PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

Program Regulations and Curriculum

2026-2030

BACHELOR OF TECHNOLOGY (B.Tech.) in

INFORMATION SCIENCE AND TECHNOLOGY - IST

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

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 CreditSystem (CBCS) with focus on Social Project Based Learning, Industrial Training, and Internship toenable 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 2026-2030.
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 2026-2030 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 2026-2027.

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, 2026-2030;*

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 2026-2030 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 2026-2030 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 inArtificial 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,

2026-2030, 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**

Table 1: Assessment Components and Weightage for different category of Courses		
Nature of Course and Structure	**Evaluation Component**	**Weightage**
Lecture-based Course	Continuous	50%

L component in the L-T-P Structure is predominant (more than 1) (Examples: 3-0-0; 3-0-2; 2-1-0; 2-0-2, 2-0-4 etc.)	Assessments	
	End Term Examination	50%
Lab/Practice-based Course P component in the L-T-P Structure is predominant (Examples: 0-0-4; 1-0-4; 1-0-2; etc.)	Continuous Assessments	75%
	End 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	Guidelines 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 **Error! Reference source not found.** 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.

12.6 Minimum Performance Criteria:

12.6.1 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.

12.6.2 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.

12.6.3 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 12.6.1 and 12.6.2 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:

13.1 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.

13.2 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.

13.3 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:

13.3.1 A student may complete SWAYAM/NPTEL/other approved MOOCs as mentioned in Clause13.3 (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.

13.3.2 SWAYAM/NPTEL/ other approved MOOCs as mentioned in Clause 13.3 (as per Academic Regulations) shall be approved by the concerned Board of Studies and placed (as Annexures) in the concerned PRC.

13.3.3 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.

13.3.4 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.

13.3.5 A student shall request for transfer of credits only from such approved Courses as mentioned in Sub-Clause 13.3.2 above.

13.3.6 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.

13.3.7 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.

13.3.8 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

13.3.9 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 forthe award ofthe concernedDegree.

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

13.4 Themaximumnumberofcreditsthatcanbetransferredbyastudentshallbelimitedtoforty percent (40%) of the mandatory minimum credit requirements specified by the concerned Program Regulations and Curriculum forthe award ofthe concernedDegree. 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.

13.5 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 (2026-2030) 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) 2026-2030: 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)	09
2	Basic Science Courses (BSC)	24
3	Engineering Science Courses (ESC)	23
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	ENG1030	Fundamentals of Communication	1	0	2	2	2	S	-
2	ENG2030	Communication in Practice	1	0	2	2	2	S	-
3	FIN1002	Essentials of Finance	3	0	0	3	3	S	-
4	APT4005	Aptitude for Employability	0	0	2	1	2	S/EM	-
5	PPS3018	Preparedness for Interview	0	0	2	1	2	S	-
Total No. of Credits						**09**			

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	MAT1211	Linear Algebra, Differential Equations and Statistical Methods	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	MAT1212	Mathematical foundations	3	1	0	4	4	F	-

		for Machine Learning and Data Analysis							
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	MAT2311	Vector Calculus, Numerical Methods and Sampling Distributions	3	1	0	4	4	EM	-
8	MAT2321	Discrete Mathematical Structures	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	EEE1251	Basics of Electrical and Electronics Engineering Lab	0	0	2	1	2	F/S	-
8	ECE1512	Design Thinking in Practice	2	0	2	3	4	F	-
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						**23**			

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	3	0	0	3	3	S	-

		and Architecture							
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	CSE2509	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	IST2505	Ethical Hacking	2	0	0	2	2	S/EM	CSE2251
21	IST2506	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	IST2002	Applied Data Science	3	0	0	3	3	S/EM	-
25	IST2003	Applied Data Science Lab	0	0	2	1	2	S/EM	-
26	CSE2287	Software Design and Development	3	0	0	3	3	S	-
27	CSE2285	Cloud Computing	2	0	0	2	2	S/EM	-
28	CSE2286	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 **	1	0	0	0
3	LAW7601	Indian Constitution	1	0	0	0
4	PPS1026	Industry Readiness Program – II	0	0	2	0
5	CIV7601	Universal Human Values and Ethics	1	0	0	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 4th and 5th 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 awardedas 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 4th and 5th Semesters or 6th and 7th Semesters or during the 5th / 6th / 7th 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 7th / 8th 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									
S.No.	Course Code	Course Name	L	T	P	C	ontact Hours	ype of Skills	re-equi ites
1	IST2501	Optimization Methods in Machine Learning	3	0	0	3	3	S	SE2264
2	CSE3426	Front End Full Stack Development *	2	0	2	3	4	S/ EM	SE2258
3	CSE3427	Java Full Stack Development *	2	0	2	3	4	S/ EM	SE2258
4	CSE342	.Net Full Stack Development	2	0	2	3	4	S/ EM	SE2

	8	*							258
5	CAI3411	Generative AI	2	0	2	3	4	S/ EM	SE2264
6	CAI3427	Language Models for Text Mining +	2	0	2	3	4	S/ EM	SE2264
7	IST3412	Business Intelligence and Analytics	3	0	0	3	3	S/ EM	SE2260
8	CAI3428	Practical Deep Learning with TensorFlow +	2	0	2	3	4	S/ EM	SE2264
9	CIT2504	AI and Deep Learning for IoT	3	0	0	3	3	S/ EM	SE2264
10	CAI3429	Deep Learning Techniques for Computer Vision +	2	0	2	3	4	S/ EM	SE2264

rack 2 – Big Data and Data Science

Sl. No.	Course Code	Course Name	L	T	P	C	ontact Hours	ype of Skills	Pre-equisites
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	/ EM	SE2258
3	CSE3427	Java Full Stack Development *	2	0	2	3	4	/ EM	SE2258
4	CSE3428	.Net Full Stack Development *	2	0	2	3	4	/ EM	SE2258
5	IST3410	Data Handling and Visualization	2	0	2	3	4	/ EM	-
6	CAI3427	Language Models for Text Mining +	2	0	2	3	4	/ EM	SE2264
7	CDV3408	Predictive Analytics for System Monitoring & Performance Optimization	3	0	0	3	3	/ EM	-
8	CSD3411	Cybersecurity and Data Privacy	3	0	0	3	3	/ EM	SE2251
9	CAI3428	Practical Deep Learning with TensorFlow +	2	0	2	3	4	/ EM	SE2264
10	CBD3403	Cloud Storage & Data Management	3	0	0	3	3	/ EM	BD2000
11	CAI3429	Deep Learning Techniques for Computer Vision +	2	0	2	3	4	/ EM	SE2264

Track 3 – Block Chain and Cyber Security

Sl. No.	Course Code	Course Name	L	T	P	C	ontac Hours	ype of Skills	Pre-equisites
1	CCS2503	Cyber Security	3	0	0	3	3	S/ EM	SE2251
2	CCS2506	Intrusion Detection and Prevention System	3	0	0	3	3	S/ EM	SE2251
3	CSE3426	Front End Full Stack Development *	2	0	2	3	4	S/ EM	SE2258
4	CSE3427	Java Full Stack Development *	2	0	2	3	4	S/ EM	SE2258
5	CSE3428	.Net Full Stack Development *	2	0	2	3	4	S/ EM	SE2258
6	CCS3400	Digital and Mobile Forensics	2	0	2	3	4	S/ EM	SE2251
7	CAI3427	Language Models for Text Mining +	2	0	2	3	4	S/ EM	SE2017

8	CCS3405	Vulnerability Assessment and Penetration Testing	3	0	0	3	3	S/ EM	SE2269
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	SE2264
11	CCS2509	Malware Analysis	3	0	0	3	3	S/ EM	SE2251
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	SE2264

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	CSE2258
4	CSE3427	Java Full Stack Development *	2	0	2	3	4	S/ EM	CSE2258
5	CSE3428	.Net Full Stack Development *	2	0	2	3	4	S/ EM	CSE2258
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	CSE2264
8	IST3411	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	CSE2264
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	CSE2264
Special Track									
1	CSE3450	Introduction to NVIDIA Accelerated Computing and AI	2	0	2	3	3	S/ EM	-
2	CSE3451	NVIDIA Applied Artificial Intelligence and Generative AI	2	0	2	3	3	S/ EM	CSE3157
3	CSE3452	Advanced AI Systems and Generative Intelligence using NVIDIA Platforms	2	0	2	3	3	S/ EM	CSE3157
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 MOOC (NPTEL) Courses

20.1 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:

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

20.1.2 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.

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

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

20.1.5 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.

20.1.6 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.

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

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

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

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

Sl#	Duration	Credits	Evaluation
1.	12 weeks	3	Continuous Assessment –50 Marks Mid Term –50 Marks End Term-100 Marks
2.	8 weeks	2	Mid Term-50 Marks End Term-100 Marks
3	4 weeks	1	End Term-100 Marks

21. 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	Basket
Semester 1 - Chemistry Cycle						**19**		
1	MAT1211	Linear Algebra, Differential Equations and Statistical Methods	3	1	0	4	F	BSC
2	CIV1200	Foundations of Integrated Engineering	2	0	0	2	S	ESC
3	CHE2501	Chemistry of Smart Materials	3	0	0	3	S	BSC
4	CSE1500	Computational Thinking using Python	2	0	2	3	S	ESC
5	ENG1030	Fundamentals of Communication	1	0	2	2	S	HSMC
6	EEE1200	Basics of Electrical and Electronics Engineering	3	0	0	3	F/S	ESC
7	CHE2502	Chemistry of Smart Materials Lab	0	0	2	1	S	BSC
8	PPS1025	Industry Readiness Program – I	0	0	2	0	S	MAC
9	EEE1251	Basics of Electrical and Electronics Engineering Lab	0	0	2	1	F/S	ESC
Semester 2 – Physics Cycle						**22**		
1	MAT1212	Mathematical foundations for Machine Learning and Data Analysis	3	1	0	4	F	BSC
2	MEC1006	Engineering Graphics	2	0	0	2	S	ESC
3	PHY2501	Optoelectronics and Quantum Physics	3	0	0	3	F/S	BSC
4	CSE2200	Problem Solving using C	2	0	0	2	S	PCC
5	ENG2030	Communication in Practice	1	0	2	2	S	HSMC
6	ECE2022	Digital Design	2	0	0	2	F/S	ESC
7	CSE2201	Problem Solving using C Lab	0	0	4	2	S	PCC
8	PHY2504	Optoelectronics and Quantum Physics Lab	0	0	2	1	F/S	BSC
9	PPS1026	Industry Readiness Program – II	0	0	2	0	S	MAC
10	ECE2052	Digital Design Lab	0	0	2	1	F/S	ESC
11	ECE1512	Design Thinking in Practice	2	0	2	3	F	ESC

