users.
- Native iOS Applications (Swift)
  - iOS Mobile Apps designed for iPhone and iPad using Swift.
  - Target audience: iOS users (Apple ecosystem).
- Cross-Platform Mobile Apps (Flutter, React Native)
  - Cross-platform apps designed to run on both Android and iOS from a single codebase using frameworks like Flutter or React Native.
  - Target audience: Users on both Android and iOS platforms.
- Mobile Web Applications (Progressive Web Apps - PWA)
  - Mobile-optimized web applications using HTML5, CSS3, and JavaScript that run in a browser with native-like functionality (offline support, push notifications).
  - Target audience: Users accessing apps via mobile browsers.

Development Tools and Frameworks

- Integrated Development Environments (IDEs)
  - Android Studio (for Android): The official IDE for Android development, supporting Java, Kotlin, and Android SDK.
  - Xcode (for iOS): The official IDE for iOS development with Swift and Objective-C, providing a comprehensive suite of development tools for iPhone/iPad applications.
  - Visual Studio Code (VS Code): Lightweight IDE for working with Flutter, React Native, and web development projects.
- Cross-Platform Development Frameworks
  - Flutter: Open-source UI framework by Google for building natively compiled applications for mobile, web, and desktop from a single codebase.

○ React Native: Open-source framework developed by Facebook for building cross-platform apps with JavaScript and React.

- Backend & Cloud Tools
  - ○ Firebase: Google's backend-as-a-service (BaaS) platform offering authentication, real-time databases, cloud storage, and push notifications for mobile apps.
  - ○ AWS Amplify: Cloud platform for backend services (API, storage, authentication) and mobile deployment.
  - ○ SQLite / Realm: Local storage solutions for mobile apps to manage data storage and retrieval on-device.
- Mobile App Testing and Debugging Tools
  - ○ Android Emulator (for Android): A virtual device to run and test Android apps without needing physical devices.
  - ○ Xcode Simulator (for iOS): A tool to simulate different iOS devices and test apps during development.
  - ○ Appium: Open-source tool for automated testing across native, hybrid, and mobile web applications.
- Version Control and Collaboration
  - ○ Git: Version control system for managing code changes and collaborating with teams.
  - ○ GitHub / GitLab / Bitbucket: Online platforms for hosting Git repositories, collaboration, and version control management.
- Mobile App Deployment Tools
  - ○ Google Play Console: For managing Android app publishing, distribution, and monitoring.
  - ○ Apple App Store Connect: For managing iOS app submissions, reviews, and releases on the Apple App Store.
- UI/UX Design Tools
  - ○ Figma / Adobe XD: Tools for UI/UX design and wireframing to create the visual elements of mobile applications before development.
  - ○ Sketch: Vector-based design tool for iOS UI design and prototyping

Text Book(s):

T1. Pradeep kothari "Android Application Development - Black Book", dreamtechpress
T2. Barry Burd (Author), "Android Application Development" ALL – IN – ONE FOR Dummies
T3. Jeff Mcherter (Author),Scott Gowell (Author), "Professional mobile Application Development" paperback, Wrox - Wiley India Private Limited
T4. Wei-Meng Lee (Author) "Beginning Android Application Development" Wrox – Wiley India Private Limited

Reference(s):

1. Bill Phillips, Chris Stewart, and Kristin Marsicano (Author) "Android Programming" 3rd edition, 2017.The Big Nerd Ranch Guide, Big Nerd Ranch LLC, 5. The Big Nerd Ranch Guide, by"
2. Erik Hellman, "Android Programming – Pushing the Limits", 1st Edition, Wiley India Pvt Ltd, 2014.
3. Dawn Griffiths and David Griffiths, "Head First Android Development", 1st Edition, O'Reilly SPD Publishers, 2015.
4. J F DiMarzio, "Beginning Android Programming with Android Studio", 4th Edition, Wiley India Pvt Ltd, 2016. ISBN-13: 978-8126565580
5. Anubhav Pradhan, Anil V Deshpande, " Composing Mobile Apps" using Android, Wiley 2014,

ISBN: 978-81-265-4660-2
6. Reto Meier “Professional Android Application Development”
E-Resources: https://puniversity.informaticsglobal.com/login Or http://182.72.188.193/

Course Code: **CSE2505**	**Course Title:** Mobile Application Development **Type of Course:** Theory	**L- T-P- C**	2	**0**	0	2
Version No.	2.0					
Course Pre-requisites	CSE1006					
Anti-requisites	**NIL**					
Course Description	The course deals with the basics of android platform and application life cycle. The goal of the course is to develop mobile applications with Android containing at least one of the following phone material components: GPS, accelerometer or phone camera, use simple GUI applications and work with database to store data locally or in a server. Topics include user interface design; user interface building; input methods; data handling; network techniques and URL loading; GPS and motion sensing. Android application framework and deployment. Power management, Screen resolution, Touch interface, Store data on the device.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of Mobile Applications and Development as mentioned above and attain Employability Skills through Experiential Learning Techniques.					
Course Outcomes	On successful completion of the course the students shall be able to: 1. Discuss the fundamentals of mobile application development and its architecture. (Comprehension) 2. Illustrate mobile applications with appropriate android view. (Application) 3. Demonstrate the use of services, broadcast receiver, Notifications and content provider.(Application) 4. Apply data persistence techniques, to perform CRUD operations. (Application) 5. Use advanced concepts for mobile application development. (Application)					
Course Content:						
Module 1	Introduction and Architecture of Android	Assignment	Simulation/Data Analysis	10 Sessions		
Topics: Android: History and features, Architecture, Development Tools, Android Debug Bridge (ADB), and Life cycle.						
Module 2	User Interfaces, Intent and Fragments	Term paper/Assignment	Simulation/Data Analysis	15 Sessions		

Topics: Views, Layout, Menu, Intent and Fragments.				
Module 3	Components of Android	Term paper/Assignment	Simulation/Data Analysis	15 Sessions
Topics: Activities, Services, Broadcast receivers, Content providers, User Navigation				
Module 4	Notifications and Data Persistence	Term paper/Assignment	Simulation/Data Analysis	15 Sessions
Topics: Notification, Shared Preferences, SQLite database, Android Room with a View, Firebase.				
Module 5	Advance App Development	Term paper/Assignment	Simulation/Data Analysis	15 Sessions
Topics: Graphics and Animation, App Widgets, Sensors, Performance, Location, Places, Mapping, Custom Views, Canvas.				
Targeted Application & Tools that can be used: Applications: Native Android Applications Native iOS Applications Cross Platform mobile Apps Mobile web Applications				
Text Book(s): T1. Pradeep kothari "Android Application Development - Black Book", dreamtechpress T2. Barry Burd (Author), "Android Application Development" ALL – IN – ONE FOR Dummies T3. Jeff Mcherter (Author),Scott Gowell (Author), "Professional mobile Application Development" paperback, Wrox - Wiley India Private Limited T4. Wei-Meng Lee (Author) "Beginning Android Application Development" Wrox – Wiley India Private Limited				

Reference(s):
1. Bill Phillips, Chris Stewart, and Kristin Marsicano (Author) "Android Programming" 3rd edition, 2017.The Big Nerd Ranch Guide, Big Nerd Ranch LLC, 5. The Big Nerd Ranch Guide, by"
2. Erik Hellman, "Android Programming – Pushing the Limits", 1st Edition, Wiley India Pvt Ltd, 2014.
3. Dawn Griffiths and David Griffiths, "Head First Android Development", 1st Edition, O'Reilly SPD Publishers, 2015.
4. J F DiMarzio, "Beginning Android Programming with Android Studio", 4th Edition, Wiley India Pvt Ltd, 2016. ISBN-13: 978-8126565580
5. Anubhav Pradhan, Anil V Deshpande, " Composing Mobile Apps" using Android, Wiley 2014, ISBN: 978-81-265-4660-2
6. Reto Meier "Professional Android Application Development"
E-Resources: https://puniversity.informaticsglobal.com/login Or http://182.72.188.193/

Course Code: **PPS3019**	**Course Title: Corporate Communication** Type of Course: Practical Only Course	L- T - P- C	0	0	2	1
Version No.	1.0					
Course Pre-requisites	• Students are expected to understand Basic English. • Students should have desire and enthusiasm to involve, participate and learn.					
Anti-requisites	NIL					
Course Description	This course is designed to enable students to understand concepts of corporate communication, improve confidence, and develop professional skills. This course will give the students a competitive advantage and increase their chances of success in the professional world by building their strong resumes and better online presence. This will benefit learners in effectively presenting their concepts and ideas in a corporate environment through various activities and learning methodologies.					
Course Objective	The objective of the course is to **familiarize the learners with the concepts of "Corporate Communication" and attain SKILL DEVELOPMENT through PARTICIPATIVE LEARNING techniques.**					

Course Out Comes	**On successful completion of this course, the students shall be able to:** CO1: Demonstrate effective communication while presenting oneself and others CO2: Design resume based on industrial standards CO3: Employ techniques to face interview and Group discussion		
Course Content:			
Module 1	Introduction to Corporate Communication	Classroom activity	6 Hours
Topics: Setting Expectations, Ice Breaker, Cs of communication, Significance of corporate communication, Communication process, barriers in communication and solutions, Structuring messages, non-verbal communication, written communication, Self introduction.			
Module 2	Presentation Skills	Group Presentation	8 Hours

Topics: Presentation Skills, Opening Body & Closing Body, Audibility, Speech Clarity, Fluency, Voice Modulation, Non-verbal Communication and Body Language.			
Module 3	Resume Writing	Individual Assessment	8 Hours
Topics: Importance of a Resume, Types of Resumes (Chronological, Functional, and Combination), Understanding Job Descriptions, Optimizing for Applicant Tracking Systems (ATS), Understanding ATS Requirements, Formatting Tips for ATS Compliance, The Digital Edge: Online Profiles (Aligning Resumes with LinkedIn Profiles)			
Module 4	Personal Interview	Individual Assessment	8 Hours
Understanding Personal Interviews, Self-Awareness and Preparation (SWOT Analysis: Knowing Your Strengths and Weaknesses), **Handling Common Questions (**Tell Me About Yourself: Structuring a Strong Response, Answering Situational and Behavioral Questions (STAR Method) **Practical Skills for Success (**Dressing for Success: Professional Appearance, Mock Interviews: Practice with Feedback, Follow-Up Etiquette: Thank You Emails and Calls)			
Targeted Application & Tools that can be used: 1. TED Talks 2. You Tube Links 3. Activities			
Project work/Assignment: Mention the Type of Project /Assignment proposed for this course			

1) Resume
2) Self-Introduction
3) Personal Interview

Targeted Application & Tools that can be used:

1. TED Talks
2. YouTube Links

Assignments proposed for this course

1. Self-Introduction
2. Resume

YouTube Links: https://youtu.be/z_jxoczNWc (Steve Jobs Introducing the iPhone 4 in June 2010)

References

1. "Talk Like TED - The 9 Public-Speaking Secrets of the World's Top Minds" By Carmine Gallo St. Martin's Press Copyright © 2014 Carmine Gallo All rights reserved. ISBN: 978-1-250-04112-8
2. "The Presentation Secrets of Steve Jobs: How to Be Insanely Great in Front of Any Audience" MP3 CD – Import, 22 April 2014
3. "The Definitive Book of Body Language: The Hidden Meaning Behind People's Gestures and Expressions" Hardcover – Illustrated, 25 July 2006
4. "Crucial Conversations: Tools for Talking When Stakes Are High" Paperback – Import, 1 July 2002

Web links:

1. https://www.wordstream.com/blog/ws/2014/11/19/how-to-improve-presentation-skills

https://www.cbs.de/en/blog/15-effective-presentation-tips-to-improve-presentation-skills/

2. https://hbr.org/2022/05/the-art-of-asking-great-questions

Topics relevant to development of "SKILL": Art of Presentation, Team building, Art of questioning and Personal Branding for Skill Development through Participative Learning Techniques. This is attained through assessment component mentioned in course handout.

Course Code: CSE2271	**Course Title: Software Design and Development** **Type of Course: Theory**		**L-T- P- C**	**3-0-0-3**
Version No.	1.0			
Course Pre-requisites	**NIL**			
Anti-requisites	**NIL**			
Course Description	The objective of this course is to provide the fundamentals concepts of Software Engineering process and principles. The course covers software requirement engineering processes, system analysis, design, implementation and testing aspects of software system development. The course covers software quality, configuration management and maintenance.			
Course Objectives	The objective of the course is to familiarize the learners with the concepts of Software Engineering and attain Skill Development through Participative Learning techniques.			
Course Out Comes	On successful completion of this course the students shall be able to: 1] Describe the Software Engineering principles, ethics and process models(Knowledge) 2] Identify the requirements, analysis and appropriate design models for a given application(Comprehension) 3] Understand the Agile Principles(Knowledge) 4] Apply an appropriate planning, scheduling, evaluation and maintenance principles involved in software(Application)			
Module 1	**Introduction to Software Engineering and Process Models (Knowledge level)**	Quiz		**10 Hours**
Introduction: Need for Software Engineering, Professional Software Development, Software Engineering Ethics, Software Engineering Practice-Essence of Practice, General Principles Software Development Life Cycle **Models:** Waterfall Model – Classical Waterfall Model, Iterative Waterfall Model, Evolutionary model-Spiral, Prototype.				
Module 2	**Software Requirements, Analysis and Design (Comprehension level)**	Assignment	Development of SRS documents for a given scenario	**12 Hours**
Requirements Engineering: Eliciting requirements, Functional and non- Functional requirements, Software Requirements Specification (SRS), Requirement Analysis and validation. Requirements modelling- Introduction to Use Cases, Activity diagram and Swim lane diagram. CASE support in Software Life Cycle, Characteristics of CASE Tools, Architecture of a CASE Environment. **Design:** Design concepts, Architectural design, Component based design, User interface design.				
Module 3	**Agile Principles & Devops (Knowledge level)**	Quiz		**10 Hours**
Agile: Scrum Roles and activities, Sprint Agile software development methods - Scaling, User Stories, Agile estimation techniques, Product backlogs, Stake holder roles, Dynamic System Development Method. **Devops:** Introduction, definition, history, tools.				
Module 4	**Software Testing and Maintenance**	Assignment	Apply the testing concepts using Programing	**13 Hours**

	(Application Level)			
Software Testing-verification and validation, Test Strategies - White Box Testing, Black box Testing. Automation Tools for Testing. **Software Quality Assurance**-Elements of software quality assurance, SQA Tasks, Goals and Metrics, Software configuration management- SCM process, SCM Tools (GitHub). **Maintenance-** Characteristics of Software Maintenance, Software Reverse Engineering, Software Maintenance Process Models.				
Targeted Application & Tools that can be used: Selenium, GitHub, CASE Tools				
Text Book 1] R1. Roger S. Pressman, "Software Engineering – A Practitioner's Approach", VII Edition, McGraw-Hill, 2017. 2] B2. Bob Hughes, Mike Cotterell, Rajib Mall, "Software Project Management", VI Edition, McGraw-Hill, 2018.				
References 1. Rajib Mall, "Fundamentals of Software Engineering", VI Edition, PHI learning private limited, 2015. 2. Ian Sommerville, "Software Engineering", IX Edition, Pearson Education Asia, 2011. 3. Agile Software Development Principles, Patterns and Practices.1st Edition, Wiley, 2002				
Topics Relevant to "Skill Development: Balck box Testing, White box Testing, Automated Testing for Skill development through Participative Learning Techniques. This is attained through assessment mentioned in the course handout				

Course Code: CSE2267	**Course Title: Machine Learning Techniques** **Type of Course: Program Core -Theory**	**L-T-P-C**	3	0	0	3
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	NIL					
Course Description	Machine Learning algorithms are the key to developing intelligent systems such as Apple's Siri, Google's self-driving cars, and more. This course introduces the core concepts and essential algorithms of various machine learning techniques, including Regression learning, Bayesian learning, Ensemble learning, Perceptron learning (foundations of Neural Networks), Unsupervised learning, Competitive learning, learning from Gaussian Mixture Models, and learning to detect outliers. The course lectures cover both the theoretical foundations and practical algorithms for these learning methods. Lab sessions complement the lectures, enabling students to develop intelligent systems for real-life problems using Python libraries.					
Course Objectives	The primary objective of this course is to enhance the **EMPLOYABILITY** of students by leveraging **EXPERIENTIAL LEARNING** techniques. Students will gain practical skills in applying machine learning to real-world challenges.					
Course Out Comes	On successful completion of this course the students shall be able to: 15. **Apply** advanced supervised machine learning methods for predictive modeling. [APPLY] 16. **Produce** machine learning models with better predictive performance using meta-learning (ensemble) algorithms. [APPLY] 17. **Create** predictive models using Perceptron learning algorithms, understanding their foundational role in neural networks. [APPLY] 18. **Employ** advanced unsupervised learning algorithms for clustering, competitive learning, and outlier detection. [APPLY] 19. **Implement** machine learning-based intelligent models using Python libraries and frameworks. [APPLY]					

Course Content:

Module 1	**Supervised Learning**	**Assignment**	**Module Tests**	**No. of Sessions: 12**

Overview of Machine Learning (ML): ML workflow, types of ML (supervised, unsupervised, reinforcement), types of features, Feature Engineering (Data Imputation Methods). Regression: Introduction, Simple Linear Regression, Loss Functions (MSE, MAE, RMSE), Polynomial Regression, Logistic Regression, Softmax Regression with Cross-Entropy as cost function. Bayesian Learning: Bayes Theorem, estimating conditional probabilities for categorical and continuous features, Naïve Bayes for supervised learning, Bayesian Belief Networks. Support Vector Machines (SVM): Soft margin and kernel tricks (Polynomial, RBF, Sigmoid). Evaluation Methodologies: Testing Dataset, Train-Validation-Testing split, N-Fold Cross Validation (K-Fold, Stratified K-Fold)..

Module 2	**Ensemble Learning**	**Assignment**	**Module Tests**	**No. of Sessions: 11**

Introduction to Ensemble Learning: Motivation, bias-variance trade-off. Bagging: Using subsets of instances (Bagging, Pasting), using subsets of features (Random Patches, Random Subspaces method). Ensemble Methods: Voting Classifier (Hard and Soft Voting), Random Forest (Algorithm, Feature Importance). Boosting: AdaBoost (Adaptive Boosting), Gradient Boosting Machines (GBM), Extremely Randomized Trees, Stacking (Meta-Learning).

Module 3	**Perceptron Learning**	**Assignment**	**Module Tests**	**No. of**

	& Neural Networks Foundation			**Sessions: 11**

From Biological to Artificial Neurons: Basic structure of a neuron. Perceptrons: Single-layer Perceptrons, Linear Threshold Units, logical computations with Perceptrons (AND, OR, NOT). Activation Functions: Common activation functions (Sigmoid, Tanh, ReLU, Leaky ReLU, ELU, Softmax) and their properties. Loss Functions: Common loss functions for classification (Binary Cross-Entropy, Categorical Cross-Entropy) and regression (MSE, MAE). Multi-layer Perceptrons (MLP): Architecture, advantages over single Perceptrons.

Backpropagation Algorithm: Understanding the concept of backpropagation, Gradient Descent and its variants (Stochastic Gradient Descent, Mini-batch Gradient Descent, Adam, RMSprop).

Module 4	**Unsupervised Learning & Advanced Topics**	**Assignment**	**Module Tests**	**No. of Sessions: 11**

Clustering: Simple K-Means Clustering (Algorithm, Simple and Mini-Batch), updating centroids incrementally. K-Means Enhancements: Finding the optimal number of clusters (Elbow Method, Silhouette Coefficient), drawbacks of K-Means, K-Means++. Hierarchical Clustering: Divisive hierarchical clustering (Bisecting K-Means), clustering using Minimum Spanning Tree (MST). Competitive Learning: Clustering using Kohonen's Self-Organizing Maps (SOM). Density-Based Clustering: DBSCAN (Density-Based Spatial Clustering of Applications with Noise). Probabilistic Clustering: Clustering using Gaussian Mixture Models (GMM) with Expectation-Maximization (EM) algorithm. Outlier Detection Methods: Isolation Forest, Local Outlier Factor (LOF). Association Rule Mining: Introduction, Apriori Algorithm (concepts only). Collaborative Filtering: Introduction (User-based, Item-based).

Targeted Application & Tools that can be used:

7. Google Colab
8. Python IDEs like PyCharm

Teaching Methodology

6. **Lectures:** Interactive lectures covering theoretical foundations and algorithmic details.
7. **Lab Sessions:** Hands-on sessions focusing on implementing algorithms using Python libraries (e.g., scikit-learn, NumPy, Pandas, Matplotlib, Seaborn, Keras/TensorFlow for Perceptron).
8. **Problem-Based Learning:** Real-world case studies and problems will be discussed and solved in labs.
9. **Experiential Learning:** Emphasis on practical application through assignments, mini-projects, and a final course project.
10. **Discussions:** Encouraging critical thinking and peer learning through in-class discussions.

Project work/Assignment:

2. Students can be assigned a mini project to develop a machine learning application for real-life problems in various domains such as health care, business intelligence, environmental modeling, etc.

Textbook(s):

T1. Aurélien Géron. *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow*, Oreilly, 3rd Edition, 2022.

References:

R1. Raschka, S., Liu, Y., & Mirjalili, V. (2022). Machine Learning with PyTorch and Scikit-Learn: Develop machine learning and deep learning models with Python. Packt Publishing Ltd.
R2. Alpaydin, E. (2021). Machine Learning, revised and updated edition. MIT Press.
R3. Deisenroth, M. P., Faisal, A. A., & Ong, C. S. (2020). Mathematics for machine learning. Cambridge University Press.
R4. Andreas C Muller, and Sarah Guido. *Introduction to Machine Learning with Python: A Guide for Data Scientists*, O'Reilly, 1st Edition, 2016.

Weblinks

W8. W1. NPTEL Courses: https://nptel.ac.in/courses/106106139 (IIT M), https://nptel.ac.in/courses/106105152 (IIT Kgp)

W9. Scikit-learn documentation: https://scikit-learn.org/stable/

W10. TensorFlow documentation: https://www.tensorflow.org/

W11. Keras documentation: https://keras.io/

W12. Pandas documentation: https://pandas.pydata.org/

W13. NumPy documentation: https://numpy.org/

W14. Kaggle: For datasets and competitions.

Course Code: CSE2268	**Course Title: Machine Learning Techniques Lab** **Type of Course: Program Core -Laboratory**	**L-T-P-C**	0	0	2	1
Version No.	1.0					
Course Pre-requisites	CSE1500 – Computational Thinking with Python CSE3157 – Artificial Intelligence and Machine Learning					
Anti-requisites	NIL					
Course Description	Machine Learning algorithms are the key to develop intelligent systems such as Apple's Siri, Google's self-driving cars etc. This course introduces the concepts of the core machine learning techniques such as Regression learning, Bayesian learning, Ensemble learning, Perceptron learning, Unsupervised learning, Competitive learning, learning from Gaussian mixture models and learning to detect outliers. Course lectures covers both the theoretical foundations as well as the essential algorithms for the various learning methods. Lab sessions complement the lectures and enable the students in developing intelligent systems for real life problems.					
Course Objectives	The objective of the course is EMPLOYBILITY of student by using EXPERIENTIAL LEARNING techniques.					
Course Out Comes	On successful completion of this course the students shall be able to: 20. **Apply** advanced supervised machine learning methods for predictive modeling. [Apply] 21. **Produce** machine learning models with better predictive performance using meta learning algorithms [Apply] 22. **Create** predictive models using Perceptron learning algorithms [Apply] 23. **Employ** advanced unsupervised learning algorithms for clustering, competitive learning and outlier detection [Apply] 24. **Implement** machine learning based intelligent models using Python libraries. [Apply]					

Course Content: **No. of Sessions: 15 (30 hours)**

Experiment No. 1: File Handling Using Python
Level 1: Read a CSV file using Python
Level 2: Read a text file using Python

Experiment No. 2: Methods for handling missing values
Level 1: Given a data set from UCI repository, implement the different ways of handling missing values in it using Scikit-learn library of Python
Level 2: Implement one of these methods using a custom defined function in Python.

Experiment No. 3: Data Visualization
Level 1: Perform Exploratory Data Analysis for a given data set by creating Scatter Plot, Pair Plot, Count Plot using Matplotlib and Seaborn
Level 2: Create Heat Maps, WordCloud

Experiment No. 4: Regression learning
Level 1: Given a data set from UCI repository, implement the simple linear regression algorithm and estimate the models parameters and the performance metrics. Plot the learning curves.
Level 2: Implement the polynomial regression algorithm. Compare the learning curves of Polynomial

and
Linear Regression.

Experiment No. 5: Logistic Regression
Level 1: Write custom code for generating the logistic/sigmoid plot for a given input
Level 2: Given a data set from UCI repository, implement the Logistic regression algorithm. Estimate the class probabilities for a given test data set. Plot and analyze the decision boundaries.

Experiment No. 6: Bayesian Learning
Level 1: Given a data set from UCI repository, implement a classification model using the Bayesian algorithm.
Level 2: Implement a Naïve Bayes classifier using 5-fold cross-validation

Experiment No. 7: Support Vector Machine (SVM)
Level 1: Given data sets from UCI repository, implement a linear SVM and a non-linear SVM based classification model.
Level 2: Construct kernels with 5-fold cross-validation for SVM.

Experiment No. 8 & 9: Ensemble Learning
Level 1: Implement Ensemble Learning algorithms such as Bagging, Pasting and Out-of Bag Evaluation
Level 2: Random Patches and Random Subspace Method, Adaboost and Gradient Boosting, Stacking.

Experiment No. 10: Perceptron Learning
Level 1: Implement the Perceptron Classifier
Level 2: An Image Classifier Using the Sequential API of Keras

Experiment No. 11 & 12: Unsupervised Learning
Level 1: K-means – simple and mini-batch. Finding the optimal number of clusters using Elbow method and Silhoutte Coefficient . Compare the inertia of both as k increases. Tuning the hyperparameter 'k' using GridSearchCV.
Level 2: Using clustering for Image segmentation and Preprocessing. Kmeans++

Experiment No. 13: Density Based Clustering
Level 1: Implement DBSCAN – clustering using the local density estimation. Perform hard and soft clustering for new instances.
Level 2: Outlier Detection using Isolation Forest and Local Outlier Factor

Experiment No. 14: Association Rule Mining
Level 1: Implement the Apriori Algorithm for Association Rule Mining
Level 2: Implement the Dynamic Itemset Counting Algorithm for Association Rule Mining.

Experiment No. 15: Collaborative Filtering
Level 1: Implement Collaborative Filtering using Item-Based Filtering
Level 2: Implement Collaborative Filtering using User-Based Filtering

Targeted Application & Tools that can be used:

9. Google Colab
10. Python IDEs like PyCharm

Project work/Assignment: Mention the Type of Project /Assignment proposed for this course

Students can be assigned a mini project to develop a machine learning application for real-life problems in various domains such as health care, business intelligence, environmental modeling, etc.

Textbook(s):

4. Aurélien Géron. *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow*, Oreilly, 3rd Edition, 2022.

Course Code: CSE 7100	**Course Title:** Mini Project **Type of Course:**	**L- T-P- C**	0	0	0	4
Version No.	**1.0**					
Course Pre-requisites	**NIL**					
Anti-requisites	**NIL**					
Course Description	Students observe science and technology in action, develop an awareness of the method of scientific experimentation, and often get an opportunity to see, study and operate sophisticated and costly equipment. They also learn about the implementation of the principles of management they have learnt in class, when they observe multidisciplinary teams of experts from engineering, science, economics, operations research, and management deal with techno-economic problems at the micro and macro levels. Finally, it enables them to develop and refine their language, communication and inter-personal skills, both by its very nature, and by the various evaluation components, such as seminar, group discussion, project report preparation, etc. The broad-based core education, strong in mathematics and science and rich in analytical tools, provides the foundation necessary for the student to understand properly the nature of real-life problems. The students have options to pursue this course as either Project Work and Dissertation at the university, or Project Work in an Industry/ Company/ Research Laboratory, or Internship Program in an Industry/Company.					
Course Objectives	The objective of the course is to familiarize the learners with the concepts of Professional Practice and attain Employability Skills through Experiential Learning techniques.					
Course Outcomes	On successful completion of this course the students shall be able to: 1. Identify the engineering problems related to local, regional, national or global needs. (Understand) 2. Apply appropriate techniques or modern tools for solving the intended problem. (Apply) 3. Design the experiments as per the standards and specifications. (Analyze) 4. Interpret the events and results for meaningful conclusions. (Evaluate) 5. Appraise project findings and communicate effectively through scholarly publications. (Create)					

Course Code: CSE 7300	**Course Title:** Capstone Project **Type of Course:**	**L- T-P- C**	0	0	0	10
Version No.	**1.0**					
Course Pre-requisites	**NIL**					
Anti-requisites	**NIL**					
Course Description	Students observe science and technology in action, develop an awareness of the method of scientific experimentation, and often get an opportunity to see, study and operate sophisticated and costly equipment. They also learn about the implementation of the principles of management they have learnt in class, when they observe multidisciplinary teams of experts from engineering, science, economics, operations research, and management deal with techno-economic problems at the micro and macro levels. Finally, it enables them to develop and refine their language, communication and inter-personal skills, both by its very nature, and by the various evaluation components, such as seminar, group discussion, project report preparation, etc. The broad-based core education, strong in mathematics and science and rich in analytical tools, provides the foundation necessary for the student to understand properly the nature of real-life problems. The students have options to pursue this course as either Project Work and Dissertation at the university, or Project Work in an Industry/ Company/ Research Laboratory, or Internship Program in an Industry/Company.					
Course Objectives	The objective of the course is to familiarize the learners with the concepts of Professional Practice and attain Employability Skills through Experiential Learning techniques.					
Course Outcomes	On successful completion of this course the students shall be able to: 1. Identify problems based on societal /research needs. (Understand) 2. Apply Knowledge and skill to solve societal problems in a group. (Apply) 3. Develop interpersonal skills to work as member of a group or leader. (Apply) 4. Analyze the inferences from available results through theoretical / Experimental / Simulations. (Analyze) 5. Analyze the impact of solutions in societal and environmental context for sustainable development. (Analyze) 6. Improve in written and oral communication. (Create) 7. Demonstrate capabilities of self-learning in a group, which leads to lifelong learning. (Understand)					

Course Code: CSE3500	Course Title: Intelligent Systems with Machine Learning Type of Course: Lab	L- T-P- C	2	0	2	3
Version No.	1.0					
Course Pre-requisites	CSE2267					
Anti-requisites	**NIL**					
Course Description	Machine Learning algorithms are the key to develop intelligent systems such as Apple's Siri, Google's self-driving cars etc. This course introduces the concepts of the core machine learning techniques such as Regression learning, Bayesian learning, Ensemble learning, Perceptron learning, Unsupervised learning, Competitive learning, learning from Gaussian mixture models and learning to detect outliers. Course lectures covers both the theoretical foundations as well as the essential algorithms for the various learning methods. Lab sessions complement the lectures and enable the students in developing intelligent systems for real life problems.					
Course Objectives	This course is designed to improve the learners 'EMPLOYABILITY SKILLS' by using EXPERIENTIAL LEARNING techniques. The supervised hands-on laboratory exercises, assessments and the group projects facilitate this learning process.					
Course Out Comes	On successful completion of the course the students shall be able to: 1] Apply advanced supervised machine learning methods for predictive modeling. [Application] 2] Produce machine learning models with better predictive performance using meta learning algorithms [Application] 3] Create predictive models using Perceptron learning algorithms[Application] 4] Employ advanced unsupervised learning algorithms for clustering, competitive learning and outlier detection[Application] 5] Implement machine learning based intelligent models using Python libraries. [Application]					
Course Content:						

Module 1	Supervised Learning	Assignment		Programming using Keras/Sklearn	**No. of Classes L – 7 P – 12**
	Topics: An overview of Machine Learning(ML); ML workflow; types of ML; Types of features, Feature Engineering -Data Imputation Methods; Regression – introduction; simple linear regression, loss functions; Polynomial Regression; Logistic Regression; Softmax Regression with cross entropy as cost function**; Bayesian Learning** – Bayes Theorem, estimating conditional probabilities for categorical and continuous features, Naïve Bayes for supervised learning; Bayesian Belief networks; Support Vector Machines – soft margin and kernel tricks.				
Module 2	Ensemble Learning	Assignment		Programming using Keras/Sklearn	**No. of Classes L-3 P-4**
	Topics: **Ensemble Learning** – using subset of instances – Bagging, Pasting, using subset of features –random patches and random subspaces method; Voting Classifier, Random Forest; Boosting – AdaBoost, Gradient Boosting, Extremely Randomized Trees, Stacking.				
Module 3	Perceptron Learning	Assignment /Quiz		Programming using Keras/Sklearn	**No. of Classes L-7 P -2**
	Topics: **Perceptron Learning** – from biological to artificial neurons, Perceptrons, Linear Threshold Units, logical computations with Perceptrons, common activation functions – sigmoid, tanh, relu and softmax, common loss functions, multi-layer Perceptrons and the Backpropagation algorithm using Gradient Descent.				
Module 4	Unsupervised Learning	Assignment		Programming using Keras/Sklearn	**No. of Classes L-6 P -6**

	Topics: **Unsupervised Learning** – simple k Means clustering- simple and mini-batch**;** updating centroids incrementally; finding the optimal number of clusters using Elbow method ; Silhoutte coefficient,drawbacks of kMeans,kMeans++ **;** Divisive hierarchical clustering – bisecting k-means, clustering using Minimum Spanning Tree (MST) **Competitive Learning** - Clustering using Kohenen's Self Organising Maps (SOM), **Density Based Spatial Clustering – DBSCAN**; clustering using Gaussian Mixture Models (GMM) with EM algorithm ; Outlier Detection methods – **Isolation Forest, Local Outlier Factor(LOF)**
	List of Laboratory Tasks: **Experiment N0 1: Methods for handling missing values** **Level 1:** Given a data set from UCI repository, implement the different ways of handling missing values in it using Scikit-learn library of Python **Level 2:** Implement one of these methods using a custom defined function in Python. **Experiment No. 2: Data Visualization** **Level 1** Perform Exploratory Data Analysis for a given data set by creating Scatter Plot, Pair Plot, Count Plot using Matplotlib and Seaborn **Level 2** Create Heat Maps, WordCloud **Experiment No. 3: Regression learning** **Level 1** Given a data set from UCI repository, implement the simple linear regression algorithm and estimate the models parameters and the performance metrics. Plot the learning curves. **Level 2** Implement the polynomial regression algorithm. Compare the learning curves of Polynomial and Linear Regression. **Experiment No.4: Logistic regression** **Level 1** Write custom code for generating the logistic/sigmoid plot for a given input **Level 2** Given a data set from UCI repository, implement the Logistic regression algorithm. Estimate the class probabilities for a given test data set. Plot and analyze the decision boundaries. **Experiment No.5: Bayesian Learning** **Level 1** Given a data set from UCI repository, implement a classification model using the Bayesian algorithm **Experiment No.6: Support Vector Machine(SVM)** **Level 1** Given data sets from UCI repository, implement a linear SVM and a non-linear SVM based classification model. **Experiment No. 7: Ensemble Learning** **Level 1** : Implement Ensemble Learning algorithms such as Bagging, Pasting and Out-of Bag Evaluation **Level 2** : Random Patches and Random Subspace Method **Experiment No. 8: Ensemble Learning** **Level 1** : AdaBoost and Gradient Boosting, Stacking **Experiment No. 9: Perceptron Learning** **Level 1** : Implement the Perceptron Classifier **Level 2** : – An Image Classifier Using the Sequential API of Keras **Experiment No. 10: Unsupervised Learning** **Level 1** : K-means – simple and mini-batch. Finding the optimal number of clusters using Elbow method and Silhoutte Coefficient . Compare the inertia of both as k increases. Tuning the hyperparameter 'k' using GridSearchCV. **Level 2** : – Using clustering for Image segmentation and Preprocessing. Kmeans++ **Experiment No. 11: Density Based Clustering** **Level 1** Implement DBSCAN – clustering using the local density estimation. Perform hard and soft clustering for new instances. **Experiment No. 12: Outlier Detection** **Level 1** Outlier Detection using Isolation Forest and Local Outlier Factor