Sl. No.	Course Code	Course Name	L	T	P	C	Type of Skill/ Focus	Basket
Semester 3						**23**		
1	MAT2311	Vector Calculus, Numerical Methods and Sampling Distributions	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	APT4002	Introduction to Aptitude	0	0	2	0	AT	MAC
12	CHE7601	Environmental Studies	1	0	0	0	F	MAC
Semester 4						**24**		
1	MAT2321	Discrete Mathematical Structures	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	CSE2509	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
12	CIV7601	Universal Human Values and Ethics*	1	0	0	0	F	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	IST2505	Ethical Hacking	2	0	0	2	S/EM	PCC
4	IST2506	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	IST2002	Applied Data Science	3	0	0	3	S/EM	PCC
8	IST2003	Applied Data Science Lab	0	0	2	1	S/EM	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	AT	MAC
12	LAW7601	Indian Constitution	1	0	0	0	F	MAC
Semester 6						**22**		
1	CSE2287	Software Design and Development	3	0	0	3	S	PCC
2	CSE2285	Cloud Computing	2	0	0	2	S/EM	PCC
3	CSE2286	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/EM	PEC
8	CSEXXXX	Professional Elective – III	2	0	2	3	S/EM	PEC
9	XXXXXXX	Open Elective – I	3	0	0	3	S	OEC
10	APT4005	Aptitude for Employability	0	0	2	1	AT	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	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

Note: MAC course offering are subject to change

22. Course Catalogue

Course Catalogue of all Courses Listed including the Courses Offered by other School / Department and Discipline / Programme Electives – Course Code, Course Name, Prerequisite, Anti-requisite, Course Description, Course Outcome, Course Content (with Blooms Level, CO, No. of Contact Hours), Reference Resources.

Humanities and Social Sciences including Management Courses (HSMC)

Course ode: APT4005	**Course Title: Aptitude For Employability Type of Course: Practical Only (HSMC)**	**L- T-P- C**	**0**	**0**	**2**	**1**	
Version No.	1.0						
CoursePre-requisites	StudentsshouldhavethebasicconceptsofQuantitativeaptitude, Verbal ability along with its applications in real life problems.						
Anti-requisites	NIL						
CourseDescription	Thiscourseisdesignedtoenablethestudentstoenhancetheirskillsin quantitative aptitude and verbal ability skills.						
CourseObjective	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.						
CourseOutcomes	On successful completion of the course, students shall be able to: **CO1:** Recall fundamental mathematical concepts. [Remember] **CO2:** Identify the appropriate principle required to solve a given problem. [Understand] **CO3:** Apply suitable concepts to solve quantitative and logical reasoning problems. [Apply]						
CourseContent:							
Module1	**QuantitativeAbility**						**20 Sessions**
Topics: Number System, Percentage, Ratio andProportion,Average,MixtureandAllegation,Time andWork, Profit and Loss, Time Speed and Distance, Simple Interest and Compound Interest, Probability, Permutation and Combination.							

Module2	**VerbalAbility**	**10 Sessions**
Topics:-PartsofSpeech,SubjectVerbAgreement,SpottingError,ClozeTest,VerbalAnalogies, ReadingComprehension,Idioms&Phrases,ParaJumbles.		
TargetedApplication&Toolsthatcanbeused: Applicationarea:PlacementactivitiesandCompetitiveexaminations. Tools: LMS		
Evaluation	**ContinuousEvaluation** Topicwise evaluation	
TextBook 1. FasttrackobjectivebyRajesh Verma 2. R S Aggarwal 3. S.PBakshi		
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 DevelopmentthroughProblemsolvingTechniques.Thisisattainedthroughassessment componentmentionedincoursehandout.		

Course Code: FIN1002	**Course Title: Essentials of Finance** **Type of Course: Theory Course (HSMC)**	**L-T-P-C**	**3**	**0**	**0**	**3**
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	NIL					
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	To develop a comprehensive understanding of financial management and taxation by enabling students to understand forms of business organization and their financial implications, analyze financial principles influencing decision-making, interpret financial statements to assess organizational performance, and determine tax liability under various heads of income as per the Income Tax Act, 1961.					
Course Outcomes	On successful completion of this course, students will be able to: **CO1:** Explain the basic concepts of finance, financial markets, and financial institutions. [Understand] **CO2:** Apply financial concepts and tools for effective business decision-making. [Apply] **CO3:** Analyze and interpret financial statements to assess the financial performance of					

	an organization. [Analyze] **CO4:** Identify various heads of income and applicable deductions under the Income Tax Act, 1961. [Understand] **CO5:** Compute taxable income and determine tax liability based on relevant financial data and regulations. [Apply]	
Course Content:		
Module 1	**Introduction to Finance**	**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**	**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**	**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: PPS3018	**Course Title: Preparedness for Interview** **Type of Course: Practical Only Course (HSMC)**	**L- T- P- C**	**0**	**0**	**2**	**1**
Version No.	1.0					
Course Pre-requisites	StudentsareexpectedtounderstandBasicEnglish. Studentsshouldhavedesireandenthusiasmtoinvolve,participateandlearn.					
Anti-requisites	NIL					
Course Description	Thiscourseisdesignedtoenablestudentstounderstandsoftskillsconceptsto be corporate ready. The modules are set to improve self-confidence, communicate effectively and Prepare for the Interview to assist in employability.Ithelpsthestudentstogetaglimpseoftheacceptablecorporate 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,flippedclassrooms,continuousfeedback,role-playandmentoring.					
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, students shall be able to: **CO1:** Develop professional resumes. [Apply] **CO2:** Illustrate and present resumes effectively. [Understand] **CO3:** Apply acquired skills and knowledge for effective participation in group discussions and interviews. [Apply]					
Course Content:						
Module 1	**ResumeBuilding**					**10 Sessions**
Topics: Resume structure,use of templates, Do's and Don'ts, ATS methods, Cover Letter and Video Resume. **Activity:**Realworldscenarios						
Module 2	**GroupDiscussion**					**9 Sessions**
Topics:-Groupdiscussionas aplacement process,GDtechniqueslike Keyword.SPELT&POVofaffected parties.Do&Don'tofGD,Case-letsandtopicsforGD,practicesessionandevaluation **Activity:-**Realworldscenarios						
Module 3	**PersonalInterview**					**9 Sessions**
Topics: Placementprocess,Differentinterviewrounds,HRinterviews,Interviewquestionsanddesired answers, Different types of interviews, Do's and Don'ts. **Activity:-**RolePlay&Real-worldscenario						
Module 4	**Recap/Revision/FeedbackSession**					**2 Sessions**

TargetedApplication&Toolsthatcanbe used: 1. TED Talks 2. YouTube Links 3. RolePlayactivities
Projectwork/Assignment:MentiontheTypeofProject/Assignmentproposedforthiscourse
ContinuousIndividualAssessment
TheTopicsrelatedtoSkillDevelopment: ArtOfPresentationandGroupDiscussionfor**SkillDevelopment**tthrough**ParticipativeLearning** Techniques. This is attained through assessment Component mentioned in course handout.

List of Basic Science Courses (BSC)

<table>
<tr><td>Course Code: CHE2501</td><td>Course Title: Chemistry of Smart Materials
Type of Course: Theory (BSC)</td><td>L-T- P- C</td><td>3</td><td>0</td><td>0</td><td>3</td></tr>
<tr><td>Version No.</td><td colspan="6">1.0</td></tr>
<tr><td>Course Pre-requisites</td><td colspan="6">NIL</td></tr>
<tr><td>Anti-requisites</td><td colspan="6">NIL</td></tr>
<tr><td>Course Description</td><td colspan="6">Theobjective 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 materialsand 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</td></tr>
<tr><td>Course Objective</td><td colspan="6">The objective of the course is ‘SKILL DEVELOPMENT’ of the student by using Participative learning techniques.</td></tr>
<tr><td>Course Outcomes</td><td colspan="6">On successful completion of this course, students shall be able to:
CO1:Apply the knowledge of computational chemistry for molecular
structure, geometry and properties prediction. [Understand]
CO2: Explain the role of organic, polymeric and hybrid materials in memory
systems and photo and electroactive materials in display systems. [Understand]
CO3: Explain the properties of Nanomaterials associated with sensors and
smart devices. [Understand]
CO4: Apply the knowledge of fundamentals of green chemistry and
sustainable chemistry for E- waste management. [Apply]
CO5: Analyze the principles of electrochemistry for performance evaluation
of batteries, supercapacitors and photovolatics for sustainable
technologies. [Analyze]</td></tr>
<tr><td colspan="7">Course Content:</td></tr>
<tr><td>Module 1</td><td colspan="4">Computational Chemistry</td><td colspan="2">09 Sessions</td></tr>
<tr><td colspan="7">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.</td></tr>
<tr><td>Module 2</td><td colspan="4">Materials for Memory and Display Systems</td><td colspan="2">09 Sessions</td></tr>
</table>

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	**Nanomaterials based Smart Sensors and Devices**	**09 Sessions**

Topics: Nanomaterials- Introduction,classificationbasedondimensionality,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**	**09 Sessio**

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 toBiomaterials- 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**	**09 Sessions**

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. 1. Essentials of computational chemistry: theories and models, Christopher J Cramer, 2013, John Wiley & Sons. ISBN: 978-0-470-09182-1.
2. F. Jensen, Introduction to Computational Chemistry, 3rd edition, Wiley, 2017.
3. Functional and smart materials, Chander Prakash, Sunpreet Singh, J. Paulo Davim, 2020, CRC Press, ISBN: 978-036-727-510-5.
4. 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.
5. 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.
6. 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.
7. Fundamentals of analytical chemistry: An introduction, Douglas A. Skooget etal., 2004 Thomson Asia pte Ltd., 8th, ISBN: 978-0-495-55828-6.

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 components are relevant to skill development through Participative Learning techniques. This is attained through assessment components mentioned in course plan.

Course Code: CHE2502	**Course Title: Chemistry of Smart Materials Lab** **Type of Course: Laboratory (BSC)**	**L-T-P- C**	**0**	**0**	**2**	**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	On successful completion of the course, students shall be able to: **CO1:** Determine the amount of acids in domestic and industrial waste using conductometric and pH sensors. [Remember] **CO2:** Calculate the amount of metal ions in electronic waste using potentiometric and optical sensors. [Apply] **CO3:** Examine the viscoelastic property of the organic solvents using viscometry method. [Analyze] **CO4:** Apply the appropriate titration method to quantify the metal ions in industrial waste. [Apply] **CO5:** Explain the preparation, identification and recovery of industrially important materials using relevant techniques [Understand]
Course Content: **Total 30 sessions**	
Experiment 1	
Determination of strength of strong acid in battery electrolyte using conductometric sensors.	
Experiment 2	
Determination of pKa of organic acid of battery electrolyte using pH sensor.	
Experiment 3	
Estimation of copper from PCBs by using colorimeter (Optical Sensor).	
Experiment 4	
Determination of viscosity coefficient of a given organic liquid using Ostwald's Viscometer (viscoelastic property).	
Experiment 5	
Estimation of iron in electronic devices using Std. Potassium permanganate solution.	

Experiment 6

Conductometric estimation of mixture of acids in recycling process of E- waste.

Experiment 7

Estimation of iron from e-waste using Electrochemical sensors.

Experiment 8

Estimation of valuable metals (copper) from e- waste by Iodometric titration.

Experiment 9

Flame photometric estimation of sodium (battery recycling- Optical Sensor).

Experiment 10

Synthesis of conducting polyaniline for gas sensor applications (Demonstration experiment).

Experiment 11

Green synthesis of nanomaterials (Demonstration experiment).

Experiment 12

Recovery of valuable metals from e- waste by eletroless method (Demonstration experiment).

Any 10 experiments will be conducted out of 12

Continuous Internal Assessment:

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

Text 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

References

1. Engineering Chemistry Laboratory Manual (English, Paperback, Dr Manoj Kumar Solanki), Educreation 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
Topics relevant to SKILL DEVELOPMENT: All the experiments are relevant to Skill Development through Experiential Learning Techniques. This is attained through assessment component mentioned in course plan.

Course Code: MAT1211	**Course Title: Linear Algebra, Differential Equations and Statistical Methods** **Type of Course:1] School Core**	**L-T- P- C**	**3**	**1**	**0**	**4**
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	NIL					
Course Description	This course introduces fundamental mathematical concepts from linear algebra, differential calculus, ordinary differential equations and statistics. The content includes matrices, systems of linear equations, eigenvalues and eigenvectors, functions of several variables, differential equations, correlation and regression analysis, and curve fitting techniques. The course combines theoretical developments with illustrative applications drawn from scientific and engineering settings. The course emphasizes the application of these mathematical tools in solving problems in science and engineering. It extends student's problem-solving abilities through analytical and computational approaches, with focus on both theoretical understanding and applications.					
Course Objective	The objective of this course is to provide students with a strong mathematical foundation required for engineering and scientific studies. The course introduces fundamental concepts from linear algebra, differential calculus, ordinary differential equations, and statistical methods that are widely used in modeling, analysis and problem solving. It aims to familiarize students with mathematical techniques for representing engineering systems, studying relationships among variables, describing dynamic processes and interpreting data. The course serves as a foundation to prepare students for advanced engineering subjects that rely on mathematical modeling, computation and analytical methods.					
Course Out Comes	On successful completion of the course the students shall be able to: **CO1:**Use linear algebra concepts to evaluate properties of matrices and quadratic forms. (L3) **CO2:** Apply matrix-based methods to solve systems of linear equations. (L3) **CO3:** Use fundamental theorems and analytical techniques of differential calculus to single variable functions. (L3) **CO4:** Apply the concepts of partial derivatives to multivariable functions. (L3) **CO5:** Solve first order ordinary differential equations using analytical methods.(L3) **CO6:**Employ statistical techniques for data analysis and to interpret					

	relationshipsbetween variables. (L3)

Course Content:

Module 1	**Fundamentals of Linear Algebra**	**(15L+5T=20 Sessions)**

Matrices, echelon forms, rank of matrix, Systems of linear equations: Consistency, solution to system of linear homogeneous and non-homogeneous equations- Gauss elimination method, Gauss-Jordan method (Self-study), LU decomposition method, Network analysis using linear systems, Design of traffic patterns, Electrical circuits.

Eigenvalue and eigenvector, diagonalization of symmetric matrices, quadratic forms.

Module 2	**Differential Calculus**	**(15L+5T=20 Sessions)**

Introduction, limit, continuity, Rolle's theorem, mean value theorems, Indeterminate forms and L'Hôpital's rule.

Partial derivatives, Euler's theorem, total derivative, Jacobians, Taylor's and Maclaurin's theorems, maxima, minima and saddle points, method of Lagrange multipliers, Current Growth in an RL Circuit, Velocity Estimation, Fuel consumption optimization, Cost function optimization.

Module 3	**Ordinary Differential Equations**	**(6L+2T=8 Sessions)**

Definition, types of differential equations-variable separable, homogeneous, exact, linear and Bernoulli's equations, Equations not of first degree: equations solvable for p, equations solvable for y, equations solvable for x and Clairaut's type, Newton's law of cooling, Current in RL circuit.

Module 4	**Statistical Methods and Curve Fitting**	**(9L+3T=12 Sessions)**

Measures of central tendency, moments, skewness, and kurtosis. Correlation - Karl Pearson's coefficient of correlation, rank correlation, multiple correlation. Regression analysis - lines of regression, multiple regression. Principle of least square, curve fitting- straight line, parabola and exponential curves, Stress and strain curve fitting, Feature selection and Prediction models.

Targeted Applications &Tools that can be used:

- Linear Algebra is applied to solve systems of equations in circuits, traffic flow, and network analysis, with eigenvalues and quadratic forms aiding stability and optimization studies.
- Differential Calculus supports process speed analysis, velocity estimation, and optimization techniques such as minimizing fuel consumption and cost functions.
- Differential Equations model dynamic behaviours like current in RL circuits, heat transfer in materials, and growth or decay in environmental and nuclear engineering.
- Statistical Methods and Curve Fitting enable effective data analysis through correlation, regression, and curve fitting, supporting predictive modeling and interpretation of experimental data such as stress-strain relationships.

Software: MATLAB, Python, R Programming, Mathematica, Scilab, Octave

Assignment:

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. B. S. Grewal, *Higher Engineering Mathematics*, 44th ed. New Delhi, India: Khanna Publishers, 2017.
2. D. C. Montgomery and G. C. Runger, *Applied Statistics and Probability for Engineers*, 7th ed. Hoboken, NJ, USA: Wiley, 2018.

References:

1. P. V. O'Neil, *Advanced Engineering Mathematics*, 8th ed. Boston, MA, USA: Cengage Learning, 2018.
2. R. I. Miller and J. E. Freund, *Probability and Statistics for Engineers*, 9th ed. Noida, India: Pearson Education Ltd., 2017.
3. E. Kreyszig, *Advanced Engineering Mathematics*, 10th ed. Hoboken, NJ, USA: John Wiley & Sons, 2011.
4. V. Henner, T. Belozerova, and M. Khenner, *Ordinary and Partial Differential Equations*. Boca Raton, FL, USA: CRC Press, 2013.

E-resources/ Web links:

1. https://nptel.ac.in/courses/111105121
2. https://nptel.ac.in/courses/111106146
3. https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_106839
4. https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_61605
5. https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_134719
6. https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=TEXTBOOK_LIBRARY01_06082022_174

Topics relevant to SKILL DEVELOPMENT: The course focuses on the concepts of differential calculus, differential equations and statistical methods 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: MAT1212	**Course Title:** Mathematical foundations for Machine Learning and Data Analysis **Type of Course:** 1] School Core	**L-T- P- C**	**3**	**1**	**0**	**4**
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	NIL					
Course Description	This course provides a comprehensive foundation in advanced mathematical concepts essential for engineering and scientific applications. It covers integral calculus and special functions, linear transformations and matrix decompositions, vector spaces and inner product spaces, and probability theory with random variables and standard probability distributions. The course emphasizes mathematical modeling, analytical problem-solving, data analysis techniques, and engineering applications, enabling students to apply mathematical tools to solve real-world engineering problems.					
Course Objective	The course aims to develop a strong conceptual understanding of vector spaces, linear algebra, and vector calculus. It introduces mathematical techniques used in engineering analysis and modelling, enabling students to apply these concepts effectively in solving engineering problems. The course also enables students to apply probability distributions and statistical methods in practical situations involving uncertainty and data analysis. Further, it strengthens analytical and problem-solving skills through applications of differential equations and statistical inference. In addition, the course familiarizes students with computational tools for solving engineering mathematics problems, thereby enhancing their ability to address real-world engineering challenges.					

Course Outcomes	On successful completion of the course the students shall be able to: **CO1:** Apply integral calculus and special functions to solve engineering problems. (L3) **CO2:** Solve problems involving linear transformations and matrix techniques. (L3) **CO3:** Apply concepts of vector spaces, basis, dimension, and orthogonality in mathematical applications. (L3) **CO4:**Apply probability concepts and random variable distributions to solve problems involving uncertainty and stochastic phenomena (L3) **CO5:**Examine discrete and continuous random variables using probability distributions such as Binomial, Poisson, Uniform, Exponential, and Normal distributions. (L3) **CO6:**Utilize computational tools such as MATLAB, Python, or similar software for mathematical computation, data analysis, and engineering applications. (L3)	
Course Content:		
Module 1	**Integral Calculus and Special Functions**	**(8L+3T=11 Sessions)**
Definite and indefinite integrals, Evaluation of definite and improper integrals; Beta and Gamma functions and their properties; Surface areas, volumes, Sterlin approximation theorem.		
Module 2	**Vector Spaces and Inner Product Spaces**	**(12L+4T=16Sessions)**
Vector Space, Examples, Properties, Subspaces, Linear Combinations, Spanning Set of a Vector Space, Linear Dependence and Independence, Basis and Dimension, Change of Basis; Normed Linear Spaces, Inner Products Spaces, Orthogonality, Orthogonal and Orthonormal Bases, Gram-Schmidt Orthogonalization Process, Orthogonal Complements, Orthogonal Projections. Applications to Difference Equations, Markov Chains.		
Module 3	**Linear Transformations and Matrix Decompositions**	**(10L+3T=13 Sessions)**
Linear Transformations, Algebra of Transformations, Singular and Nonsingular Linear Mappings, Isomorphisms, Matrix Representation of Linear Transformations, The Range and Null Space, Rank-Nullity Theorem. Least Squares Approximation, Singular Value Decomposition: Singular Values, Computing Singular Value Decomposition, Data Compression Using Singular Value Decomposition and Introduction to Principal Component Analysis.		
Module 4	**Probability and Random Variables for machine learning**	**(15L+5T =20Sessions)**
Introduction to Probability, Conditional Probability, Bayes' Theorem and Problems. Random Variables (discrete and continuous), Probability Mass/Density Functions, Mathematical Expectations, discrete probability distributions - Binomial distribution, Poisson distribution, Continuous uniform distribution - Exponential distribution, Normal distribution. Machine learning: large language models		
Targeted Applications & Tools that can be used: • Evaluation of definite and improper integrals for computing areas, volumes, surface areas, engineering design calculations, and scientific modeling using Beta and Gamma functions. • Representation and analysis of high-dimensional vector spaces, feature extraction, orthogonal projections, signal processing, computer graphics, pattern recognition, and machine learning. • Least-squares regression, image compression, recommendation systems, dimensionality reduction, Principal Component Analysis (PCA), data mining, and machine learning applications. • Probabilistic modeling, Bayesian inference, uncertainty quantification, predictive analytics,		

classification algorithms, statistical learning, and foundations of large language models (LLMs).