	Targeted Application & Tools that can be used : 1. Execution of the ML algorithms will be done using the Google's cloud service namely "Colab", available at https://colab.research.google.com/ or Jupyter Notebook. 2. The data sets will be from the bench marking repositories such as UCI machine learning repository available at : https://archive.ics.uci.edu/ml/index.php 3. Laboratory tasks will be implemented using the libraries available in Python such as Scikit learn, matplotlib, seaborn, perceptron and the deep learning framework namely Keras.
	Project work/Assignment: Mention the Type of Project /Assignment proposed for this course
	Students can be assigned a mini project to develop a machine learning application for real-life problems in various domains such as health care, business intelligence, environmental modeling, etc.
	Text Book There are a number of useful textbooks for the course, but each cover only a part of the course syllabus. Following is an indicative list of textbooks. 1. Aurélien Géron, "Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow", Oreilly, Second Edition, 2019. 2. Andreas C Muller, Sarah Guido, "Introduction to Machine Learning with Python :A Guide for Data Scientists", Oreilly, First Edition, 2018 3. Giuseppe Bonaccorso, "Machine Learning Algorithms: A reference guide to popular algorithms from data science and machine learning", Packt Publishing, 2017.
	References In references apart from the books and web links, mention a few standards &Hand books relevant to the Laboratory tasks used by the professionals. 1. Tan P. N., Steinbach M & Kumar V. "***Introduction to Data Mining",*** Pearson Education, 2016. 2. **https://towardsdatascience.com/machine-learning/home** 3. MITopencourseware:https://ocw.mit.edu/courses/6-0002-introduction-to-computational-thinking-and-data-science-fall-2016/resources/lecture-11-introduction-to-machine-learning/ 4. https://onlinecourses.nptel.ac.in/noc21_cs85/preview

Course Code: CSE3501	**Course Title: Advanced Deep Learning Techniques** **Type of Course: Theory**	**L- T-P- C**	**3**	**0**	**0**	**3**
Version No.	1.0					
Course Pre-requisites	CSE2267					
Anti-requisites	NIL					
Course Description	This course introduces students to the concepts of deep neural networks and state of the art approaches to develop deep learning models. In this course students will be given an exposure to the details of neural networks as well as deep learning architectures and to develop end-to-end models for such tasks. It will help to design and develop an application-specific deep learning models and also provide the practical knowledge handling and analyzing end user realistic applications. Topics include Fundamental concepts of deep neural networks, Convolutional Neural Networks, Recurrent Network structures, Deep Unsupervised Learning, Generative Adversarial Networks and applications in various problem domains.					
Course Objective	This course is designed to improve the learners EMPLOYABILITY SKILLS by using EXPERIENTIAL LEARNING techniques.					
Course Outcomes	On successful completion of this course the students shall be able to: 1. Learn the Fundamental Principles of Deep Learning. (Remember). 2. Identify the Deep Learning Algorithms for learning tasks in various related domains (Apply). 3. To understand and apply deep generative models. (Understand). 4. Apply deep learning architectures to image and audio data. (Apply)					
Course Content:						
Module 1	**Introduction to Deep Learning and Neural Networks**	Assignment				**13[7L+6P] Sessions**
Topics: Fundamentals of Deep Learning, Perceptron, Multilayer Perceptron, Optimizing Perceptions using Activation Functions, Loss Functions, Gradient Descent. Feedforward Neural Network, Training Neural Network with Back-propagation, Hyper parameters, Regularization, Dropouts, Batch Normalization, Practical Issues in Neural Network Training -The Problem of Overfitting, The Vanishing and Exploding Gradient Problems						
Module 2	**Common Deep Learning Architectures:**	Assignment				**18[8L+10P] Sessions**
Topics: Convolutional Neural Network, Transfer learning Techniques, Variants of CNN: DenseNet, ResNet Sequence Modelling: Recurrent Neural Network and its variants - Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU)						
Module 3	**Deep Generative Models**	Assignment				**16[8L+8P] Sessions**

Topics:
Generative Adversarial Networks, Kohonen Networks, Autoencoders, Boltzmann Machine, Restricted Boltzmann Machine, Deep Belief Network

Module-4	**Advanced Deep Learning Architectures**	Assignment		**13[7L+6P] Sessions**

Topics:
Hopfield Network, Probabilistic Neural Network, Deep Reinforcement Learning - The Basic Framework of Reinforcement Learning

Deep Learning applications: Image segmentation, Object detection,Speech Recognition, Video Analytics

Project work/Assignment:

1. **Assignment 1 on (Module 1 and Module 2)**
2. **Assignment 2 on (Module 3 and Module 4)**

List of Laboratory Tasks:

Lab 1: Working with Deep Learning Frameworks
Objective: Explore various Deep Learning Frameworks
Tasks: Identify deep learning frameworks (Keras, Tensorflow, Matplotlib, etc)
Activity: Practice with various methods available in DL Frameworks to develop a Model.

Lab 2: Build a Basic Artificial Neural Network
Objective: Create a ANN with DL frameworks.
Task: Identify suitable ANN Layers using Keras and Tensorflow.
Activity: Design a basic Artificial Neural Networks using Keras with TensorFlow (pima-indians-diabetes)

Lab 3 and Lab 4: Build a MultiLayer Perceptron
Objective: Create a MLP for classification task.
Task: Identify suitable model for house price prediction.
Activity: Design a MLP for implementing classification and fine-tuning using House price.csv

Lab 5: Build a Convolutional Neural Network
Objective: Create a CNN model.
Task: Build CNN architecture for Dog-Cat classification problem.
Activity: Implement a Convolution Neural Network (CNN) for dog/cat classification problem using keras

Lab 6 and Lab 7: Build a Time-Series Model
Objective: Create a RNN and LSTM Model
Task: Build RNN/LSTM Model for predicting time series data.
Activity Train a sentiment analysis model on IMDB dataset, use RNN layers with LSTM/GRU notes

Lab 8: Build a Gated Recurrent Unit architecture.
Objective: Create a Time Series Model.
Task: Build GRU Architecture for predicting time series data.
Activity: Implement a GRU architecture for language translations.

Lab 9 and Lab 10: Build a Transfer Learning Model.
Objective: Create a Seq2Seq Model
Task: Create Hugging-face API using Transfer learning model.
Activity: Implement Transfer Learning models for classification problems Exploring Hugging-face API

Lab 11: Build an Auto-Encoder model
Objective: Create an Unsupervised Deep Learning Model.
Task: Create AutoEncoder network Output Translations.
Activity: implement an Encoder-Decoder Recurrent neural network model for Neural Machine Translation.

Lab 12: Build Generative Adversarial Networks.

Objective: Create an Unsupervised Deep Learning Model.
Task: Design GAN Architecture for Image generations.
Activity: Design a Age Prediction model by Applying Generative Adversarial

REFERENCE MATERIALS:

TEXTBOOKS

1. François Chollet, “Deep Learning with Python”, 2nd Edition, Manning Publications, 2022
2. Ian Goodfellow, Yoshua Bengio, Aaron Courville, “Deep Learning”, MIT Press, 2017.

REFERENCES

1. Amlan Chakrabarti Amit Kumar Das, Saptarsi Goswami, Pabitra Mitra , “Deep Learning”, Pearson Publication, 2021.
2. David Foster, “Generative Deep Learning” O’Reilly Publishers, 2020.
3. John D Kellehar, “Deep Learning”, MIT Press, 2020.

JOURNALS/MAGAZINES

1. IEEE Transactions on Neural Networks and Learning Systems
https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=5962385
2. IEEE Transactions on Pattern Analysis and Machine Intelligence
https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=34http://ijaerd.com/papers/special_papers/IT032.pdf
3. International Journal of Intelligent Systems https://onlinelibrary.wiley.com/journal/1098111x

SWAYAM/NPTEL/MOOCs:

4. Swayam Nptel – Deep Learning – IIT Ropar https://onlinecourses.nptel.ac.in/noc21_cs35/preview
5. Coursera – Neural Networks and Deep Learning Andrew Ng
6. Coursera - Neural Networks for Machine Learning by Geoffrey Hinton in Coursera

Course Code: CSE3502	**Course Title: Computational Optimization for Intelligent Systems** **Type of Course: Theory (Discipline Elective)**	**L-T--P- C**	3	0	0	3

Version No.		1.0
Course Pre-requisites		CSE2267
Anti-requisites		**NIL**
Course Description		This course introduces a range of machine learning models and optimization tools that are used to apply these models in practice. Course will introduce what lies behind the optimization tools often used as a black box as well as an understanding of the trade-offs of numerical accuracy and theoretical and empirical complexity. For the students with some optimization background this course will introduce a variety of applications arising in machine learning and statistics as well as novel optimization methods targeting these applications.
Course Objective		The objective of the course is to familiarize the learners with the concepts of **Optimization Techniques for Machine Learning** and attain **Skill Development** through **Participative Learning** techniques.
Course Outcomes		On successful completion of this course the students shall be able to: 1. **Demonstrate** simple examples to illustrate how Machine Learning is applied in real-world scenarios. [**Understand].** 2. **Implement** Machine Learning models (e.g., decision trees, linear regression, neural networks) using tools or programming languages. **[Apply].** 3. **Determine** the suitability of convex optimization in solving problems like portfolio optimization, machine learning, or network design. **[Apply].** 4. **Solve** convex optimization problems with real or simulated data, such as minimizing a cost function or optimizing resource allocation. **[Apply].**
Course Content:		

Module 1:	**Fundamentals of Machine learning**	**Quiz**		Knowledge based Quiz	**10 Sessions**
	Topics: Machine Learning Paradigm, Empirical Risk Minimization, Structural Risk Minimization, Learning Guarantees, Introduction of VC-Dimension, Dimensionality Reduction Techniques.				
Module 2:	**Machine learning models**	**Quiz**		Comprehension based Quiz	**12 Sessions**
	Topics: Logistic Regression, Support Vector Machines, Sparse Regression, Low Dimensional Embedding, Low Rank Matrix Factorization, Sparse PCA, Multiple Kernel Learning, Loss Functions, Entropy, Cross-Entropy Loss				
Module 3	**Convex optimization models**	**Assignment**		Batch-wise Assignments	**13 Sessions**
	Topics: Linear Optimization, Convex Quadratic Optimization, Second Order Cone Optimization, Semi-definite Optimization, Convex Composite Optimization.				
Module 4:	**Methods for convex optimization**	Assignment and Presentation		Batch-wise Assignment and Presentations	**10 Sessions**

	Topics: gradient descent, Newton method, interior point methods, active set, prox methods, accelerated gradient methods, coordinate descent, cutting plances, stochastic gradient.
	Targeted Application & Tools that can be used: Use of Google Colab
	Project work/Assignment: **Survey on Methods for convex optimization** **Survey on Machine learning models related to optimization**
	Text Book T1. Charu C. Aggarwal, "*Linear Algebra and Optimization for Machine Learning*", Springer, 2020. T2. Sra Suvrit, Nowozin Sebastian, and Wright Stephen J, "*Optimization for Machine Learning*", The MIT Press,2012.
	References R1.Guanghui Lan, "*First-order and Stochastic Optimization Methods for Machine Learning*", Springer Cham, 2020. **Web References** W1. https://sm-nitk.vlabs.ac.in/ W2. https://nptel.ac.in/courses/
	Topics relevant to SKILL DEVELOPMENT: Concepts of Convex optimization models and Methods for convex optimization f**or Skill Development through Problem Solving methodologies. This is attained through assessment component mentioned in course handout.**

Course Code: CSE3503	Course Title: Reinforcement Learning for AI Systems Type of Course: Theory Only	L-T-P-C	2	0	2	3

Version No.		1.0
Course Pre-requisites	•	CSE2267
Anti-requisites		NIL
Course Description		The goal of this class is to provide an introduction to reinforcement learning, a very active research sub-field of machine learning. Reinforcement learning is concerned with building programs that learn how to predict and act in a stochastic environment, based on past experience. Applications of reinforcement learning range from classical control problems, such as power plant optimization or dynamical system control, to game playing, inventory control, and many other fields. Notably, reinforcement learning has also produced very compelling models of animal and human learning. During this course, we will study theoretical properties and practical applications of reinforcement learning. We will follow the second edition of the classic textbook by Sutton & Barto (available online for free, or from MIT Press), and supplement it as needed with papers and other materials.
Course Objective		The objective of the course is to familiarize the learners with the concepts of **Reinforcement Learning** and attain **Skill Development** through **Problem Solving** Methodologies.
Course Out Comes		On successful completion of the course the students shall be able to: 1. Knowledge of basic and advanced reinforcement learning techniques. 2. Identification of suitable learning tasks to which these learning techniques can be applied. 3. Appreciation of some of the current limitations of reinforcement learning techniques. 4. Formulation of decision problems, set up and run computational experiments, evaluation of results from experiments.
Course Content:		

Module 1	Introduction	Assignment		Programming	**No. of Classes:10**
	Topics: Course logistics and overview. Origin and history of Reinforcement Learning research. Its connections with other related fields and with different branches of machine learning. **Probability Primer** Brush up of Probability concepts - Axioms of probability, concepts of random variables, PMF, PDFs, CDFs, Expectation. Concepts of joint and multiple random variables, joint, conditional and marginal distributions. Correlation and independence.				
Module 2	Markov Decision Process	Assignment		Programming	**No. of Classes:10**
	Topics: Introduction to RL terminology, Markov property, Markov chains, Markov reward process (MRP). Introduction to and proof of Bellman equations for MRPs along with proof of existence of solution to Bellman equations in MRP. Introduction to Markov decision process (MDP), state and action value functions, Bellman expectation equations, optimality of value functions and policies, Bellman optimality equations.				
Module 3	Prediction and Control by Dynamic Programing	Assignment		Programming	**No. of Classes:10**
	Topics: Overview of dynamic programing for MDP, definition and formulation of planning in MDPs, principle of optimality, iterative policy evaluation, policy iteration, value iteration, Banach fixed point theorem, proof of contraction mapping property of Bellman expectation and optimality operators, proof of convergence of policy evaluation and value iteration algorithms, DP extensions **Monte Carlo Methods for Model Free Prediction and Control** Overview of Monte Carlo methods for model free RL, First visit and every visit Monte Carlo, Monte Carlo control, On policy and off policy learning, Importance sampling.				

Module 4	TD Methods and Policy Gradients	Assignment		Programming	**No. of Classes:10**

Topics:

Incremental Monte Carlo Methods for Model Free Prediction, Overview TD(0), TD(1) and TD(λ), k-step estimators, unified view of DP, MC and TD evaluation methods, TD Control methods - SARSA, Q-Learning and their variants.

Getting started with policy gradient methods, Log-derivative trick, Naive REINFORCE algorithm, bias and variance in Reinforcement Learning, Reducing variance in policy gradient estimates, baselines, advantage function, actor-critic methods.

Targeted Application & Tools that can be used:

While Convolution Neural Network (CNN) and Recurrent Neural Network (RNN) are becoming more important for businesses due to their applications in Computer Vision (CV) and Natural Language Processing (NLP), Reinforcement Learning (RL) as a framework for computational neuroscience to model decision making process seems to be undervalued. Besides, there seems to be very little resources detailing how RL is applied in different industries. Despite the criticisms about RL's weaknesses, RL should never be neglected in the space of corporate research given its huge potentials in assisting decision making.

Tools: Torch, Google Colaboratory, Spider, Jupiter Notebook

Project work/Assignment:

This part is written for general readers. At the same time, it will be of greater value for readers with some knowledge about RL.

- **Resources management in computer clusters**

Designing algorithms to allocate limited resources to different tasks is challenging and requires human-generated heuristics. The paper "Resource Management with Deep Reinforcement Learning" [2] showed how to use RL to automatically learn to allocate and schedule computer resources to waiting jobs, with the objective to minimize the average job slowdown.

State space was formulated as the current resources allocation and the resources profile of jobs. For action space, they used a trick to allow the agent to choose more than one action at each time step. Reward was the sum of (-1/duration of the job) over all the jobs in the system. Then they combined REINFORCE algorithm and baseline value to calculate the policy gradients and find the best policy parameters that give the probability distribution of actions to minimize the objective.

- **Traffic Light Control**

Researchers tried to design a traffic light controller to solve the congestion problem. Tested only on simulated environment though, their methods showed superior results than traditional methods and shed a light on the potential uses of multi-agent RL in designing traffic system.

Five agents were put in the five-intersection traffic network, with a RL agent at the central intersection to control traffic signalling. The state was defined as eight-dimensional vector with each element representing the relative traffic flow of each lane. Eight choices were available to the agent, each representing a phase combination, and the reward function was defined as reduction in delay compared with previous time step. The authors used DQN to learn the Q value of the {state, action} pairs.

- **Robotics**

There are tremendous works on applying RL in Robotics. Readers are referred to for a survey of RL in Robotics. In particular, trained a robot to learn policies to map raw video images to robot's actions. The RGB images were fed to a CNN and outputs were the motor torques. The RL component was the guided policy search to generate training data that came from its own state distribution.

- **Web System Configuration**

There are more than 100 configurable parameters in a web system and the process of tuning the parameters requires a skilled operator and numerous trail-and-error tests. The paper "A Reinforcement Learning Approach to Online Web System Auto-configuration" showed the first attempt in the domain on how to do autonomic reconfiguration of parameters in multi-tier web systems in VM-based dynamic environments.

The reconfiguration process can be formulated as a finite MDP. The state space was the system configuration, action space was {increase, decrease, keep} for each parameter, and reward was defined as the difference between the given targeted response time and measured response time. The authors used the model-free Q-learning algorithm to do the task.

Text Book

1. "Reinforcement Learning: An Introduction", Richard S. Sutton and Andrew G. Barto, 2nd Edition
2. "Probability, Statistics, and Random Processes for Electrical Engineering", 3rd Edition, Alberto Leon-Garcia
3. "Machine Learning: A Probabilistic Perspective", Kevin P. Murphy

	References 1. Richard S. Sutton and Andrew G. Barto, "Reinforcement learning: An introduction", Second Edition, MIT Press, 2019. 2. Li, Yuxi. "Deep reinforcement learning." arXiv preprint arXiv:1810.06339 (2018). 3. Wiering, Marco, and Martijn Van Otterlo. "Reinforcement learning." Adaptation, learning, and optimization 12 (2012):
	E-Resources NPTEL course – https://onlinecourses.nptel.ac.in/noc19_cs55/preview https://archive.nptel.ac.in/courses/106/106/106106143/ https://www.digimat.in/nptel/courses/video/106106143/L35.html
	Topics relevant to "SKILL DEVELOPMENT": Real time Data Analysis using Reinforcement learning **for Skill Development through Problem Solving techniques. This is attained through assessment component mentioned in course handout.**

Course Code: CSE3504	Course Title: Computational Linguistics and Natural Language Processing Type of Course: Theory Only	L- T-P- C	3	0	0	3

Version No.	1.0
Course Pre-requisites	CSE2267
Anti-requisites	NIL
Course Description	The purpose of this course is to introduce students to the science of natural language processing (NLP). NLP is the science of extracting information from unstructured text. It is basically how we can teach machines to understand human languages and extract meaning from text. In addition to regular theory, the course also involves: 1. Programming Assignments 2. Regular Quiz Tests (once a week and once after every module)
Course Objective	The objective of the course is to familiarize the learners with the concepts of Fundamentals of Natural language Processing and attain **Skill Development** through **Participative Learning** techniques.
Course Out Comes	**On successful completion of the course the students shall be able to:** • **Understand** the fundamental concepts of Natural Language Processing. [Knowledge] • **Read** corpora and **train** models for different NLP tasks. [Application] • **Use** word embeddings for solving an NLP Application. [Application] • **Understand** sequence to sequence modeling as used in machine translation. [Application]
Course Content:	

Module 1	**Introduction**	Quizzes			**7 Sessions**
	Topics: Introduction. History. Text Analytics. Various tasks in NLP. Sentence boundary Detection. Edit distance. Introduction to word embeddings, PoS tagging, chunking, parsing, machine translation.				
Module 2	**Word and Text Representations**	Quizzes		Assignments	**8 Sessions**
	Topics: Logistic Regression and Naïve Bayes classification. Vector semantics and embeddings. Neural Networks and Neural Language Models. Text representations and classification. Deep learning architectures for sequence processing (CNN and LSTM).				
Module 3	**PoS Tagging, NER Tagging and Parsing**	Quizzes		Assignments	**12 Sessions**
	Topics: Part-of-Speech Tagging – using NLTK and spacy. Building a PoS Tagger using existing data and Hidden Markov Model. Named Entity Recognition. Relationship between NER tagging and PoS tagging. Constituency Parsing.				
Module 4	**NLP Applications**	Quizzes			**9 Sessions**
	Topics: Lexical Resource Creation. Sentiment Analysis. Machine Translation. Word Sense Disambiguation and WordNet. Question Answering.				
	Targeted Application & Tools that can be used: 1. **Python Libraries (Eg. NLTK, Spacy, etc.)** 2. **Java (Stanford CoreNLP)** 3. **Google Colab**				
	Project work/Assignment:				

	Assignment: Students will have to do group assignments for Modules 2 & 3. As a part of their assignments, they will have to implement the solution to particular problems.
	Text Book **T1**Daniel Jurafsky, and James Martin."*Speech and Language Processing*" (3rd edition draft, 2022)
	References **R1**Chris Manning and HinrichSchutze, "*Foundations of Statistical Natural Language Processing*", 1st Edition, MIT Press. 1999. **R2**PawanGoyal, "*Natural Language Processing*". NPTEL. **E-Book Link for R2: https://drive.google.com/file/d/10nbwAJd-dv6htOOZVBgAvLd1WscI0RqC/view** **Web resources:https://web.stanford.edu/~jurafsky/slp3/** **NPTEL Course: https://onlinecourses.nptel.ac.in/noc22_cs98/course**
	Topics relevant to "SKILL DEVELOPMENT": Assignment implementations in software, batch wise presentations for developing Skill Development through Participative Learning techniques. This is attained through assessment component mentioned in course handout.

Course Code: CSE3505	**Course Title: Synergistic Neural Fuzzy Computing** **Type of Course: Theory** **(Discipline Elective in AI & ML Basket)**	**L-T-P-C**	2	0	2	3

Version No.		1.0
Course Pre-requisites		CSE2267
Anti-requisites		NIL
Course Description		This course aims to introduce the basic concepts of Neural Networks and Fuzzy Logic. Neural networks reflect the behavior of the human brain, allowing computer programs to recognize patterns and solve common problems in the fields of AI, machine learning, and deep learning. Fuzzy Logic is a method of reasoning that resembles human reasoning. The approach of Fuzzy Logic imitates the way of decision-making in humans that involves all intermediate possibilities between digital values YES and NO. This course introduces fundamental concepts in Neural Networks and Fuzzy Logic Theory.
Course Objective		The objective of the course is to familiarize the learners with the concepts of **Neural Networks and Fuzzy Logic** and attain **Skill Development** through **Participative Learning** techniques.
Course Outcomes		**On successful completion of this course the students shall be able to:** 1. Define the concept of Neural Networks. [Knowledge] 2. Define the ideas behind most common learning algorithms in Neural Network.[Knowledge] 3. Discuss the concepts of Fuzzy Sets and Relations. [Comprehension] 4. Demonstrate the Fuzzy logic concepts and its applications.[Application]
Course Content:		

Module 1	**Introduction to Neural Network**	Quiz		Single Layer Perceptron	**9Classes**
	Topics: Introduction to NN: History, Artificial and biological neural networks, Artificial intelligence and neural networks. Neurons and Neural Networks: Biological neurons, Models of single neurons, Different neural network models. Single Layer Perceptron: Least mean square algorithm, Learning curves, Learning rates, Perceptron.				
Module 2	**Multilayer Perceptron**	Quiz		Multilayer Perceptron	**10 Classes**
	Topics: Multilayer Perceptron: The XOR problem, Back-propagation algorithm, Heuristic for improving the back-propagation algorithm, Some examples. Radial-Basis Function Networks: Interpolation, Regularization, Learning strategies. Kohonen Self-Organising Maps: Self-organizing map, The SOM algorithm, Learning vector quantization.				
Module 3	**Fuzzy Sets, Operations and Relations**	Quiz		Fuzzy Operations	**10Classes**
	Topics: Fuzzy Sets: Crisp Sets - an Overview, Fuzzy Sets - Definition and Examples, α - Cuts and its Properties, Representations of Fuzzy Sets, Extension Principles of Fuzzy Sets. Fuzzy Operations: Operations on Fuzzy Sets - Fuzzy Complements, Fuzzy Intersections, Fuzzy Unions, Combinations of Operations, Aggregation Operations. Fuzzy Relations: Binary Fuzzy relations, Fuzzy Equivalence Relations, Fuzzy Compatibility Relations.				
Module 4	**Fuzzy Logic and Fuzzy Logic Controller**	Assignment		Developing Fuzzy Logic Controller	**10Classes**

	Fuzzy Logic: Classical Logic, Multivalued Logic, Fuzzy Propositions, Fuzzy Quantifiers, Linguistic Hedges, Inference from Conditional Fuzzy Propositions, Conditional and Qualified Propositions and Quantified Propositions. Fuzzy Controllers: An Overview, Fuzzification Module, Fuzzy Rule Base, Fuzzy Inference Engine, Defuzzification Module, An Example.
	Targeted Application & Tools that can be used: 1. Python Libraries and Software (Eg.,Tensorflow, Scikit-Learn etc.) 2. Matlab (Neural Network Toolbox, Fuzzy Logic Toolbox)
	Project work/Assignment:
	Students will have to do group assignments for Modules 2 & 4. As a part of their assignments, they will have to implement the solution to particular problems.
	Textbook(s): 1. Haykin, Simon. "*Neural networks and learning machines"*, 3/E. Pearson Education India, 2011. https://www.pearson.com/en-us/subject-catalog/p/Haykin-Neural-Networks-and-Learning-Machines-3rd-Edition/P200000003278/9780133002553 2. George J. Klir and Bo Yuan, "*Fuzzy Sets and Fuzzy Logic- Theory and Applications*", Prentice Hall of India, 2015. https://www.worldcat.org/title/fuzzy-sets-and-fuzzy-logic-theory-and-applications/oclc/505215200
	References: 1. Shivanandam, Deepa S, "*Principles of Soft computing*", N Wiley India, 3rd Edition, 2018.https://www.wileyindia.com/principles-of-soft-computing-3ed.html 2. Timothy J. Ross, "*Fuzzy Logic with Engineering Applications*", Third Edition, Wiley, 2011. https://onlinelibrary.wiley.com/doi/book/10.1002/9781119994374 3. Kumar S., "*Neural Networks - A Classroom Approach*", Tata McGraw Hill, 2nd Edition 2017.https://www.worldcat.org/title/neural-networks-a-classroom-approach/oclc/56955342 4. Fakhreddine O. Karray, and Clarence W. De Silva. "*Soft computing and intelligent systems design: theory, tools, and applications*". Pearson Education, 2009. **Weblinks** https://www.pearson.com/en-gb/search.html?q=Karray%20Soft-Computing-and-Intelligent-Systems-Design-Theory-Tools-and-Applications
	Topics relevant to "Skill Development": Assignment implementations in software, batch wise presentationsare usedforSkill Development through Participative Learning techniques. This is attained through assessment component mentioned in course handout.

Course Code: CSE3506	**Course Title: Introduction to Bioinformatics** **Type of Course: General CSE Basket, Theory based**		**L- T- P- C**	3	0	0	3
Version No.	2.0						
Course Pre-requisites	NIL						
Anti-requisites	NIL						
Course Description	This course is designed to provide the knowledge of the concepts related to bioinformatics. The course is aimed at understanding the DNA and Protein sequences and databases. It also deals with Pairwise comparison and calculating the scoring matrix. Further, it focuses on Sequence Alignment techniques, discovering the Motifs in the sequence. Students will also learn the overview of Structural Bioinformatics and Genome sequencing.						
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Introduction to Bioinformatics** and attain **Employability** through **Participative Learning** techniques.						
Course Outcomes	**C.O.1:** Understand the DNA Protein sequence and structures. (Bloom's Level: **Knowledge)** **C.O.2:** Explain the file formats and sequence alignments of DNA sequence. (Bloom's Level: **Comprehension)** **C.O.3:** Apply the techniques of the motifs discovery for the analysis of Protein Sequence. (Bloom's Level: **Application)**						
Course Content:							
Module 1	**Fundamentals of Bioinformatics**	Quiz	Comprehension based Quizzes and assignments;			**9 Classes**	
Topics: **Introduction to Bioinformatics**: Introduction to molecular biology, Cell, DNA, RNA, Transcription, Translation, Folding, Gene Structure, Introduction to Bioinformatics, Components and fields of bioinformatics, Omics, basic principles of structural/functional analysis of biological molecules, Biological Data Acquisition, Types of DNA sequences,Genomic DNA,Mitochondrial DNA,DNA Sequencing tools, Protein sequencing and structure determination methods, Finding Reverse complement of a sequence.							
Module 2	**Genome databases and Sequence Similarity**	Quizzes and assignments	Comprehension based Quizzes and assignments			**8 Classes**	
Topics: Types and classification of genome databases, DNA sequence retrieval system, various DNA and protein sequence file formats, Common sequence file formats; Files for multiple sequence alignment; Files for structural data, Frequent words and k-mers in Text, String Reconstruction problem, Sequence Similarity searching, Sequence Similarity searching tools, NCBI BLAST, PSI BLAST, Significance of sequence alignments, Alignment scores and gap penalties.							
Module 3	**DNA sequence analysis and applications**	Quizzes and assignments	Comprehension based Quizzes and assignments			**10 Classes**	
Sequence similarity searches and alignment tools, Finding alignment using Needleman-Wunsch and Smith-Waterman algorithm, Heuristic Methods of sequence alignment, Pair-wise and multiple sequence alignments, DNA sequence analysis, Motif in protein sequence ,Motif discovery using Gibbs sampling,Motif finding, Gene Prediction models: Hidden Markov model(HMM), Generalized Hidden Markov model(GHMM), Bayesian method.							
Targeted Application & Tools that can be used: BLAST, FastA, , ClustalW, MEGA							
Project work/Assignment:							
Each batch of students (self-selected batch mates – up to 4 in a batch) will be allocated case studies/assignments							
Textbook(s): 1. Bioinformatics: Sequence and Genome Analysis, David W. Mount, Cold Spring Harbor Laboratory Press, 2004. 2. Introduction to Bioinformatics, Arthur Lesk, Fifth Edition, Oxford University Press, 2019							
References 1. Bioinformatics Methods and Applications, S. C. Rastogi, N.Mendiratta, P.Rastogi, Fourth Edition, Prentice Hall India. 2.Bioinformatics Algorithms- An Active Learning Approach, Phillip Compeau & Pavel Pevzner, 2nd Edition, Vol. I &							

II, Active Learning Publishers, 2015

E-References
1. https://puniversity.informaticsglobal.com:2229/login.aspx

Topics related to development of "Employability skills": Batch wise presentations on selected topics

1. String Reconstruction problem
2. Sequence Similarity searching
3. Alignment scores and gap penalties
4. Protein sequencing
5. Gene Prediction models: Hidden Markov model(HMM)
6. Finding similarities by performing pairwise and multiple sequence alignment,
7. Evaluating phylogenetic trees.

fordeveloping **Employability Skills** through **Participative Learning techniques**. This is attained through assessment component mentioned in course handout.

Course Code: CSE3507	**Course Title: Algorithms in Computational Biology** **Type of Course: PCC** **Lab Integrated**	**L-T-P-C**	2	0	2	3

Version No.		1.0
Course Pre-requisites	•	CSE3506
Anti-requisites		NIL
Course Description		This course introduces core algorithms used in computational biology to solve biological problems efficiently. It covers sequence analysis, dynamic programming, genome assembly, and string matching techniques. Students will learn how to apply algorithmic thinking to biological datasets and evaluate the performance of solutions.
Course Object		The objective of the course is to familiarize the learners with the concepts of Algorithams in Computational Biology and attain **Skill Development** through **Experiential Learning** techniques.
Course Out Comes		On successful completion of the course the students shall be able to: **CO1:** Define key concepts in computational biology and bioinformatics. *(Remember)* **CO2:** Explain the significance of algorithms in analyzing biological data. *(Understand)* **CO3**: Apply basic algorithmic strategies to solve simple biological problems. *(Apply)* **CO4:** Explore different types of biological data and their computational needs. *(Apply)*
Course Content:		

Module 1	**Introduction**	Assignment			**5L+6P Sessions**
	Topics: History and principles of algorithms, Types of algorithms, Development and computational complexity .				
Module 2	Algorithms Issues and Problems	Assignment			**8L+8P Sessions**
	Topics: Asymptotic analysis of algorithms, NP-complete problems, Polynomial reducibility,				

	Traveling Salesman Problem (TSP), Consecutive Integer Problem (CIP), Sorting problems and Fibonacci problem					
Module 3	Algorithmic Approaches	Assignment			**10L+8PSessions**	
	Topics: Linear, exhaustive search, branch and bound, divide and conquer, Expectation and Maximization (EM), Forward and backward algorithms, Discriminative learning, Knuth-Morris-Pratt and Boyer-Moore (string matching), Graph algorithms and maximum likelihood algorithms					
Module 4	Dynamic Programming & Methods	Assignment			**7L+8P Sessions**	
	Topics: Principles and applications, Heuristics tools (BLAST, FASTA, ClustalW), Probabilistic/statistical methods, Models of evolution and relevant algorithms, Partial and double digest problems, Graph algorithms for DNA sequence assembly (CASP3, Phrap, Phred), Protein structure prediction – Chou-Fasman algorithm.					
	Project work/Assignment:					
3.	**4. Assignment 1 on (Module 1 and Module 2)** 5. **Assignment 2 on (Module 3 and Module 4)**					
	Text Book 1) **Phillip Compeau & Pavel Pevzner, *Bioinformatics Algorithms: An Active Learning Approach*, Vol. 1, 2nd Ed., 2019**. 2) **Michael T. Goodrich & Roberto Tamassia, *Algorithm Design and Applications*, Wiley, 2015**. 3) **Jason Kinser,** *Computational Biology: A Hypertextbook*, 2nd Ed., Jones & Bartlett Learning, 2021 4) **Gautam B. Singh**, *Fundamentals of Bioinformatics and Computational Biology*, Springer, 2015.					
	References 1. Zhumur Ghosh & Bibekanand Mallick, *Bioinformatics: Principles and Applications*, Oxford University Press, 2014. 2. Steven Skiena, *The Algorithm Design Manual*, Springer, 2nd Ed., 2008 3. Pavel Pevzner, Computational Molecular Biology: An Algorithmic Approach, MIT Press, 2000. 4. T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein, *Introduction to Algorithms*, 3rd Ed., MIT Press, 2009. 5. Arthur Lesk, *Introduction to Bioinformatics*, 5th Ed., Oxford University Press, 2019 **Web Resources** W1. https://onlinecourses.nptel.ac.in/noc25_cs06/preview W2. https://ocw.mit.edu/courses/6-096-algorithms-for-computational-biology-spring-2005/pages/lecture-notes/					
	Module I: Introduction to Algorithms **Experiment 1: *Implement Sorting and Searching Algorithms*** • **Implement Bubble, Merge Sort, Linear and Binary Search**					

	• **Compare time complexity using real biological data (e.g., gene lengths)** **Experiment 2:** ***Time Complexity and Recursion using Fibonacci Series*** • **Compare recursive and dynamic programming approaches** • **Visualize time/memory usage with time and memory_profiler modules** **Module II: Algorithmic Problem Solving** **Experiment 3:** ***Solve the Travelling Salesman Problem (TSP)*** o **Use brute force or greedy algorithms** o **Simulate sequencing fragment reassembly as a path problem** **Experiment 4:** ***Knapsack Problem in Bioinformatics*** o **Apply knapsack logic to protein interaction weighting or resource allocation** **Module III: String Matching and Sequence Analysis** **Experiment 5:** ***Naive Pattern Matching Algorithm on DNA Sequence*** • **Identify motifs like start/stop codons** **Experiment 6:** ***KMP and Boyer-Moore Algorithms*** • **Compare performance on large FASTA datasets** **Experiment 7:** ***Regular Expression Matching in Genomic Data*** • **Search for specific motifs** **Module IV: Dynamic Programming** **Experiment 8:** ***BLAST Query using Biopython (Online)*** **Submit a BLAST query and parse top hits** **Experiment 9:** ***DNA Read Assembly Simulation***
	Topics relevant to development of "Employability": Proficiency in bioinformatics algorithms, data analysis **Topics relevant to "PROFESSIONAL ETHICS":** Maintaining professional integrity in computational biology.