Software: MATLAB, Mathematica, Maple, Python, GeoGebra, Octave, R, Scilab.

Text Book

1. E. Kreyszig, *Advanced Engineering Mathematics*, 10th ed. Hoboken, NJ, USA: John Wiley & Sons, 2015.
2. B. S. Grewal, *Higher Engineering Mathematics*, 45th ed. New Delhi, India: Khanna Publishers, 2017.sley Publishing Company.

Assignment:

Assignments will involve theoretical problem-solving and computational implementation using suitable programming tools. The assignments may be undertaken individually or in groups, as decided by the instructor.

References:

1. G. Strang, *Introduction to Linear Algebra*, 6th ed. Wellesley, MA, USA: Wellesley-Cambridge Press, 2023.
2. D. C. Lay, S. R. Lay, and J. J. McDonald, *Linear Algebra and Its Applications*, 6th ed. Harlow, U.K.: Pearson, 2021.
3. E. Kreyszig, *Advanced Engineering Mathematics*, 10th ed. Hoboken, NJ, USA: John Wiley & Sons, 2015.
4. B. S. Grewal, *Higher Engineering Mathematics*, 45th ed. New Delhi, India: Khanna Publishers, 2017.
5. C. D. Meyer, *Matrix Analysis and Applied Linear Algebra*, 2nd ed. Philadelphia, PA, USA: SIAM, 2023.

E-resources/ Web links:

1. https://open.umn.edu/opentextbooks/textbooks/24
2. https://open.umn.edu/opentextbooks/textbooks/introduction-to-probability
3. https://open.umn.edu/opentextbooks/textbooks/linear-transformations-on-vector-spaces

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: MAT2311	**Course Title: Vector Calculus, Numerical Methods and Sampling Distributions** **[Type of Course:1] School Core**	**L-T- P- C**	**3**	**1**	**0**	**4**
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	NIL					
Course Description	This course introduces the fundamental concepts and methods of ordinary differential equations, numerical techniques, and statistical inference used in engineering and scientific applications. The course covers analytical methods for solving differential equations, numerical methods for solving linear and nonlinear equations, interpolation, numerical differentiation and integration, and numerical solutions of ordinary differential equations. It also introduces sampling distributions and hypothesis testing for statistical decision-making. The course enables students to apply mathematical and					

	statistical techniques to solve computational and real-world engineering problems.	
Course Objective	The objectives of this course are to develop the ability to solve ordinary differential equations arising in scientific and engineering applications. Introduce numerical techniques for solving systems of linear equations and nonlinear algebraic equations. Apply interpolation, differentiation, and integration methods for approximating mathematical functions and evaluating definite integrals. Employ numerical methods for obtaining approximate solutions of ordinary differential equations. Apply concepts of sampling distributions and statistical inference to estimate population parameters and support decision-making. Apply hypothesis testing techniques to real-world problems involving means, proportions, variances, and associations among variables.	
Course Outcomes	After successful completion of the course, students will be able to: **CO1:** Employ vector differential operators to solve problems involving scalar and vector fields and classify them as solenoidal or irrotational (L3) **CO2:** Apply vector integration techniques and integral theorems to a solve engineering problems. (L3) **CO3:** Use numerical methods to solve systems of linear and nonlinear equations. (L3) **CO4:** Implement numerical techniques for interpolation, differentiation, integration, and ordinary differential equations. (L3) **CO5:** Utilize sampling distributions and statistical inference techniques to solve statistical problems. (L3) **CO6:** Perform hypothesis testing for statistical decision-making. (L3)	
Course Content:		
Module 1	**Vector Calculus**	**(14L+5T=19 Sessions)**
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, Surface and Volume Integrals, Green's Theorem in the plane, Divergence Theorem of Gauss, Stokes's Theorem. Problems on Image Edge Detection, Robot Path Planning and 3D Surface Modelling		
Module 2	**Solution of Linear Systems and non-linear of Equation**	**(9L+3T=12 Sessions)**
Introduction, Jacobi iteration method, Gauss-Seidel iteration method. Solution of Algebraic and Transcendental Equations: Bisection method, Regula-Falsi method, Newton-Raphson method, secant method, fixed point iteration method. Page Ranking, Network Analysis, Circuit Simulation, Root Finding, Load Balancing, and Computational Optimization Problems.		
Module 3	**Numerical Interpolations and Differential Equations**	**(10L+3T=13 Sessions)**
Newton's method, Divided difference method, Lagrange's method, Numerical differentiation, Numerical integration: Trapezoidal rule, Simpson's one-third rule, Simpson's three-eighth rule, Gaussian quadrature rule. Solution of ordinary differential equations: Taylor series method, modified Euler's method, Runge-Kutta method for 4th order. Problems on Signal Reconstruction, Sensor Data Interpolation and Trajectory Prediction.		

Module 4	Sampling Distributions	(12L+4T=16 Sessions)

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, test for ratio of variances - Chi-square test for goodness of fit and independence of attributes. A/B Testing, Fraud Detection, Spam Classification Evaluation, and Network Traffic Analysis.

Targeted Applications & Mathematical Tools:

- Analysis of fluid flow in open channels and pipelines, groundwater movement, stress and strain fields, terrain modelling, and 3D surface representation in civil engineering projects.
- Structural analysis, load distribution in trusses and frames, hydraulic network analysis, foundation design computations, and optimization problems in transportation and water resource systems.
- Estimation of terrain elevations, survey data interpolation, rainfall-runoff modelling, settlement prediction, traffic flow analysis, and numerical solutions of engineering differential equations.
- Quality control of construction materials, reliability analysis, traffic studies, environmental data analysis, hydrological investigations, and hypothesis testing in engineering experiments.
- Analysis of vector fields in fluid flow, electromagnetics, and heat transfer; image edge detection, robot path planning, and 3D surface modelling using gradients, divergence, curl, and integral theorems.

Software: MATLAB, Python, Microsoft Excel Solver, LINDO/LINGO, AutoCAD, ETABS, Scilab.

Assignment:

Assignments will involve theoretical problem-solving and computational implementation using suitable programming tools. The assignments may be undertaken individually or in groups, as decided by the instructor.

Text Book

1. M. K. Jain, S. R. K. Iyengar, and R. K. Jain, *Numerical Methods for Scientific and Engineering Computation*, 7th ed. New Delhi, India: New Age International Publishers, 2019.
2. B. S. Grewal, *Higher Engineering Mathematics*, 45th ed. New Delhi, India: Khanna Publishers, 2022.
3. S. C. Gupta and V. K. Kapoor, *Fundamentals of Mathematical Statistics*, 12th ed. New Delhi, India: Sultan Chand & Sons, 2020.

References:

1. M. K. Jain, S. R. K. Iyengar, and R. K. Jain, *Numerical Methods for Scientific and Engineering Computation*, 7th ed. New Delhi, India: New Age International Publishers, 2019.
2. S. S. Sastry, *Introductory Methods of Numerical Analysis*, 5th ed. New Delhi, India: PHI Learning Pvt. Ltd., 2012.
3. J. Kiusalaas, *Numerical Methods in Engineering with Python 3*, 3rd ed. Cambridge, U.K.: Cambridge University Press, 2025.
4. D. C. Montgomery and G. C. Runger, *Applied Statistics and Probability for Engineers*, 8th ed. Hoboken, NJ, USA: John Wiley & Sons, 2023.
5. D. D. Wackerly, W. Mendenhall, and R. L. Scheaffer, *Mathematical Statistics with Applications*, 7th ed. Belmont, CA, USA: Cengage Learning, 2008.
6. E. Kreyszig, *Advanced Engineering Mathematics*, 10th ed. Hoboken, NJ, USA: John Wiley & Sons, 2011.
7. R. L. Burden, J. D. Faires, and A. M. Burden, *Numerical Analysis*, 11th ed. Boston, MA, USA: Cengage Learning, 2021.
8. S. C. Chapra, *Applied Numerical Methods with MATLAB for Engineers and Scientists*, 4th ed. New York, NY, USA: McGraw-Hill Education, 2018.

E-resources/ Web links:

1. https://presiuniv.knimbus.com/portal/v2/default/viewDetail?searchResultType=ECATALOGUE_

BASED&uniqueId=CUSTOM_PACKAGE_EBSCO_29052023_53751

2. https://presiuniv.knimbus.com/portal/v2/default/viewDetail?searchResultType=ECATALOGUE_BASED&uniqueId=TEXTBOOK_LIBRARY01_06082022_117
3. https://presiuniv.knimbus.com/portal/v2/default/viewDetail?searchResultType=ECATALOGUE_BASED&uniqueId=DOAB_1_06082022_7313
4. https://presiuniv.knimbus.com/portal/v2/default/viewDetail?searchResultType=ECATALOGUE_BASED&uniqueId=CUSTOM_PACKAGE_EBSCO_29052023_228965
5. https://presiuniv.knimbus.com/portal/v2/default/viewDetail?searchResultType=ECATALOGUE_BASED&uniqueId=DOAB_1_06082022_8838
6. https://presiuniv.knimbus.com/portal/v2/default/viewDetail?searchResultType=ECATALOGUE_BASED&uniqueId=SPRINGER_OPEN_ACCESS_BOOKS_10022025_1861
7. https://presiuniv.knimbus.com/portal/v2/default/viewDetail?searchResultType=ECATALOGUE_BASED&uniqueId=DOAB_1_06082022_7312

Topics relevant to SKILL DEVELOPMENT: Identification and solution of differential equations, application of numerical methods for algebraic and differential equations, implementation of interpolation and numerical integration techniques, statistical data analysis and hypothesis testing, and use of Python/MATLAB for engineering problem solving.