Course Code: CSE3508	**Course Title: Statistical Methods for BioInformatics** **Type of Course: PCC** **Lab Integrated**	**L-T-P-C**	2	0	2	3

Version No.		1.0
Course Pre-requisites	•	CSE3506
Anti-requisites		NIL
Course Description		This course provides an introduction to the statistical methods commonly used in bioinformatics and biological research. The course briefly reviews basic probability and statistics including events, conditional probabilities, Bayes theorem, random variables, probability distributions, and hypothesis testing and then proceeds to topics more specific to bioinformatics research, including Markov chains, hidden Markov models, Bayesian statistics, and Bayesian networks. Students will learn the principles behind these statistical methods and how they can be applied to analyze biological sequences and data..
Course Object		The objective of the course is to familiarize the learners with the concepts of Statistical Methods for BioInformatics and attain **Skill Development** through **Experiential Learning** techniques.
Course Out Comes		On successful completion of the course the students shall be able to: **CO1:** Understand the basic concepts of bioinformatics including databases. Understand) **CO2:** Evaluate methods to characterize and manage the different types of biological sequence data. (Apply) **CO3**: Apply basic concepts in biostatistics exemplifying sampling methods, (Apply) **CO4:** Evaluate hypothesis testing using statistical methods(Apply)
Course Content:		

Module 1	Bioinformatics Fundamentals and Database Systems	Assignment			**6L+6P Sessions**
	Topics: Bioinformatics definition, history, scope and applications, Bioinformatics web portals: NCBI, EBI, ExPASy, Biological databases: Classification of databases - primary (Genbank), secondary (PIR) and tertiary or composite (KEGG) databases, Sequence databases - DNA sequence databases (ENA,DDBJ), Protein sequence databases (Swissprot, PROSITE) .				
Module 2	Sequence Alignment	Assignment			**7L+8P Sessions**

	Topics: Basics of sequence alignment - match, mismatch, gaps, gap penalties, scoring alignment, Types of sequence alignment - pairwise and multiple alignment, local and global alignment, Dot matrix comparison of sequences, Scoring matrices - PAM and BLOSUM, Pairwise sequence similarity search by BLAST and FASTA					
Module 3	Basic Concepts In Biostatistics	Assignment			**10L+8PSessions**	
	Topics: Introduction to Biostatistics, kinds of data and variables - based on nature (numericaldiscrete and continuous, categorical-ordinal and nominal) - based on source (primary and secondary data), sample size, sampling methods and sampling errors, Data tabulation and representation methods: graphical methods– stem and leaf plot, line diagram, bar graphs, histogram, frequency polygon, frequency curves; diagrammatic method- pie diagram, Measures of central tendency- mean, median, mode; merits and demerits, Measures of dispersion- range, variance, standard deviation, standard error and coefficient of variation; merits and demerits, Correlation and regression analysis and their applications to biolog.					
Module 4	Biostatistics-Applications	Assignment			**7L+8P Sessions**	
	Topics: Introduction to Probability- definition; Normal distribution: definition and properties, Hypothesis testing- steps in testing for statistical hypothesis, null and alternative hypothesis, level of significance- type-1 and type-2 errors, Test of significance for large samples- Z-test for means and proportions, Test of significance for small samples- student's t-test(one sample and two samples), Chi-square test and its applications- goodness of fit (not based on distribution), test of independence , Analysis of variance (One-way ANOVA) and their applications to biology .					
	Project work/Assignment:					
6.	**7. Assignment 1 on (Module 1 and Module 2)** 8. **Assignment 2 on (Module 3 and Module 4)**					
	Text Book 5) **Arthur Lesk**, *Introduction to Bioinformatics*, Oxford University Press, 5th Edition, 2019. 6) **Wayne W. Daniel & Chad L. Cross**, *Biostatistics: A Foundation for Analysis in the Health Sciences*, Wiley, 11th Edition, 2019. 7) **Gautam B. Singh**, *Fundamentals of Bioinformatics and Computational Biology*, Springer, 2015.					
	References **1.** Zhumur Ghosh & Bibekanand Mallick, *Bioinformatics: Principles and Applications*, Oxford University Press, 2014. **2.** David W. Mount, *Bioinformatics: Sequence and Genome Analysis*, Cold Spring Harbor, 2nd Edition, 2004**.** **3.** Marcello Pagano & Kimberlee Gauvreau, *Principles of Biostatistics*, CRC Press, 2nd Edition, 2018. 4. **S. C. Gupta & V. K. Kapoor**, *Fundamentals of Mathematical Statistics*, Sultan Chand **Web Resources** W3. https://www.ncbi.nlm.nih.gov					

	W4. https://archive.nptel.ac.in/courses/102/101/102101056/ W5. https://onlinecourses.nptel.ac.in/noc25_bt06/preview W6. https://www.ebi.ac.uk W7. https://www.expasy.org
	Module I: Bioinformatics & Biological Databases **1. Experiment 1: Introduction to NCBI – Searching for DNA & protein sequences** **2. Experiment 2: Exploring the EBI and ExPASy portals** **3. Experiment 3: Retrieving gene information from GenBank and ENA** **4. Experiment 4: Identifying protein domains using PROSITE and SwissProt** **Module II: Sequence Alignment** **5. Experiment 5: Pairwise sequence alignment using EMBOSS Needle** **6. Experiment 6: Local sequence alignment using BLAST** **7. Experiment 7: Global sequence alignment using Clustal Omega** **8. Experiment 8: Constructing phylogenetic trees using MEGA or Phylogeny.fr** **Module III: Basic Biostatistics** **9. Experiment 9: Data collection and classification of variables** **10. Experiment 10: Creating bar charts, pie charts, histograms using MS Excel or Python (matplotlib/seaborn)** **11. Experiment 11: Calculating mean, median, mode, standard deviation using statistical software (R/SPSS/Excel)** **Module IV: Statistical Analysis** **12. Experiment 12: Performing correlation and regression analysis in R** **13. Experiment 13: Hypothesis testing using t-test and Z-test in SPSS/R** **14. Experiment 14: Conducting chi-square test for independence** **15. Experiment 15: One-way ANOVA application on biological data s**
	Topics relevant to development of "Employability": Training in bioinformatics tools, biological databases, sequence analysis **Topics relevant to "PROFESSIONAL ETHICS":** Data privacy in genomic research, ethical use of biological databases

Course Code: CSE3509	**Course Title:** Emerging Technologies in Big Data **Type of Course:** **Lab Integrated**	L-T- P- C	2 -0	2	3
Version No.	1.0				
Course Pre-requisites	CSE2500				
Anti-requisites	NIL				
Course Description	The purpose of the course is to provide the fundamentals of Big data technology, to emphasize the importance of choosing suitable tools for processing and analyzing big data to gain insights. The student should have knowledge and skill to select and use most appropriate big data tools to solve business problems. The associated laboratory provides an opportunity to implement the concepts and enhance critical thinking and analytical skills. With a good knowledge in the fundamentals of Big data technology the student can gain practical experience in implementing them, enabling the student to be an effective solution provider for applications that involve huge volume of data.				
Course Objectives	The objective of the course is to familiarize the learners with the concepts of Big Data Technologies and attain SKILL DEVELOPMENT through EXPERIENTIAL LEARNING techniques.				
Course Outcomes	On successful completion of the course the students shall be able to: Apply Map-Reduce programming on the given datasets to extract required insights. (Application). Employ appropriate Hadoop Ecosystem tools such as scoop, Hbase, Hive, to perform data analytics for a given problem. (Application). Use Spark tool to analyze the given dataset for a given problem. (Application).				
Course Content:					
Module 1	Introduction to Hadoop	Programming Assignment	Data Collection and Analysis	10 Classes	
Introduction to Big Data and its importance: Basics of Distributed File System, Four Vs, Drivers for Big data, Big data applications, Structured, unstructured, semi-structured and quasi structured data. Big data Challenges-Traditional versus big data approach, The Big Data Technology Landscape: No-SQL. The Hadoop: History of Hadoop-Hadoop use cases, The Design of HDFS, Blocks and replication management, Rack awareness, HDFS architecture, HDFS Federation, Name node and data node, Anatomy of File write. Anatomy of File read, Hadoop Map Reduce paradigm, Map and reduce tasks, Job Tracker and task tracker, Map reduce execution pipeline, Key value pair, Shuffle and sort, Combiner and Partitioner, APIs used to Write/Read files into/from Hadoop, Need for Flume and Sqoop. Anatomy of a YARN: Hadoop 2.0 Features, Name Node High Availability, YARN Architecture,					

Introduction to Schedulers, YARN scheduler policies, FIFO, Fair And Capacity scheduler.				
Module 2	Hadoop Ecosystem Tools	Programming Assignment	Data Collection and Analysis	8 Classes
Introduction to SQOOP: SQOOP features, Sqoop Architecture, Sqoop Import All Tables, Sqoop Export All Tables, Sqoop Connectors, Sqoop Import from MySQL to HDFS, Sqoop vs flume. Hive: Apache Hive with Hive Installation, Hive Data Types, Hive Table partitioning, Hive DDL commands, Hive DML commands, and Hive sort by vs. order by, Hive Joining tables, Hive bucketing. Hbase: Introduction to HBase and its working architecture- Commands for creation and listing of tables-disabled and is disabled of table - enable and is enabled of table- describing and dropping of table-Put and Get command - delete and delete all command-commands for scan, count, truncate of tables.				
Module 3	Spark	Programming Assignment	Data analysis	8 Classes
Introduction to Apache Spark A unified Spark, Who uses Spark and for what?, A Brief History of Spark, Spark version and releases, Storage layers for Spark. Programming with RDDs: RDD Basics, Creating RDDs, RDD Operations, Passing functions to Spark, Common Transformations and Actions, Persistence. Spark SQL: Linking with Spark SQL, Using Spark SQL in Applications, Loading and Saving Data, JDBC/ODBC Server, User-defined functions, Spark SQL Performance. Scala: The Basics, Control Structures and functions, Working with arrays, Maps and Tuples.				
List of Laboratory Tasks: 1. Level 1: To install the Hadoop in pseudo cluster mode. Level 1: HDFS Shell Commands – Files and Folders. Level 2: HDFS Shell Commands – Management. 2. Run a basic Word Count Map Reduce program to understand Map Reduce Paradigm. Level 1: Find the number of occurrence of each word appearing in the input file(s) Level 2: Performing a Map Reduce Job for word search count (look for specific keywords in a file). 3. Write a Map Reduce program that mines weather data. Weather sensors collecting data every hour at many locations across the globe gather large volume of log data, which is a good candidate for analysis with Map Reduce, since it is record-oriented. Data available at: https://github.com/tomwhite/hadoopbook/tree/master/input/ncdc/all. Level 1: Find average, max and min temperature for each year in NCDC data set? Level 2: Programming assignment to analyze the social media data for business analytics. 4. Level 1: Finding out Number of Products Sold in Each Country using map reduce with sample dataset				

Level 2: Find matrix multiplication using map reduce

5. Level 1: Installation of Hive, working on basic hive commands. (Create, Alter and Drop tables)
   Level 2: Apply Hive commands to student database/employee database.

6. Level 1: Working on advance hive commands. (Static Partitioning & Dynamic partitioning)
   Level 2: Continue the previous experiment, select and apply suitable partitioning technique.

7. Level 1: Working on advance hive commands-2. (Bucketing)
   Level 2: Continue the previous experiment, apply bucketing technique to bring out the difference between partitioning and bucketing.

8. Level 1: Installing Ecosystem tools such as Scoop, Hbase.
   Level 2: Scoop – Move Data into Hadoop.

9. Level 1: Working on basic Hbase commands (General commands, DDL Commands)
   Level 2: Apply Hbase commands on Insurance database/employee dataset.

10. Level 1: Working on advanced Hbase commands. (DML).
    Level 2: Continue the previous experiment to demonstrate CRUD operations.

11. Level 1: Install, Deploy & configure Apache Spark.
    Level 2: Using RDD and FlatMap count how many times each word appears in a file and write out a list of words whose count is strictly greater than 4 using Spark

12. Level 1: Write a program in Apache spark to count the occurrences words in a given text file and display only those words starting with ‘a’ in ascending order of count.
    Level 2: Apache access logs are responsible for recording data for all web page requests processed by the Apache server. An access log record written in the Common Log Format will look something like this: 127.0.0.1 - Scott [10/Dec/2019:13:55:36 – 0700] "GET /server-status HTTP/1.1" 200 2326 Where, HTTP 200 status response code indicates that the request has succeeded. Write a program to read the records of

access log file log.txt and display the number of successful requests using Spark.

13. Level 1: Chess king moves horizontally, vertically or diagonally to any adjacent cell. Given two different cells of the chessboard, determine whether a king can go from the first cell to the second in one move.

Write a scala program that receives input of four numbers from 1 to 8, each specifying the column and row number, first two - for the first cell, and then the last two - for the second cell. The program should output YES if a king can go from the first cell to the second in one move, or NO otherwise.

Level 2: Data analytics using Apache Spark on Amazon food dataset, find all the pairs of items frequently reviewed together.

Write a single Spark application that:

Transposes the original Amazon food dataset, obtaining a Pair RDD of the type:

Counts the frequencies of all the pairs of products reviewed together;

Writes on the output folder all the pairs of products that appear more than once and their frequencies. The pairs of products must be sorted by frequency.

Targeted Application & Tools that can be used:

Business Analytical Applications

Social media Data Analysis

Predictive Analytics

Tools: Hadoop Framework tools like map reduce, Hive, Hbase, Scoop, Spark.

Text Book

Seema Acharya, Subhashini Chellappan. 2015. Big Data and Analytics. Wiley Publication.

Matei Zaharia, Bill Chambers. 2018. SPARK: The Definitive Guide. Oreilly.

References

Tom White. 2016. Hadoop: The Definitive Guide. O'Reilley.

Cay S. Horstmann. 2017. Scala for the Impatient. Wesley.

Topics relevant to development of “Skill Development”: Real time application development using Hadoop Ecosystem tools through Experiential Learning as mentioned in the course handout.

Course Code: CSE3510	**Course Title: Statistical Techniques for Data Science** **Type of Course: Theory**	**L-T-P-C**	2	0	2	3
Version No.	1.0					
Course Pre-requisites	MAT1003					
Anti-requisites	NIL					
Course Description	This course provides an in-depth introduction to statistics and machine learning theory, methods, and algorithms for data science. Topics include multiple regression, kernel learning, sparse regression, generalized linear models, supervised and unsupervised learning, deep learning, covariance learning, factor models, principal component analysis, and more. The course emphasizes the applicability and limitations of these methods using mathematical statistics and real-world data sets.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Statistical Foundations for Data Science** and attain **Employability** through **Participative Learning** techniques.					
Course Out Comes	On successful completion of the course the students shall be able to: 4) Understand the rise and significance of Big Data in various fields such as Biological Sciences, Health Sciences, Computer and Information Sciences, Economics and Finance, Business and Program Evaluation, Earth Sciences, and Astronomy. 4) Develop a strong foundation in multiple linear regression and the Gauss-Markov theorem. 4) Apply linear regression with random design and partial linear regression. 4) Apply the power method and learn about factor models and structured covariance learning.					
Course Content:						
Module 1	Introduction	Assignment	Programming	**No. of Classes:10**		
Topics: Introduction to bigdata, Rise of Big Data and Dimensionality in -Biological Sciences ,Health Sciences , Computer and Information Sciences , Economics and Finance, Business and Program Evaluation, Earth Sciences and Astronomy - Impact of Big Data - Impact of Dimensionality , Computation of Noise Accumulation , Spurious Correlation , Statistical theory - Aim of High-dimensional Statistical Learning.						
Module 2	Multiple Linear Regression	Assignment	Programming	**No. of Classes:12**		
Topics: Multiple Linear Regression, The Gauss-Markov Theorem , Statistical Tests - Weighted Least-Squares , Box-Cox Transformation , Model Building and Basis Expansions, Polynomial Regression - Spline Regression , Multiple Covariates , Ridge Regression - Bias-Variance Tradeoff - Penalized Least Squares - Bayesian Interpretation - Ridge Regression Solution Path - Kernel Ridge Regression , Exponential family 231 5.1.2 Elements of generalized linear models , Maximum likelihood , Computing MLE: Iteratively reweighed least squares , Deviance and Analysis of Deviance, Regularization parameters, Refitted Cross-validation, Extensions to Nonparametric Modeling.						

Module 3	Inference in linear regression	Assignment	Programming	**No. of Classes:14**

Topics:

Inference in linear regression - Debias of regularized regression estimators , Choices of weights , Inference for the noise level , Inference in generalized linear models , Desparsified Lasso , Decorrelated score estimator - Test of linear hypotheses , Numerical comparison - Asymptotic efficiency 345 7.3.1 Statistical efficiency and Fisher information, Linear regression with random design , Partial linear regression , Gaussian graphical models - Inference via penalized least squares , Sample size in regression and graphical models , General solutions , Local semi-LD decomposition , Data swap , Gradient approximation

Module 4	Principal Component Analysis	Assignment	Programming	**No. of Classes:9**

Topics:

Principal Component Analysis -Introduction to PCA , Power Method , Factor Models and Structured Covariance Learning , Factor model and high-dimensional PCA-Cluster Analysis - K-means clustering , Hierarchical clustering , Model-based clustering , Spectral clustering , Data-driven choices of the number of clusters , Variable Selection in Clustering , Sparse K-means clustering , Sparse model-based clustering , Sparse Mixture of Experts Model, Correlation Screening, Generalized and Rank Correlation Screening, Nonparametric Screening, Sure Screening and False Selection.

Targeted Application & Tools that can be used:

Tools: Torch, Google Colaboratory, Spider, Jupiter Notebook

Project work/Assignment:

Text Book

TextBook(s):

T1 Fan, J., Li, R., Zhang, C.-H., and Zou, H. (2020). Statistical Foundations of Data Science. CRC Press.

T2 Wainwright, M. J. (2019). *High-dimensional statistics: A non-asymptotic viewpoint.* Cambridge University Press.

References

R1. James, G., Witten, D., Hastie, T.J., Tibshirani, R. and Friedman, J. (2013). *An Introduction to Statistical Learning with Applications in R* . Springer, New York.

R2. Hastie, T.J., Tibshirani, R. and Friedman, J. (2009). *The elements of Statistical Learning: Data Mining, Inference, and Prediction* (2nd ed). Springer, New York.

R3. Buehlmann, P. and van de Geer, S. (2011). *Statistics for High-Dimensional Data: Methods, Theory and Applications*. Springer, New York.

Book link

R1: Fan, J., Li, R., Zhang, C.-H., and Zou , Statistical Foundations of Data Science. CRC Press.

E book link

R2: W. N. Venables, D. M. Smith and the R Core Team, https://cran.r-roject.org/doc/manuals /R-intro.pdf, October,2022

Web resources:

W1. https://www.youtube.com/playlist?list=PLOU2XLYxmsIK9qQfztXeybpHvru-TrqAP

https://presiuniv.knimbus.com/user#/
Topics relevant to "EMPLOYABILITY SKILLS": - Asymptotic efficiency 345 7.3.1 Statistical efficiency and Fisher information, Linear regression with random design , Partial linear regression **for developing Employability Skills through Participative Learning techniques. This is attained through assessment component mentioned in course handout..**

<table>
<tr><td>Course Code: CSE3511</td><td colspan="3">Course Title: Predictive Analytics and Applications
Type of Course: Program Core</td><td>L-T- P- C</td><td>2</td><td>0</td><td>2</td><td>3</td></tr>
<tr><td>Version No.</td><td colspan="8">1</td></tr>
<tr><td>Course Pre-requisites</td><td colspan="8">MAT1003</td></tr>
<tr><td>Anti-requisites</td><td colspan="8">NIL</td></tr>
<tr><td>Course Description</td><td colspan="8">Predictive Analytics subject is conceptual in nature. The students will be benefited in this course to know about modern data analytic concepts and develop the skills for analyzing and synthesizing data sets for decision making in the firms.</td></tr>
<tr><td>Course Objective</td><td colspan="8">The objective of the course is skill development of student by using Learning techniques</td></tr>
<tr><td>Course Out Comes</td><td colspan="8">On successful completion of the course the students shall be able to:
• CO 1: Define the nature of analytics and its applications. (Remember)
• CO 2: Summarize the concepts of predictive analytics and data mining.(Understand)
• CO 3: Construct the analytical tools in business scenarios to achieve competitive advantage.(Apply)
• CO 4: Build the real-world insights in decision trees and time series analysis methods in dynamic business environment.(Apply)</td></tr>
<tr><td>Course Content:</td><td colspan="8"></td></tr>
<tr><td>Module 1</td><td>Introduction to Predictive Analytics</td><td>Self-Learning</td><td colspan="4">Applications of analytics</td><td>7 Sessions</td></tr>
<tr><td colspan="9">Topics:Analytics- Definition, importance, Analytics in decision making, Applications, Challenges, Experts perception on analytics; Popularity in Analytics; Predictive analytics in business Scenarios-case studies</td></tr>
<tr><td colspan="9"></td></tr>
<tr><td colspan="9"></td></tr>
<tr><td>Module 2</td><td>Principles and Techniques</td><td>Case analysis</td><td colspan="4"></td><td>8 Sessions</td></tr>
<tr><td colspan="9">Topics:
Predictive modeling: Propensity models, cluster models, collaborative filtering, applications and limitations - Statistical analysis: Univariate Statistical analysis, Multivariate Statistical analysis</td></tr>
<tr><td>Module 3</td><td>Model Selection</td><td>Participative Learning & Case Analysis</td><td colspan="4"></td><td>7 Sessions</td></tr>
</table>

Preparing to model the data: supervised versus unsupervised methods, statistical and data mining methodology, cross-validation, overfitting, bias-variance trade-off, balancing the training dataset, establishing baseline performance.
Measuring Performance in Regression Models - Linear Regression and Its Cousins - Non-Linear Regression Models - Regression Trees and Rule-Based Models
Measuring Performance in Classification Models - Discriminant Analysis and Other Linear Classification Models - Non-Linear Classification Models

Module 4	**Time Series Analysis**	Discussion & Presentation		8 Sessions

Time series Model: ARMA, ARIMA, ARFIMA - Temporal mining - Box Jenkinson method, temporal reasoning, temporal constraint networks

Text Book

1.Jeffrey Strickland, Predictive analytics using R, Simulation educators, Colorado Springs, 2015
2.Max Kuhn and Kjell Johnson, Applied Predictive Modeling, 1st edition Springer, 2013.

References

R1 Dinesh Kumar, U. (2021). Business Analytics: The Science of data-Driven Decision Making.
R2 Business Analytics - Data Analysis & Decision Making", S. Christian Albright and Wayne L. Winston, Cengage Publication, 5th Edition, 2012
E book link R1: Raman, R., Bhattacharya, S., & Pramod, D. (2018). Predict employee attrition by using predictive analytics. Benchmarking: An International Journal. https://www-emerald-com-presiuniv.knimbus.com/insight/content/doi/10.1108/BIJ-03-2018- 0083/full/html
2. **E book link R2:** Jing, Z., Luo, Y., Li, X., & Xu, X. (2022). A multi-dimensional city data embedding model for improving predictive analytics and urban operations. Industrial Management & Data Systems, (ahead-of-print). https://www-emerald-com-presiuniv.knimbus.com/insight/content/doi/10.1108/IMDS-01-2022- 0020/full/html
3. **E book link R3:** Singh, R., Sharma, P., Foropon, C., & Belal, H. M. (2022). The role of big data and predictive analytics in the employee retention: a resource-based view. International Journal of Manpower. https://www-emerald-com-presiuniv.knimbus.com/insight/content/doi/10.1108/IJM-03-2021- 0197/full/html
4. **E book link R4:** Mishra, D., Luo, Z., Hazen, B., Hassini, E., & Foropon, C. (2018). Organizational capabilities that enable big data and predictive analytics diffusion and organizational performance: A resource-based perspective. Management Decision. https://www-emerald-com-presiuniv.knimbus.com/insight/content/doi/10.1108/MD-03-2018- 0324/full/html

Web resources:

W1.https://www.sas.com/en_in/insights/analytics/predictive-analytics.html
W2. https://www.techtarget.com/searchbusinessanalytics/definition/predictive-analytics
W3. https://www.cio.com/article/228901/what-is-predictive-analytics-transforming-data-intofuture-insights.html
W4. https://www.simplilearn.com/what-is-predictive-analytics-article
W5. https://www.northeastern.edu/graduate/blog/predictive-analytics/
W6.https://www.marketingevolution.com/knowledge-center/the-role-of-predictive-analyticsin-data-driven-marketing
Swayam & NPTEL Video Lecture Sessions on Predictive Analytics

1. https://onlinecourses.swayam2.ac.in/imb20_mg19/preview
2. https://onlinecourses.nptel.ac.in/noc19_mg42/preview

Case References

1. Predictive Analytics Industry Use cases.
2. https://www.rapidinsight.com/blog/11-examples-ofpredictive-analytics/
3. Srinivasan Maheswaran (2017). Predictive Analytics – Employee Attrition Case center.

Topics relevant to development of "Skill Development": ": Application of Business Analytics to enhances customer satisfaction and firms' success

Topics relevant to development of "Environment and sustainability: Focus on Predictive analytics to minimize the errors in decision making

Course Code: CSE3512	**Course Title: Data Mining** **Type of Course: Discipline Elective/ Theory Only**	**L- T-P-C**	3	0	0	3
Version No.	2.0					
Course Pre-requisites	MAT1003					
Anti-requisites	**NIL**					
Course Description	Introduction, Applications, issues in data mining, data pre-processing techniques, data mining tasks, association rules, advanced association rules, classification, different approaches for classification, clustering, outlier detection. Recent trends in data mining.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Data Mining** and attain **Employability** through **Problem Solving** Methodologies					
Course Out Comes	**On successful completion of the course the students shall be able to:** • Apply the various pre-processing techniques needed for a data mining task. • Understand the functionality of the various data mining algorithms. • Appreciate the strengths and limitations of various data mining models. • Understand the advances in data mining for real life applications.					
Course Content:						

Module 1	**Introduction to Data Mining**	Assignment		Data Collection	**5 Sessions**
	Topics: Introduction to Data mining – Data Mining Goals– Stages of the Data Mining Process–Data Mining Techniques– Merits and Demerits.				
Module 2	**Data preprocessing**	Quiz		Problem Solving	**9 Sessions**
	Topics: Types of data – Pre Processing steps – Data Preprocessing Techniques – Similarity and Dissimilarity measures.				
Module 3	**Data Mining – Frequent Patterns**	Assignment		Problem Solving	**7 Sessions**
	Topics: Market Basket Analysis, item sets – Generating frequent item sets and rules efficiently – Apriori Algorithm– FPGrowth.				
Module 4	**Classification and clustering**	Assignment		Problem Solving	**11 Sessions**
	Classification and Clustering Decision tree Induction – Bayesian classification –Classification by Back Propagation - Lazy learners – Modern evaluation and selection techniques to improve classification accuracy. Clustering Analysis – portioning method – Hierarchical methods – Density based method				
Module 5	**Outlier detection & Data mining trends**	Assignment		Problem Solving	**5 Sessions**
	Anomaly detection preliminaries - Different Outlier detection techniques-Web mining- Text mining-Demonstration of Weka tool.				
	Project work/Assignment:				
	Assignments 1. From the dataset given, find the Entropy, Gain value of the attributes and also draw the decision tree using entropy for the given dataset. 2. Transactional Data Base, D given below which contains set of items find the frequent item set using the Apriori Algorithm and generate the Association Rules. Minimum Support count is 2%. Minimum confidence is 60%.				

T_{id}	Items
10	1, 3, 4
20	2, 3, 5
30	1, 2, 3, 5
40	2, 5

Text Book

T1 T1. Tan P. N., Steinbach M & Kumar V. "Introduction to Data Mining" ,Pearson Education, 2016.

References

R1 *Han J & Kamber M, "Data Mining: Concepts and Techniques", Elsevier, Second Edition, 2006*

R2 *G K Gupta, "Introduction to Data Mining with Case Studies", PHI, Third Edition, 2014.*

R3 *Alex Berson and Stephen J. Smith, "Data Warehousing, Data Mining and OLAP", Tata McGraw – Hill*

Additional web-based resources

W1. https://onlinecourses.swayam2.ac.in/cec20_cs12/previewText book of Data Mining: Concepts and Techniques, Jiawei Han, Micheline Kamber and Jian Pei, Morgan Kaufmann Publishers, 2012.

W2.https://puniversity.informaticsglobal.com:2284/ehost/detail/detail?vid=7&sid=e2d7362a-fd3049a98f0393e963521dbd%40redis&bdata=JnNpdGU9ZWhvc3QtbGl2ZQ%3d%3d#AN=377411&db=nlebk

3. https://nptel.ac.in/courses/105105157

Topics relevant to "EMPLOYABILITY SKILLS": Data Mining Techniques, FP Growth for developing **Employability Skills** through **Participative Learning** techniques. This is attained through the assessment component mentioned in the course handout.

Course Code: CSE3513	**Course Title: No SQL Data Management** **Type of Course: Lab Integrated**	**L-T-P-C**	2	0	2	3

Version No.		1.0
Course Pre-requisites		CSE2261
Anti-requisites		NIL
Course Description		The **NoSQL Data Management** course provides a comprehensive understanding of non-relational database systems, emphasizing their emergence as scalable and flexible alternatives to traditional relational databases. It covers various NoSQL data models, including key-value, document, column-family, and graph databases, exploring their structure, use cases, and design principles. Students will learn about the challenges of data persistence, concurrency, and integration, and how aggregate-oriented models address these issues. The course delves into distribution models such as sharding and replication, the implications of the CAP theorem, and consistency trade-offs. Additionally, it introduces Map-Reduce for large-scale data processing and guides students in modeling data for access efficiency. Practical applications and limitations of each NoSQL type are discussed, preparing students to design robust, scalable data solutions for real-world applications.
Course Objectives		The objective of the course is to introduce students to the principles, models, and practical applications of NoSQL databases. It aims to equip students with the knowledge and skills necessary to understand the advantages of NoSQL over traditional relational databases, comprehend various data models (key-value, document, column-family, and graph), explore distribution models for scalability and availability, and apply Map-Reduce and other techniques for data processing.
Course Out Comes		1. Understand the limitations of relational databases and explain the motivation behind the emergence of NoSQL databases. 2. Differentiate between various NoSQL data models such as key-value, document, column-family, and graph databases based on structure, use cases, and performance. 3. Analyze and apply appropriate distribution strategies including sharding, replication, and consistency models in distributed NoSQL systems. 4. Design and implement data access patterns using aggregate-oriented modeling and schema-less approaches for scalable NoSQL applications. 5. Utilize Map-Reduce and other data processing techniques to handle large-scale data operations efficiently in NoSQL environments.
Course Content:		

Module 1	**Introduction to NoSQL and Aggregate-Oriented Data Models**	**Quiz**		**Knowledge based quiz**	**No. of sessions:8**
	Why NoSQL? The Value of Relational Databases, Getting at Persistent Data, Concurrency, Integration, A (Mostly) Standard Model, Impedance Mismatch, Application and Integration Databases, Attack of the Clusters, The Emergence of NoSQL, Aggregate Data Models; Aggregates, Example of Relations and Aggregates, Consequences of Aggregate Orientation, Key-Value and Document Data Models, Column-Family Stores, Summarizing AggregateOriented Databases. More Details on Data Models; Relationships, Graph Databases, Schema less Databases, Materialized Views, Modelling for Data Access.				

Module 2	**Distributed Data Systems and Consistency Models**	**Assignment**		**Data Visualization**	**No. of sessions:10**
	Distribution Models; Single Server, Sharding, Master-Slave Replication, Peer-to-Peer Replication, Combining Sharding and Replication Consistency, Update Consistency, Read Consistency, Relaxing Consistency, The CAP Theorem, Relaxing Durability, Quorums. Version Stamps, Business and System Transactions, Version Stamps on Multiple Nodes				

Module 3	**Key-Value Stores and Map-Reduce Framework**	**Design an algorithm using Example**		**Random Forest**	**No. of sessions:10**
	Map-Reduce, Basic Map-Reduce, Partitioning and Combining, Composing Map-Reduce Calculations, A Two Stage Map-Reduce Example, Incremental Map-Reduce Key-Value Databases, What Is a Key-Value Store, Key-Value Store Features, Consistency, Transactions, Query Features, Structure of Data, Scaling, Suitable Use Cases, Storing Session Information, User Profiles, Preference, Shopping Cart Data, When Not to Use, Relationships among Data, Multi operation Transactions, Query by Data, Operations by Sets.				
Module 4	**Document-Oriented Databases and Use Cases**	**Case Study**		**Conduct a case study on how data sets can be gathered and implemented in real time application.**	**No. of sessions:10**
	Document Databases, What Is a Document Database?, Features, Consistency, Transactions, Availability, Query Features, Scaling, Suitable Use Cases, Event Logging, Content Management Systems, Blogging Platforms, Web Analytics or Real-Time Analytics, E- Commerce Applications, When Not to Use, Complex Transactions Spanning Different Operations, Queries against Varying Aggregate Structure				
Module 5	**Graph Databases and Connected Data Solutions**	**Case Study**			**No. of sessions:10**
	Graph Databases, What Is a Graph Database?, Features, Consistency, Transactions, Availability, Query Features, Scaling, Suitable Use Cases, Connected Data, Routing, Dispatch, and Location-Based Services, Recommendation Engines, When Not to Use.				
	TEXTBOOKS: 1. Sadalage, P. & Fowler, NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence, Pearson Addision Wesley, 2012 REFERENCE BOOKS: 1. Dan Sullivan, "NoSQL For Mere Mortals", 1st Edition, Pearson Education India, 2015. (ISBN- 13: 978-9332557338) 2. Dan McCreary and Ann Kelly, "Making Sense of NoSQL: A guide for Managers and the Rest of us", 1st Edition, Manning Publication/Dreamtech Press, 2013. (ISBN-13: 978-9351192022) 3. Kristina Chodorow, "Mongodb: The Definitive Guide- Powerful and Scalable Data Storage", 2nd Edition, O'Reilly Publications, 2013. (ISBN-13: 978-9351102694) VIDEO LINKS: 1. https://www.geeksforgeeks.org/introduction-to-nosql/ (and related links in the page) 2. https://www.youtube.com/watch?v=0buKQHokLK8 (How do NoSQL databases work? Simply explained)				

Course Code: CSE3514	Course Title: Applied Data Intelligence Type of Course: Program Core	L-T-P-C	2	0	2	3

Version No.	1.0
Course Pre-requisites	CSE2264
Anti-requisites	NIL
Course Description	The aim of the course is to give complete overview of Python's data analytics tools and techniques. Learning python is a crucial skill for many data science roles, and this course helps to understand and develop feature engineering. With a blended learning approach, Python for data science along with concepts like data wrangling, mathematical computing, and more can be learnt.
Course Objectives	The objective of the course is to familiarize the learners with the concepts of **Applied Data Science** and attain **Employability** through E**xperiential Learning** techniques.
Course Out Comes	On successful completion of this course the students shall be able to: 1. Understand Numpy and Matrix Operations [Knowledge] 2. Analyze the need for data preprocessing and visualization techniques. [Comprehensive] 3. Demonstrate the performance of different supervised learning algorithms like decision Tree, Random Forest, Linear Regression, Logistic Regression etc. [Application] 4. Apply unsupervised learning algorithms like K-Means, K-Medoids etc for grouping the given data. [Applicaion]
Course Content:	

Module 1	**Introduction to Data Science, Python Data Structures, Python Numpy Package**	**Quiz**		**Knowledge based quiz**	**No. of sessions:8**
	Data Science - Need, Applications, Difference between data analysis and data analytics. Python- Variables, data types, control structures, Operators, Simple operations, Array and its operations, Numpy operations, Matrix and its operations				
Module 2	**Data preparation and preprocessing using Pandas dataframe, Exploratory Data Analysis, Data Visualization**	**Assignment**		**Data Visualization**	**No. of sessions:10**
	Dealing missing values, Normalization, statistical description about the data, Accessing the data, Summary of the data, Relationship between the data, Data Visualization using matplotlib				
Module 3	**Supervised Learning Algorithms**	**Design an algorithm using Example**		**Random Forest**	**No. of sessions:10**
	Decision Tree Algorithm, ID3 Classifier, Random Forest, Classifier Accuracy, Linear Prediction, Logistic Regression – Case study				
Module 4	**Unsupervised Learning Algorithms**	**Case Study**		**Conduct a case study on how data sets can be gathered and implemented in real time application.**	**No. of sessions:10**
	Various distance Function, Dissimilarity between the mixed types of data, K-Means Algorithm, K- Medoids Algorithm -Case Study				
	List of Laboratory Tasks: 1. Introduction to R tool for data analytics science 2. Basic Statistics and Visualization in R 3. K-means Clustering				

	4. Association Rules 5. Linear Regression 6. Logistic Regression 7. Naive Bayesian Classifier 8. Decision Trees 9. Simulate Principal component analysis 10. Simulate Singular Value Decomposition
	Targeted Application & Tools that can be used: • IBM SPSS • Julia and Jupyter Notebook • Matplotlib
	Project work/Assignment:
	1. Design forest fire and wildfire prediction system. 2. Driver Drowsiness Detection System with OpenCV & Keras 3. Credit Card Fraud Detection using Python.
	Textbook(s): 1. Applied Data Science with Python and Jupyter-Alex Galea,Packt Publishing,October2018 2. Data Visualization in Python with Pandas and Matplotlib Paperback –DavidLandup, June 16, 2021
	References: 1.Data Science with Python and Dask- Jesse Daniel,1st Edition,July30,2019 **Weblinks:** • Udemy: https://www.udemy.com/course/applied-data-science-with-python-specialization-mhm/ • NPTEL online course : https://nptel.ac.in/courses/106106179 • https://presiuniv.knimbus.com/user#/home
	Topics relevant to "EMPLOYABILITY SKILLS": Data Science, Decision Tree Algorithm **for developing Employability Skills through Experiential Learning techniques. This is attained through assessment component mentioned in course handout.**

Course Code: CSE3515	**Course Title: Cloud Data Engineering Type of Course : Theory**	**L-T- P-C**	2	0	2	3

Version No.		1.0
Course Pre-requisites		CSE2272
Anti-requisites		nil
Course Description		This Course is designed to introduce the concepts of Cloud Computing as a new computing paradigm. Cloud Computing has emerged in recent years as a new paradigm for hosting and delivering services over the Internet. The students can explore various Cloud Computing terminology, principles and applications. Understanding different views of the Cloud Computing such as theoretical, technical and commercial aspects. **Topics include:** Evolution of cloud computing and its services available today, Introduction, Architecture of cloud computing, Infrastructure, platform, software, Types of cloud, Business models, cloud services, Collaborating using cloud services, Virtualization for cloud, Security, Standards and Applications.
Course Objective		The objective of the course is to familiarize the learners with the concepts of **Could computing and Virtualization** and attain **Employability** through **Participative Learning** techniques.
Course Out Comes		**On successful completion of the course the students shall be able to:** • Describe fundamentals of cloud computing, virtualization and cloud computing services. • Discuss high-throughput and data-intensive computing. • Explain security and standards in cloud computing. • Demonstrate the installation and configuration of virtual machine.
Course Content:		

Module 1		**10 Sessions**
	Introduction to Cloud and Virtualization Cloud Computing at a Glance, Historical Developments, Building Cloud Computing Environments, Computing Platforms and Technologies, Virtualization, Characteristics of Virtualized Environments Taxonomy of Virtualization Techniques, Virtualization and Cloud Computing, Technology Examples, Cloud Computing Architecture, IaaS, PaaS, SaaS, Types of Clouds, Economics of Cloud	
Module 2		**10 Sessions**
	High Throughput and Data Intensive Computing: Task computing, MPI applications, Task based programming, Introduction to DIC, Technologies for DIC, Aneka Map Reduce Programming	
Module 3		**09 Sessions**
	Cloud Security and Standards : Cloud Security Challenges, Software-as-a-Service Security, Application standards, Client standards, Infrastructure and Service standards.	
Module 4		**09 Sessions**
	Cloud Platforms, Advances in cloud: introduction to **Amazon Web Services**: Introduction to **Google App Engine,** Introduction to **Microsoft Azure.** **Media Clouds - Security Clouds - Computing Clouds - Mobile Clouds – Federated Clouds – Hybrid Cloud**	
	Text Book 1. John Rittinghouse and James Ransome, "Cloud Computing, Implementation, Management and Security", CRC Press. 2. Rajkumar Buyya, Christian Vecchiola, and Thamarai Selvi, "Mastering Cloud Computing", McGraw	

	Hill Education.
	References 1. David E.Y. Sarna, "Implementing and Developing Cloud Applications", CRC Press. 2. Anthony T Velte, Toby J Velte, Robert Elsenpeter, "Cloud Computing: A Practical Approach", Tata McGraw-Hill. **Web resources:** https://presiuniv.knimbus.com/user#/home
	Topics relevant to "EMPLOYABILITY SKILLS": Aws, Azure, APIs, Aneka Cloud Platform, EC2, Installation of VM Workstation**,** Infrastructure Security Challenges for developing **Employability Skills** through **Participative Learning techniques**. This is attained through assessment component mentioned in course handout..

Course Code: CSE3516	**Course Title:** Federated Learning **Type of Course: Theory**	**L- T-P- C**	2	0	2	3

Version No.		1
Course Pre-requisites		CSE2272
Anti-requisites		
Course Description		Federated Learning can improve the performance of models by leveraging the diversity of the data across different devices. In this course, students will learn basics of Federated Learning and will be able to apply the real-time updates of the model in various practical scenarios.
Course Objective		The objective of the course is to get exposure to need for distributed model updates and understand the importance of privacy and security in machine learning techniques.
Course Out Comes		**On successful completion of the course the students shall be able to:** • Describe the key concepts and architecture of Federated Learning. (Knowledge) • Apply different methods to develop federated learning systems. (Comprehension) • Apply optimization techniques in Federated Learning (Application) • Construct and scale a simple federated system (Application) • Evaluate privacy and security concerns in Federated Learning and implement privacy-preserving techniques (Application)
Course Content:		

Module 1	**Introduction to Federated Learning:**	Assignment		Data Collection/Interpretation	**10 Sessions**
	Topics: Introduction to Federated Learning – Overview of Federated Learning: Definition, History, and Applications – Concepts and Terminology – Federated Learning Architecture -Machine Learning Perspective - Security & Privacy in Federated Learning – Federated Learning vs Centralized Learning: Comparison and Contrast.				
Module 2	**Horizontal and Vertical Federated Learning**	Case studies / Case let		Case studies / Case let	**13 Sessions**
	Topics: Horizontal Federated Learning (HFL) -Definition and Architecture of Horizontal Federated Learning – Federated Averaging (FedAvg) Algorithm – Improvements on the FedAvg Algorithm. Vertical Federated Learning (VFL) – Definition and Architecture of Vertical Federated Learning – VFL Algorithms: Secure Federated Linear Regression, Secure Federated Tree Boosting. Federated Learning with Non-IID Data – Heterogeneity in Federated Learning -Stratification and Local Updated Rules – Advanced Optimization Techniques in Federated Learning -Adaptive Learning Rate -Momentum and Weight Decay				
Module 3	**Federated Transfer Learning and Security**	Case studies / Case let		Case studies / Case let	**14 Sessions**
	Topics: Federated Transfer Learning (FTL) – Framework of Federated Transfer Learning – Homomorphic Encryption in FTL – FTL Training Process -FTL Prediction Process – Security Analysis of FTL – Secret Sharing based FTL Security in Federated Learning – Protecting Against Data Leakage in FL -Private Parameter Aggregation for FL – Data Leakage in FL Advanced Security Issues -Dealing with Byzantine Threats to Neural Networks in FL.				
	Targeted Application & Tools that can be used: Building and simulating federated learning systems using tensorFlowFederated (TFF), PySyft, Google Colab / Jupyter Notebook.				

	Simulating and deploying FL across nodes using Cloud Platforms (AWS, GCP, Azure) It can help small, medium and large businesses in any sector keep information assets secure.
	Project work/Assignment:
	Assignment: Practical Applications and Case Studies -Real-world Applications of Federated Learning
	Text Book **T1** Federated learning comprehensive overview of methods and applications Springer Nature Switzerland AG; 1st ed. 2022 edition By Heiko Ludwig (Editor), Nathalie Baracaldo **T2** Federated Learning (Synthesis Lectures on Artificial Intelligence and Machine Learning) , by Ronald J. Brachman, Francessa Rossi, and Peter Stone, Series Editors, Released 30 December 2019. Publisher(s): Morgan & Claypool Publishers. .
	References **R1 Federated Learning with Python by Kiyoshi Nakayama PhD, George Jeno, O'Reilly Media, Inc. Pub.** **R2 What-is-federated learning? By Emily Glanz, Nova Fallen, O'Reilly Media, Inc. Pub.** **E book link R1:** http://www.iso.org/iso/home/standards/management- standards/iso27001.html **E book link R2:** https://www.oreilly.com/library/view/what-is-federated/9781098107253/ch03.html
	Topics relevant to development of "SKILL DEVELOPMENT": Security Policy Implementation, Security Roles, for development of Skill Development through Participative Learning Techniques. This is attained through assessment component mentioned in course handout.

Course Code: CSE3517	Course Title: Edge Computing Type of Course: Theory Only	L-T-P-C	2	0	2	3
Version No.	1.0					
Course Pre-requisites	CSE2272					
Anti-requisites	**Nil**					
Course Description	In this course, we will study significant tools and applications that comprise today's cloud computing platform, with a special focus on using the cloud for big data applications. The course covers various topics such as the evolution of computing industry, cloud computing basics and edge computing. The course provides information on the different types of edge compute deployments, different types of edge compute services (such as CDN Edge, IOT Edge, and Multi-access Edge (MEC)). The course also educates the students on the different vendor platforms, software services, standard bodies and open source communities available for edge computing. Students will also create a research project of their choosing.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of Edge Computing and attain **Employability** through **Problem Solving** Methodologies.					
Course Out Comes	On successful completion of the course the students shall be able to: CO1 Understand the principles, architectures of edge computing (Knowledge) CO2 Describe IoT Architecture and Core IoT Modules (Comprehension) CO3 Summarize edge to Cloud Protocols (Comprehension) CO4 Describe Edge computing with RaspberryPi (Comprehension)					
Course Content:						

Module 1	IoT and Edge Computing Definition and Use Cases	Term paper/Assignment/Case Study	Programming/Simulation/Data Collection/any other such associated activity		**9 Sessions**
	Topics: Introduction to Edge Computing Scenario's and Use cases - Edge computing purpose and definition, Edge computing use cases, Edge computing hardware architectures, Edge platforms, Edge vs Fog Computing, Communication Models - Edge, Fog and M2M.				
Module 2	IoT Architecture and Core IoT Modules	Term paper/Assignment/**Case Study**	Programming/Simulation/Data Collection/any other such associated activity		**9 Sessions**
	Topics: A connected ecosystem,IoT versus machine-to-machine versus, SCADA, The value of a network and Metcalfe's and Beckstrom's laws, IoT and edge architecture, Role of an architect, Understanding Implementations with examples-Example use case and deployment, Case study – Telemedicine palliative care, Requirements, Implementation, Use case retrospective.				
Module 3	RaspberryPi	Term paper/Assignment/Case Study	**Programming/Simulation**/Data Collection/any other such associated activity		**10 Sessions**
	Topics: Introduction to RaspberryPi, About the RaspberryPi Board: Hardware Layout and Pinouts, Operating Systems on RaspberryPi, Configuring RaspberryPi, Programming RaspberryPi, Connecting Raspberry Pi via SSH, Remote access tools, Interfacing DHT Sensor with Pi, Pi as Webserver, Pi Camera, Image & Video Processing using Pi.				
Module 4	Edge to Cloud Protocols	Term paper/Assignment/Case Study	Programming/Simulation/Data Collection/any other such associated activity		**7 Sessions**

	Topics: Implementation of Microcomputer RaspberryPi and device Interfacing, Edge to Cloud Protocols-Protocols,MQTT, MQTT publish-subscribe, MQTT architecture details, MQTT state transitions,MQTT packet structure, MQTT data types, MQTT communication formats, MQTT 3.1.1 working example.			
Module 5	Edge computing with RaspberryPi	Term paper/Assignment/Case Study	**Programming/Simulation**/Data Collection/any other such associated activity	**7 Sessions**
	Topics: Edge computing with RaspberryPi, Industrial and Commercial IoT and Edge, Edge computing and solutions.			
	Targeted Application & Tools that can be used: • **Application** : Smart Surveillance Video Stream Processing at the Edge for Real-Time Human Objects Tracking. • **Tools** :Eclipse ioFog : An integrated development environment built by the Eclipse Foundation, backed by IBM. Eclipse ioFog is the organization's open-source edge computing platform.			
	Project work/Assignment: Mention the Type of Project /Assignment proposed for this course			
	Exploring topics such as developing scalable architectures, moving from closed systems to open systems, and ethical issues rising from data sensing, addresses both the challenges and opportunities of Edge computing presents. Students can harness federating Edge resources, middleware design issues, data management and predictive analysis, smart transportation and surveillance applications, and more. A coordinated and integrated solutions can be provided by thorough knowledge of the foundations, applications, and issues that are central to Edge computing.			
	Text Book 10. IoT and Edge Computing for Architects - Second Edition, by Perry Lea, Publisher: Packt Publishing, 2020, ISBN: 9781839214806 2. Raspberry Pi Cookbook, 3rd Edition, by Simon Monk, Publisher: O'Reilly Media, Inc., 2019, ISBN: 978149204322.			
	Topics relevant to "EMPLOYABILITY SKILLS": Implementation of Microcomputer RaspberryPi and device Interfacing for developing **Employability Skills** through **Problem Solving methodologies**. This is attained through assessment component mentioned in course handout.			

Course Code: CSE3518	**Course Title: Network Security and Firewall Management** **Type of Course: Lab Integrated**		**L-T- P- C**	2	0	2	3
Version No.		1					
Course Pre-requisites		CSE2502					
Anti-requisites							
Course Description		This course provides an in-depth study of various network attacks techniques and methods to defend against them. A number of threats and vulnerabilities of the Internet will be covered, including various vulnerabilities of TCP/IP protocols, denial of service (DOS), attacks on routing, attacks on DNS servers, TCP session hijacking, and so on. This course will also cover defending mechanisms, including intrusion detection, firewalls, tracing the source of attacks, anonymous communication, IPsec, virtual private network, and PKI. To make it easy for students to understand these attacks, basics of the TCP/IP protocols will also be covered in the course.					
Course Objective		The objective of the course is to familiarize the learners with the concepts of **Firewall and Internet security** and attain **Skill Development** through **Problem Solving** Methodologies.					
Course Out Comes		**On successful completion of the course the students shall be able to:** • To identify elements of firewall design, types of security threats and responses to security attacks. • Examine security incident postmortem reporting and ongoing network security activities. • Construct code for authentication algorithms. • Develop a signature scheme using Digital signature standard. • Demonstrate the network security system using open source tools					
Course Content:							

Module 1	Introduction to Firewall	Assignment		Data Collection/Interpretation	**12 Sessions**
	Introduction of Firewall in computer network,Categories of firewall,How firewall works,Types of firewall, Firewall location and Configuration,Firewall Policies,Firewall Biasing,Network Architecture,Net masks,Packet filters,Stateful firewalls,Resources				
Module 2	**Computer security**	Case studies / Case let		Case studies / Case let	**12 Sessions**
	Topics: Attacks on Computers and Computer Security: Need for Security, Security Approaches, Principles of Security Types of Attacks. Transport Level Security: Web Security Considerations, Secure Sockets Layer, Transport Layer Security, HTTPS, Secure Shell (SSH)				
Module 3	**Network Security**	Quiz		Case studies / Case let	**10 Sessions**
	Topics: Overview of Network Security:Elements of Network Security , Classification of Network Attacks ,Security Methods ,Symmetric-Key Cryptography :Data Encryption Standard (DES),Advanced Encryption Standard (AES) , Public-Key Cryptography :RSA Algorithm ,Diffie-Hellman Key-Exchange Protocol , Authentication :Hash Function , Secure Hash Algorithm (SHA) , Digital Signatures.				
Module 4	Cyber laws and Compliance Standards	Quiz		Case studies / Case let	**11 Sessions**
	Topics: Kerberos:Working ,ASS,TGS,SS-Internet security protocols-AH,ESP,Models-Transport and tunnel-Email security,Public key Infrasturcture,Certificates,certificates authority.Cyber Crime: Introduction,Hacking,Digital forgery,Cyber Stalking,Identify theft and Fraud,Cyber terrorism,Cyber defamation,Crime against individual,Government,Property.				
	List of Laboratory Tasks: 1. Perform encryption, decryption using the following substitution techniques				

	(i) Ceaser cipher, (ii) playfair cipher iii) Hill Cipher iv) Vigenere cipher 2. Perform encryption and decryption using following transposition techniques i) Rail fence ii) row & Column Transformation 3. Apply DES algorithm for practical applications. 4. Apply AES algorithm for practical applications. 5. Implement RSA Algorithm using HTML and JavaScript 6. Implement the Diffie-Hellman Key Exchange algorithm for a given problem. 7. Calculate the message digest of a text using the SHA-1 algorithm. 8. Implement the SIGNATURE SCHEME – Digital Signature Standard. 9. Demonstrate intrusion detection system (ids) using any tool eg. Snort or any other s/w. 10. Automated Attack and Penetration Tools Exploring N-Stalker, a Vulnerability Assessment Tool 11. Defeating Malware i) Building Trojans ii) Rootkit Hunter
	Targeted Application & Tools that can be used
	Text Book **T1 :** Behrouz A Forouzan, Data and Communications and Networking, Fifth Edition, McGraw Hill, Indian Edition **T2:** James F Kurose and Keith W Ross, Computer Networking, A Top-Down Approach, Sixth edition, Pearson,2017
	References **R1:** Andrew S Tanenbaum, Computer Networks, fifth edition, Pearson Edition **R2**: Nader F Mir, Computer and Communication Networks, 2nd Edition, Pearson, 2014. **Web resources:** 1. https://networklessons.com/cisco/asa-firewall 2. https://www.udemy.com/course/cisco-asa-firewall-lab-guide 3. https://geekflare.com/learn-network-security
•	**Topics relevant to development of "Skill Development": AES, Network Security for Skill Development through Problem Solving methodologies. This is attained through assessment component mentioned in course handout.**

Course Code: CSE3519	**Course Title:** Information Security and Management **Type of Course: Theory Only**	**L- T-P-C**	3	0	0	3
Version No.	1					
Course Pre-requisites	CSE2502					
Anti-requisites						
Course Description	The course explores information security through some introductory material and helps gain an appreciation of the scope and context of information security. It includes a brief introduction to cryptography, security management, network and computer security. It allows a student to begin a fascinating journey into the study of information security and develop an appreciation of some key security concepts. The course concludes with a discussion of a simple model of the information security in industry and explores skills, knowledge and roles required for employability. A student will be able to determine and analyze potential career opportunities in this profession.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of Information Security and Management and attain **Employability** through **Participative Learning** techniques.					
Course Out Comes	**On successful completion of the course the students shall be able to:** • Describe the basic concept of information security. (Knowledge) • Explain the concepts and methods of cryptography. (Comprehension) • Demonstrate the aspects of risk management. (Application)					
Course Content:						

Module 1	**Information Security Management:**	Assignment		Data Collection/Interpretation	**10 Sessions**
	Topics: Information Security Overview, Threat and Attack Vectors, Types of Attacks, Common Vulnerabilities and Exposure (CVE), Security Attacks, Fundamentals of Information Security, Computer Security Concerns, Information Security Measures.				
Module 2	**Fundamentals of Information Security and Data Leakage**	Case studies / Case let		Case studies / Case let	**13 Sessions**
	Topics: Key Elements of Networks, Logical Elements of Networks, Critical Information Characteristics, Information States. What is Data Leakage and Statistics, Data Leakage Threats, Reducing the Risk of Data Loss, Key Performance Indicators (KPI), Database Security.				
Module 3	**Information Security Policies and Management**	Case studies / Case let		Case studies / Case let	**14 Sessions**
	Topics: Information Security Policies-Necessity-Key Elements and Characteristics, Security Policy Implementation, Configuration, Security Standards-Guidelines and Frameworks, Security Roles and Responsibilities, Accountability, Roles and Responsibilities of Information Security Management, Team Responding to Emergency Situation- Risk Analysis Process.				
	Targeted Application & Tools that can be used: An ISMS is a systematic approach to managing sensitive company information so that it remains secure. It includes people, processes and IT systems by applying a risk management process. It can help small, medium and large businesses in any sector keep information assets secure. The ISO 27000 family of standards helps organizations keep information assets secure.				

	Using this family of standards will help your organization manage the security of assets such as financial information, intellectual property, employee details or information entrusted to you by third parties. ISO/IEC 27001 is the best-known standard in the family providing requirements for an information security management system (ISMS).
	Project work/Assignment:
	Assignment:
	Text Book **T1** Management of Information Security by Michael E.Whilman and Herbert J.Mattord **T2** Information Security: The Complete Reference, Second Edition, 2nd Edition. by Mark Rhodes-Ousley. Released April 2013. Publisher(s): McGraw-Hill. .
	References **R1 Title, Cryptography & Network Security (Sie) 2E. Author, Forouzan. Publisher, McGraw-Hill Education (India) Pvt Limited.** **R2 Information Systems Security, 2ed: Security Management, Metrics, Frameworks and Best Practices. Nina Godbole.** **E book link R1:** http://www.iso.org/iso/home/standards/management- standards/iso27001.html **E book link R2:** http://csrc.nist.gov/publications/nistpubs/800-55-Rev1/SP800-55-rev1.pdf WEBLINKS: pu.informatics.global , https://sm-nitk.vlabs.ac.in.
	Topics relevant to development of "SKILL DEVELOPMENT": Security Policy Implementation, Security Roles, for development of Skill Development through Participative Learning Techniques. This is attained through assessment component mentioned in course handout.

Course Code: CSE3520	Course Title: Network Intrusion Detection and Prevention Type of Course:1] PCC 2] Theory Only	L- T-P-C	3	0	0	3

Version No.		1.0
Course Pre-requisites		CSE2502
Anti-requisites		**NIL**
Course Description		Objective of the course is to Understand when, where, how, and why to apply Intrusion Detection tools and techniques in order to improve the security posture of an enterprise. Apply knowledge of the fundamentals and history of Intrusion Detection in order to avoid common pitfalls in the creation and evaluation of new Intrusion Detection Systems and Analyze intrusion detection alerts and logs to distinguish attack types from false alarms.
Course Objectives		The objective of the course is to familiarize the learners with the concepts of **Intrusion Detection and Prevention System** and attain **Skill Development** through **Participative Learning** techniques.
Course Out Comes		On successful completion of the course the students shall be able to: • Understand about the intruders. • Define intrusion detection and prevention policies • Explain the fundamental concepts of Network Protocol Analysis and demonstrate the skill to capture and analyze network packets. • Use various protocol analyzers and Network Intrusion Detection Systems as security tools to detect network attacks and troubleshoot network problems.
Course Content:		

Module 1	**Introduction to Intrusion Detection and Prevention System**	Assignment	Programming Task	**10 Sessions**

Topics

Understanding Intrusion Detection – Intrusion detection and prevention basics – IDS and IPS analysis schemes, Attacks, Detection approaches –Misuse detection – anomaly detection – specification based detection – hybrid detection. Internal and external threats to data, Need and types of IDS, Information sources,Host based information sources, Network based information sources.

Assignment: Demonstrating the skills to capture and analyze network packets using network packet analyzer.

Module 2	**Intrusion Prevention System**	Assignment	Programming Task	**10 Sessions**

Topics:

Intrusion Prevention Systems, Network IDs protocol based IDs, Hybrid IDs, Analysis schemes, thinking about intrusion. A model for intrusion analysis, techniques, Responses, requirement of responses, Types of responses, mapping responses to policy Vulnerability analysis, credential analysis, non-credential analysis. Architecture models of IDs and IPs.

Assignment: Applying Intrusion detection in security applications.

Module 3	**Applications and tools**	Assignment	Programming/Data analysis task	**12 Sessions**

Topics:
Tool Selection and Acquisition Process – Bro Intrusion Detection – Prelude Intrusion Detection – Cisco Security IDS – Snorts Intrusion Detection – NFR security. Introduction to Snort, Snort Installation Scenarios, Installing Snort, Running Snort on Multiple Network Interfaces, Snort Command Line Options. Step-By-Step Procedure to Compile and Install Snort Location of Snort Files, Snort Modes Snort Alert Modes

Assignment: Demonstrate the working with Snort Rules, Rule Headers, Rule Options and The Snort Configuration File.

Module 4	Legal issues and organizations standards	Assignment	Programming/Data analysis task	**9 Sessions**

Law Enforcement / Criminal Prosecutions – Standard of Due Care – Evidentiary Issues, Organizations and Standardizations.

Assignment: Addressing common legal concerns and myths about Intrusion Detection system

Textbooks

T1. Carl Endorf, Eugene Schultz and Jim Mellander " Intrusion Detection & Prevention", 1st Edition, Tata McGraw-Hill, 2004.
T2. Earl Carter, Jonathan Hogue, "Intrusion Prevention Fundamentals", Pearson Education, 2006.

References

R1. Rafeeq Rehman : " Intrusion Detection with SNORT, Apache, MySQL, PHP and ACID," 1st Edition, Prentice Hall , 2003.
R2. Christopher Kruegel, Fredrik Valeur, Giovanni Vigna: "Intrusion Detection and Correlation Challenges and Solutions", 1st Edition, Springer, 2005.
R3. Paul E. Proctor, "The Practical Intrusion Detection Handbook ",Prentice Hall , 2001.

Weblinks:
https://www.youtube.com/watch?v=RYB4cG8G2xo
https://www.coursera.org/lecture/detecting-cyber-attacks/intrusion-detection-systems-UeDqJ

Topics relevant to "SKILL DEVELOPMENT": Agent development for intrusion detection for Skill Development through **Participative Learning techniques**. This is attained through assessment component mentioned in course handout.

Course Code: CSE3521	Course Title: Principles and Practices of Web Security Type of Course: Lab Integrated	L- T-P- C	2	0	2	3
Version No.	1					
Course Pre-requisites	CSE2502					
Anti-requisites	NIl					
Course Description	The purpose of this course this course is to introduce you to the field of web security by understanding web functionality and various security validations. The web is our gateway to many critical services and is quickly evolving as a platform to connect all our devices. Web vulnerabilities are growing on a year-to-year basis and designing secure web applications is challenging. The course covers fundamental concepts of web security principles, web vulnerability and exploitation, various attacks on web applications, and a few basic topics on web encryption.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Web Security** and attain **Skill Development** through **Experiential Learning** techniques.					
Course Out Comes	**On successful completion of the course the students shall be able to:** • Define the fundamentals of web applications and validation [Knowledge] • Recognize the significance of password and authentication in web applications[Comprehension] • Explain the importance of session management in web [Comprehension] • Apply web attack techniques to find vulnerabilities in web applications [Application]					
Course Content:						

Module 1	**Introduction**	Quiz		Comprehension based Quiz on web fundamentals	**10 Sessions**
	Topics: Web Functionality, Encoding Schemes, Mapping the Application - Enumerating the Content and Functionality, Analyzing the Application Bypassing, Client-Side Controls: Transmitting Data Via the Client, Capturing User Data, Handling Client-Side Data Securely - Input Validation, Blacklist Validation - Whitelist Validation - The Defense in-Depth Approach - Attack Surface Reduction, Rules of Thumb, Classifying and Prioritizing Threats.				
Module 2	**Web Application Authentication**	Assignment		Comprehensive based assignment on Web authentication	**11 Sessions**
	Topics: Authentication Fundamentals- Two Factor and Three Factor Authentication, Web Application Authentication- Password Based, Built-in, HTTP, Single Sign-on, Custom Authentication, Validating credentials - Secured Password Based Authentication: Attacks against Password, Importance of Password Complexity - Design Flaws in Authentication Mechanisms - Implementation Flaws in Authentication Mechanisms - Securing Authentication.				
Module 3	**Session Management &Web Security Principles**	Quiz		Comprehension based Quiz on web security techniques.	**11 Sessions**
	Topics: Need for Session Management, Weaknesses in Session Token Generation, Weaknesses in Session Token Handling, Securing Session Management; Access Control: Access Control Overview, Common Vulnerabilities, Attacking Access Controls, Securing Access Control. Origin Policy, Exceptions, Browser security Principles- Cross Site Scripting and Cross Site Request Forgery, File Security Principles: Source Code Security, Forceful Browsing, Directory Traversals.				
Module 4	**Web Application Vulnerability**	Assignment		Comprehension based assignment on web vulnerabilities	**10 Sessions**

	Topics: Attacking data-stores and backend components- Injecting into Interpreted Contexts, injecting into SQL, NoSQL, XPath, LDAP, Injecting OS Commands, Manipulating File Paths, Injecting into XML Interpreters, Injecting into Back-end HTTP Requests, Injecting into Mail Services, Attacking application logic-real world logic flaws, Attacking users-Cross site scripting-varieties of XSS,XSS attacks in action, finding and exploiting XSS vulnerabilities, preventing XSS attacks, Other techniques-cookie based Attacks, HTTP Header Injection
	List of Laboratory Tasks: **Task 01:** Practical knowledge of known vulnerabilities in CGI, LAMP stacks, REST APIs cross-site scripting **Task 02:** HTTP and setting up stacks, the various types of databases Access Controls, Vulnerabilities **Task 03:** SQL injection and prevention **Task 04:** Study of web authoring tools **Task 05:** Testing web applications **Task 06:** Cross site request forgery attack lab **Task 07:** Web tracking
	Targeted Application & Tools that can be used 1. Wordpress tool can be used for building websites with possible vulnerabilities. 2. Tools such as Nmap and Nessus can be used for web attack demonstration**.**
	Project work/Assignment:
	Assignment: Group assignment to identify and write different web exploits to demonstrate vulnerabilities in web applications.
	Text Book **T1**Dafydd Stuttard, Marcus Pinto, "The Web Application Hacker's Handbook", Willey Publishing Inc.
	References **R1** B. Sullivan, V. Liu, and M. Howard, "Web Application Security", A B Guide. New York: McGraw-HillEducation, 2011**R2**Web Application Security: Exploitation and Countermeasure for Modern Web Applications, by AndrewHoffman **E book link R1:** https://presiuniv.knimbus.com/user#/home **E book link R2 :** https://presiuniv.knimbus.com/user#/home R3 **Web resources:** NPTEL / Swayam Link: Introduction to Information Security I, IIT Madras https://nptel.ac.in/courses/106106129 PU Library Link :https://puniversity.informaticsglobal.com/login
	Topics relevant to "EMPLOYABILITY SKILLS": Session Management &Web Security Principles and Web Application vulnerability for **Skill Development** through **Experiential Learning Techniques.** This is attained through the assessment component mentioned in the course handout.