Course Code: MAT2321	**Course Title:** Discrete Mathematical Structures **Type of Course:1] School Core**	**L-T- P- C**	**3**	**1**	**0**	**4**
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	NIL					
Course Description	This course introduces the fundamental concepts of Discrete Mathematics, providing the mathematical foundations essential for computer science and related disciplines. It covers sets, relations, functions, closures, equivalence relations, and primitive recursive functions, followed by partially ordered sets, lattices, and Boolean algebras. The course also explores graph theory, including graph types, isomorphism, Euler and Hamiltonian paths, circuits, and graph coloring. In addition, students study algebraic structures such as semigroups, monoids, and groups, residue arithmetic, and group codes. Emphasis is placed on developing logical reasoning, mathematical modeling, and problem-solving skills applicable to computing, information systems, and network analysis.					
Course Objective	The primary objective of this course is to provide students with a strong foundation in discrete mathematical structures and logical reasoning that underpin computer science and related disciplines. It aims to develop the ability to construct mathematical arguments, analyze discrete problems, and apply formal methods to problem solving. The course introduces fundamental concepts of logic, sets, relations, functions, Boolean algebra, graph theory, and algebraic structures such as semigroups, monoids, and groups, while also exploring formal languages and grammar. Through these topics, students will acquire the analytical and mathematical skills necessary for applications in algorithm design, digital systems, coding theory, cryptography, and other areas of computing.					
Course Outcomes	On successful completion of the course the students shall be able to: **CO1:** Apply propositional and predicate logic, logical equivalences, normal forms, inference rules, and proof techniques to analyze arguments and solve problems in					

	mathematical reasoning and computer science. (L3) **CO2:** Apply the concepts of sets, relations, functions, closures, equivalence relations, and primitive recursive functions to model and solve problems in discrete mathematics and computer science. (L3) **CO3:** Employ partially ordered sets, lattices, Boolean algebras, and Boolean functions, and apply their properties to solve problems in discrete mathematical structures and logical systems. (L3) **CO4:** Apply the concepts of lattices, Boolean algebras, and Boolean functions to solve logical and computational problems. (L3) **CO5:** Apply graph theoretic concepts and techniques to solve problems in networks, connectivity, and optimization applications. (L3) **CO6:** Employ group theory and residue arithmetic techniques in coding, computation, and information processing. (L3)	
Course Content:		
Module 1	**Fundamentals of Logic**	**(11L+4T=14 Sessions)**
Propositional Logic, Propositional Logic, Equivalences, Normal forms, Inference rules, Introduction to Proofs, Conversion to clausal form, Predicate calculus, The Statement function, Inference theory of the Predicate Calculus. Compiler optimization and program correctness verification, Cybersecurity and cryptographic protocol verification.		
Module 2	**Relations and Functions**	**(10L+3T=13 Sessions)**
Sets and set-operations, Relations, Functions, Properties of Relations & Representations of Relation, closure of different types of relations, equivalence relations, primitive recursive function. Database Management Systems, Artificial Intelligence and Machine Learning.		
Module 3	**Boolean Algebra and Graph Theory**	**(13L+4T=17 Sessions)**
Partially Ordered Sets, Lattices, Boolean Algebras, Functions of Boolean Algebras, Boolean Functions as Boolean Polynomials. Computer Networks: Routing and switching logic. Robotics: Decision-making using Boolean logic. Graph Theory-Basic Concept of Graph Theory, Types of Graphs, Isomorphism, Euler Paths and Circuits, Hamiltonian Paths and Circuits, Colouring of Graph. Social Networks (Facebook, Instagram, LinkedIn).and GPS & Navigation: Shortest path and route optimization.		
Module 4	**Groups**	**(11L+4T=15 Sessions)**
Semi Groups, Monoids, Groups, The applications of Residue Arithmetic to Computers, Group codes. Functional programming and distributed systems. Compiler design and programming languages.		
Targeted Application & Tools that can be used: • Compiler optimization, program correctness verification, cybersecurity protocol verification, automated reasoning, formal software verification. • Database Management Systems (DBMS), Artificial Intelligence (AI), Machine Learning (ML), knowledge representation, data modeling and query processing. • Digital circuit design, computer networks (routing and switching), robotics decision-making systems, logic controller design, social network analysis, GPS and navigation systems, route optimization, communication networks, and recommendation systems. • Cryptography, error-correcting codes, compiler design, distributed systems, residue arithmetic in		

computer architecture.

Software: MATLAB, Lean 4, Mathematica, Maple, Python, GeoGebra, Octave, R, Scilab.

Assignment:

Assignments will involve theoretical problem-solving and computational implementation using suitable programming tools. The assignments may be undertaken individually or in groups, as decided by the instructor. Additionally, experiential learning activities will be conducted using Lean 4, enabling students to develop formal proofs, verify logical propositions, and gain hands-on experience in theorem proving and formal verification.

Text Book

6. K. H. Rosen, *Discrete Mathematics and Its Applications*, 8th ed. New York, NY, USA: McGraw-Hill Education, 2019.
7. F. Harary, *Graph Theory*. Reading, MA, USA: Addison-Wesley Publishing Company, 1969.

References:

1. S. Epp, *Discrete Mathematics with Applications*, 5th ed. Boston, MA, USA: Cengage Learning, 2019.
2. K. A. Ross and C. R. B. Wright, *Discrete Mathematics*, 6th ed. New Delhi, India: Pearson Education, 2018.
3. L. Mott, A. Kandel, and T. P. Baker, *Discrete Mathematics for Computer Scientists and Mathematicians*, 3rd ed. New Delhi, India: PHI Learning, 2016.

E-resources/ Web links:

4. https://presiuniv.knimbus.com/portal/v2/default/viewDetail?searchResultType=ECATALOGUE_BASED&uniqueId=TEXTBOOK_LIBRARY01_06082022_289
5. https://presiuniv.knimbus.com/portal/v2/default/viewDetail?searchResultType=ECATALOGUE_BASED&uniqueId=DOAB_1_06082022_8919
6. https://nptel.ac.in/courses/106108227
7. https://nptel.ac.in/courses/111106102
8. https://nptel.ac.in/courses/111106113

Topics relevant to SKILL DEVELOPMENT: The course develops skills in logical reasoning, mathematical modeling, problem-solving, algorithmic thinking, and graph analysis through the study of logic, relations, Boolean algebra, and graph theory. It also enhances computational and analytical skills by applying algebraic structures, formal languages, recursion, and coding techniques to real-world computing problems.

Course Code: **PHY2501**	**Course Title:** Optoelectronics and Quantum Physics **Type of Course:** **BSC**	**L-T-P-C**	**3**	**0**	**0**	**3**
Version No.	1.0					
Course Pre-requisites	NIL					
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 Outcomess	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**	**12 Sessions**
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**	**12 Sessions**
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**	**11 Sessions**
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**	**10 Sessions**
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.

This is attained through the Assignment/ Presentation as mentioned in the assessment component in course handout.

Course Code: PHY2504	**Course Title: Optoelectronics and Quantum Physics Lab** **Type of Course: BSC**	**L-T-P-C**	**0**	**0**	**2**	**1**
Version No.	1.0					
Course Pre-requisites	**NIL**					
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 Outcomess	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: **30 Sessions**

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: Determination of dielectric constant of given materials.
Level 1: Determination of Stefan's constant

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.

List of Engineering Science Courses (ESC)

CourseCode: ECE1512	**CourseTitle:DesignThinkinginPractice**	**L-T-P-C**	2	0	2	3
Version No.	1.0					
CoursePre-requisites	NIL					
Anti-requisites	NIL					
Course Description	This course introduces students to Design Thinking frameworks for identifying real-world user needs and transforming them into innovative technology solutions using embedded systems, sensors, IoT platforms, and AI-enabled computing devices. The course blends human-centeredinnovationmethodologieswithhands-on developmentusing Arduino, Aries, ESP32, Raspberry Pi, NVIDIA Jetson, and GPU-based embedded platforms. Students will develop problem-solving skills while learning problem identification, ideation, rapid prototyping,sensorintegration,edgeAI,andintelligentsystemdevelopmentforapplications insmartautomation,healthcare,agriculture,robotics,andIndustry5.0.Thecourseemphasizes experientialandproject-basedlearningtobuildfuture-readyinnovatorscapableofdeveloping scalable, intelligent,andsocially impactfulproducts.					

Course Objective	The objective of the course is to enhance students' employabilityskillsthroughparticipative learning techniques by integrating Design Thinking, embedded systems, IoT, and AI technologies to develop innovative, intelligent, and sociallyimpactful real-world solutions.	
Course Outcomes	**Onsuccessfulcompletionofthecourse,thestudentsshallbeableto** 1. **Explain**the principles, framework,andprocess ofdesignthinking 2. **Formulate**EngineeringProblemfromuserchallenges,customerexpectations,and market opportunities using design thinking process. 3. **Analyze**theEngineeringProblemusingthebasicconceptsofmicrocontrollerand sensory devices. 4. **Find** alternate methods to solve the engineering problem by working with embedded computing platforms. 5. **Develop**functionalprototypesusingembeddedboards,sensors,andanintelligent system for the chosen method 6. **Validate**prototypeperformancethroughtesting,userfeedback,anditerative improvements.	
Course Content		
Module 1	**Introduction to Design Thinking**	**6 Sessions**
Introduction to Design ThinkingImportance of Design ThinkingHistory of Design Thinking- Design ThinkingFramework-DesignThinkingprocess-EmpathiseDefineUnderstandingPurpose-Ideate-What isCreativity?-ThinkingDifferentlyBasicRulesofDesignThinkingTheCreativityHeartSetEveryoneisa Designer - Activity-based with case studies		
Module 2	**Basic Concepts of Microcontrollers and Sensory Devices**	**8 Sessions**
Introduction to Arduino, ESP and NodeMCU - Pin Configuration and Architecture - Device and Platform Features-ConceptofDigitalandAnalogPorts-FamiliarizingwithArduinoInterfacingBoardProgramming using Arduino IDE and Embedded C - Datatypes and Variables - I/O FunctionsSensors, types and its applications-HumiditySensor-TemperatureSensor-WaterDetector/Sensor -PIRSensor -UltrasonicSensor -ConnectingSwitchesandActuators-SensorInterfacewithESPMicrocontroller-VariousCloudPlatforms		
Module 3	**Working with Embedded Computing Platforms and 3D printing technology**	**8 Sessions**
IntroductiontoRaspberryPiboards,pindiagram,differenttypesofRaspberryPiboardsandtheirapplications, ConfigurationofRaspberryPiboards-LEDandswitchcontrolExploringvariouslibrariesandtheirfunctions. GPUfundamentalsConfigurationofJetsonNanoboard,WorkingwithH200GPUServers.Introductionto3D Printer:3DPrintertechnologyanditsworkingPrinciples,Applications.IntroductiontoonlineSimulators: WorkingwithTinkerCADSimulator.		
Module 4	**Prototype Development and Validation Techniques**	**8 Sessions**

ImportanceofPrototypinginDesignThinking – ConceptRefinementUsingPrototypes From–Ideato FunctionalPrototype UserTestingandValidation IterationandDesignOptimization – PrototypeReadiness – – for ProductDevelopment(DFM&MarketFeasibility – overview) – Activity-basedwith casestudies.