Course Code: CSE3522	**Course Title: Penetration Testing and Risk Assessment** **Type of Course: Theory**	**L-T- P- C**	3	0	0	3
Version No.	1.0					
Course Pre-requisites	CSE2502					
Anti-requisites	**NIL**					
Course Description	This course explores the tools that can be used to perform information gathering. This course also covers how vulnerability can be carried out by means of tools or manual investigation, and analysis of common attacks in data, mobile applications and wireless networks.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Vulnerability Assessment and Penetration Testing** and attain **Employability** through **Problem Solving** Methodologies.					
Course Out Comes	**On successful completion of the course the students shall be able to:** • Understand the basic principles for information gathering and detecting vulnerabilities in the system. • Determine the security threats and vulnerabilities in SDN networks and web applications. • Able to use the exploits in mobile applications and wireless networks • Understand the metasploit and metrepreter are used to automate the attacks and penetration testing techniques.					
Course Content:						
Module 1	**Information Gathering, Host Discovery and Evading Techniques**	Assignment		Theory		**9 Sessions**
	Topics: Introduction - Terminologies - Categories of Penetration Testing - Phases of Penetration Test -Penetration Testing Reports - Information Gathering Techniques - Active, Passive and Sources of Information Gathering – Approaches, Host discovery - Scanning for open ports and services- Types of Port, Vulnerability Scanner Function, pros and cons - Vulnerability Assessment with NMAP - Testing, SCADA environment with NMAP					
Module 2	**Vulnerability Scanner in SDN Networks and Web application**	Quiz		Theory		**10 Sessions**
	Topics: Nessus Vulnerability Scanner - Safe check – Silent dependencies - Port Range Vulnerability Data Resources, SDN Data plane, Control Plane, Application Plane. SDN security attack vectors and SDN Harderning, Authentication Bypass with Insecure Cookie Handling - XSS Vulnerability - File inclusion vulnerability - Remote file Inclusion -Patching file Inclusions - Testing a website for SSI Injection.					
Module 3	**Mobile Application Security and wireless network Vulnerability analysis**	Quiz		Theory		**11 Sessions**
	Topics: Types of Mobile Application Key challenges in Mobile Application and Mobile application penetration testing methodology, Android and ios Vulnerabilities - OWASP mobile security risk - Exploiting WM - BlackBerry Vulnerabilities - Vulnerability Landscape for Symbian - Exploit Prevention -Handheld Exploitation, WLAN and its inherent insecurities Bypassing WLAN Authentication uncovering hidden SSIDs MAC Filters Bypassing open and shard authentication - Advanced WLAN Attacks Wireless eavesdropping using MITM session hijacking over wireless – WLAN Penetration Test Methodology.					
Module 4	**Exploits**	Quiz		Theory		**8 Sessions**

	Topics: Architecture and Environment- Leveraging Metasploit on Penetration Tests, Understanding - Metasploit Channels, Metasploit Framework and Advanced Environment configurations – Understanding the Soft Architecture, Configuration and Locking, Advanced payloads and add on modules Global datastore, module datastore, saved environment Meterpreter.
	Targeted Application & Tools that can be used: This course helps the students to understand the threats and vulnerabilities using NMAP.
	Project work/Assignment:
	Project Assignment:
	Text Book 1. Rafay Baloch, Ethical Hacking and Penetration Testing Guide, CRC Press, 2015. ISBN : 78-1-4822-3161-8. 2. Dr. Patrick Engebretson, The Basics of Hacking and Penetration Testing Ethical Hacking and Penetration Testing made easy , Syngress publications, Elsevier, 2013. ISBN :978-0-12-411644-3. 3. Mayor, K.K.Mookey, Jacopo Cervini, Fairuzan Roslan, Kevin Beaver, Metasploit Toolkit for Penetration Testing, Exploit Development and Vulnerability Research, Syngress publications, Elsevier, 2007. ISBN : 978-1-59749-074-0
	References 1. Mastering Modern Web Penetration Testing By Prakhar Prasad,October 2016 PacktPublishing. 2. SQL Injection Attacks and Defense 1st Edition, by Justin Clarke-Salt, Syngress Publication **Web resources:**https://onlinecourses.nptel.ac.in/noc19_cs68/preview - **IIT Kharagpur,** Prof. Indranil Sen Gupta
	Topics relevant to development of "EMPLOYABILITY SKILLS": Exploitation, Penetration testing techniques, for development of Employability skills through the Participative Learning Techniques. This is attained through the assessment components mentioned in course handout.

Course Code: CSE3531	**Course Title: Go Programming** **Type of Course: Theory Only**			**L- T-P- C**	3	0	0	3
Version No.		1.0					
Course Pre-requisites		Nil					
Anti-requisites		NIL					
Course Description		Go is an open source programming language created by Google. Go is expressive, concise, clean, and efficient. Its concurrency mechanisms make it easy to write programs that get the most out of multicore and networked machines. Go compiles quickly to machine code yet has the convenience of garbage collection and the power of run-time reflection. It's a fast, statically typed, compiled language that feels like a dynamically typed, interpreted language. It is gaining popularity and it is continuing to grow rapidly in industries such as Dropbox, Uber etc. This course will provide an introduction to the Go programming essentials to students of Engineering through lecture hours with demonstrations. Topics: Topics covered in this course are go program structure; data types and control statements; Composite Types – arrays, slices, strings, runes, bytes, hash maps; functions; methods; garbage collection essentials – pointers, structs, interfaces; error handling; Concurrency – go routines and channels, Packages – import and create custom packages and applications of Go					
Course Objective		The objective of the course is to familiarize the learners with the concepts of **GO Programming** and attain Employability Skills through Problem Solving techniques.					
Course Out Comes		**On successful completion of the course the students shall be able to:** CO1: Identify primitive programming constructs in GO. (**Knowledge**) CO2: Discuss composite data types with concepts of modular programming. (**Comprehension**) CO3: Implement garbage collection using pointers, structs, interfaces and modules. (**Application**) CO4: Apply concurrent programming and test routines with applications. **(Application)**					
Course Content:							
Module 1	**Introduction to Go Programming Language**	Assignment	Data Collection/Interpretation			**10 Sessions**	
	Topics: **[Knowledge]** Feature of Go language, Installing and Configuring the development environment- Go tools and playground. Structure of Go program; Basic types-numbers, boolean, strings, runes. Variables- declaration, zero values, naming, rules, conversions, constants, multiple variables. Introduction to packages, functions from other packages, println, reading input, Control Structures - if, switch, for, programming exercises using control statements.						
Module 2	**Composite types and functions**	Assignment	Data Collection/Interpretation			**9 Sessions**	
	Topics: **[Comprehension]** Composite types - arrays, slices, slices with overlapping storage, Structs. Functions-declaring, parameters, returning multiple values, variadic functions; Programming exercises						
Module 3	**Pointers, Structs, Interfaces and modules**	Quiz	Case studies / Case let			**9 Sessions**	
	Topics: **[Application]** Pointers: *and & operator, types, pointers with functions, garbage collector – history, Methods and Interfaces, Modules,packages – importing and creating custom packages; Programming exercises.						
Module 4	**Concurrency and Applications**	Quiz	Case studies / Case let			**7 Sessions**	
	Topics: **[Application]** Concurrency using Go routines, multiple go routines, channels – channel operations, Testing- writing test, Go test						

	command, Core Packages for – strings, containers and lists, Writing Web Applications, Basic Statistical Computations, histogram plotting, encryption and decryption.
	Targeted Application & Tools that can be used: 1. https://go.dev/play/ 11. https://go.dev/doc/install
	Project work/Assignment:
	Text Book **T1** 1. John Badner,"Learning Go: An Idiomatic Approach to Real World Go Programming", Oreilly, California,2021.
	References **R1.** 1. Alan A.A. Donovan and Brian W. Kernighan, "The Go Programming Language", Pearson Education, India,2016. **R2**. Tsoukalos M. Mastering Go: Create Golang production applications using network libraries, concurrency, machine learning, and advanced data structures. Packt Publishing Ltd; 2019 Aug 29. **Web resources:** https://www.golangprograms.com/go-language.html EBSCO database of Presidency University:https://puniversity.informaticsglobal.com/login W3**.** GO document: https://go.dev/doc/ **Online tool for program execution:** • GO Play Ground - https://go.dev/play/ • Download and install: https://go.dev/doc/install
	Topics relevant to development of "Employability": Go Programming basics for developing **Employability Skills** through Problem Solving methodologies. This is attained through assessment component mentioned in course handout.

Course Code: CSE3532	**Course Title: Advanced DBMS** **Type of Course: Lab Integrated**	**L-T-P-C**	2	0	2	3
Version No.	**1.0**					
Course Pre-requisites	**CSE2260**					
Anti-requisites	**NIL**					
Course Description	**The purpose of this course is to make the students revisit RDBMS transactions first. Then introduce them with Distributed, Parallel, and NoSQL database concepts. They include the main characteristics, advantages, and disadvantages of each one of them. Importance and differences among them are noted. Need to transit from RBMS to NoSQL is discussed. The striking features of distributed, parallel and NoSQL are considered and studied.** **The associated laboratory provides a chance to have hands-on concepts learned during this course.**					
Course Objective	**This course is designed to improve the learners' <u>EMPLOYABILITY SKILLS</u> by learning the working on Database using MySQL.**					
Course Outcomes	**On successful completion of this course the students shall be able to:** (1) Recall the transactions in RDMS **(2)** Explain advanced features of distributed, parallel, and NoSQL databases. **(3)** Illustrate the features in Distributed database **(4)** Employ Parallel database concepts in real life applications.					
Course Content:						

Module 1	**Transactions in RDBMS**	Quiz	Comprehension based Quizzes and assignments.	**06Classes**

Topics:
RDBMS -Transaction control state diagram, ACID properties of transaction, Schedules in transactions - Serial, Non-Serial and Serializable, Serializability-Conflict and View, Conflict Serializability check by Precedency Graph, Concurrency Control – Lock Based and Time Stamp Based.

Module 2	**NoSQL Databases**	Programming and Mini Project	Laboratory experiments and Mini Projects on NoSQL Topics using MongoDB/ Casandra.	**06Classes**

Topics:
NoSQL Introduction – Scale Out, Commodity Hardware, Brief History, Features – Non-Relational, Schema Free, Simple API, and Distributed. NoSQL Architectures/Data Models - Document, Columnar, Key-Value, and Graph. Transaction in NoSQL- BASE for reliable database transactions, Achieving Horizontal Scalability with Database Sharding, CAP theorem.
Case Study: MongoDB/Casandra/ AWS/ HBase

Module 3	**Distributed Databases**	Assignment	Assignment on main topics of Distributed Databases	**06Classes**

Topics:
Loosely Coupled, Characteristics of Distributed Databases, Local and Global view of applications, Distributed Processing, Types – Homogeneous and Heterogeneous, Distributed Data Storage – Replication and Fragmentation, Fragmentation – Horizontal and Vertical Type, Difference between Centralized and Distributed Databases.

Module 4	Parallel Databases	Assignment	Assignment on main topics of Parallel Databases	**06 Classes**

Topics:
Tightly Coupled, Features of parallel databases, Shared Memory, Shared Disk, Shared Nothing Systems. Advantages of each of these schemes, Advantages and Disadvantages of Parallel Databases, Differences between Parallel and Distributed Databases.

Install MONGODB
https://www.javatpoint.com/mongodb-create-database
Create any one of the following databases.
Employee, Student, University, Banking, or Online Shopping
Drop database

Create Collection: In MongoDB db.createCollection(name,option) is used to create collection.
Drop Collection

List of Laboratory Tasks:(7 X 2= 14 Sessions)

Level 1: Perform CRUD operations (Insert, Update, Delete and Query Documents) on 'Student' Database.
Level 2: Do MongoDB text search on 'Employee' Database.

Experiment No. 2: Try experiments on MongoDB Operators
Level 1: Perform queries involving MongoDB Query and Projection Operators using 'Student' Database.
Level 2: Do queries involving MongoDB update operator on 'Employee' Database.

Experiment No. 3:Explore different query modifiers.
Level 1: Perform different query modifiers on 'Student' Database.
Level 2: Try various query modifiers on 'Employee' Database.

Experiment No. 4:Explore Aggregation commands.
Level 1: Implement different aggregation commands on 'Student' Database.
Level2: Perform various aggregation commands on 'Employee' Database.

Experiment No. 5:Explore Authentication commands.
Level 1: Try authentication commands on 'Student' Database.
Level 2: NA

Experiment No. 6:Explore Replication Commands
Level 1: Try all replication commands on 'Student' Database.
Level2: Implement replication commands on 'Employee' Database.

Experiment No.7:Try Sharding Commands.
Level1: Explore Sharding Commands on 'Student' Database.
Level 2: Implement Sharding Commands on 'Employee' Database.

Targeted Application & Tools that can be used:
MongoDB is to be installed and used.

Project work/Assignment:

Each batch of students (self-selected batch mates) will identify projects, such as, Library, Banking, and Reservation etc.,and do it. Concepts of NoSQL, like, CRUD operations, supporting ad hoc queries, indexing flexibility, assisting replication, creating capped collections, and Retrieving data from multiple documents.
Sample Mini Projects:
1. Content Management System
Clubbing the content assets like text and HTML into a single database helps provide a better user experience. MongoDB has an excellent toolset not only for storing and indexing but also for controlling the structure of a content management system. You can easily design a web-based CMS by using the model proposed by "Metadata and Asset Management" in MongoDB. Additionally, you can use "Storing Comments" to model user comments on blog posts.
2. Gaming Project
Data is an essential part of making video games work. Some typical examples of gaming data include player profiles, matchmaking, telemetry, and leaderboards.
The common thread between all games is that they all have a specific goal. And you have to achieve multiple objectives or pay your way out to reach the end goal. This may involve steps like watering your plants, growing vegetables, serving food in a restaurant, and so on.

Textbook(s):
1. **Sadalage, P. & Fowler, NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence, 1st Edition, 2019(Wiley Publications).**
2. **Stefano Ceri, Giuseppe Pelagatti , Distributed Databases: Principles and Systems,, 2017(McGraw Hill Education).**

References 1. **Elmasri R and Navathe S B, "Fundamentals of Database System",7th Edition, 2017(Pearson Publication).** 2. **Pivert. *NoSQL Data Models: Trends and Challenges*, 1st edition(Wiley).**
Topics related to development of "FOUNDATION":Transaction, CRUD Operations, Replication, and Sharding Topics related to development of "EMPLOYABILITY": Project implementations in software, batch wise presentations Topics related to development of "HUMAN VALUES AND PROFESSIONAL ETHICS": Team Dynamics during Mini Project Development.

Course Code: CSE3533	**Course Title:** Programming in C# and .NET **Type of Course:** Theory Integrated Lab	**L-T-P-C**	1	0	4	3
Version No.	**1.0**					
Course Pre-requisites	**CS2258**					
Anti-requisites	**NIL**					
Course Description	.NET is a software framework which is designed and developed by Microsoft. It is used to develop Form-based applications, Web-based applications, and Web services. There is a variety of programming languages available on the .Net platform, VB.Net, C# etc. C# is one of the most popular languages of .NET framework. It is used to build applications for Windows, phone, web etc. It provides a lot of functionalities and also supports industry standards. This course intends to provide the basic concepts of .NET framework and various components of the .NET framework architecture. This course also provides the features of C# programming language to design and implement console, desktop-based applications and web based applications.					
Course Objective	**This course is designed to improve the learners' <u>EMPLOYABILITY SKILLS</u> by learning the working on Database using MySQL.**					
Course Outcomes	On successful completion of this course the students shall be able to: 1. Understand the fundamental concepts and benefits of .NET framework and its components.[Comprehension] 2. Illustrate the Object-Oriented paradigm using C# Language.[Knowledge] 3. Develop different types of applications by applying the C# programming concepts and database connectivity.[Application] 4. Demonstrate the use of event handling mechanism[Application]					
Course Content:						
Module 1	Concepts of .NET Technology, Implementation of .NET, Components of .NET framework	Quiz	Demonstration of Assembly, Introduction to IDE	**Th + Pr 4+ 1(6)**		
Topics: Understand the motivation behind the .NET platform, Common Language Infrastructure (CLI). Know the role of the Common Type System (CTS), the Common Language Specification (CLS) and the Common Language Runtime (CLR), Understand the assembly, metadata, namespace, type distinction, Contrast single-file and multi-file assemblies, Know the role of the Common Intermediate Language (CIL),Introduction to .NET Core						
Module 2	C# Programming constructs, OOPS concepts	Lab Based Assignments,	Hands on Session for the Concepts, Creating a Console application	**3+6(15)**		
Topics: Program structures, Types and Variables, Expressions, Statements, Structs, Class, Objects Name Spaces - Constructor and Destructors, Function Overloading & Inheritance, Operator Overloading, Array, Interfaces, Modifiers -, Working with Console input & Output, Property and Indexers , Enum ,Interfaces						
Module 3	Event handling, Data base Connectivity	Lab based Assignments	Practice Exercises using Visual Studio.NET	**3+3(9)**		
Topics: Delegates and events, Exception handling, Advantage of ADO.NET, ADO.NET Architecture, .NET Providers, Connection, Command, Data Set, Data Reader. Working with Connection Oriented and Connection less approach, Programming Window Forms Applications: The notifies - subscribers paradigm for handling						

events. .NET framework for handling GUI events				
Module 4	P.NET	o based Assignments	ctice Exercises using Visual dio.NET	**2(6)**
Topics: Introduction to Web Forms, Basic working of WebForms, Introduction to Web Forms, Connectivity with the database , validation Controls				

List of Practical Tasks:

Experiment 1:[Module 1]

Level 1: Demonstrate the .NET framework and Visual studio IDE for writing C# code.

Level 2: Demonstrate the .NET framework and inside of Assembly using ildasm tool.

Experiment 2: [Module 2]

Level 1: University wants to gift for those date of birth falls on February 29th . Create a C# program that will accept the employee's birth year. Check the leap year and issue them with surprise gift.

Level 2: A developer wants to check the given input is in Fibonacci series or not.

Experiment 3: [Module 2]

Level 1 : A teacher is asked to create mark list of her class students. The class consists of 10 students and they have 5 different subjects. Store the student's name and five subject marks also. Calculate the total of all subject marks and display them.

Level 2: A class teacher is storing the students 'name and Roll number. Write a program to help to sort out the roll number using different sorting techniques.

Experiment 4: [Module 2]

Level 1: Design a class to represent a bank account. Include the following members: Data Members: - Name of the depositor, Account Number, Type of Account, Balance amount in the account and methods : To assign initial values, To deposit an amount, To withdraw an amount after checking balance, To display name and the balance. Write a C# program to demonstrate the working of the various class members.

Level 2: Define a class 'Person' with data members name and age. Also include following: Default Constructor and parameterized constructor, Input method which takes values from user and assigns to data members, Output method to display all data . Create 5 objects of 'Person' class using array of objects and call all the methods of a class.

Experiment 5: [Module 2]

Level 1: Write a C# program to show single and multilevel inheritance.

Level 2: Create a class 'Emp' by extending Person class with additional data member empno, position with following features:

a. Default constructor
b. Parameterized constructor
c. Input method which takes values from user and assigns to data members and calls input method of Person
d. Output method to display all data and calls output method of Person

Define a class Manager by extending Emp with data member bonus. Provide necessary constructors and override input and output methods. Create objects of manager in main.

Experiment 6: [Module 2]

Level 1: Calculate the area of different shapes using method overloading.

Level 2: Class teacher created different groups in a class and store the data in that. In order to make common announcements and activities, teacher merged all data into a single group. Write a code to merge two groups into one.

Experiment 7: [Module 2]

Level 1: Class Teacher stores students marks in an array. Teacher is searching for highest and lowest marks of the class and number of students scored those marks. Write a program to help teacher to do the same.

Level 2: Create an application for currency converter.

Experiment 8: [Module 3]

Level 1: EC is updating their database of new voters. If the user's age is less than 18, application should raise the exception.

Level 2: Develop a desktop based application for displaying employees salary and leave balance.

Experiment 9: [Module 3]

Level 1: Create a login screen and prompts for the user name and Password. If the user exists in the database , show welcome message to the user when the button is clicked.

Level 2: Company wants to create a calculator application using C#.

Experiment 10: [Module 3]

Level 1: University stores the CSE students' data in the database and display the student details whenever required. Administrator allows to insert, update and modify the data. Implement this.

Level 2: University decides to conduct online quiz for CSE students. Create a windows-based application to implement this.

Experiment 11: [Module 4]

Level 1: University is organizing a cultural festival and organizing teams wants to collect registration for various events with the help of web page. Design a registration form for collecting the participant details.

Level 2: University is decided to display all information about the various departments in their website. Design a web site to show the above mentioned.

Experiment 12: [Module 4]

Level 1: XYZ corporation wants to review their product. So company is creating a feedback form and validating the data with the help of validation controls. Design a feedback form with validation controls.

Level 2: XYZ corporation wants to review their product. So company is creating a feedback form and validating the data with the help of validation controls. Design a feedback form with validation controls and display the comparison chart of various months.

Targeted Application & Tools that can be used:
Microsoft Visual studio.NET 2022, Visual Studio Code.

Project work/Assignment:
Problem Solving: Design of Algorithms and implementation of programs.
Programming: Implementation of given scenario using .NET.

Textbook(s):

1. **Herbert Schildt , "C# 4.0 The Complete Reference", Fourth Edition, TMH**
2. **2. Matthew Macdonald,"ASP.NET: The Complete Reference", McGraw Hill Education**

References:

1. Joseph Albahari and Ben Albhari, "C# 3.0/4.0 in NUTSHELL", O'REILLY.
2. Andrew Troelsen, "C# and the .NET Platform" 1st edition Apress
3. Matthew Macdonald,"Beginning ASP.NET 4.5 in C#", Wiley India

Online References

1.C# Tutorial (C Sharp) (w3schools.com)
2. https://docs.microsoft.com/en-us/dotnet/csharp/tour-of-csharp/tutorials/
3.https://docs.microsoft.com/en-us/aspnet/tutorials

Topics relevant to development of "Employability": Web Application developing Employability Skills through Experiential Learning techniques. This is attained through assessment component mentioned in course handout.

Course Code: FIN1002	**Course Title: Essentials of Finance Type of Course: HSMC**	**L-T-P-C**	3	0	0	3
Version No.	1.0					
Course Pre-requisites	This course is designed to be accessible to all students, regardless of their prior financial knowledge.					
Anti-requisites						
Course Description	This course is designed to equip students with a **foundational understanding of key financial concepts and principles**. It will enable them to comprehend the **core functions of finance**, delve into the intricacies of **financial management within organizations**, and gain insights into the **fundamental aspects of taxation**. The course aims to develop students' abilities to **interpret financial statements, evaluate investment opportunities, understand capital structure decisions**, and navigate the **basics of tax implications**.					
Course Objective	Upon successful completion of this course, students will be able to: • **Understand the basic forms of business organization and their financial implications.** • **Understand the fundamental principles and concepts** that influence financial decision-making in various contexts. • **Analyse and interpret financial statements** to assess the financial health and performance of an organization. • **Identify income under various heads of income as per Income Tax Act, 1961 and determine the tax liability.**					
Course Outcomes	**List the course outcomes** **On successful completion of this course the students shall be able to:** 1. **Understand the basic concepts of finance and financial markets and organizations.** 2. **Apply and interpret financial information for business decision making.** 3. **Identify various heads of income and deduction under Income Tax Act, 1961.**					
Course Content:						
Module 1	**Introduction to Finance**	Assignment/ Quiz	Numerical solving Task	**10 Sessions**		
Definition and Scope of Finance, Areas of Finance: Corporate Finance, Investments, Financial Institutions, International Finance; Types of Financial Markets: Money Markets vs. Capital Markets, Primary vs. Secondary Markets; Forms of Business Organization and Financial Goals: Shareholder Wealth Maximization vs. Profit Maximization; Understanding Financial Statements: Balance Sheet and Income Statement- Simple Numerical.						
Module 2	**Financial Management**	Assignment/ Quiz	Numerical solving Task	**18 Sessions**		
Capital Budgeting Decisions: Payback Period, Net Present Value (NPV), Profitability Index (PI), Internal Rate of Return (IRR); Leverage- Basic Numerical; Capital						

Structure Decisions: Optimal Capital Structure, Trade-off Theory of Capital Structure; Cost of Capital: Equity, Debt, WACC; Dividend Policy: Factors influencing Dividend Policy.				
Module 3	**Taxation**	Assignment/ Quiz	Numerical solving Task	**17 Sessions**
Principles of a Good Tax System: Equity, Certainty, Convenience, Economy; Direct vs. Indirect Taxes; Residential Status of an Individual- Basic Problems; Heads of Income; Salary, House Property- Basic Numerical; Deductions under Chapter VI-A; Computation of Taxable Income and Tax Liability; E-Filing procedure.				
Targeted Application & Tools that can be used: **Textbooks, PPT, Spreadsheet Software (e.g., Microsoft Excel), Official Website of Income Tax Department.**				
Project Work/ Assignment:				
1. Presentation: There will be a group presentation, where the students will be given a topic. They will have to explain/demonstrate the working and discuss the applications for the same. **2. Case Study**: - At the end of the course students will be given a 'real-world' cases like business models of successful companies or tax evasion by reputed companies on which they have to come up with detailed analysis and assessment.				
Text Book(s): 1. **Dr. Vinod K. Singhania & Dr. Monica Singhania.** (Latest Assessment Year Edition). *Students' Guide to Income Tax including GST*. Taxmann Publications. **2. Pandey, I. M.** (2025). *Financial Management*. Vikas Publishing House.				
Reference Book (s): 1. **Bhole, L.M., & Mahakud, J.** (Current Edition). *Financial Institutions and Markets: Structure, Growth and Innovations*. McGraw Hill Education India. 2. **Mehrotra, H.C., & Goyal, S.P.** (Latest Assessment Year Edition). *Income Tax Law & Practice*. Sahitya Bhawan Publications. 3. **Gordon, E., & Natarajan, K.** (Current Edition). *Financial Markets and Services*. Himalaya Publishing House. **Online Resources (e-books, notes, ppts, video lectures etc.):** 1. https://presidencyuniversity.linways.com 2. https://onlinecourses.nptel.ac.in/noc24_ec01/preview 3. https://www.incometax.gov.in/iec/foportal/				
Topics relevant to "SKILL DEVELOPMENT": This course is designed to provide practical financial skills through participative learning techniques. Students will engage in performing suitable calculations to determine financial parameters (e.g., time value of money, investment returns, tax liabilities) and analysing financial statements to assess organizational performance and make informed decisions.				

Course Code: CSE3525	**Course Title: Blockchain Technology** **Type of Course: Theory**	**L-T-P-C**	2	0	2	3

Version No.		1.0
Course Pre-requisites	•	CBC2000
Anti-requisites		NIL
Course Description		This course provides a comprehensive introduction to the fundamental concepts and applications of blockchain technology. Students will explore the principles of decentralization, understand the mechanics of Bitcoin, delve into the world of smart contracts and alternative cryptocurrencies, and gain practical experience in developing and deploying smart contracts using industry-standard tools. The course emphasizes both the theoretical underpinnings and the practical implementation of blockchain solutions.
Course Object		The objective of the course is to familiarize the learners with the concepts of Blockchain Technology attain **Skill Development** through **Experiential Learning** techniques.
Course Out Comes		On successful completion of the course the students shall be able to: **CO1: Understand** blockchain and decentralization principles. (Understand) **CO2: Analyze** Bitcoin's operation and transactions. (Analyze) **CO3: Apply** smart contract concepts and explore altcoins. (Apply) **CO4: Utilize** Truffle for smart contract development and deployment. (Apply)
Course Content:		

Module 1	**Fundamentals of Blockchain and Decentralization**	Assignment			**14 Sessions**
	Topics: Introduction to Blockchain: the growth of blockchain technology, the history of blockchain and Bitcoin, distributed systems, blockchain, consensus. Decentralization: decentralization using blockchain, methods of decentralization, routes to decentralization, blockchain and full ecosystem decentralization, pertinent terminology, platforms for decentralization, innovative trends				
Module 2	**Bitcoin: The First Cryptocurrency**	Assignment			**15 Sessions**
	Topics: Introduction to Bitcoin: bitcoin — an overview, cryptographic keys, transactions, blockchain, mining. The Bitcoin Network and Payments: the bitcoin network, wallets,				

	bitcoin payments, innovation in bitcoin, advanced protocols, bitcoin investment and buying and selling Bitcoin. Bitcoin Clients and APIs: bitcoin client installation, experimenting further with bitcoin-cli, bitcoin programming				
Module 3	**Beyond Bitcoin: Altcoins and Smart Contracts**	Assignment			**16 Sessions**
	Topics: Consensus Algorithms: introducing the consensus problem, analysis and design, classification, algorithms, choosing an algorithm. Alternative Coins: introducing altcoins, theoretical foundations, difficulty adjustment and retargeting algorithms, bitcoin limitations, extended protocols on top of bitcoin, development of altcoins, Initial Coin Offerings (ICOs). Smart Contracts: history, definition, ricardian contracts, smart contract templates, oracles, deploying smart contracts, the DAO				
Module 4	**Ethereum and the Decntralized Web (Web3)**	Assignment			**15 Sessions**
	Topics: Ethereum 101: ethereum – an overview, the ethereum network, components of the ethereum ecosystem, Ethereum Virtual Machine (EVM), smart contracts, ethereum development environment . Further Ethereum: blocks and blockchain, wallets and client software, nodes and miners, APIs, tools, and DApps, supporting protocols, programming languages. Introducing Web3: contract deployment, exploring Web3 with Geth				
	Project work/Assignment:				
1.	**2. Assignment 1 on (Module 1 and Module 2)** **3. Assignment 2 on (Module 3 and Module 4)**				
	Text Book 1. Banafa, A. (2024). *Blockchain technology and applications*. River Publishers. 2. Ramachandran, M. (2025). *Blockchain engineering: Secure, sustainable frameworks for healthcare applications*. Springer. 3. Tanwar, S. (2022). *Blockchain technology: From theory to practice*. Springer. 4. Vyas, S., Shukla, V. K., Gupta, S., & Prasad, A. (Eds.). (2022). *Blockchain technology: Exploring opportunities, challenges, and applications*. CRC Press.				
	References 1. Chuen, D. L. K. (Ed.). (2024). *Handbook of digital currency: Bitcoin, innovation, financial instruments and big data* (2nd ed.). Academic Press. 2. Idrees, S. M. & Nowostawski, M. (Eds.). (2023). *Transformations through blockchain technology: The new digital revolution*. Springer. 3. Jena, A. K., Panda, S. K., & Swain, S. K. (Eds.). (2022). *Blockchain technology: Applications and challenges* (Vol. 203). Springer.				

	4. Maleh, Y., Zhang, J., & Hansali, A. (2024). *Advances in emerging financial technology and digital money*. Routledge. 5. Rahman, H. (Ed.). (2025). *Blockchain technology applications in knowledge management*. IGI Global. **Web Resources** 1. Blockgeeks. Retrieved from https://www.google.com/search?q=blockgeeks.com 2. Bitcoin.org. Retrieved from https://bitcoin.org/ 3. CoinDesk. Retrieved from https://www.coindesk.com/ 4. Ethereum.org. Retrieved from https://ethereum.org/ 5. Investopedia. Retrieved from https://www.investopedia.com/ 6. Medium. Retrieved from https://medium.com/ 7. Solidity Documentation. Retrieved from https://docs.soliditylang.org/ 8. Truffle Suite Documentation. Retrieved from https://trufflesuite.com/docs 9. Web3.js Documentation. Retrieved from https://web3js.readthedocs.io/ 10. GitHub. Retrieved from https://github.com/ **YouTube Channels:** 1. Andreas Antonopoulos. Retrieved from https://www.youtube.com/@aantonop 2. Chainlink. Retrieved from https://www.youtube.com/@chainlink 3. Coin Bureau. Retrieved from https://www.youtube.com/@CoinBureau 4. Eat The Blocks. Retrieved from https://www.youtube.com/@EatTheBlocks 5. freeCodeCamp.org. Retrieved from https://www.youtube.com/@freecodecamp 6. MetaMask. Retrieved from https://www.youtube.com/@MetaMask 7. Patrick Collins. Retrieved from https://www.youtube.com/@patrickdcollins 8. Simply Explained. Retrieved from https://www.youtube.com/@SimplyExplained 9. The Defiant. Retrieved from https://www.youtube.com/@TheDefiant 10. Whiteboard Crypto. Retrieved from https://www.youtube.com/@WhiteboardCrypto
	Experiment 1: Integrated Development Environments (IDEs) for Smart Contracts • **Level 1:** Explore the features and interface of Remix IDE. Deploy a simple "Hello World" smart contract on the in-browser JavaScript VM. Observe the transaction details and contract interaction options. • **Level 2:** Install and configure MetaMask browser extension. Connect MetaMask to the Remix IDE. Deploy the same "Hello World" contract to the Ganache private network via MetaMask. Examine the transaction process in both Remix and MetaMask. **Experiment 2: MetaMask in a Private Network** • **Level 1:** Set up a local Ganache private network. Add a custom network in MetaMask, configuring the RPC URL and Chain ID to connect to your Ganache instance. Create a new account in MetaMask and observe its balance. • **Level 2:** Deploy a simple token contract (e.g., ERC-20 minimal) using Remix IDE and MetaMask on your private Ganache network. Transfer some tokens between the accounts you created in MetaMask and observe the balance changes. **Experiment 3: Smart Contract with Solidity - Basic Data Types and Structures** • **Level 1:** Write a Solidity smart contract that declares and initializes variables of different basic data types (uint, string, bool, address). Implement functions to read and modify these variables. Deploy and interact with the contract in Remix IDE.

	• **Level 2:** Create a Solidity smart contract that utilizes structs and arrays. Implement functions to add, retrieve, and update elements within these data structures. Deploy and test the contract with various inputs in Remix IDE. **Experiment 4: Smart Contract with Solidity - Control Flow and Functions** • **Level 1:** Write a Solidity smart contract that uses if-else statements and for loops within its functions. Implement a function that performs a simple calculation based on input parameters. Deploy and test the different control flow paths in Remix. • **Level 2:** Design and implement a Solidity smart contract with multiple functions, including internal and private functions. Demonstrate how these functions can be called and how visibility modifiers affect their accessibility. **Experiment 5: Contract Deployment** • **Level 1:** Deploy a pre-written simple smart contract (provided by the instructor) using MetaMask connected to the Ganache network. Observe the deployment transaction details (gas used, transaction hash, contract address). • **Level 2:** Explore different deployment parameters in Remix IDE (e.g., setting gas limit and gas price). Deploy the same contract multiple times with varying gas settings and analyze the impact on deployment cost and confirmation time in Ganache. **Experiment 6: MetaMask and Remix IDE Interaction** • **Level 1:** Deploy a simple counter smart contract using Remix IDE on the Ganache network via MetaMask. Use the Remix interface to call the contract's functions (e.g., increment, decrement, get count) and observe the state changes reflected in both Remix and MetaMask (balance changes for transactions). • **Level 2:** Deploy a more complex smart contract (e.g., a simple voting contract) using Remix and MetaMask. Interact with the contract through MetaMask's custom interaction interface (sending transactions to specific functions with appropriate arguments). **Experiment 7: Use of Geth - Installation and Account Management** • **Level 1:** Install the Geth Ethereum client on your local machine. Use Geth commands to create new Ethereum accounts and list the available accounts. Observe the keystore directory where private keys are stored. • **Level 2:** Use Geth commands to export and import Ethereum account private keys. Understand the security implications of managing private keys. Connect the Geth console to a running private network (e.g., Ganache or a custom Geth network). **Experiment 8: Genesis Block Creation in Geth** • **Level 1:** Understand the structure of a Genesis Block JSON file. Modify a sample Genesis Block configuration (e.g., changing the initial coin distribution). Initialize a new Geth data directory using this modified Genesis Block. • **Level 2:** Create a custom Genesis Block for a private Ethereum network with specific pre-allocated accounts, custom gas limit, and difficulty. Start a Geth node using this custom Genesis Block and connect to it using the Geth console. **Experiment 9: Interacting with a Private Geth Network** • **Level 1:** Start a Geth node using a previously initialized data directory. Use the Geth console to check the node's peer count and block number. Create a transaction to send Ether between two accounts within your private network using Geth commands. • **Level 2:** Deploy a simple smart contract to your private Geth network using the Geth console and web3.js (or similar library). Interact with the deployed contract's functions using the Geth console. **Experiment 10: Exploring Ethereum Transaction Structure**

	• **Level 1:** Send a transaction (Ether transfer) using MetaMask on the Ropsten test network. Examine the transaction details on a block explorer (e.g., Etherscan for Ropsten) and identify key fields like to, from, value, gas limit, gas price, and nonce. • **Level 2:** Construct and sign a raw Ethereum transaction using web3.js (or similar library) without relying on MetaMask. Broadcast this signed transaction to a test network (e.g., Ropsten) and analyze its details on a block explorer. **Experiment 11: Working with Smart Contract Events** • **Level 1:** Write a Solidity smart contract that emits events when certain actions occur (e.g., a value is updated). Deploy the contract in Remix and trigger the actions. Observe the emitted events in the Remix console. • **Level 2:** Modify the previous contract to include indexed event parameters. Write a simple web3.js script (or use the Remix event listener) to filter and listen for specific events based on the indexed parameters. **Experiment 12: Understanding Smart Contract Security - Common Vulnerabilities (Part 1)** • **Level 1:** Study a simple smart contract with a known vulnerability (e.g., integer overflow/underflow - using an older Solidity version). Deploy the contract in Remix and attempt to exploit the vulnerability through function calls. • **Level 2:** Research and demonstrate another common smart contract vulnerability (e.g., reentrancy - using a simplified example). Write a vulnerable contract and a separate "attacker" contract to exploit it on a local test network. **Experiment 13: Understanding Smart Contract Security - Common Vulnerabilities (Part 2)** • **Level 1:** Analyze a smart contract with access control implemented using onlyOwner modifier. Deploy the contract and attempt to call restricted functions from a non-owner account. • **Level 2:** Explore the concept of gas limits and denial-of-service (DoS) attacks in smart contracts. Write a contract that could be susceptible to a simple gas-based DoS attack and demonstrate how it can be exploited. **Experiment 14: Interacting with Standard ERC-20 Tokens** • **Level 1:** Deploy a standard ERC-20 token contract (using OpenZeppelin library in Remix or a pre-written contract) on a local test network. Interact with the token contract's functions (e.g., totalSupply, balanceOf, transfer) using Remix. • **Level 2:** Write a simple Solidity smart contract that interacts with the deployed ERC-20 token contract. Implement a function in your contract that allows users to spend a certain amount of the deployed tokens (requiring approval). **Experiment 15: Introduction to Truffle Framework** • **Level 1:** Install Truffle and Node.js. Create a new Truffle project. Understand the basic directory structure of a Truffle project (contracts, migrations, test). Compile a simple Solidity contract using Truffle commands. • **Level 2:** Write a simple test case for your smart contract using Truffle's testing framework (Chai and Mocha). Run the tests to ensure the contract functions as expected. Deploy your compiled contract to a local Ganache network using Truffle migrations.