PrototypeDevelopmentTasks:
TotalSessions:30

1. SmartilluminationsystemusingArduino
2. IntelligentPlantHealthMonitoringSystem.
3. AriesRoverDriveandControlSystemDevelopment
4. IoT-BasedEnvironmentalIntelligenceSystemusing ESP32
5. EdgeComputing PlatformSetup using RaspberryPi
6. RaspberryPi-basedsmartlightingController
7. VoiceX:AI-Powered SmartHomeAutomation System
8. VisionAI:IntelligentObjectRecognitionSystem.
9. SmartTrafficManagementSystemusingNVIDEAJetson Nano
10. High-PerformanceAIVision AnalyticsusingH200-GPUComputing.
11. RapidPrototype Developmentusing3DDesignandprinting

Topics: Types of embedded boards, Thonny Python, Python IDLE, sensors, 3D Printer, Jetson Nano board, GPUs

TargetedApplication&Toolsthatcanbeused:

ApplicationArea:

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 examplesofthemanyapplicationareaswhereArduino, Aries,ESP32,RaspberryPi,andsensorscanbeapplied. The flexibility and affordability of all embedded boards, 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, such as Arduino IDE, Blynk console, Tinker CAD, Thonny Python, Python IDLE, etc.

Project work/Assignment:

- Projects: At the end of the course, students will be completing the project work on solving many real-time problem statements on different domains, such as women-centric, agritech, medtech, climate cell, robotics, drones, and smart cities etc.
- 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 to the library resources and write a report on their understanding of the assigned article in an appropriate format. Presidency University Library Link
- Presentation: There will be a review presentation (Phase 1, Phase 2, and Phase 3) from among the students' groups, where the students will be given a project, and they have to demonstrate the working and discuss the applications for the same

Textbook(s):

1. Thinking Design by S Balaram. New Delhi [India]: Sage Publications Pvt. Ltd. 2010. eBook. Database: eBook Collection (EBSCOhost)
2. Monk Simon, "Programming Arduino: Getting Started with Sketches", McGraw-Hill Publications, Second Edition
3. Monk Simon "Raspberry Pi Cookbook: Software and Hardware Problems and Solutions", Publisher O'Reilly Media, Inc., ISBN: 9781098130923, Fourth Edition.
4. CUDA by Example: An Introduction to General-Purpose GPU Programming. CUDA by Example: An Introduction to General-Purpose GPU Programming. Addison-Wesley Professional, 2010.

References:

1. Design Thinking by Clarke, Rachel Ivy. Series: Library Futures, Vol. 4. Chicago: ALA Neal-Schuman. 2020. eBook., Database: eBook Collection (EBSCOhost)
2. Neerparaj Rai, "Arduino Projects for Engineers," BPB Publishers, first edition, 2016.
3. Stewart Watkiss "Learn Electronics with Raspberry Pi" Apress Berkeley, CA. Second edition,2020.

ISBN978-1-4842-6348-8 of 3D printing", Prusa Research, 3

4. Jo Prusa, "Basics, " rd edition.
5. Volker Ziemann, "A Hands-On Course in Sensors Using the Arduino and Raspberry Pi (Series in Sensors)", CRC Press, 1st Edition. 2018.
6. Professional CUDA C Programming. Professional CUDA C Programming. Wrox Press, 2014.

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 thingshttps://nptel.ac.in/courses/106105166>
6. Getting Started with Jetson Nano - https://www.youtube.com/watch?v=DVfmw3Ftu3I

Topics relevant to the development of "SKILL": Systemdesign forachievingSustainableDevelopment Goals.

Course Code: CIV1200	**Course Title:** Foundations of Integrated Engineering **Type of Course: ESC**	**L- 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**					**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**					**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**					**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**	**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**	**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

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: CSE1500	**Course Title: Computational Thinking using Python** **Type of Course: Lab Integrated Course (ESC)**	**L-T-P-C**	**2**	**0**	**2**	**3**
Version No.	1.0					
Course Pre-requisites	Nil					
Anti-requisites	NIL					
Course Description	The course efficiently introduces fundamental ideas including conditionals, loops, functions, lists, strings, and tuples through some inspiring examples. It then discusses dynamic programming like handling exceptions and file usage. In terms of data structures, the course covers Python dictionaries, classes, and objects for constructing user-defined datatypes like linear and binary search.					
Course Objective	To develop computational thinking and problem-solving skills using Python by enabling students to design algorithms, implement programs using core programming constructs and build efficient solutions using data structures and file handling techniques for real-world applications.					
Course Outcomes	On successful completion of this course, students will be able to: **CO1:** Apply computational thinking principles to design algorithms for solving basic and intermediate computing problems. [Understand] **CO2:** Implement Python programs using data types, operators, and expressions to perform computational tasks. [Apply] **CO3:** Develop programs using control structures, functions, and recursion to solve structured problems. [Apply] **CO4:** Utilize data structures such as lists, tuples, and dictionaries for efficient data					

	representation and manipulation. [Apply] **CO5:** Design and implement Python programs for file handling and exception management to handle real-world data. [Apply] **CO6:** Build modular and reusable programs using functions and standard Python libraries to enhance code efficiency and readability. [Apply]

Course Content:

Module 1	**Computational Thinking And Problem Solving**	**6 Sessions**

Topics:
Fundamentals of Computing– Identification of Computational Problems Algorithms, building blocks of algorithms (statements, state, control flow, functions), notation (pseudo code, flow chart, programming language), algorithmic problem solving, simple strategies for developing algorithms (iteration, recursion). Illustrative problems: find minimum in a list, insert a card in a list of sorted cards, guess an integer number in a range, Towers of Hanoi

Module 2	**Datatypes, Expressions, Statements**	**6 Sessions**

Topics:
Python interpreter and interactive mode,debugging; values and types: int, float, boolean, string , and list; variables, expressions, statements, tuple assignment, precedence of operators, comments; Illustrative programs: exchange the values of two variables, circulate the values of n variables, distance between two points.

Module 3	**Control flow, Functions, Strings**	**6 Sessions**

Topics:
Conditionals:Boolean values and operators, conditional (if), alternative (if else),chained conditional (if-elif-else);Iteration: state, while, for, break, continue, pass; Fruitful functions: return values,parameters, local and global scope, function composition, recursion; Strings: string slices,immutability, string functions and methods, string module; Lists as arrays. Illustrative programs: square root, gcd, exponentiation, sum an array of numbers, linear search, binary search.

Module 4	**Lists, Tuples, Dictionaries**	**6 Sessions**

Topics:
Lists: list operations, list slices, list methods, list loop, mutability, aliasing, cloning lists, list parameters; Tuples: tuple assignment, tuple as return value; Dictionaries: operations and methods; advanced list processing- list comprehension; Illustrative programs: simple sorting, histogram, Students marks statement, Retail bill preparation.

Module 5	**Files**	**6 Sessions**

Files and exceptions: text files, reading and writing files, format operator; command line arguments, errors and exceptions, handling exceptions, modules, packages; Illustrative programs: word count, copy file, Voter's age validation, Marks range validation (0-100).

List of Practical Tasks **30 Sessions**

WEEK 1: Computational Thinking & Algorithms

Lab Sheet 1: Problem Solving & Algorithm Design

Level 1: Write an algorithm and Python program to find the minimum element in a list of

numbers.

Level 2: Write an algorithm and Python program to insert an element into a sorted list at the correct position.

Lab Sheet 2: Algorithm Strategies (Iteration & Recursion)

Level 1: Write a Python program to guess a number within a given range using iteration.

Level 2: Write a Python program to solve the Tower of Hanoi problem using recursion.

WEEK 2: Data Types & Expressions

Lab Sheet 3: Data Types & Variables

Level 1: Write a Python program to demonstrate different data types (int, float, boolean, string, list).

Level 2: Write a Python program to perform type conversions and display results.

Lab Sheet 4: Expressions & Operators

Level 1: Write a Python program to swap two variables using tuple assignment.

Level 2: Write a Python program to circulate the values of n variables.

WEEK 3: Expressions & Problem Solving

Lab Sheet 5: Mathematical Expressions

Level 1: Write a Python program to calculate the distance between two points.

Level 2: Write a Python program to evaluate complex arithmetic expressions with operator precedence.

Lab Sheet 6: Debugging & Interactive Mode

Level 1: Write a Python program with intentional errors and debug it.

Level 2: Write a Python program to demonstrate interactive mode operations.

WEEK 4: Conditional Statements

Lab Sheet 7: Decision Making

Level 1: Write a Python program to check whether a number is positive, negative, or zero.

Level 2: Write a Python program to find the largest of three numbers using if-elif-else.

Lab Sheet 8: Advanced Conditions

Level 1: Write a Python program to check whether a number is even or odd.

Level 2: Write a Python program to implement a grading system using chained conditions.

WEEK 5: Iteration (Loops)

Lab Sheet 9: Looping Basics

Level 1: Write a Python program to print numbers from 1 to N using a loop.

Level 2: Write a Python program to compute factorial using loops.

Lab Sheet 10: Loop Control

Level 1: Write a Python program demonstrating break and continue.

Level 2: Write a Python program to generate number patterns using nested loops.

WEEK 6: Functions

Lab Sheet 11: Functions Basics

Level 1: Write a Python function to calculate square root of a number.

Level 2: Write a Python function to find GCD of two numbers.

Lab Sheet 12: Recursion & Scope

Level 1: Write a recursive function to calculate exponentiation.

Level 2: Write a Python program demonstrating local and global variables.

WEEK 7: Strings

Lab Sheet 13: String Operations

Level 1: Write a Python program to demonstrate string slicing.

Level 2: Write a Python program to check whether a string is a palindrome.

Lab Sheet 14: String Methods

Level 1: Write a Python program using string functions (upper, lower, replace).

Level 2: Write a Python program to count vowels and consonants in a string.

WEEK 8: Lists

Lab Sheet 15: List Basics

Level 1: Write a Python program to create and display a list.

Level 2: Write a Python program to find the sum of elements in a list.

Lab Sheet 16: List Operations

Level 1: Write a Python program to demonstrate list slicing.

Level 2: Write a Python program to perform linear search on a list.

WEEK 9: Advanced Lists

Lab Sheet 17: List Methods

Level 1: Write a Python program to demonstrate list methods (append, remove).

Level 2: Write a Python program to sort a list.

Lab Sheet 18: List Comprehension

Level 1: Write a Python program using list comprehension to generate squares.

Level 2: Write a Python program to create a histogram of list values.

WEEK 10: Tuples

Lab Sheet 19: Tuple Basics

Level 1: Write a Python program to create and access tuple elements.

Level 2: Write a Python program to demonstrate tuple assignment.

Lab Sheet 20: Tuples as Return Values

Level 1: Write a Python function returning multiple values using tuple.

Level 2: Write a Python program to swap values using tuples.

WEEK 11: Dictionaries

Lab Sheet 21: Dictionary Basics

Level 1: Write a Python program to create and access dictionary elements.

Level 2: Write a Python program to count frequency of elements using dictionary.

Lab Sheet 22: Dictionary Applications

Level 1: Write a Python program to store student marks using dictionary.

Level 2: Write a Python program to generate a retail bill using dictionary.