Course Code: CSE3528	**Course Title: Blockchain Development and Programming** **Type of Course:** **Theory & Lab Integrated**	**L-T-P-C**	2	0	2	3

Version No.		1.0
Course Pre-requisites	•	CBC2000
Anti-requisites		NIL
Course Description		This course provides a comprehensive introduction to the fundamental concepts and applications of blockchain technology. Students will explore the principles of decentralization, understand the mechanics of Bitcoin, delve into the world of smart contracts and alternative cryptocurrencies, and gain practical experience in developing and deploying smart contracts using industry-standard tools. The course emphasizes both the theoretical underpinnings and the practical implementation of blockchain solutions.
Course Object		The objective of the course is to familiarize the learners with the concepts of Blockchain Development and Programming and attain **Skill Development** through **Experiential Learning** techniques.
Course Out Comes		On successful completion of the course the students shall be able to: **CO1:** Understand blockchain and decentralization principles. (Understand) **CO2:** Analyze Bitcoin's operation and transactions. (Apply) **CO3**: Apply smart contract concepts and explore altcoins. (Apply) **CO4:** Utilize Truffle for smart contract development and deployment. (Apply)
Course Content:		

Module 1	**Fundamentals of Blockchain and Decentralization**	Assignment			**6L+6P Sessions**
	Introduction to Blockchain: the growth of blockchain technology, the history of blockchain and Bitcoin, distributed systems, blockchain, consensus. Decentralization: decentralization using blockchain, methods of decentralization, routes to decentralization, blockchain and full ecosystem decentralization, pertinent terminology, platforms for decentralization, innovative trends.				
Module 2	**Bitcoin: The First Cryptocurrency**	Assignment			**7L+8P Sessions**
	Topics:				

	Introduction to Bitcoin: bitcoin — an overview, cryptographic keys, transactions, blockchain, mining. The Bitcoin Network and Payments: the bitcoin network, wallets, bitcoin payments, innovation in bitcoin, advanced protocols, bitcoin investment and buying and selling Bitcoin. Bitcoin Clients and APIs: bitcoin client installation, experimenting further with bitcoin-cli, bitcoin programming				
Module 3	**Beyond Bitcoin: Altcoins and Smart Contracts**	Assignment			**10L+8PSessions**
	Topics: Consensus Algorithms: introducing the consensus problem, analysis and design, classification, algorithms, choosing an algorithm. Alternative Coins: introducing altcoins, theoretical foundations, difficulty adjustment and retargeting algorithms, bitcoin limitations, extended protocols on top of bitcoin, development of altcoins, Initial Coin Offerings (ICOs). Smart Contracts: history, definition, ricardian contracts, smart contract templates, oracles, deploying smart contracts, the DAO. .				
Module 4	**Ethereum and the Decentralized Web (Web3)**	Assignment			**7L+8P Sessions**
	Topics: Ethereum 101: ethereum – an overview, the ethereum network, components of the ethereum ecosystem, Ethereum Virtual Machine (EVM), smart contracts, ethereum development environment. Further Ethereum: blocks and blockchain, wallets and client software, nodes and miners, APIs, tools, and DApps, supporting protocols, programming languages. Introducing Web3: contract deployment, exploring Web3 with Geth				
	Project work/Assignment:				
9.	**10. Assignment 1 on (Module 1 and Module 2)** 11. **Assignment 2 on (Module 3 and Module 4)**				
	Text Books 1. Banafa, A. (2024). *Blockchain technology and applications*. River Publishers. 2. Ramachandran, M. (2025). *Blockchain engineering: Secure, sustainable frameworks for healthcare applications*. Springer. 3. Tanwar, S. (2022). *Blockchain technology: From theory to practice*. Springer. 4. Vyas, S., Shukla, V. K., Gupta, S., & Prasad, A. (Eds.). (2022). *Blockchain technology: Exploring opportunities, challenges, and applications*. CRC Press.				

References

1. Chuen, D. L. K. (Ed.). (2024). *Handbook of digital currency: Bitcoin, innovation, financial instruments and big data* (2nd ed.). Academic Press.
2. Idrees, S. M. & Nowostawski, M. (Eds.). (2023). *Transformations through blockchain technology: The new digital revolution*. Springer.
3. Jena, A. K., Panda, S. K., & Swain, S. K. (Eds.). (2022). *Blockchain technology: Applications and challenges* (Vol. 203). Springer.
4. Maleh, Y., Zhang, J., & Hansali, A. (2024). *Advances in emerging financial technology and digital money*. Routledge.
5. Rahman, H. (Ed.). (2025). *Blockchain technology applications in knowledge management*. IGI Global

Web Resources

W1. Blockgeeks. Retrieved from https://www.google.com/search?q=blockgeeks.com
W2. Bitcoin.org. Retrieved from https://bitcoin.org/
W3. CoinDesk. Retrieved from https://www.coindesk.com/
W4. Ethereum.org. Retrieved from https://ethereum.org/
W5. Investopedia. Retrieved from https://www.investopedia.com/
W6. Medium. Retrieved from https://medium.com/
W7. Solidity Documentation. Retrieved from https://docs.soliditylang.org/
W8. Truffle Suite Documentation. Retrieved from https://trufflesuite.com/docs
W9. Web3.js Documentation. Retrieved from https://web3js.readthedocs.io/
W10. GitHub. Retrieved from https://github.com/

Experiment 1: Integrated Development Environments (IDEs) for Smart Contracts

- **Level 1:** Explore the features and interface of Remix IDE. Deploy a simple "Hello World" smart contract on the in-browser JavaScript VM. Observe the transaction details and contract interaction options.
- **Level 2:** Install and configure MetaMask browser extension. Connect MetaMask to the Remix IDE. Deploy the same "Hello World" contract to the Ganache private network via MetaMask. Examine the transaction process in both Remix and MetaMask.

Experiment 2: MetaMask in a Private Network

- **Level 1:** Set up a local Ganache private network. Add a custom network in MetaMask, configuring the RPC URL and Chain ID to connect to your Ganache instance. Create a new account in MetaMask and observe its balance.
- **Level 2:** Deploy a simple token contract (e.g., ERC-20 minimal) using Remix IDE and MetaMask on your private Ganache network. Transfer some tokens between the accounts you created in MetaMask and observe the balance changes.

Experiment 3: Smart Contract with Solidity - Basic Data Types and Structures

- **Level 1:** Write a Solidity smart contract that declares and initializes variables of different basic data types (uint, string, bool, address). Implement functions to read and modify these variables. Deploy and interact with the contract in Remix IDE.
- **Level 2:** Create a Solidity smart contract that utilizes structs and arrays. Implement functions to add, retrieve, and update elements within these data structures. Deploy and test the contract with various inputs in Remix IDE.

Experiment 4: Smart Contract with Solidity - Control Flow and Functions

- **Level 1:** Write a Solidity smart contract that uses if-else statements and for loops within its functions. Implement a function that performs a simple calculation based on input parameters. Deploy and test the different control flow paths in Remix.
- **Level 2:** Design and implement a Solidity smart contract with multiple functions, including internal and private functions. Demonstrate how these functions can be called and how visibility modifiers affect their accessibility.

Experiment 5: Contract Deployment

- **Level 1:** Deploy a pre-written simple smart contract (provided by the instructor) using MetaMask connected to the Ganache network. Observe the deployment transaction details (gas used, transaction hash, contract address).
- **Level 2:** Explore different deployment parameters in Remix IDE (e.g., setting gas limit and gas price). Deploy the same contract multiple times with varying gas settings and analyze the impact on deployment cost and confirmation time in Ganache.

Experiment 6: MetaMask and Remix IDE Interaction

- **Level 1:** Deploy a simple counter smart contract using Remix IDE on the Ganache network via MetaMask. Use the Remix interface to call the contract's functions

(e.g., increment, decrement, get count) and observe the state changes reflected in both Remix and MetaMask (balance changes for transactions).

- **Level 2:** Deploy a more complex smart contract (e.g., a simple voting contract) using Remix and MetaMask. Interact with the contract through MetaMask's custom interaction interface (sending transactions to specific functions with appropriate arguments).

Experiment 7: Use of Geth - Installation and Account Management

- **Level 1:** Install the Geth Ethereum client on your local machine. Use Geth commands to create new Ethereum accounts and list the available accounts. Observe the keystore directory where private keys are stored.
- **Level 2:** Use Geth commands to export and import Ethereum account private keys. Understand the security implications of managing private keys. Connect the Geth console to a running private network (e.g., Ganache or a custom Geth network).

Experiment 8: Genesis Block Creation in Geth

- **Level 1:** Understand the structure of a Genesis Block JSON file. Modify a sample Genesis Block configuration (e.g., changing the initial coin distribution). Initialize a new Geth data directory using this modified Genesis Block.
- **Level 2:** Create a custom Genesis Block for a private Ethereum network with specific pre-allocated accounts, custom gas limit, and difficulty. Start a Geth node using this custom Genesis Block and connect to it using the Geth console.

Experiment 9: Interacting with a Private Geth Network

- **Level 1:** Start a Geth node using a previously initialized data directory. Use the Geth console to check the node's peer count and block number. Create a transaction to send Ether between two accounts within your private network using Geth commands.
- **Level 2:** Deploy a simple smart contract to your private Geth network using the Geth console and web3.js (or similar library). Interact with the deployed contract's functions using the Geth console.

Experiment 10: Exploring Ethereum Transaction Structure

- **Level 1:** Send a transaction (Ether transfer) using MetaMask on the Ropsten test network. Examine the transaction details on a block explorer (e.g., Etherscan for Ropsten) and identify key fields like to, from, value, gas limit, gas price, and nonce.
- **Level 2:** Construct and sign a raw Ethereum transaction using web3.js (or similar library) without relying on MetaMask. Broadcast this signed transaction to a test network (e.g., Ropsten) and analyze its details on a block explorer.

Experiment 11: Working with Smart Contract Events

- **Level 1:** Write a Solidity smart contract that emits events when certain actions occur (e.g., a value is updated). Deploy the contract in Remix and trigger the actions. Observe the emitted events in the Remix console.

- **Level 2:** Modify the previous contract to include indexed event parameters. Write a simple web3.js script (or use the Remix event listener) to filter and listen for specific events based on the indexed parameters.

Experiment 12: Understanding Smart Contract Security - Common Vulnerabilities (Part 1)

- **Level 1:** Study a simple smart contract with a known vulnerability (e.g., integer overflow/underflow - using an older Solidity version). Deploy the contract in Remix and attempt to exploit the vulnerability through function calls.
- **Level 2:** Research and demonstrate another common smart contract vulnerability (e.g., reentrancy - using a simplified example). Write a vulnerable contract and a separate "attacker" contract to exploit it on a local test network.

Experiment 13: Understanding Smart Contract Security - Common Vulnerabilities (Part 2)

- **Level 1:** Analyze a smart contract with access control implemented using onlyOwner modifier. Deploy the contract and attempt to call restricted functions from a non-owner account.
- **Level 2:** Explore the concept of gas limits and denial-of-service (DoS) attacks in smart contracts. Write a contract that could be susceptible to a simple gas-based DoS attack and demonstrate how it can be exploited.

Experiment 14: Interacting with Standard ERC-20 Tokens

- **Level 1:** Deploy a standard ERC-20 token contract (using OpenZeppelin library in Remix or a pre-written contract) on a local test network. Interact with the token contract's functions (e.g., totalSupply, balanceOf, transfer) using Remix.
- **Level 2:** Write a simple Solidity smart contract that interacts with the deployed ERC-20 token contract. Implement a function in your contract that allows users to spend a certain amount of the deployed tokens (requiring approval).

Experiment 15: Introduction to Truffle Framework

- **Level 1:** Install Truffle and Node.js. Create a new Truffle project. Understand the basic directory structure of a Truffle project (contracts, migrations, test). Compile a simple Solidity contract using Truffle commands.
- **Level 2:** Write a simple test case for your smart contract using Truffle's testing framework (Chai and Mocha). Run the tests to ensure the contract functions as expected. Deploy your compiled contract to a local Ganache network using Truffle migrations.

	Topics relevant to development of "Employability": Hands-on experience with blockchain platforms, smart contract development, and decentralized applications **Topics relevant to "PROFESSIONAL ETHICS":** Understanding ethical considerations in blockchain use, such as data privacy, transparency, and responsible innovation, promotes integrity in decentralized systems

Course Code: CAI3427	**Course Title: Language Models for Text Mining** **Type of Course: Discipline Elective - Theory**	**L-T-P-C**	2	0	0	2
Version No.	1.0					
Course Pre-requisites	CSE2264					
Anti-requisites	NIL					
Course Description	This course introduces the basics of Text Mining and Natural Language Processing. The course will teach students different concepts such as text mining, NLP, Sequence Labeling, etc. Topics: Text Mining, NLP, Tokenization, Lemmatization, Stemming, One-hot encoding, Language modelling, Bag-of-words, Term-document Matrix, Cosine similarity, Viterbi Algorithm, etc.					
Course Objectives	The objective of the course is EMPLOYBILITY of student by using EXPERIENTIAL LEARNING techniques.					
Course Out Comes	On successful completion of this course the students shall be able to: 25. **Process** text data to derive information from text. [Apply] 26. **Apply** insights from textual information to real-world business. [Apply] 27. **Develop** solutions for a particular NLP problem using different machine learning and deep learning techniques. [Apply] 28. **Utilize** different NLP tools and packages. [Apply]					

Course Content:

Module 1	**Text Mining**	**Adversarial Quiz Tests**	**Module Tests**	**No. of Sessions: 09**
Introduction to Text Mining. Text Mining vs. NLP. Text Mining Algorithms. Steps in Text Mining - Extraction, Preprocessing, Analysis and Evaluation. Lexical Resource Creation (NEW). Data collection. String Manipulation to Clean Data. Natural Language Processing. Research Paradigms in NLP. Sequential Data. Sequence Labeling (NEW). Viterbi Algorithm (NEW). Corpus. Building a HMM using a Corpus (NEW). Unknown word handling (NEW).				
Module 2	**Text Preprocessing**	**Adversarial Quiz Tests**	**Module Tests**	**No. of sessions: 06**
Introduction to Preprocessing. Tokenization. Stop Words Removal. Lemmatization and Stemming. PoS Tagging. Integer Encoding. Padding. One-Hot Encoding.				
Module 3	**Text Representations**	**Adversarial Quiz Tests**	**Module Tests**	**No. of sessions: 08**
Language Modeling. N-Gram Language Model. Bag-of-Words Model. Term-Document Matrix. Term Frequency. Inverse Document Frequency. TF-IDF. Cosine Similarity. Naive Bayes Classifier using Bag-of-Words. Topic Modeling. Latent Semantic Analysis. Singular Value Decomposition. Truncated SVD and Topic Vector. LDA Algorithm.				
Module 4	**Natural Language Processing with Keras**	**Adversarial Quiz Tests**	**Module Tests**	**No. of Sessions: 06**
Word Embeddings vs. One-Hot Encoding. Contextual Bag of Words (CBOW). Skipgram. Deep Learning for Document Classification.				

List of Laboratory Tasks:

Experiment No. 1: File Handling
Level 1: Read text files using Python and extract meaningful content.
Level 2: Parse text files using Python to preprocess the data for NLP tasks.

Experiment No. 2: Introduction to NLP Tools
Level 1: Install and use NLTK for basic text processing.
Level 2: Install and use SpaCy for tokenization, PoS tagging, and Named Entity Recognition.

Experiment No. 3: Corpus Cleaning Techniques
Level 1: Use NLTK for corpus cleaning techniques such as tokenization, stopword removal, and stemming.
Level 2: Prepare cleaned text data for downstream NLP tasks like classification or translation.

Experiment No. 4: Word Vector Usage
Level 1: Download and use pre-trained word vectors (e.g., Word2Vec, GloVe, or FastText).
Level 2: Compute similarity between two words, find the most similar word, and complete word analogies (e.g., king - man + woman = queen).

Experiment No. 5 & 6: Language Identification
Level 1: Build a simple language identifier using Bag-of-Words (BoW) features.
Level 2: Predict the language of a given text using the trained model.

Experiment No. 7 & 8: Lexical Simplification
Level 1: Implement a lexical simplifier to replace complex words with simpler alternatives.
Level 2: Generate a simplified version of a given word or sentence while preserving meaning.

Experiment No. 9 & 10: Sentiment Analysis
Level 1: Implement a basic sentiment classifier using a lexicon-based or machine learning approach.
Level 2: Compare the performance of an existing sentiment classifier (e.g., VADER, TextBlob, or a pre-trained Transformer model).

Experiment No. 11: Named Entity Recognition (NER)
Level 1: Extract named entities from a text using NLTK.
Level 2: Extract named entities using SpaCy and compare results.

Experiment No. 12 & 13: Implement a Hidden Markov Model (HMM)
Level 1: Implement a generic HMM for sequence prediction.
Level 2: Calculate the forward probability of a given sequence using HMM.

Experiment No. 14: Linguistic HMM
Level 1: Develop a Hidden Markov Model (HMM) for NLP tasks such as PoS tagging.
Level 2: Evaluate the performance of the HMM on a specific NLP task (e.g., Named Entity Recognition or Chunking).

Experiment No. 15: Machine Translation
Level 1: Implement Machine Translation (MT) using a pre-trained model from Hugging Face Transformers.
Level 2: Evaluate the quality of MT output via Round-Trip Translation (translate text to another language and back to check accuracy).

Targeted Application & Tools that can be used:

11. Google Colab
12. Python IDEs like PyCharm

Project work/Assignment: Mention the Type of Project /Assignment proposed for this course

3. Group project on some NLP Task like text classification (Creating a Simple Text Classifier: Use Scikit-learn to classify positive vs. negative reviews from a dataset), sentiment analysis, etc.

Textbook(s):

1. Daniel Jurafsky, James H. Martin. “Speech and Language Processing: An Introduction to Natural Language Processing”, Computational Linguistics and Speech, Pearson Publication, 2025 (3rd Edition Draft).
2. Aditya Joshi, Pushpak Bhattacharyya. “Natural Language Processing”, Wiley Publication, 2023 (1st Edition).

References:

R1. Chris Manning and Hinrich Schutze, “Foundations of Statistical Natural Language Processing”, 1st Edition, MIT Press. 1999.

R2. Pawan Goyal. “Natural Language Processing”. 1st Edition, 2016.

Weblinks

W1. E-Book link or R2: https://drive.google.com/file/d/10nbwAJd-dv6htOOZVBgAvLd1WscI0RqC/view

W2. Web Resource for T1: https://web.stanford.edu/~jurafsky/slp3/ - VERY VERY IMPORTANT!!!

W3. NPTEL Courses: https://nptel.ac.in/courses/106106211 CMI), https://nptel.ac.in/courses/106105158 (IIT Kgp), https://nptel.ac.in/courses/106101007 (IITB), https://nptel.ac.in/courses/106105572 (IIT Kgp - NEW)

<table>
<tr><td>Course Code: CAI3428</td><td colspan="2">Course Title: Practical Deep Learning with TensorFlow
Type of Course: Discipline Elective - Theory & Integrated Laboratory</td><td>L- T-P- C</td><td>2</td><td>0</td><td>2</td><td>3</td></tr>
<tr><td>Version No.</td><td colspan="7">1.0</td></tr>
<tr><td>Course Pre-requisites</td><td colspan="7">CSE2264</td></tr>
<tr><td>Anti-requisites</td><td colspan="7">NIL</td></tr>
<tr><td>Course Description</td><td colspan="7">This course introduces students to the concepts of deep neural networks and state of the art approaches to develop deep learning models. In this course students will be given an exposure to the details of neural networks as well as deep learning architectures and to develop end-to-end models for such tasks. It will help to design and develop an application-specific deep learning models and also provide the practical knowledge handling and analyzing end user realistic applications.</td></tr>
<tr><td>Course Objective</td><td colspan="7">This course is designed to improve the learners EMPLOYABILITY SKILLS by using EXPERIENTIAL LEARNING techniques.</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">On successful completion of this course the students shall be able to:
5. Implement backpropagation and gradient descent techniques to train neural networks effectively. (Apply)
6. Build and train deep learning models using Python libraries such as TensorFlow and Keras for real-world applications. (Apply)
7. Utilize deep learning techniques for image classification, object detection, sentiment analysis, and language modeling. (Apply)</td></tr>
<tr><td colspan="8">Course Content:</td></tr>
<tr><td>Module 1</td><td colspan="2">Basics of Neural Networks</td><td>Assignment</td><td colspan="3"></td><td>18[8L+10P] Sessions</td></tr>
<tr><td colspan="8">Topics:
Understanding Perceptron with Excel, Understanding Multilayer Perceptron with Excel, From Multilayer Perceptron to Deep Learning, Error Backpropagation and Gradient Descent to reduce errors, Activation Functions, Deep Learning, Problems with Deep Learning with solutions.</td></tr>
<tr><td>Module 2</td><td colspan="2">TensorFlow Basics</td><td>Assignment</td><td colspan="3"></td><td>14[7L+7P] Sessions</td></tr>
<tr><td colspan="8">Topics:
Introduction to TensorFlow, TensorFlow dataset, Machine Learning with TensorFlow</td></tr>
<tr><td>Module 3</td><td colspan="2">Deep Learning methods with Tensor Flow and Keras</td><td>Assignment</td><td colspan="3"></td><td>14[6L+8P] Sessions</td></tr>
<tr><td colspan="8">Topics:
Main Features of TensorFlow, Keras basics, AI with Keras.</td></tr>
<tr><td colspan="8">Project work/Assignment:</td></tr>
</table>

12. Assignment 1 on (Module 1 and Module 2)

13. Assignment 2 on (Module 3)

List of Laboratory Tasks:

Lab 1: Working with Deep Learning Frameworks
Objective: Explore various Deep Learning Frameworks
Tasks: Identify deep learning frameworks (Keras, Tensorflow, Matplotlib, etc)
Activity: Practice with various methods available in DL Frameworks to develop a Model.

Lab 2: Build a Basic Artificial Neural Network
Objective: Create a ANN with DL frameworks.
Task: Identify suitable ANN Layers using Keras and Tensorflow.
Activity: Design a basic Artificial Neural Networks using Keras with TensorFlow (pima-indians-diabetes)

Lab 3: Build a MultiLayer Perceptron
Objective: Create a MLP for classification task.
Task: Identify suitable model for house price prediction.
Activity: Design a MLP for implementing classification and fine-tuning using House price.csv

Lab 4: Create a Tensor in TensorFlow using List or Numpy array.
Objective: To understand how to create a tensor in TensorFlow using a Python list or NumPy array
Task: Create a simple tensor using both a Python list and a NumPy array in TensorFlow.
Activity: Create a tensor using a Python list and Numpy array

Lab 5: Apply math operations on tensor using various mathematical functions.
Objective: To learn how to apply mathematical operations on tensors using various TensorFlow mathematical functions.
Task: Perform basic mathematical operations (addition, subtraction, multiplication, division) and advanced functions (square, square root, exponential) on tensors.
Activity: Perform basic math operations: Add, Subtract, Multiply, Divide and Apply advanced math functions: Square, Square root, Exponential.

Lab 6: Connecting two tensors in dataset.
Objective: Combine two tensors using concatenation and stacking operations in TensorFlow.
Task: Combine two tensors using concatenation and stacking operations in TensorFlow
Activity: Concatenate them along a specific axis and Stack them along a new axis.

Lab 7: Building dataset from a file stored in a local drive
Objective: To learn how to build a dataset in TensorFlow from a file stored in a local drive.
Task: Load a dataset from a CSV file stored on the local drive and process it using TensorFlow
Activity: Load the file using TensorFlow's tf.data API and Process the dataset (e.g., convert it into tensors)

Lab 8: Loading Dataset from TensorFlow.dataset Library
Objective: To learn how to load a dataset from the tensorflow_datasets library and use it in machine learning models.
Task: Load a dataset from TensorFlow Datasets (tfds), preprocess it, and display sample data
Activity: Load a dataset (e.g., MNIST, CIFAR-10, IMDB Reviews) and Split the dataset into training and testing sets.

Lab 9: Build a Convolutional Neural Network
Objective: Create a CNN model.
Task: Build CNN architecture for Dog-Cat classification problem.

Activity: Implement a Convolution Neural Network (CNN) for dog/cat classification problem using keras.

Lab 10: **Build a Time-Series Model**
Objective: Create a RNN and LSTM Model
Task: Build RNN/LSTM Model for predicting time series data.
Activity Train a sentiment analysis model on IMDB dataset, use RNN layers with LSTM/GRU notes.

REFERENCE MATERIALS:

TEXTBOOKS

3. François Chollet, "Deep Learning with Python", 2nd Edition, Manning Publications, 2022
4. Ian Goodfellow, Yoshua Bengio, Aaron Courville, "Deep Learning", MIT Press, 2017.

REFERENCES

4. Amlan Chakrabarti Amit Kumar Das, Saptarsi Goswami, Pabitra Mitra , "Deep Learning", Pearson Publication, 2021.
5. David Foster, "Generative Deep Learning" O'Reilly Publishers, 2020.
6. John D Kellehar, "Deep Learning", MIT Press, 2020.

JOURNALS/MAGAZINES

1. IEEE Transactions on Neural Networks and Learning Systems

https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=5962385

2. IEEE Transactions on Pattern Analysis and Machine Intelligence

https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=34http://ijaerd.com/papers/special_papers/IT032.pdf

3. International Journal of Intelligent Systems https://onlinelibrary.wiley.com/journal/1098111x

SWAYAM/NPTEL/MOOCs:

4. Swayam Nptel – Deep Learning – IIT Ropar https://onlinecourses.nptel.ac.in/noc21_cs35/preview

5. Coursera – Neural Networks and Deep Learning Andrew Ng

6. Coursera - Neural Networks for Machine Learning by Geoffrey Hinton in Coursera

<table>
<tr><td>COURSE CODE: CAI3429</td><td colspan="2">Course Title: Deep Learning for Computer Vision
Type of Course: Discipline Elective - Theory & Integrated Laboratory</td><td>L-T-P-C</td><td>2</td><td>0</td><td>2</td><td>3</td></tr>
<tr><td>Version No.</td><td colspan="7">1.0</td></tr>
<tr><td>Course Pre-requisites</td><td colspan="7">MAT1003</td></tr>
<tr><td>Anti-requisites</td><td colspan="7">NIL</td></tr>
<tr><td>Course Description</td><td colspan="7">This course covers the fundamentals and advanced concepts of deep learning for computer vision applications. Students will explore convolutional neural networks (CNNs), object detection, image segmentation, and generative models. Hands-on lab experiments will reinforce theoretical concepts using frameworks like TensorFlow and PyTorch.</td></tr>
<tr><td>Course Out Comes</td><td colspan="7">On successful completion of the course the students shall be able to:

1. Understand the Fundamentals of Deep Learning for Vision

Explain the core concepts of neural networks and deep learning architectures for image processing.

Implement and optimize convolutional neural networks (CNNs) for classification tasks.

2. Apply Object Detection and Image Segmentation Techniques

Implement and analyze state-of-the-art object detection algorithms such as YOLO, Faster R-CNN, and SSD.

Develop and evaluate image segmentation models like U-Net and Mask R-CNN.

3. Explore Advanced Deep Learning Techniques for Vision

Utilize Vision Transformers (ViTs) and attention mechanisms for image classification.

Generate and manipulate images using Generative Adversarial Networks (GANs).

4. Deploy and Optimize Deep Learning Models for Real-World Applications</td></tr>
<tr><td>Course Content:</td><td colspan="7"></td></tr>
<tr><td>Module 1</td><td>Fundamentals of Deep Learning for Vision</td><td colspan="3">Assignment</td><td colspan="3">Practical</td><td>No. of Classes:8</td></tr>
<tr><td colspan="8">Introduction to Deep Learning & Neural Networks, Convolutional Neural Networks (CNNs) Architecture</td></tr>
</table>

Backpropagation & Optimization in CNNs, Transfer Learning & Pretrained Models.				
Module 2	**Object Detection & Image Segmentation**	Assignment	Practical	**No. of Classes:14**
Introduction to Object Detection (R-CNN, SSD, YOLO), Region Proposal Networks (Faster R-CNN) Semantic & Instance Segmentation (U-Net, Mask R-CNN), Real-time Object Detection Applications				
Module 3	**Advanced Topics in Vision**	Assignment	Practical	**No. of Classes:8**
Attention Mechanisms & Vision Transformers (ViTs), Generative Adversarial Networks (GANs) for Image Generation, Self-supervised Learning for Vision, Multi-modal Learning (CLIP, DALL·E)				
Module 4	**Applications & Deployment**	Assignment	Practical	**No. of Classes:8**
Edge AI & Mobile Deployment (TensorFlow Lite, ONNX), Adversarial Attacks & Robustness in Vision Models, Explainability & Interpretability of Vision Models, Case Studies & Industry Applications				

Lab Experiments are to be conducted on the following topics:-

Lab Sheet 1:

Keras Sequential API model

1. Read in the data and explore
2. Define a Sequential API model
3. Define the hyperparameters and optimizer
4. Train the model and visualize the history
5. Testing

Keras Functional API model:

1. Define a Functional API model
2. Train the model and visualize the history

Lab Sheet 2:

Softmax regression with Keras

1. Read in the data and prepare
2. Define a Sequential API model
3. Define the hyperparameters and optimizer
4. Train the model and visualize the history
5. Testing

Lab Sheet 3:

Convolutional Neural Network with Keras (grayscale images)

1. Read in the data:
2. Visualize the data:

3. Prepare the data:
4. Define a CNN model:
5. Define the hyperparameters and optimizer:
6. Train the model and visualize the history:
7. Testing:

Lab Sheet 4:

Convolutional Neural Network with Keras (color images):

1. Read in the data:
2. Visualize the data:
3. Prepare the data:
4. Define a CNN model:
5. Define the hyperparameters and optimizer:
6. Train the model and visualize the history:
7. Testing:

Lab Sheet 5:

Time series and prediction:

1. Read in the data and explore:
2. Apply the exponential smoothing method and predict

Recurrent neural network (RNN):

1. Pre-processing:
2. Do the necessary definitions: (Hyper parameters, Model,
3. Train the model:
4. Predict the future:

Lab Sheet 6:

Document classification with LSTM network:

1. Read in the data:
2. Explore the data:
3. Data preprocessing:
4. Define the model:
5. Define the optimizer and compile:
6. Train the model and visualize the history:
7. Testing:

Lab Sheet 7:

Document classification with LSTM network (Binary):

1. Read in the data:
2. Explore the data:
3. Data preprocessing:
4. Define the model:
5. Define the optimizer and compile:
6. Train the model and visualize the history:
7. Testing:

Lab Sheet 8:

Document classification with LSTM + CNN network (Binary):

1. Read in the data:

2. Explore the data:
3. Data preprocessing:
4. Define the model:
5. Define the optimizer and compile:
6. Train the model and visualize the history:
7. Testing:

Lab Sheet 9:

Softmax regression to recognize the handswritten digits:

1. Download the MNIST data:
2. Take a look at the dataset:
3. Do the necessary definitions:
4. Training and Testing:

Multi-layer neural network to recognize the handswritten digits:

1. Download the MNIST data:
2. Take a look at the dataset:
3. Do the necessary definitions:

Training and Testing:

Lab Sheet 10:

Object Detection using YOLOv5

Lab Sheet 11:

Image Segmentation using U-Net
Custom Object Detection using Faster R-CNN

Lab Sheet 12:

Implementing Vision Transformers for Image Classification
Generating Images using GANs (DCGAN, StyleGAN)

(Group Project)

8. Object Detection and Recognition:

a. Haar cascade object detection (e.g., face detection or object detection using pre-trained classifiers).

b. Feature-based object detection using techniques like Speeded-Up Robust Features (SURF) or Scale-Invariant Feature Transform (SIFT).

c. Deep learning-based object detection using Convolutional Neural Networks (CNNs) or You Only Look Once (YOLO) algorithm.

9. Optical Character Recognition (OCR):

a. Preprocessing of text images (e.g., binarization, noise removal, or skew correction).

b. Text localization using techniques like connected component analysis or Stroke Width Transform (SWT).

c. Character recognition using machine learning algorithms like Support Vector Machines (SVM) or Convolutional Neural Networks (CNNs).

10. Gesture Recognition:

a. Hand segmentation using techniques like background subtraction or skin color detection.

b. Feature extraction from hand regions (e.g., finger counting, hand shape descriptors).

c. Classification of gestures using machine learning algorithms (e.g., k-Nearest Neighbors or

Support Vector Machines).