WEEK 12: Advanced List Processing

Lab Sheet 23: Sorting & Searching

Level 1: Write a Python program to implement simple sorting.

Level 2: Write a Python program to implement binary search.

Lab Sheet 24: Data Processing

Level 1: Write a Python program to generate student marks statement.

Level 2: Write a Python program to analyze list data (min, max, average).

WEEK 13: File Handling Basics

Lab Sheet 25: File Operations

Level 1: Write a Python program to create and write to a text file.

Level 2: Write a Python program to read and display file contents.

Lab Sheet 26: File Processing

Level 1: Write a Python program to count words in a file.

Level 2: Write a Python program to copy contents from one file to another.

WEEK 14: Exceptions & Validation

Lab Sheet 27: Exception Handling

Level 1: Write a Python program using try-except.

Level 2: Write a Python program to handle multiple exceptions.

Lab Sheet 28: Validation Programs

Level 1: Write a Python program to validate voter's age.

Level 2: Write a Python program to validate marks range (0–100).

WEEK 15: Modules & Advanced Concepts

Lab Sheet 29: Modules & Packages

Level 1: Write a Python program to create and use a module.

Level 2: Write a Python program to use built-in modules.

Lab Sheet 30: Command Line Arguments

Level 1: Write a Python program to accept command-line arguments.

Level 2: Write a Python program to perform operations using command-line inputs

Project work/Assignment:

1. Assignment 1 on (Module 1 and Module 2)
2. Assignment 2 on (Module 3 and Module 4 & 5)

Text Book

1. Paul Deitel and Harvey Deitel, "Python for Programmers", Pearson Education, 1st Edition, 2021
2. Eric Matthes, Python Crash Course,: A Hands-On, Project-Based Introduction to Programming, 3rd Edition, 2023

References

1. Allen B. Downey, "Think Python: How to Think like a Computer Scientist", 2nd Edition, O'Reilly Publishers, 2016.
2. Karl Beecher, "Computational Thinking: A Beginner's Guide to Problem Solving and Programming", 1st Edition, BCS Learning & Development Limited, 2017.

Web Resources

W1. https://onlinecourses.nptel.ac.in/noc20_cs70/preview

1. **Topics relevant to development of "Employability":** Data structures using python.
2. **Topics relevant to "PROFESSIONAL ETHICS":** Naming and coding convention for simple programs using python.

Course Code: CSE2264	**Course Title: Essentials of AI** **Type of Course: -Theory Course (ESC)**	**L-T-P-C**	**3**	**0**	**0**	**3**
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	NIL					
CourseDescription	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.					
Course Objectives	To develop a foundational understanding of Artificial Intelligence by enabling students to explore search strategies and problem-solving techniques, study knowledge representation and reasoning and analyze real-world AI applications.					
Course Outcomes	On successful completion of this course, students will be able to: **CO1:** Explain various search techniques, problem-solving methods, and inference mechanisms in Artificial Intelligence. [Understand] **CO2:** Explain graphical and adversarial search algorithms used in AI. [Understand] **CO3:** Apply first-order logic and resolution techniques to solve reasoning problems. [Apply] **CO4:** Apply Hidden Markov Models (HMM) to solve sequence labeling problems. [Apply] **CO5:** Apply deep learning techniques to analyze and solve AI-based problems. [Apply]					
CourseContent:						
Module 1	**Search Methods for Problem-Solving**					**12 Sessions**
Topics: 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.						
Module 2	**Advanced Search Methods**					**12 Sessions**
Topics: **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						

Module 3	Knowledge-Based Logic Representation	12 Sessions

Topics:

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

Module 4	Uncertainty in AI	09 Sessions

Topics:

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.

Project 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).

Textbook(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.

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.
3. George Luger. *Artificial Intelligence: Structures and Strategies for Complex Problem Solving*. 6th Edition. Pearson Education. 2021.

Weblinks

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: Laboratory Course (ESC)	L-T-P-C	0	0	2	1
Version No.	1.0					
Course Pre-requisites	NIL					

Anti-requisites	NIL
CourseDescription	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	To develop practical skills in Artificial Intelligence by enabling students to implement basic AI algorithms, apply search and optimization techniques, develop simple intelligent systems, and analyze AI-based problem-solving approaches.
Course Outcomes	On successful completion of this course, students will be able to: **CO1:** Explain various search, reasoning, and problem-solving methods in Artificial Intelligence. [Understand] **CO2:** Implement graphical and adversarial search algorithms to solve AI problems. [Apply] **CO3:** Apply first-order logic and resolution techniques for logical inference. [Apply **CO4:** Implement probabilistic models such as Hidden Markov Models (HMM) for sequence labeling tasks. [Apply] **CO5:** Analyze and evaluate the performance of AI algorithms in solving real-world problems. [Analyze]

CourseContent:

List of Laboratory Tasks:30 Sessions

WEEK 1: State Space Search

Lab Sheet 1: Problem Formulation

Level 1: Represent a problem (e.g., water jug or puzzle) using state space.

Level 2: Define initial state, goal state, and operators for a real-world problem.

Lab Sheet 2: Data Structures in Search

Level 1: Implement stack and queue for search problems.

Level 2: Compare their use in BFS and DFS.

WEEK 2: Uninformed Search

Lab Sheet 3: BFS

Level 1: Implement Breadth First Search.

Level 2: Analyze time and space complexity.

Lab Sheet 4: DFS & IDDFS

Level 1: Implement Depth First Search.

Level 2: Implement Iterative Deepening DFS and compare with DFS.

WEEK 3: Uniform Cost Search

Lab Sheet 5: UCS

Level 1: Implement Uniform Cost Search.

Level 2: Compare UCS with BFS.

Lab Sheet 6: Dijkstra's Algorithm

Level 1: Implement Dijkstra's algorithm.

Level 2: Analyze shortest path for weighted graphs.

WEEK 4: Heuristic Search

Lab Sheet 7: Greedy Best First Search

Level 1: Implement Greedy Best First Search.

Level 2: Analyze heuristic quality.

Lab Sheet 8: A Search*

Level 1: Implement A* algorithm.

Level 2: Compare A* and Weighted A*.

WEEK 5: Local Search

Lab Sheet 9: Hill Climbing

Level 1: Implement Hill Climbing.

Level 2: Analyze local maxima problem.

Lab Sheet 10: Genetic Algorithm

Level 1: Implement simple Genetic Algorithm.

Level 2: Analyze crossover and mutation effects.

WEEK 6: Optimization Techniques

Lab Sheet 11: Gradient Descent

Level 1: Implement gradient descent.

Level 2: Analyze convergence rate.

Lab Sheet 12: Local vs Global Search

Level 1: Compare local search techniques.

Level 2: Apply to optimization problems.

WEEK 7: Adversarial Search

Lab Sheet 13: Minimax

Level 1: Implement minimax algorithm.

Level 2: Apply to simple game (tic-tac-toe).

Lab Sheet 14: Alpha-Beta Pruning

Level 1: Implement alpha-beta pruning.

Level 2: Compare with minimax.

WEEK 8: Constraint Satisfaction

Lab Sheet 15: CSP Basics

Level 1: Solve map coloring problem.

Level 2: Solve N-Queens problem.

Lab Sheet 16: CSP Optimization

Level 1: Implement backtracking search.

Level 2: Apply arc consistency.

WEEK 9: Propositional Logic

Lab Sheet 17: Logic Representation

Level 1: Represent logical expressions.

Level 2: Convert to CNF/DNF.

Lab Sheet 18: Inference Rules

Level 1: Apply inference rules.

Level 2: Solve logical problems.

WEEK 10: First Order Logic

Lab Sheet 19: FOL Representation

Level 1: Represent statements using FOL.

Level 2: Translate real-world problems into FOL.

Lab Sheet 20: FOL Inference

Level 1: Apply inference rules.

Level 2: Solve problems using FOL reasoning.

WEEK 11: Resolution

Lab Sheet 21: Propositional Resolution

Level 1: Apply resolution on propositional logic.

Level 2: Solve logical queries using resolution.

Lab Sheet 22: FOL Resolution

Level 1: Apply resolution in FOL.

Level 2: Solve story problems using resolution.

WEEK 12: Probability & Bayes

Lab Sheet 23: Probability

Level 1: Compute basic probabilities.

Level 2: Apply conditional probability.

Lab Sheet 24: Bayes Theorem

Level 1: Apply Bayes theorem.

Level 2: Solve real-world classification problems.

WEEK 13: Naïve Bayes

Lab Sheet 25: Classification

Level 1: Implement Naïve Bayes classifier.

Level 2: Apply on dataset.

Lab Sheet 26: Evaluation

Level 1: Evaluate classifier accuracy.

Level 2: Analyze performance.

WEEK 14: Hidden Markov Models

Lab Sheet 27: Forward Algorithm

Level 1: Implement forward probability.

Level 2: Analyze sequence probability.

Lab Sheet 28: Viterbi Algorithm

Level 1: Implement Viterbi algorithm.

Level 2: Apply to POS tagging.

WEEK 15: Deep Learning Basics

Lab Sheet 29: Neural Networks

Level 1: Implement simple perceptron.

Level 2: Train multilayer perceptron.

Lab Sheet 30: Activation Functions

Level 1: Implement activation functions.

Level 2: Compare performance of different functions.

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: CSE2274	**Course Title: Competitive Programming and Problem Solving** **Type Of Course: Laboratory Course (ESC)**	**L-T-P-C**	**0**	**0**	**4**	**2**
VersionNo.	1.0					
CoursePre-requisites	NIL					
Anti-requisites	NIL					
Course Description	TheCompetitive Programming and Problem Solvingcourse equips students with efficient problem-solving skills for coding competitions and real-world challenges. Startingwithbrute-forcesolutions,studentslearntooptimizetimeandspacecomplexity using advanced techniques like dynamic programming, greedy algorithms, and backtracking.Hands-onpracticeonplatformslikeCodeChefandCodeforceshelpstackle 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 Objective	TheobjectiveofthecourseistofamiliarizethelearnerswiththeconceptsofCompetitive Programming and Problem Solving and attain Skill Development through Experiential Learning techniques.					