Tools/Software Required :

1. OpenCV 4
2. Python 3.7
3. MATLAB

Text Books

1. "Deep Learning for Computer Vision Image Classification, Object Detection and Face Recognition in Python"
Jason Brownlee (2019)
2. "Deep Learning for Computer Vision with python" **Adrian Rosebrock (2017)**

References

3. **Goodfellow, I., Bengio, Y., & Courville, A. (2016).** *Deep Learning.* MIT Press.
A foundational book covering deep learning principles, including CNNs, optimization, and generative models.
4. **Raschka, S., & Mirjalili, V. (2022).** *Machine Learning with PyTorch and Scikit-Learn.* Packt Publishing.
Covers practical deep learning techniques using PyTorch, including CNNs and transfer learning.
5. **Geron, A. (2022).** *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow (3rd Edition).* O'Reilly Media.
Provides hands-on implementations of deep learning for computer vision using TensorFlow and Keras.
6. **Zhang, A., Lipton, Z. C., Li, M., & Smola, A. J. (2021).** *Dive into Deep Learning.* Available online (https://d2l.ai).
Open-access book covering CNNs, object detection, and advanced vision techniques with PyTorch and TensorFlow.
7. **Chollet, F. (2021).** *Deep Learning with Python (2nd Edition).* Manning Publications.
Explains deep learning fundamentals and applications with Keras, including image classification and segmentation.
8. **Ballé, J., Laparra, V., & Simoncelli, E. P. (2017).** *Deep Learning for Computer Vision: A Brief Introduction.*
A concise introduction to CNNs, object detection, and generative models.

Course Code: CSE3426	**Course Title: Front-end Full Stack Development** **Course Type : Lab Integrated**	**L- T-P- C**	2	0	2	3

Version No.	1.0
Course Pre-requisites	CSE2258
Anti-requisites	**NIL**
Course Description	This intermediate course enables students to perform front-end full stack development, with emphasis on employability skills. The course covers key technologies and architectures that enables the student to design and implement front-end. On successful completion of this course, the student shall be able to pursue a career in full-stack development. The students shall develop strong problem-solving skills as part of this course.
Course Objectives	The objective of the course is to familiarize the learners with the concepts Front end Full Stack Development and attain Employability through experiential Learning techniques.
Course Outcomes	**On successful completion of the course the students shall be able to:** **1] Describe the fundamentals of DevOps and Front-end full stack development. [Comprehension]** **2] Illustrate a basic web design using HTML, CSS, Javascript. [Application]** **3] Illustrate development of a responsive web. [Application]** **4] Apply concepts of Angular.js to develop a web front-end. [Application]**
Course Content:	

Module 1	Fundamentals of DevOps	Project		Programming	**04 Sessions**
	Topics: Introduction to Agile Methodology; Scrum Fundamentals; Scrum Roles, Artifacts and Rituals; DevOps – Architecture, Lifecycle, Workflow & Principles; DevOps Tools Overview – Jenkins, Docker, Kubernetes. Review of GIT source control.				
Module 2	Web Design & Development	Project		Programming	**03 Sessions**
	Topics: HTML5 – Syntax, Attributes, Events, Web Forms 2.0, Web Storage, Canvas, Web Sockets; CSS3 – Colors, Gradients, Text, Transform; **Assignment:** Develop a website for managing HR policies of a department.				
Module 3	Responsive web design	Project		Programming	**08 Sessions**
	Topics: BootStrap for Responsive Web Design; JavaScript – Core syntax, HTML DOM, objects, classes, Async; Ajax and jQuery Introduction **Assignment:** Design and develop a website that can actively keep track of entry-exit information of a housing society.				
Module 4	Fundamentals of Angular.js	Project		Programming	**15 Sessions**
	Topics: Setting up Development & Build Environment: Node.js and NPM; Introduction to TypeScript; Working with OOP concepts with TypeScript; Angular Fundamentals; Angular CLI; Introduction to TypeScript; Debugging Angular applications; Components & Databinding in Depth; Angular Directives; Using Services & Dependency Injection; Angular Routing; Observables; Handling Forms in Angular Apps; Output transformation using Pipes; Making Http Requests; Authentication & Route Protection; Dynamic Components; Angular Modules & Optimizing Angular Apps; Deploying an Angular App; Angular Animations; Adding Offline Capabilities with Service Workers; Unit Testing in Angular Apps (Jasmine, Karma). Overview of React.js **Assignment:** Develop a software tool to do inventory management in a warehouse.				
	Targeted Application & Tools that can be used: **Application Area is to Design and Analyzing the efficiency of Algorithms. This fundamental course is used by all application developers.**				

	Professionally Used Software: GCC compiler.
	Text Book: T1. Fender, Young, "*Front-end Fundamentals",* Leanpub, 2015 T2. Northwood, Chris, "*The Full Stack Developer: Your Essential Guide to the Everyday Skills Expected of a Modern Full Stack Web Developer",* APress, 2018
	References: R1. Flanagan D S, *"Javascript : The Definitive Guide"* 7th Edition. 7th ed. O'Reilly Media; 2020. R2. Alex Libby, Gaurav Gupta, and Asoj Talesra. *"Responsive Web Design with HTML5 and CSS3 Essentials",* Packt Publishing, 2016 R3. Duckett J Ruppert G Moore J. *"Javascript & Jquery : Interactive Front-End Web Development.";* Wiley; 2014. R4. Web Reference: https://www.youtube.com/watch?v=JGNTYXkVCVY&list=PLd3UqWTnYXOkTSBCBNyyhxo_jxlY_uTWA&index=2 R5. Web Reference: https://www.freecodecamp.org/news/frontend-web-developer-bootcamp/ https://puniversity.informaticsglobal.com:2229/login.aspx?direct=true&db=nlebk&AN=2233842&site=ehost-live https://nptel.ac.in/courses/106102064
	Topics relevant to development of "Employability": DevOps Tools Overview – Jenkins, Docker, Kubernetes for developing Employability Skills through Experiential Learning techniques. This is attained through assessment component mentioned in course handout.

Course Code: CSE3427	**Course Title: Java Full Stack Development** **Course Type: Lab Integrated**	**L- T-P-C**	2	0	2	3

Version No.		1.0
Course Pre-requisites		**CSE1006**
Anti-requisites		
Course Description		This advanced level course enables students to perform full stack development using Java, with emphasis on employability skills. The key technologies used for Full Stack development is based on either Java technology or .NET technology. In this course, the focus is on using Java, and the related technologies/tools like Java EE, Java Persistence, Hibernate, Maven, Spring Core, etc. On successful completion of this course, the student shall be able to pursue a career in full-stack development. The students shall develop strong problem-solving skills as part of this course.
Course Objectives		This course is designed to improve the learners' EMPLOYABILITY SKILLS by using PROBLEM SOLVING Methodologies.
Course Outcomes		**On successful completion of the course the students shall be able to:** **1] Practice the use of Java for full stack development [Application]** **2] Show web applications using Java EE. [Application]** **3] Solve simple applications using Java Persistence and Hibernate [Application]** **4] Apply concepts of Spring to develop a Full Stack application. [Application]** **5] Employ automation tools like Maven, Selenium for Full Stack development. [Application]**
Course Content:		

Module 1	Introduction	Project		Programming	**03 Sessions**
	Topics: Review of Java; Advanced concepts of Java; Java generics; Java IO; New Features of Java. Unit Testing tools.				
Module 2	Java EE Web Applications	Project		Programming	**05 Sessions**
	Topics: Introduction to Eclipse & Tomcat; JSP Fundamentals; Reading HTML form Data with JSP; State Management with JSP; JSP Standard Tag Library - Core & Function Tags; Servlet API Fundamentals; ServletContext, Session, Cookies; Request Redirection Techniques; Building MVC App with Servlets & JSP; Complete App - Integrating JDBC with MVC App **Assignment:** Develop an application for managing HR policies of a department.				
Module 3	Java Persistence using JPA and Hibernate	Project		Programming	**06 Sessions**
	Topics: Fundamentals of Java Persistence with Hibernate; JPA for Object/Relational Mapping, Querying, Caching, Performance and Concurrency; First & Second Level Caching, Batch Fetching, Optimistic Locking & Versioning; Entity Relationships, Inheritance Mapping & Polymorphic Queries; Querying database using JPQL and Criteria API (JPA) **Assignment:** Design and develop a website that can actively keep track of entry-exit information of a housing society.				
Module 4	Spring Core	Project		Programming	**10 Sessions**
	Topics: Spring Core, Spring MVC, Spring Boot REST API; Understanding Spring Framework; Using Spring MVC; Building a Database Web App with Spring and Hibernate o Spring AOP (Aspect Oriented Programming); Implementing Spring Security; Developing Spring REST API; Using Spring Boot for Rapid Development **Assignment:** Develop a software tool to do inventory management in a warehouse.				

Module 5	Automation tools	Project		Programming	**06 Sessions**
	Topics: Introduction to Automation Tools; Apache Maven: Maven Fundamentals, Software Setup - Commandline and Eclipse, pom.xml and Directory Structure, Multi-Module Project Creation, Scopes, Dependency Management, Profiles; Functional/BDD Testing using Selenium, Selenium Fundamentals and IDE, Selenium WebDriver, Installation and Configuration, Locating WebElements, Driver Commands, WebElement Commands **Assignment:** Illustrate the use of automation tools in the development of a small software project.				
	Targeted Application & Tools that can be used: **Application Area is to Design and Analyzing the efficiency of Algorithms. This fundamental course is used by all application developers.** **Professionally Used Software: Eclipse, NetBeans, Hibernate, Selenium, Maven, GIT.**				
	Project work/Assignment:				
	Problem Solving: Design of Algorithms and implementation of programs. **Programming: Implementation of given scenario using Java.**				
	Text Book: T1. Fender, Young, "*Front-end Fundamentals",* Leanpub, 2015				
	References R1. Soni, Ravi Kant. *"Full Stack AngularJS for Java Developers: Build a Full-Featured Web Application from Scratch Using AngularJS with Spring RESTful."* , Apress, 2017. R2. Mardan, Azat. *"Full Stack JavaScript: Learn Backbone.js, Node.js and MongoDB."*, Apress, 2015				

Course Code: CSE3428	**Course Title: .NET Full Stack Development** Course Type: Lab Integrated		**L- T-P- C**	2	0	2	3
Version No.		1.0					
Course Pre-requisites		CSE2258					
Anti-requisites		**CSE3427**					
Course Description		This advanced level course enables students to perform full stack development using .NET, with emphasis on employability skills. The key technologies used for Full Stack development is based on either Java technology or .NET technology. In this course, the focus is on using .NET and the related technologies/tools like C#, ASP.NET, Entity Framework Core, etc. On successful completion of this course, the student shall be able to pursue a career in full-stack development. The students shall develop strong problem-solving skills as part of this course.					
Course Objectives		This course is designed to improve the learners' EMPLOYABILITY SKILLS by using PROBLEM SOLVING Methodologies.					
Course Outcomes		**On successful completion of the course the students shall be able to:** **1] Practice the use of C# for developing a small application [Application]** **2] Show web applications using Entity Framework. [Application]** **3]Solve simple web applications that use SQL and ASP.NET [Application]** **4] Apply concepts of ASP.NET to develop a Full Stack application. [Application]**					
Course Content:							
Module 1	C# Programming for Full Stack Development	Project		Programming		**10 Sessions**	
	Topics: .NET Framework Fundamentals, Visual Studio IDE Fundamentals, C# Language Features, Working with arrays and collections, Working with variables, operators, and expressions, Decision and iteration statements, Managing program flow and events, Working with classes and methods, OOP concepts, Properties, Auto Implemented, Delegates, Anonymous Methods and Anonymous Types, Extension methods, Sealed Classes/Methods, Partial Classes/Methods, Asynchronous programming and threading, Data validation and working with data collections including LINQ, Handling errors and exceptions, Working with Files, Unit Testing – Nunit framework **Assignment:** Develop a small application for managing library using C#.						
Module 2	Entity Framework Core 2.0	Project		Programming		**06 Sessions**	
	Topics: Entity Framework Core 2.0 Code First Approach; Introduction To Entity Framework and EDM; Querying the EDM; Working With Stored Procedures; Advanced Entity Framework - DbContext [EF6]; Advanced Operations; Performance Optimization; Data Access with ADO.NET **Assignment:** Develop an application for managing HR policies of a department.						
Module 3	ASP.NET	Project		Programming		**06 Sessions**	
	Topics: ASP.NET Core, ASP.Net Core 3.1 MVC, ASP.NET Core Middleware and Request pipeline, Review of SQL using MS SQL, Working With Data In Asp.Net, Razor View Engine, State Management In Asp. Net MVC & Layouts; **Assignment:** Develop a web application to mark entry/exit of guests in a building.						
Module 4	ASP.NET	Project		Programming		**08 Sessions**	

	Topics: Introduction To Models, Validations In Asp.Net MVC, Authentication and Authorization In Asp.Net MVC, Advanced Asp. Net MVC - Ajax Action Link In MVC, Advanced Asp.Net MVC - Ajax Forms In MVC, Microsoft Testing Framework – Unit Testing the .NET Application **Assignment:** Develop a software tool to do inventory management in a warehouse.
	Targeted Application & Tools that can be used: **Application Area is to Design and Analyzing the efficiency of Algorithms. This fundamental course is used by all application developers.** **Professionally Used Software: Visual Studio**
	Project work/Assignment:
	Problem Solving: Design of Algorithms and implementation of programs. **Programming: Implementation of given scenario using .NET.**
	Text Book: T1. Fender, Young, "*Front-end Fundamentals",* Leanpub, 2015 T2. Valerio De Sanctis, *"ASP.NET Core 5 and Angular: Full-stack web development with .NET 5 and Angular 11",* 4th Edition, Packt, 2021.
	References R1. Benjamin Perkins, Jon D. Reid, *"Beginning C# and .NET",* Wiley, 2021 Reid, 2021. R2. Piotr Gankiewicz, *"Full Stack .NET Web Development",* Packt Publishing, 2017. R3. Tamir Dresher, Amir Zuker, Shay Friedman, *"Hands-On Full-Stack Web Development with ASP.NET Core",* Packt Publishing, 2018. R4. Dustin Metzgar, *"Exploring .NET core with microservices, ASP.NET core, and Entity Framework Core",* Manning, 2017.

Course Code: CSE3534	**Course Rust Programming** **Type of Course: Lab Integrated**	**L-T-P-C**	2	0	2	3
Version No.	**1.0**					
Course Pre-requisites	CSE1006					
Anti-requisites	**NIL**					
Course Description	Rust is for students and those who are interested in learning about systems concepts. Using Rust, many people have learned about topics like operating systems development. The Rust programming language helps students write faster, more reliable software. High-level ergonomics and low-level control are often at odds in programming language design; Rust challenges that conflict. Through balancing powerful technical capacity and a great developer experience, Rust gives developers the option to control low-level details such as memory usage without all the hassle traditionally associated with such control. The associated laboratory provides a chance to have hands-on concepts learned during this course.					
Course Objective	**This course is designed to improve the learners' EMPLOYABILITY SKILLS by learning the Rust Programming language.**					
Course Outcomes	**Upon the successful completion of this course students will be able to:** • **Create a Rust project, including proper Cargo configuration.** • **Translate a design into a working Rust program.** • **Explain and remedy type and lifetime errors encountered during Rust programming.** • **Use structs, enums and traits as intended in the construction of Rust programs.** • **Apply references, boxes, cells and reference counting in Rust programming.** • **Divide a Rust crate into multiple source files using the module system.** • **Write tests and documentation using the Rust infrastructure.**					
Course Content:						
Module 1	Introduction to Rust Programming	Quiz			**10 sessions**	
Topics: Introduction to Rust Programming: Reasons to adopt Rust –Use cases – Opportunities- Language Features-Advantages- Installation- First Example-Rust Data types- Variables – Constants-String-Operators-Branching and Looping						
Module 2	Features of Rust	Programming and Mini Project			**12 sessions**	
Topics: Unique Features of Rust: Tuple- Common types (Option, Result)- Array-Ownership – Borrowing- Slices Structures- Enums-Collections (vector, string, hashmap, iterator) - Modules - Rust Collections-Error Handling- Input Output Generics- Package Manager-Iterator- Closure-Smart Pointers – Concurrency						
Module 3	**Ownership**	Assignment			**11 sessions**	
Topics: Ownership + move semantics - Borrowing (References) – Lifetimes - Smart pointers (Box, Rc/Arc, Cell/Refcell) - Modularising projects - Documenting code - Unit testing - Documentation testing						
Module 4	Generics and Concurrency	Assignment			**12 sessions**	

Topics:
Generics (and monomorphization) – Traits - Static vs dynamic dispatch – Closures - Function types (fn, FnOnce, FnMut, Fn) – Macros - Fearless concurrency – Threads - Sync primitives (Mutex, RwLock, mpsc, etc.) - Current state and future of Rust

Install Rustup
https://www.rust-lang.org/tools/install
Experiment No. 1
Create and run a new project using the conventions of Cargo
Create a project to Covert Fahrenheit to Celsius.
Experiment No. 2
Create a project to display binary equivalent of an integer, perform AND, and shift operations.
Create a dollar pattern
Create a guessing game program that will ask for user input, process that input, and check that the input is in the expected form.
Experiment No. 3
Create a project to display the magnitude and angle of the given complex numbers.

Create a project with functions to check a number is prime or not, count primes, add primes and display primes within the given range.

Experiment No. 4
Create a simple programs using simple struct, associated functions and Enums
Implement a Turtle builder struct to allow building a Turtle object. Perform rotation with angle, move forward and backward.
Experiment No. 5
Write a program to solve the quadratic equation using command line arguments.
Write a program on Word histogram on accepting a file, count the occurrences using HashMap, sort and display.
Experiment No. 6
Create a program to simulate John Conway's Game of Life using array and thread sleep.

Create a project to make changes to Turtle type so that it is placed in a library module and main function uses the module.
Experiment No. 7
Write a program to handle error that main returns a Result type, making the necessary code changes.
Create a new library project named generics to implement stack and queue operations.

Targeted Application & Tools that can be used:
Rust is to be installed and used.
https://www.rust-lang.org

Project work/Assignment:

Problem Solving: Design of Algorithms and implementation of programs.
Programming: Implementation of given scenario using Rust.

Textbook(s):
1. **Klabnik, Steve, and Carol Nichols. *The Rust programming language*. No Starch Press, 2023. Publisher: William Pollock**

References
1. **Jim Blandey, Jason Orendorff and Leonora F.S. Tindall, "Programming Rust – Fast, safe system Development",2nd Edition, 2021(O'Reilly Publication).**

https://rustbook.cs.brown.edu
Topics related to development of "FOUNDATION":Features of Rust Topics related to development of "EMPLOYABILITY": Project implementations in software, batch wise presentations Topics related to development of "HUMAN VALUES AND PROFESSIONAL ETHICS": Team Dynamics during Mini Project Development.

Course Code: CCS2504	**Course Title:** Ethical Hacking **Type of Course:** Core Subject	**L-T- P- C**	**2**	**0**	**0**	**2**
Version No.	1.3					
Course Pre-requisites	Basic networking tools					
Anti-requisites	NIL					
Course Description	This course introduces students to a wide range of topics related to ethical hacking. It also provides an in-depth understanding of how to effectively protect computer networks. These topics cover some of the tools and penetration testing methodologies used by ethical hackers and provide a thorough discussion of what and who an ethical hacker is and how important they are in protecting corporate and government data from cyber-attacks					
Course Objective	The objective of the course is to familiarize the learners with the concepts of Ethical Hacking and attain to improve the learners' Employability Skills by using Experiential Learning techniques.					
Course Out Comes	On successful completion of this course the students shall be able to: 1. Extrapolate the importance of ethical hacking. 2. Determine the various techniques for performing reconnaissance Categorize various types of system scanners and their functions. 4. Identify the function of sniff on a network.					
Course Content:						

Module 1	**Introduction to Hacking**	Assignment		Programming activity	**7 Hours**
Topics: Introduction to Hacking-Important Terminologies - Asset - Vulnerability - Penetration Test - Vulnerability Assessments versus Penetration Test - Penetration Testing Methodologies - Categories of Penetration Test. **Assignment:** Different phase methodologies on penetration testing					
Module 2	**Linux Basics**	Assignment		Programming activity	**8 Hours**
Topics: Major Linux Operating Systems - File Structure inside of Linux - BackTrack - Changing the Default Screen Resolution - Some Unforgettable Basics. **Assignment:** Penetration testing distribution					
Module 3	**Information Gathering Techniques**	Assignment		Programming activity	**8 Hours**

Topics: Sources of Information Gathering - Copying Websites Locally - NeoTrace - Xcode Exploit Scanner - Interacting with DNS Servers - DNS Cache Snooping - DNS Lookup with Fierce - SNMP - SMTP. **Assignment:**Domain internet groper					
Module 4	**Target Enumeration and Port Scanning Techniques**	Assignment		Programming activity	**7 Hours**
Topics: Target Enumeration and Port Scanning Techniques - Host Discovery - Scanning for Open Ports and Services - Types of Port Scanning - Vulnerability Assessment. **Assignment:** Demonstrations for port scanning					
Text Book 1.Rafay Baloch, 2014: "Ethical Hacking and Penetration Testing Guide" Apple Academic Press Inc**.**					
References 1.Gary Hall, Rrin Watson, 2016: "Hacking: Computer Hacking, Security Testing,Penetration Testing, and Basic Security". 2.James Corley, Kent Backman, Michael Simpson, 2010: "Hands-On Ethical Hacking and Network Defense", 2nd Edition, Cengage Learning. **E-Resources**: (1) Ethical Hacking in 12 Hours - Full Course - Learn to Hack! - YouTube					
Topics relevant to "EMPLOYABILITY SKILLS": CEH Certification Ethical hacking techniques for **Employability skills** through **Experiential Learning techniques**. This is attained through the assessment component mentioned in course handout.					

Course Code: CCS2505	**Course Title:** Ethical Hacking Lab	**L-T- P- C**	**0**	**0**	**4**	**2**
Version No.	1.3					
Course Pre-requisites	Data Communication and Computer Networks					
Anti-requisites	NIL					
Course Description	This course introduces students to a wide range of topics related to ethical hacking. It also provides an in-depth understanding of how to effectively protect computer networks. These topics cover some of the tools and penetration testing methodologies used by ethical hackers and provide a thorough discussion of what and who an ethical hacker is and how important they are in protecting corporate and government data from cyber-attacks					
Course Objective	The objective of the course is to familiarize the learners with the concepts of Ethical Hacking and attain to improve the learners' Employability Skills by using Experiential Learning techniques.					
Course Out Comes	On successful completion of this course the students shall be able to: 1. Extrapolate the importance of ethical hacking.					

	2. Determine the various techniques for performing reconnaissance 3. Categorize various types of system scanners and their functions. 4. Identify the function of sniff on a network.
Course Content:	

List of Laboratory Tasks:
Experiments:

1. Command Prompt
2. Wireshark
3. Netscantool
4. OWZAP
5. Neotrace
6. NMAP
7. AngryIPScanner
8. Maltigo
9. Readnotify
10. HTTRACK
11. Yougetsignal
12. CAPSA Portable Network Analyzer
13. Samspade
14. Shodan
15. Oputils
16. Brupsuit
17. Zenmap
18. OSINT
19. John the ripper

Targeted Application & Tools that can be used: Application Software and open source tools like SQL Injection and NIDS,HIDS.

Text Book

1.Rafay Baloch, 2014: “Ethical Hacking and Penetration Testing Guide” Apple Academic Press Inc.

References

1.Gary Hall, Rrin Watson, 2016: “Hacking: Computer Hacking, Security Testing,Penetration Testing, and Basic Security”.
2.James Corley, Kent Backman, Michael Simpson, 2010: “Hands-On Ethical Hacking and Network Defense”, 2nd Edition, Cengage Learning.

E-Resources:
(1) Ethical Hacking in 12 Hours - Full Course - Learn to Hack! - YouTube

Topics relevant to “EMPLOYABILITY SKILLS”: CEH Certification
Ethical hacking techniques for **Employability skills** through **Experiential Learning techniques**. This is attained through the assessment component mentioned in course handout.

Course Code: CSD2001	**Course Title: Applied Data Science** **Type of Course: Program Core**	**L-T-P-C**	3	0	0	3
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	NIL					
Course Description	The aim of the course is to give complete overview of Python’s data analytics tools and techniques. Learning python is a crucial skill for many data science roles, and this course helps to understand and develop feature engineering. With a blended learning approach, Python for data science along with concepts like data wrangling, mathematical computing, and more can be learnt.					
Course Objectives	The objective of the course is to familiarize the learners with the concepts of **Applied Data Science** and attain **Skill Development** through E**xperiential Learning** techniques.					
Course Out Comes	On successful completion of this course the students shall be able to: 1. Describe Numpy and Matrix Operations [Remember] 2. Summarize the need for data preprocessing and visualization techniques. [Understand] 3. Demonstrate the performance of different supervised learning algorithms [Apply] 4. Apply unsupervised learning algorithms for grouping the given data. [Apply]					
Course Content:						
Module 1	**Introduction to Data Science, Python Data Structures, Python Numpy Package**	**Quiz**	**Knowledge based quiz**	**No. Of sessions:11**		
Data Science: Basics of Data Science, Sources of Data, Data Science Project Life Cycle: OSEMN Framework, Difference between data analysis and data analytics. Python- Variables, data types, control						

structures, Operators, Simple operations, Array and its operations, Numpy operations, Matrix and its operations.				
Module 2	**Data preparation and preprocessing using Pandas dataframe, Exploratory Data Analysis, Data Visualization**	**Assignment**	**Data Visualization**	**No. Of sessions:12**
Data Quality Assessment, Feature Aggregation, Feature Encoding, Dealing with missing values, Normalization techniques, Statistical description about the data, Relationship between the data, Data Visualization using matplotlib.				
Module 3	**Supervised Learning Algorithms**	**Design an algorithm using Example**	**Random Forest**	**No. Of sessions:11**
Supervised learning techniques: Regression Models- Linear and Logistic Model, Classification Models – Decision Tree, Naïve Bayes, Model Selection and Evaluation criteria: Accuracy, F1 score – Sensitivity – Specificity – AUC.				
Module 4	**Unsupervised Learning Algorithms**	**Case Study**	**Conduct a case study on how data sets can be gathered and implemented in real time application.**	**No. Of sessions:11**
The Clustering Models – K Means algorithm, K- Medoids Algorithm, types of clustering models, Hierarchical clustering techniques, drawbacks of K Means, case study for different algorithms.				

Textbook(s):

1. Data Science Using Python and R- Chantal D.L & Daniel T.L John Wiley & Sons, Inc. - 2019
2. Applied Data Science with Python and Jupyter-Alex Galea,Packt Publishing,October2018
1. Data Visualization in Python with Pandas and Matplotlib Paperback –DavidLandup, June 16, 2021

References:

1.Data Science with Python and Dask- Jesse Daniel,1st Edition,July30,2019

Weblinks:

- https://presiuniv.knimbus.com/user#/home
- Udemy: https://www.udemy.com/course/applied-data-science-with-python-specialization-mhm/
- NPTEL online course : https://nptel.ac.in/courses/106106179

Topics relevant to "SKILLS Development": Data Science, Decision Tree Algorithm for developing **Skills development** through **Experiential Learning techniques.**
This is attained through assessment component mentioned in course handout.

Course Code: CSD2002	**Course Title: Applied Data Science Lab** **Type of Course: Program Core**	**L-T-P-C**	0	0	2	1
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	NIL					
Course Description	The aim of the course is to give complete overview of Python's data analytics tools and techniques. Learning python is a crucial skill for many data science roles, and this course helps to understand and develop feature engineering. With a blended learning approach, Python for data science along with concepts like data wrangling, mathematical computing, and more can be learnt.					
Course Objectives	The objective of the course is to familiarize the learners with the concepts of **Applied Data Science** and attain **Skill Development** through E**xperiential Learning** techniques.					
Course Out Comes	On successful completion of this course the students shall be able to: 1. Describe Numpy and Matrix Operations [Remember] 2. Summarize the need for data preprocessing and visualization techniques. [Understand] 3. Demonstrate the performance of different supervised learning algorithms [Apply] 4. Apply unsupervised learning algorithms for grouping the given data. [Apply]					
Course Content:	**List of Laboratory Tasks:** 1. **Basic operations using Python** 2. **Reading and writing different types of datasets.** 3. **Descriptive statistics in python** 4. **Visualizations** 5. **Simple linear Regression** 6. **Simple logistic Regression**					

7. **Decision trees classifier**
8. **Support vector machine classifier**
9. **Naive Bayes classifier**
10. **Clustering model**

Targeted Application & Tools that can be used:

- Anaconda- Jupyter Notebook
- Google-Colab

Project work/Assignment:

1. Design forest fire and wildfire prediction system.
1. Driver Drowsiness Detection System with OpenCV & Keras
1. Credit Card Fraud Detection using Python.

Textbook(s):

1. Data Science Using Python and R- Chantal D.L & Daniel T.L John Wiley & Sons, Inc. - 2019
2. Applied Data Science with Python and Jupyter-Alex Galea,Packt Publishing,October2018

Data Visualization in Python with Pandas and Matplotlib Paperback –DavidLandup, June 16, 2021

References: 1.Data Science with Python and Dask- Jesse Daniel,1st Edition,July30,2019 **Weblinks:** • https://presiuniv.knimbus.com/user#/home • Udemy: https://www.udemy.com/course/applied-data-science-with-python-specialization-mhm/ • NPTEL online course : https://nptel.ac.in/courses/106106179	
Topics relevant to "SKILLS Development": Data Science, Decision Tree Algorithm for developing **Skills development** through **Experiential Learning techniques.** **This is attained through assessment component mentioned in course handout.**	

Course Code: CSE7000	**Course Title:** Internship **Type of Course:**	**L- T-P- C**	-	-	-	2
Version No.	**1.0**					
Course Pre-requisites	**Knowledge and Skills related to all the courses studied in previous semesters.**					
Anti-requisites	**NIL**					
Course Description	Students observe science and technology in action, develop an awareness of the method of scientific experimentation, and often get an opportunity to see, study and operate sophisticated and costly equipment. They also learn about the implementation of the principles of management they have learnt in class, when they observe multidisciplinary teams of experts from engineering, science, economics, operations research, and management deal with techno-economic problems at the micro and macro levels. Finally, it enables them to develop and refine their language, communication and inter-personal skills, both by its very nature, and by the various evaluation components, such as seminar, group discussion, project report preparation, etc. The broad-based core education, strong in mathematics and science and rich in analytical tools, provides the foundation necessary for the student to understand properly the nature of real-life problems.					
Course Objectives	The objective of the course is to familiarize the learners with the concepts of Professional Practice and attain Employability Skills through Experiential Learning techniques.					
Course Outcomes	On successful completion of this course the students shall be able to: 1. Identify the engineering problems related to local, regional, national or global needs. (Understand) 2. Apply appropriate techniques or modern tools for solving the intended problem. (Apply) 3. Design the experiments as per the standards and specifications. (Analyze) 4. Interpret the events and results for meaningful conclusions. (Evaluate)					

Course Code: **CSE7100**	**Course Title:** **Mini Project** **Type of Course:**	**L- T-P- C**	0	0	0	4
Version No.	**1.0**					
Course Pre-requisites	**Knowledge and Skills related to all the courses studied in previous semesters.**					
Anti-requisites	**NIL**					
Course Description	Students observe science and technology in action, develop an awareness of the method of scientific experimentation, and often get an opportunity to see, study and operate sophisticated and costly equipment. They also learn about the implementation of the principles of management they have learnt in class, when they observe multidisciplinary teams of experts from engineering, science, economics, operations research, and management deal with techno-economic problems at the micro and macro levels. Finally, it enables them to develop and refine their language, communication and inter-personal skills, both by its very nature, and by the various evaluation components, such as seminar, group discussion, project report preparation, etc. The broad-based core education, strong in mathematics and science and rich in analytical tools, provides the foundation necessary for the student to understand properly the nature of real-life problems. The students have options to pursue this course as either Project Work and Dissertation at the university, or Project Work in an Industry/ Company/ Research Laboratory, or Internship Program in an Industry/Company.					
Course Objectives	The objective of the course is to familiarize the learners with the concepts of Professional Practice and attain Employability Skills through Experiential Learning techniques.					
Course Outcomes	On successful completion of this course the students shall be able to: 1. Identify the engineering problems related to local, regional, national or global needs. (Understand) 2. Apply appropriate techniques or modern tools for solving the intended problem. (Apply) 3. Design the experiments as per the standards and specifications. (Analyze) 4. Interpret the events and results for meaningful conclusions. (Evaluate) 5. Appraise project findings and communicate effectively through scholarly publications. (Create)					

Course Code: **CSE7300**	**Course Title:** **Capstone Project** **Type of Course:**	**L- T-P- C**	0	0	0	10
Version No.	**1.0**					
Course Pre-requisites	**Knowledge and Skills related to all the courses studied in previous semesters.**					

Anti-requisites	**NIL**
Course Description	Students observe science and technology in action, develop an awareness of the method of scientific experimentation, and often get an opportunity to see, study and operate sophisticated and costly equipment. They also learn about the implementation of the principles of management they have learnt in class, when they observe multidisciplinary teams of experts from engineering, science, economics, operations research, and management deal with techno-economic problems at the micro and macro levels. Finally, it enables them to develop and refine their language, communication and inter-personal skills, both by its very nature, and by the various evaluation components, such as seminar, group discussion, project report preparation, etc. The broad-based core education, strong in mathematics and science and rich in analytical tools, provides the foundation necessary for the student to understand properly the nature of real-life problems. The students have options to pursue this course as either Project Work and Dissertation at the university, or Project Work in an Industry/ Company/ Research Laboratory, or Internship Program in an Industry/Company.
Course Objectives	The objective of the course is to familiarize the learners with the concepts of Professional Practice and attain Employability Skills through Experiential Learning techniques.
Course Outcomes	On successful completion of this course the students shall be able to: 1. Identify problems based on societal /research needs. (Understand) 2. Apply Knowledge and skill to solve societal problems in a group. (Apply) 3. Develop interpersonal skills to work as member of a group or leader. (Apply) 4. Analyze the inferences from available results through theoretical / Experimental / Simulations. (Analyze) 5. Analyze the impact of solutions in societal and environmental context for sustainable development. (Analyze) 6. Improve in written and oral communication. (Create) 7. Demonstrate capabilities of self-learning in a group, which leads to lifelong learning. (Understand)

Course Code: CIV7601	Course Title: Universal Human Values and Ethics Type of Course: MAC course	L-T-P-C	-	-	-	0
Course Pre-requisites	NIL					
Anti-requisites	NIL					

Course Description	The purpose of the course is to develop a holistic perspective in students' life. The course adopts a self-reflective methodology of teaching and is designed to equip the students to explore their role in all aspects of living as a part of the society. It presents a universal approach to value education by developing the right understanding of reality through the process of self-exploration. This self-exploration develops more confidence and commitment in students enabling them to critically evaluate their pre-conditioning and present beliefs. As an outcome of the holistic approach, the students will be able to practice the ethical conduct in the social and professional life. The prime focus throughout the course is toward affecting a qualitative transformation in the life of the student rather than just a transfer of information. This course is designed to cater to Human Values and Professional Ethics.			
Course Objective	The objective of the course is 'SKILL DEVELOPMENT' of the student by using 'SELF LEARNING' techniques			
Course Outcomes	On successful completion of this course the students shall be able to: CO.1 Recognize the importance of Value Education through the process of self-exploration CO.2 Explain the human being as the co-existence of the self and the body in harmony. CO.3 Describe the role of foundational values in building harmonious relationships. CO.4 Summarize the importance of a holistic perspective in developing ethical professional behavior.			
Course Content:				
Module 1	Introduction to Value Education	Online Assessment	MCQ Quiz	5 Sessions
Topics: Right Understanding, Relationship and Physical Facility (Holistic Development and the Role of Education), Understanding Value Education, Self-exploration as the Process for Value Education, Continuous Happiness and Prosperity – the Basic Human Aspirations, Happiness and Prosperity – Current Scenario, Method to Fulfil the Basic Human Aspirations.				
Module 2	Harmony in the Human Being	Online Assessment	MCQ Quiz	5 Sessions
Topics: Understanding Human being as the Co-existence of the Self and the Body, Distinguishing between the Needs of the Self and the Body, The Body as an Instrument of the Self, Understanding Harmony in the Self, Harmony of the Self with the Body, Programme to ensure self-regulation and Health				
Module 3	Harmony in the Family and Society	Online Assessment	MCQ Quiz	5 Sessions
Topics: Harmony in the Family – the Basic Unit of Human Interaction, 'Trust' – the Foundational Value in Relationship, 'Respect' – as the Right Evaluation, Other Feelings, Justice in				

Human-to-Human Relationship, Understanding Harmony in the Society, Vision for the Universal Human Order.

Module 4	Implications of the Holistic Understanding – A Look at Professional Ethics	Online Assessment	MCQ Quiz	5 Sessions

Topics:
Natural Acceptance of Human Values, Definitiveness of (Ethical) Human Conduct, A Basis for Humanistic Education, Humanistic Constitution and Universal Human Order, Competence in Professional Ethics, Holistic Technologies, Strategies for Transition towards Value-based Life and Profession

Targeted Application & Tools that can be used:
Application areas are Personal life, Education and Career, Workplace , Society and Environmental Responsibility
Tools: Online Tools – NPTEL and Swayam.

Project work/Assignment:

Assessment Type
- Online exams (MCQs) will be conducted by the Department of Civil Engineering through Linways.

Online Link*:
3) UHV II - https://www.youtube.com/watch?v=NhFBzn5qKIM&list=PLWDeKF97v9SO8vvjC1KyqteziTbTjN1So&pp=0gcJCWMEOCosWNin
4) Lecture by Dr. Kumar Sambhav, NPTEL course: Universal Human Values, https://onlinecourses.swayam2.ac.in/aic22_ge23/preview
5) Lecture by Dr. Padmavati, Dr Narendran Thiruthy, NPTEL Course: Biodiversity Protection, Farmers and Breeders Rights, https://nptel.ac.in/courses/129105008, 2024.

* Other source links are available in below Resources link.

Text Book

12. A Foundation Course in Human Values and Professional Ethics, R R Gaur, R Asthana, G P Bagaria, 2nd Revised Edition, Excel Books, New Delhi, 2019. ISBN 978-93-87034-47-1
13. Human Values, A.N. Tripathi, New Age Intl. Publishers, New Delhi, 2019.
14. Premvir Kapoor, Professional Ethics and Human Values, Khanna Book Publishing, New Delhi, 2022.

Reference Books

6. E.F. Schumacher, 1973, Small is Beautiful: a study of economics as if people mattered, Blond & Briggs, Britain.
7. Sussan George, 1976, How the Other Half Dies, Penguin Press. Reprinted 1986.
8. Donella H. Meadows, Dennis L. Meadows, Jorgen Randers, William W. Behrens III, 1972, Limits to Growth – Club of Rome's report, Universe Books.
9. A Nagraj, 1998, Jeevan Vidya Ek Parichay, Divya Path Sansthan, Amarkantak.
10. P L Dhar, RR Gaur, 1990, Science and Humanism, Commonwealth Publishers.
11. A N Tripathy, 2003, Human Values, New Age International Publishers.
12. E G Seebauer& Robert L. Berry, 2000, Fundamentals of Ethics for Scientists & Engineers, Oxford University Press

13. M Govindrajran, S Natrajan& V.S. Senthil Kumar, Engineering Ethics (including Human Values), Eastern Economy Edition, Prentice Hall of India Ltd.
14. B P Banerjee, 2005, Foundations of Ethics and Management, Excel Books.
15. William P. Cunningham and Mary Ann Cunningham (2020), Principles of Environmental Science: Inquiry & Applications, 9th Edition, McGraw-Hill Education, USA.

Resources:

16. https://onlinecourses.swayam2.ac.in/imb25_mg195/preview
17. https://onlinecourses.nptel.ac.in/noc25_mg141/preview
18. https://onlinecourses.swayam2.ac.in/ini25_hs52/preview
19. https://onlinecourses.nptel.ac.in/noc25_hs219/preview
20. https://onlinecourses.swayam2.ac.in/cec25_mg14/preview
21. https://onlinecourses.swayam2.ac.in/imb25_mg195/preview
22. https://onlinecourses.swayam2.ac.in/imb25_mg196/preview

Topics relevant to Skill Development:

3. An attitude of enquiry.
4. Write reports

The topics related to Human values and Professional ethics:

All topics in are relevant to Human values and Professional ethics.

Course Code: APT4002	Course Title: Introduction to Aptitude (Audited)	L- P- C	0	2	0
Version No.	1.0				
Course Pre-requisites	Students should know the basic Mathematics & aptitude along with understanding of English				
Anti-requisites	Nil				
Course Description	The objective of this course is to prepare the trainees to tackle the questions on various topics and various difficulty levels based on Quantitative Ability, and Logical Reasoning asked during the placement drives. There will be sufficient focus on building the fundamentals of all the topics, as well as on solving the higher order thinking questions. The focus of this course is to teach the students to not only get to the correct answers, but to get there faster than ever before, which will improve their employability factor.				
Course Objective	The objective of the course is to familiarize the learners with the concepts of Aptitude and attain Skill Development through Problem Solving techniques.				
Course Outcomes	On successful completion of the course the students shall be able to: CO1] **Recall** all the basic mathematical concepts they learnt in high school. CO2] **Identify** the principle concept needed in a question. CO3] **Solve** the quantitative and logical ability questions with the appropriate concept. CO4] **Analyze** the data given in complex problems. CO5] **Rearrange** the information to simplify the question				
Course Content:					
Module 1	Quantitative Ability	Assignment	Bloom's Level : Application		**12 Hours**
Topics: Introduction to Aptitude, working of Tables, Squares, Cubes					
Module 2	Logical Reasoning	Assignment	Bloom's Level : Application		**18 Hours**
Topics: Linear & Circular Arrangement Puzzle, Coding & Decoding, Blood Relations, Directions, Ordering and Ranking, Clocks and Calendars, Number Series, Wrong number series, Visual Reasoning					

Targeted Application & Tools that can be used: Application area: Placement activities and Competitive examinations. Tools: LMS						
Text Book 1. Quantitative Aptitude by R S Aggarwal 2. Verbal & Non-Verbal Reasoning by R S Aggarwal						
References 1. www.indiabix.com 2. www.youtube.com/c/TheAptitudeGuy/videos						
Topics relevant to Skill development: Quantitative and reasoning aptitude for Skill Development through Problem solving Techniques. This is attained through assessment component mentioned in course handout.						
Course Code: **APT4004**	**Course Title: Aptitude Training-Intermediate** Type of Course: Practical Only Course	L- T - P- C	0	0	2	0
Version No.	1.0					
Course Pre-requisites	Students should have the basic concepts of Quantitative aptitude along with its applications in real life problems.					
Anti-requisites	NIL					
Course Description	This is a skill-based training program for the students. This course is designed to enable the students to enhance their skills in Quantitative Aptitude.					
Course Objective	**The objective of the course is to familiarize the learners with the concepts of Aptitude and attain Skill Development through Problem Solving techniques.**					

<table>
<tr><td>Course Out Comes</td><td colspan="3">On successful completion of this course the students shall be able to:

CO1: Recall all the basic mathematical concepts.

CO2: Identify the principle concept needed in a question.

CO3: Solve the quantitative and logical ability questions with the appropriate concept.

CO4: Analyze the data given in complex problems.</td></tr>
<tr><td>Course Content:</td><td colspan="3"></td></tr>
<tr><td>Module 1</td><td>Quantitative Ability 1</td><td>Assignment</td><td>16 Hours</td></tr>
<tr><td colspan="4">Topics:

Number System, Percentage, Ratio and Proportion, Average, Mixture and Allegation, Time and Work, Profit and Loss</td></tr>
<tr><td>Module 2</td><td>Quantitative Ability 2</td><td>Assignment</td><td>14 Hours</td></tr>
<tr><td colspan="4">Topics:
Time Speed and Distance, Boats and Streams, Simple Interest, Compound Interest, Probability, Permutation and Combination</td></tr>
<tr><td colspan="4">Targeted Application & Tools that can be used:
Application area: Placement activities and Competitive examinations.
Tools: LMS</td></tr>
<tr><td colspan="4">Continuous Evaluation:</td></tr>
<tr><td colspan="4">CA1 – Online Test
CA2 – Online Test
CA3 – Online Test
Assignment</td></tr>
</table>

Text Book:

1. Fast Track Objective by Rajesh Verma
2. R S Aggarwal
3. Rakesh Yadav

References:

1. **www.indiabix.com**
2. **www.testbook.com**
3. **www.youtube.com/c/TheAptitudeGuy/videos**

Topics relevant to Skill Development: Quantitative aptitude for Skill Development through Problem solving Techniques. This is attained through components mentioned in course handout.

Course Code: APT4005	**Course Title:** Aptitude For Employability **Type of Course:** Practical Only	**L- T-P- C**	0	0	2	1

Version No.		1.0
Course Pre-requisites		Students should have the basic concepts of Quantitative aptitude, Verbal ability along with its applications in real life problems.
Anti-requisites		Nil
Course Description		This course is designed to enable the students to enhance their skills in quantitative aptitude and verbal ability skills.
Course Objective		The objective of the course is to familiarize the learners with concepts in Quantitative Aptitude and Verbal ability through problem solving techniques suitable for their career development.
Course Outcomes		On successful completion of the course the students shall be able to: CO1] **Recall** all the basic mathematical concepts CO2] **Identify** the principle concept needed in a question CO3] **Solve** the quantitative and logical ability questions with the appropriate concept.
Course Content:		

Module 1	Quantitative Ability	Lab-10hrs		Platform Assessment-10hrs	**20 Hours**

	Topics: Number System, Percentage, Ratio and Proportion, Average, Mixture and Allegation, Time and Work, Profit and Loss, Time Speed and Distance, Simple Interest and Compound Interest, Probability, Permutation and Combination.				
Module 2	Verbal Ability	Lab-5hrs		Platform Assessment-5hrs	**10 Hours**
	Topics: - Parts of Speech, Subject Verb Agreement, Spotting Error, Cloze Test, Verbal Analogies, Reading Comprehension, Idioms & Phrases, Para Jumbles				
	Targeted Application & Tools that can be used: Application area: Placement activities and Competitive examinations. Tools: LMS				
Evaluation	**Continuous Evaluation** • Topic wise evaluation				

	Text Book 1. Fast track objective by Rajesh Verma 2. R S Aggarwal 3. S.P Bakshi
	References 1. www.indiabix.com 2. www.testbook.com 3. www.youtube.com/c/TheAptitudeGuy/videos
	Topics relevant to Skill development: Quantitative and reasoning aptitude for Skill Development through Problem solving Techniques. This is attained through assessment component mentioned in course handout.

<table>
<tr><td>Course Code: APT4006</td><td colspan="3">Course Title: Logical and Critical Thinking

Type of Course: Audited</td><td>L- T-P- C</td><td>0</td><td>0</td><td>2</td><td>0</td></tr>
<tr><td>Version No.</td><td colspan="8">1.0</td></tr>
<tr><td>Course Pre-requisites</td><td colspan="8">Students should have the basic concepts of Logical reasoning and Critical thinking, along with its applications in real life problems.</td></tr>
<tr><td>Anti-requisites</td><td colspan="8">Nil</td></tr>
<tr><td>Course Description</td><td colspan="8">This is a skill-based training program for the engineering students (Undergraduate). This course is designed to enable the students to enhance their skills in Logical reasoning and Critical thinking.</td></tr>
<tr><td>Course Objective</td><td colspan="8">The objective of the course is to familiarize the learners with concepts in in Logical reasoning and Critical thinking through problem solving techniques suitable for their career development.</td></tr>
<tr><td rowspan="4">Course Outcomes</td><td colspan="8">On successful completion of the course the students shall be able to:</td></tr>
<tr><td colspan="8">CO1] Understand all the concepts.</td></tr>
<tr><td colspan="8">CO2] Apply the concepts in problem solving (Bloom’s taxonomy Level 3)</td></tr>
<tr><td colspan="8">CO3] Analyze and structure the reasoning techniques and spatial visualization skills</td></tr>
<tr><td colspan="9">Course Content:</td></tr>
<tr><td>Module 1</td><td colspan="3">Logical Thinking</td><td>Assignment</td><td colspan="3"></td><td>16 Hours</td></tr>
<tr><td rowspan="2"></td><td colspan="8">Topics:</td></tr>
<tr><td colspan="8">Syllogisms, Cubes and Dices, Mirror and Water images, Paper cutting and Folding, Embedded figures & Completion of figures, Data Interpretation, Data sufficiency</td></tr>
<tr><td>Module 2</td><td colspan="3">Critical Thinking</td><td>Assignment</td><td colspan="3"></td><td>14 Hours</td></tr>
<tr><td rowspan="2"></td><td colspan="8">Topics:</td></tr>
<tr><td colspan="8">Analogy, Symbol and Notations, Statement and assumption, Cause of action, Statement and conclusion, Puzzles</td></tr>
<tr><td rowspan="3"></td><td colspan="8">Targeted Application & Tools that can be used:</td></tr>
<tr><td colspan="8">Application area: Placement activities and Competitive examinations.</td></tr>
<tr><td colspan="8">Tools: LMS</td></tr>
<tr><td rowspan="3">Evaluation</td><td colspan="8">Continuous Evaluation</td></tr>
<tr><td colspan="8">· Topic wise evaluation</td></tr>
<tr><td colspan="8">· Internal Assessments</td></tr>
<tr><td></td><td colspan="8">Text Book</td></tr>
</table>

	1. A new approach to reasoning verbal, non-verbal & analytical by BS Sijwali
	2. R S Aggarwal
	3. Kiran publications
	References
	1. www.indiabix.com
	2. www.testbook.com
	3. www.youtube.com/c/TheAptitudeGuy/videos
	Topics relevant to Skill Development Logical reasoning and Critical thinking for Skill Development through Problem solving Techniques. This is attained through assessment component mentioned in course handout.

Course Code: PPS3018	**Course Title: Preparedness for Interview** Type of Course: Practical Only Course	L- T- P- C	0	0	2	1

Version No.		1.0
Course Pre-requisites		Students are expected to understand Basic English. Students should have desire and enthusiasm to involve, participate and learn.
Anti-requisites		NIL
Course Description		This course is designed to enable students to understand soft skills concepts to be corporate ready. The modules are set to improve self-confidence, communicate effectively and Prepare for the Interview to assist in employability. It helps the students to get a glimpse of the acceptable corporate readiness and equip them with the fundamental necessities of being able to confidently deal with the highly competitive corporate environment and helps in crafting different types of resumes. The pedagogy used will be group discussions, flipped classrooms, continuous feedback, role-play and mentoring.
Course Objective		The objective of the course is to familiarize the learners with the concepts of "**Preparing for Interview**" and attain SKILL DEVELOPMENT through PARTICIPATIVE LEARNING techniques.

Course Out Comes		**On successful completion of this course the students shall be able to:** CO1: **Develop** professional Resumes CO2: **Illustrate** Resumes effectively CO3: **Apply** skills and knowledge learnt for active and effective Group Discussions and Interview			
Course Content:					
Module 1		**Resume Building**	**Classroom activity**		10 Hours
	Topics: Resume structure, use of templates, Do's and Don'ts, ATS methods, Cover Letter and Video Resume **Activity:** Real world scenarios				
Module 2		**Group Discussion**	**Mock G D**		9 Hours
	Topics: -Group discussion as a placement process, GD techniques like Keyword. SPELT & POV of affected parties. Do & Don't of GD, Case-lets and topics for GD, practice session and evaluation **Activity:-** Real world scenarios				
Module 3		**Personal Interview**	**Grooming checks + Evaluation + Mock Interview+ Role Play**		9 Hours
	Topics: Placement process, Different interview rounds, HR interviews, Interview questions and desired answers, Different types of interviews, Do's and Don'ts. **Activity: -** Role Play & Real-world scenario				
Module 4		**Recap/Revision /Feedback Session**	**Practice sessions**		2 Hours
	Targeted Application & Tools that can be used: 1. TED Talks 2. You Tube Links 3. Role Play activities				
	Project work/Assignment: Mention the Type of Project /Assignment proposed for this course				
	Continuous Individual Assessment				

	The Topics related to Skill Development: Art Of Presentation and Group Discussion for **Skill Development** through **Participative Learning** Tech- niques. This is attained through assessment Component mentioned in course handout.

Course Code: CSE2255	**Course Title:** Object Oriented Programming Using Java **Type of Course: Theory**	**L-T- P-C**	3	0	0	3
Version No.	2.0					
Course Pre-requisites	CSEXXXX – Problem Solving Using C					
Anti-requisites	Nil					
Course Description	This course introduces the core concepts of object-oriented programming. This course has theory and lab component which emphasizes on understanding the implementation and application of object-oriented programming paradigm. It helps the student to build real time secure applications by applying these concepts and also for effective problem solving. The students interpret and understand the need for object oriented programming to build applications.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of Problem-Solving using JAVA and attain **SKILL DEVELOPMENT** through **EXPERIENTIAL LEARNING** techniques					
Course Out Comes	**On successful completion of the course the students shall be able to:** **CO1:** Describe the basic programming concepts. [Understand] **CO2:** Apply the concept of classes, objects and methods to solve problems. [Application] **CO3:** Apply the concept of arrays and strings. [Appy] **CO4:** Implement inheritance and polymorphism building secure applications. [Apply] **CO5:** Apply the concepts of interface and error handling mechanism. [Apply]					
Course Content:						
Module 1	**Basic Concepts of Programming and Java**	Assignment	Problem Solving	**9 Sessions**		
Topics: Introduction to Principles of Programming: Process of Problem Solving, Java program structure, Download Eclipse IDE to run Java programs, Sample program, Data types, Identifiers, Variables, Constants in java, Operators, Assignments and Expression, Basic Input/ Output functions, Control Statements: Branching and Looping.						
Module 2	**Classes, objects, methods and Constructors**	Assignment	Problem Solving	**10 Sessions**		

Topics: Classes, Objects and Methods: Introduction to object Oriented Principles, defining a class, adding data members and methods to the class, access specifiers, instantiating objects, reference variable, accessing class members and methods.
Static Polymorphism: Method overloading, constructors, constructor overloading, this keyword, static keyword, Nested classes, Accessing members in nested classes.

Module 3	**Arrays, String and String buffer**	Assignment	Problem Solving	**8 Sessions**

Topics: Arrays: Defining an Array, Initializing & Accessing Array, Multi –Dimensional Array, Array of objects. String: Creation & Operation. String builder class, methods in String Buffer.

Module 4	**Inheritance and Polymorphism**	Assignment	Problem Solving	**10 Sessions**

Topics: Inheritance: Defining a subclass, Types of Inheritance, super keyword. Dynamic Polymorphism: Method overriding. Final keyword: with data members, with member functions and with class. Abstract keyword: with data members, with member functions and with class, Exception handling.

Module 5	**Input & Output Operation in Java**	Assignment	Problem Solving	**8 Sessions**

Input/output Operation in Java(java.io Package), Streams and the new I/O Capabilities, Understanding Streams, working with File Object, File I/O Basics, Reading and Writing to Files, Buffer and Buffer Management, Read/Write Operations with File Channel, Serializing Objects, Observer and Observable Interfaces.

Text Book

T1 Herbert Schildt, "The Complete Reference Java 2", Tata McGraw Hill Education, 11th Edition,2019.

References

R1. Cay S Horstmann and Cary Gornell, "CORE JAVA volume I-Fundamentals", Tenth Edition, Pearson 2015.

R2: James W. Cooper, "Java TM Design Patterns – A Tutorial", Addison-Wesley Publishers.4th Edition, 2000.

R3. E. Balagurusamy, "Programming with Java", Tata McGraw Hill Education, 6th Edition, 2019.

E book link R1: http://rmi.yaht.net/bookz/core.java/9780134177373-Vol-1.pdf

E book link R2: Java(tm) Design Patterns: A Tutorial([PDF] [7qmsenjl97t0] (vdoc.pub)

Web **resources**

ps://youtube.com/playlist?list=PLu0W_9lII9agS67Uits0UnJyrYiXhDS6q

ps://puniversity.informaticsglobal.com:2229/login.aspx

Topics relevant to development of "Skill Development":

1. Static Polymorphism
2. Method overloading, constructors
3. constructor overloading
4. this keyword

5. static keyword and Inner classes
6. Inheritance and Polymorphism.

for **Skill Development** through **Experiential Learning** techniques. This is attained through assessment component mentioned in course handout.

Course Code: CSE2256	**Course Title:** Object Oriented Programming Using Java Lab **Type of Course: Lab**	**L-T- P-C**	0	0	4	2
Version No.	2.0					
Course Pre-requisites	CSEXXXX – Problem Solving Using C					
Anti-requisites	Nil					
Course Description	This course introduces the core concepts of object-oriented programming. This course has theory and lab component which emphasizes on understanding the implementation and application of object-oriented programming paradigm. It helps the student to build real time secure applications by applying these concepts and also for effective problem solving. The students interpret and understand the need for object oriented programming to build applications.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of Problem-Solving using JAVA and attain **SKILL DEVELOPMENT** through **EXPERIENTIAL LEARNING** techniques					
Course Out Comes	**On successful completion of the course the students shall be able to:** **CO1:** Demonstrate basic programming concepts. [Apply] **CO2:** Apply the concept of classes, objects and methods to solve problems. [Application] **CO3:** Apply the concept of arrays and strings. [Appy] **CO4:** Implement inheritance and polymorphism building secure applications. [Apply] **CO5:** Apply the concepts of interface and error handling mechanism. [Apply]					
Course Content:						
Module 1	**Basic Concepts of Programming and Java**	Assignment	Problem Solving	**12 Sessions**		
Download Eclipse IDE to run Java programs, Sample programs on Data types, Identifiers, Variables, Constants in java, Operators, Assignments and Expression, Basic Input/ Output functions, Control Statements: Branching and Looping.						
Module 2	**Classes, objects, methods and Constructors**	Assignment	Problem Solving	**14 Sessions**		
Problem solving using Classes, Objects and Methods: defining a class, adding data members and methods to the class, access specifiers, instantiating objects, reference variable, accessing class members and methods. Use Static Polymorphism: Method overloading, constructors, constructor overloading, this keyword, static keyword, Nested classes, Accessing members in nested classes.						

Module 3	**Arrays, String and String buffer**	Assignment	Problem Solving	**10 Sessions**
Using Arrays and Strings : Defining an Array, Initializing & Accessing Array, Multi –Dimensional Array, Array of objects. String: Creation & Operation. String builder class, methods in String Buffer.				
Module 4	**Inheritance and Polymorphism**	Assignment	Problem Solving	**12 Sessions**
Inheritance: Defining a subclass, Types of Inheritance, super keyword. Dynamic Polymorphism: Method overriding. Final keyword: with data members, with member functions and with class. Abstract keyword: with data members, with member functions and with class, Exception handling.				
Module 5	**Input & Output Operation in Java**	Assignment	Problem Solving	**12 Sessions**
Input/output Operation in Java(java.io Package), Streams and the new I/O Capabilities, Understanding Streams, working with File Object, File I/O Basics, Reading and Writing to Files, Buffer and Buffer Management, Read/Write Operations with File Channel, Serializing Objects, Observer and Observable Interfaces.				
P1: Programming Exercises on Basic Concepts. LEVEL 1: Discuss about datatypes and variables. LEVEL 2: Demonstrate a simple java program P2: Programming Exercises on Basic Concepts. LEVEL 1: Discuss about datatypes and variables. LEVEL 2: Demonstrate a simple java program P3: Programming Exercises on operators, expressions based on a given scenario. LEVEL 1: Explain operators, expressions. LEVEL 2: Demonstrate operators P4: Programming Exercises Command Line Arguments based on a given scenario. LEVEL 1: Explain command line arguments LEVEL 2: Demonstrate command line arguments P5: Programming Exercises on basic Input/ Output functions and Control Statements: Branching LEVEL 1: Explain Input/ Output functions LEVEL 2:Demonstrate Control Statements: Branching P6: Programming Exercises on Control Statements: Looping LEVEL 1: Explain variour loops. LEVEL 2:Demonstrate Control Statements: Looping P7: Programming Exercises on Creating Objects, classes on a given scenario. LEVEL 1: Illustrate class, object and methods. LEVEL 2: Execute java program using class and objects P8: Programming Exercises on Adding methods and Constructors to the class based on a given scenario. LEVEL 1: Illustrate methods and constructors LEVEL 2: Execute java program using methods and constructors P9: Programming Exercises on methods based on a given scenario. LEVEL 1: Illustrate method overloading LEVEL 2: Apply method overloading for the given scenario. P10: Programming Exercises on methods based on a given scenario. LEVEL 1: Illustrate constructors overloading				

LEVEL 2: Apply constructor overloading for the given scenario
P11: Programming Exercises on methods for static members bassed on a given scenario.
LEVEL 1: Benefits of usage static members
LEVEL 2: Usage of Static Members for the given scenario
P12: Programming Exercises on static methods based on a given scenario.
LEVEL 1: Benefits of usage static methods
LEVEL 2: Usage of Static Methods for the given scenario.
P13: Programming Exercises on nested Classes based on a given scenario.
LEVEL 1: Benefits of usage nested classes
LEVEL 2: Apply the concept of usage of nested classes for the given scenario
P14: Programming Exercises on Arrays and its built-in functions based on a given scenario.
LEVEL 1: Illustrate one dimensional arrays and its functions.
LEVEL 2: Demonstrate programs with single-dimensional arrays and operations.
P15: Programming Exercises on Arrays and its built-in functions based on a given scenario.
LEVEL 1: Illustrate multi dimensional arrays and its functions.
LEVEL 2: Demonstrate programs with multi-dimensional arrays and operations.
P16: Programming Exercises on String Class and its built-in functions based on a given scenario.
LEVEL 1: Explain about String class and String methods.
LEVEL 2: Execute simple java applications for String and StringBuffer operations
P17: Programming Exercises on String Buffer Class and its built-in functions based on a given scenario.
LEVEL 1: Explain about StringBuffer class and String methods.
LEVEL 2: Execute simple java applications for String and StringBuffer operations
P18: Programming Exercises on String Builders and its built-in functions based on a given scenario.
LEVEL 1: Explain about String Builders.
LEVEL 2: Execute java applications for String Builders
P19: Programming Exercises on single, multi level Inheritance and super keyword based on given scenario.
LEVEL 1: Explain single and multi level inheritance.
LEVEL 2: Demonstrate simple applications for the different types of inheritance
P20: Programming Exercises hierarchical Inheritance and super keyword based on given scenario.
LEVEL 1: Explain hierarchical inheritance.
LEVEL 2: Demonstrate simple applications for hierarchical inheritance
P21: Programming Exercises on Overriding.
LEVEL 1: Differentiate method overloading and method overriding.
LEVEL 2: Demonstrate simple program with dynamic method dispatch.
P22: Programming Exercises on Final based on given scenario.
LEVEL 1: Implement programs using concept of final.
LEVEL 2: Use final keyword for the given problem
P23: Programming Exercises on Abstract keyword based on given scenario.
LEVEL 1: Implement programs using concept of Abstract.
LEVEL 2: Use abstract keyword for the given problem
P24: Programming Exercises on Interface based on a given scenario.
LEVEL 1: Differentiate abstract class about interface
LEVEL 2: Implement interfaces in the given problem
P25: Programming Exercises on Exception Handling based on a given scenario.
LEVEL 1: Explain exception handling
LEVEL 2: Solve the given problem using exception handling mechanism.
P26: Programming Exercises on Character Stream Classes based on a given scenario.

LEVEL 1: Explain Character Stream Classes
LEVEL 2: Solve the given problem using Character Stream Class.
P27: Programming Exercises on Read/Write Operations with File Channel based on a given scenario.
LEVEL 1: Explain Read/Write Operations with File Channel
LEVEL 2: Solve the given problem using Read/Write Operations with File Channel.
P28: Programming Exercises on Read/Write Operations with File Channel based on a given scenario.
LEVEL 1: Explain Read/Write Operations with File Channel
LEVEL 2: Solve the given problem using Read/Write Operations with File Channel.
P29: Programming Exercises on Read/Write Operations with File Channel based on a given scenario.
LEVEL 1: Explain Read/Write Operations with File Channel
LEVEL 2: Solve the given problem using Read/Write Operations with File Channel.
P30: Programming Exercises on Read/Write Operations with File Channel based on a given scenario.
LEVEL 1: Explain Read/Write Operations with File Channel
LEVEL 2: Solve the given problem using Read/Write Operations with File Channel.

Targeted Application & Tools that can be used : JDK /Eclipse IDE/Visual Studio Code / net Beans IDE.

Text Book

T1 Herbert Schildt, "The Complete Reference Java 2", Tata McGraw Hill Education, 11th Edition,2019.

References

R1. Cay S Horstmann and Cary Gornell, "CORE JAVA volume I-Fundamentals", Tenth Edition, Pearson 2015.

R2: James W. Cooper, "Java TM Design Patterns – A Tutorial", Addison-Wesley Publishers.4th Edition, 2000.

R3. E. Balagurusamy, "Programming with Java", Tata McGraw Hill Education, 6th Edition, 2019.

E book link R1: http://rmi.yaht.net/bookz/core.java/9780134177373-Vol-1.pdf

E book link R2: Java(tm) Design Patterns: A Tutorial([PDF] [7qmsenjl97t0] (vdoc.pub)

Web **resources**

ps://youtube.com/playlist?list=PLu0W_9lII9agS67Uits0UnJyrYiXhDS6q

ps://puniversity.informaticsglobal.com:2229/login.aspx

Topics relevant to development of "Skill Development":

1. Static Polymorphism
2. Method overloading, constructors
3. constructor overloading
4. this keyword
5. static keyword and Inner classes
6. Inheritance and Polymorphism.

for **Skill Development** through **Experiential Learning** techniques. This is attained through assessment component mentioned in course handout.

Course Code: DES1146	**Course Title:** Introduction to Design Thinking **Type of Course:** Theory		**L-T-P- C**	1	0	0	1
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	NIL					
Course Description	The course aims to introduce students to the fundamental principles and processes of Design Thinking and will learn to apply Design Thinking methodologies to real-world challenges. The course emphasizes empathy, creativity, and collaboration, equipping students with essential skills for successful engineering practice.					
Course Objective	This course is designed to develop and familiarize the learners with the concepts of creating thinking and attain Entrepreneurship by using Participative Learning techniques.					
Course Outcomes	On successful completion of the course the students shall be able to: 1) Understand the concept and importance of Design Thinking. 2) Differentiate between traditional problem-solving and Design Thinking. 3) Identify the core stages of the Design Thinking process.					
Course Content:	All assignments and projects must be developed using the reference materials available from the PU e-resource database – JSTOR, EBSCO, Library OPAC, NPTEL Videos, etc.					
Module 1	Introduction to Design Thinking	Visual journal, book of essays, context-specific assignment/project	Visual output generation, by Visual Journal and narrative development.			**3 hours**
Topic 1) Definition and Introduction to Design Thinking 2) Understand the Design Thinking Process						
Module 2	Design Thinking in Action	Visual journal, book of essays, context-specific assignment/project	Visual output generation, by visual journal and narrative development.			**12 hours**

Topics:

1) Introduction to the steps of Design Thinking Process
2) Understand use cases of Design thinking
3) Design Thinking and Research Tools pertaining to Consumer Tech. , Home Tech. , Personal Tech. , Auto Tech. or Extended Reality.

Targeted Application & Tools that can be used:

1) Design ideation tools like Miro , SCAMPER etc.
2) Research Tools for Human Centric Design using forecasting tools like WGSN
3) Feedback tools like Google Forms , etc.
4) Expert Lectures

1. **Text Book**
2. Thinking Design by S Balaram. New Delhi [India]: Sage Publications Pvt. Ltd. 2010. eBook., Database: eBook Collection (EBSCOhost)
3. https://puniversity.informaticsglobal.com:2284/ehost/detail/detail?vid=6&sid=18ab1f43-1f92-4d02-ae2e-a9c06dc06d8c%40redis&bdata=JnNpdGU9ZWhvc3QtbGl2ZQ%3d%3d#AN=354920&db=nlebk

References

1. Design Thinking by Clarke, Rachel Ivy. Series: Library Futures, Vol. 4. Chicago: ALA Neal-Schuman. 2020. eBook., Database: eBook Collection (EBSCOhost)
2. https://puniversity.informaticsglobal.com:2282/ehost/detail/detail?vid=4&sid=c80a7d79-eda4-4b7e-a0d6-afafe437962b%40redis&bdata=JnNpdGU9ZWhvc3QtbGl2ZQ%3d%3d#AN=2433506&db=nlebk
3. The Pocket Universal Methods of Design: 100 Ways to Research Complex Problems, Develop Innovative Ideas, and Design Effective Solutions by Bruce Hanington; Bella Martin. Minneapolis: Rockport Publishers. 2017. eBook., Database: eBook Collection (EBSCOhost)

https://puniversity.informaticsglobal.com:2282/ehost/detail/detail?vid=11&sid=f086b8c2-260e-4caa-8c48-d732c21a7724%40redis&bdata=JnNpdGU9ZWhvc3QtbGl2ZQ%3d%3d#AN=1638693&db=nlebk

4. What Is Design Thinking and Why Is It Important? By Rim Razzouk and Valerie Shute - Review of Educational Research, Vol. 82, No. 3 (September 2012), pp. 330-348 (19 pages), Published by: American Educational Research Association

https://puniversity.informaticsglobal.com:2054/stable/23260048?Search=yes&resultItemClick=true&searchText=design+thinking&searchUri=%2Faction%2FdoBasicSearch%3FQuery%3Ddesign%2Bthinking%26so%3Drel&ab_segments=0%2FSYC-6168%2Ftest&refreqid=fastly-default%3Acb1be24976e25734cb5fc13a8af6fdfb&seq=1#metadata_info_tab_contents

5. Abductive Thinking and Sensemaking: The Drivers of Design Synthesis by John Kolko, Design Issues, Vol. 26, No. 1 (Winter, 2010), pp. 15-28 (14 pages), Published by: The MIT Press

https://puniversity.informaticsglobal.com:2054/stable/20627839?Search=yes&resultItemClick=true&searchText=design+thinking&searchUri=%2Faction%2FdoBasicSearch%3FQuery%3Ddesign%2Bthinking%26so%3Drel&ab_segments=0%2FSYC-6168%2Ftest&refreqid=fastly-

default%3A0b89336ea274d63c010536b01316d7bb&seq=1#metadata_info_tab_contents
6. Designerly Ways of Knowing: Design Discipline versus Design Science by Nigel Cross, Design Issues, Vol. 17, No. 3 (Summer, 2001), pp. 49-55 (7 pages), Published by: The MIT Press
https://puniversity.informaticsglobal.com:2054/stable/1511801?Search=yes&resultItemClick=true&searchText=design+thinking&searchUri=%2Faction%2FdoBasicSearch%3FQuery%3Ddesign%2Bthinking%26so%3Drel&ab_segments=0%2FSYC-6168%2Ftest&refreqid=fastly-default%3A0d5b607b163f60876ca973ed90e22b1c&seq=1#metadata_info_tab_contents