Course Outcomes	Onsuccessfulcompletionofthecoursethestudentsshallbeableto: **CO1:**Understandthechallengesofonlinecompetitiveprogrammingplatforms,interpret problemconstraints andunderstandbrute-forcesolutionapproaches usedincompetitive programming. [Understand] **CO2:**Analyzethetimeandspacecomplexityofnumber-theoreticsolutionstodetermine the need for optimization under competitive programming constraints. [Analyze] **CO3:** Apply appropriate algorithmic paradigms and data structures to solve problems within defined time and memory constraints. [Apply] **CO4:** Analyze execution failures and identify algorithmic inefficiencies in solutions involvingadvanceddatastructureconceptstoredesignoptimizedsolutions withingiven time and memory constraints. [Analyze] **CO5:** Analyze advanced problem-solving techniques using dynamic programming to evaluate their effectiveness in large-scale competitive programming problems. [Analyze]

Course Content:

Module 1	**IntroductiontoCompetitiveProgramming**	**NA**

Topics:
Overview of Efficient Codingfor ProblemSolvingand CP: Introductiontocompetitiveprogramming(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 asTLE; use of STL.
CPProblemSolvingusingAdvancedTopics:conceptofdisjointsetsandtheirefficientrepresentation,algorithmic approachessuch asGreedy,Backtracking,Dynamic Programming and applying them for CP problems using bottom-up dynamic programming.

Module 2:	**NumberTheoryforProblem-Solving**	**NA**

Topics:
UseofNumberTheoryforproblem-solving:reducingtime/spacecomplexityofbruteforcecodingsolutionofSieveMethod,InverseModule,EuclidianMethodoffactorization;efficientcoding for Permutation Combination; XORing based and pattern-based solutions.

Module 3:	**OptimizingTime&SpaceUsingSequentialStorage**	**NA**

Topics:
CodingforOptimizingtimeandSpaceusingSequentialStorage:twopointerapproach; problem-solving using arrays and strings such as rotation on sorted arrays, duplicate removal, string matchingalgorithms;Kadane'salgo,stacks,priority-queuesandhashingbasedefficientcoding;median based problems and alternate solutions.

Module 4:	**Non-LinearDataStructures**	**NA**

Topics:
ApplyingNon-LinearDataStructuresforreal-lifeproblems:designofefficientsolutionsfor problems such as finding loops in a linked list, memory efficient DLL, block reversal in LL; problem solvingusingtreesandbinarytrees,Catalannumbers,applicationsofgraphs,spanningtreeandpath algos for CP problems with reduced time/space complexity.

Module 5:	**ProblemSolvingusingAdvancedTopics**	**NA**

Topics:
CPProblemSolvingusingAdvancedTopics:conceptofdisjointsetsandtheirefficientrepresentation,algorithmic approachessuch asGreedy,Backtracking,Dynamic Programming and applying them for CP problems using bottom-up dynamic programming.

List of Practical Tasks **60 Sessions**

WEEK 1

Lab Sheet 1: Sorting & Ranking

Level 1: Write a program to find the runner who finished in the third position from the given finishing times of N runners.

Level 2: Write a program to find the top 10 runners along with their bib numbers efficiently for large datasets.

Lab Sheet 2: Searching Techniques

Level 1: Write a program to check whether a given book ID exists in a library database.

Level 2: Design and implement a system to search books efficiently by title using an appropriate data structure.

WEEK 2

Lab Sheet 3: Cost Calculation

Level 1: Write a program to calculate the total cost of products in a shopping cart.

Level 2: Design an algorithm to apply multiple discount rules and compute the final bill.

Lab Sheet 4: Modular Arithmetic

Level 1: Write a program to compute (a^m \mod p).

Level 2: Implement binary exponentiation for efficient computation of large powers modulo p.

WEEK 3

Lab Sheet 5: Combinatorics

Level 1: Write a program to calculate the number of ways to choose K cards from N cards.

Level 2: Design an algorithm to calculate probabilities of drawing specific card combinations.

Lab Sheet 6: Graph Connectivity

Level 1: Write a program to determine if a path exists between two nodes in a network.

Level 2: Design an algorithm to detect anomalies in a network using XOR-based comparison techniques.

WEEK 4

Lab Sheet 7: Array Processing

Level 1: Write a program to find the minimum time required for cars to pass a point based on their speeds.

Level 2: Implement a two-pointer approach to simulate movement of cars with different lengths.

Lab Sheet 8: String Matching

Level 1: Write a program to count occurrences of a substring in a given string.

Level 2: Implement the KMP algorithm to efficiently find all occurrences of a pattern in a text.

WEEK 5

Lab Sheet 9: Priority Queue

Level 1: Write a program to track the highest bid in an auction system.

Level 2: Design a system to handle concurrent bids efficiently and update the highest bid.

Lab Sheet 10: Shortest Path

Level 1: Write a program to check if two users are connected in a social network.

Level 2: Implement Dijkstra's algorithm to find the shortest path between users.

WEEK 6

Lab Sheet 11: Tree Traversal

Level 1: Write a program to traverse a file system and display all files and directories.

Level 2: Design a file system supporting operations like create, delete, and search.

Lab Sheet 12: Tree Applications

Level 1: Write a program to calculate the total price of items in a shopping cart represented as a tree.

Level 2: Design a system to apply discounts efficiently in a hierarchical shopping cart.

WEEK 7

Lab Sheet 13: Disjoint Set

Level 1: Write a program to check whether all users in a group are connected.

Level 2: Find the minimum number of connections required to connect all users.

Lab Sheet 14: Greedy Algorithms

Level 1: Write a program to find the cheapest path to reach a destination.

Level 2: Design an algorithm to find the fastest path considering time constraints.

WEEK 8

Lab Sheet 15: BFS Applications

Level 1: Write a program to find the minimum number of moves for a rook to reach a target square.

Level 2: Implement BFS to find shortest path in a grid.

Lab Sheet 16: Game Algorithms

Level 1: Represent a game state using a tree structure.

Level 2: Implement minimax algorithm with alpha-beta pruning.

WEEK 9

Lab Sheet 17: Sorting Algorithms

Level 1: Write a program to implement insertion sort.

Level 2: Write a program to implement merge sort or quick sort.

Lab Sheet 18: Searching Algorithms

Level 1: Write a program to perform linear search.

Level 2: Write a program to perform binary search efficiently.

WEEK 10

Lab Sheet 19: Algorithm Analysis

Level 1: Write a program to compare execution time of two algorithms.

Level 2: Optimize a given algorithm to improve performance.

Lab Sheet 20: Recursion

Level 1: Write a program to solve Tower of Hanoi.

Level 2: Compare recursive and iterative solutions for a problem.

WEEK 11

Lab Sheet 21: Graph Representation

Level 1: Represent a graph using adjacency list.

Level 2: Represent a graph using adjacency matrix.

Lab Sheet 22: Graph Traversal

Level 1: Implement Depth First Search (DFS).

Level 2: Implement Breadth First Search (BFS).

WEEK 12

Lab Sheet 23: Data Structure Integration

Level 1: Combine arrays and lists to solve a problem.

Level 2: Use multiple data structures (tree + hash table) in one solution.

Lab Sheet 24: Real-world Application

Level 1: Design a simple recommendation system.

Level 2: Optimize it for large-scale data.

WEEK 13

Lab Sheet 25: String Algorithms

Level 1: Write a program for basic string matching.

Level 2: Implement efficient pattern matching algorithm.

Lab Sheet 26: Network Algorithms

Level 1: Check connectivity in a network.

Level 2: Detect anomalies using data comparison techniques.

WEEK 14

Lab Sheet 27: Algorithm Design

Level 1: Solve a problem using brute-force approach.

Level 2: Improve the solution using greedy or dynamic approach.

Lab Sheet 28: Performance Evaluation

Level 1: Analyze runtime of a program.

Level 2: Compare different algorithmic approaches.

WEEK 15

Lab Sheet 29: Case Study

Level 1: Implement a real-world problem solution.

Level 2: Optimize and document the solution.

Lab Sheet 30: Mini Project

Level 1: Combine multiple algorithms to build a system.

Level 2: Present optimized solution with performance analysis.

TargetedApplication&Toolsthatcanbeused:

1. CorC++Compiler(g++):ThestandardcompilerforCP.Familiarizestudentswithcompilationflags (e.g., -O2 for optimization).
2. IDE(IntegratedDevelopment Environment):Code::Blocks, VisualStudio, CLion, orsimilar IDEs. These provide debugging capabilities, code completion, and other helpful features.
3. OnlineJudges(CodeChef,Codeforces,LeetCode,HackerRank):Essentialforpracticingandsubmitting 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. WolframAlpha:Ausefultoolforverifyingnumber theorycalculationsandexploringconcepts.
7. StringLibraries:Familiarizestudentswiththestringmanipulationfunctionsavailableinc++.
8. GraphVisualizationTools:ToolslikeGraphvizcanbehelpfulforvisualizinggraphsand understanding graph algorithms.
9. DPDebuggingTechniques: PracticedebuggingDP solutions, astheycanbecomplex. Visualizing the DP table can be helpful

TextBooks:

1. Guide to Competitive Programming: Learning and Improving Algorithms Through Contests" (3rd Edition), *Antti Laaksonen, springer, 2024*
2. "DataStructuresandAlgorithmsinJava:AProject-BasedApproach"–*DanS.Myers, Cambridge University Press*

ReferenceBooks:

1. DataStructuresandAlgorithmicThinkingwithPython/C++/Java",NarasimhaKarumanchi,5th Edition, Career Monk, 2017.
2. IntroductiontoAlgorithms,ThomasH.Cormen(Author), CharlesE.Leiserson(Author),RonaldL.Rivest,fourtheditionApril2022 .

Web Resources

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

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

Assessment Type

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

Course Code: ECE2022	**Course Title: DIGITAL DESIGN** **Type of Course: School Core [Theory Only]**	**L-T-P-C**	**2**	**0**	**0**	**2**
Version No.	1.0					
Course Pre-requisites	**NIL**					
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. Further to focusing on combinational and sequential digital logic circuits. This course is analytical in nature and needs a fundamental knowledge on logical computation with Boolean algebra. The focus of the course will be to discuss the minimization techniques for making low-cost canonical digital circuit implementations. In this course, we emphasize design procedures of digital electronic circuits. Additionally, this course will create a foundation for future courses that includes Computer Architecture, Microprocessors, Microcontrollers and Embedded Systems etc.					
Course Objectives	The objective of the course is to familiarize the learners with Design and Analysis of Digital Logic Circuits concepts and to improve the learner's **SKILL DEVELOPMENT** by using **PARTICIPATIVE LEARNING TECHNIQUES** through Assessments**.**					
Course Outcomess	On successful completion of this course the students shall be able to: **CO1. Explain** the fundamental concepts of Number system, Boolean algebra and Logic gates **CO2. Determine** the minimum literal Boolean Expression using K-Map minimization techniques **CO3. Make use of**relevant logic gates for constructing the combinational circuit. **CO4. Produce** standard and real world combinational logic circuits using minimum number of logic gates **CO5. Demonstrate** the working principles of flip-flops and latches within the					

	context of sequential logic circuits. **CO6. Design** synchronous and asynchronous logic circuit using Flip-flops	
Module 1	**Number System, Logic Gates & Boolean function simplification**	**10 Sessions**
Review of Number systems -Overview of Boolean Algebra- Boolean Laws, Rules & Theorems. Boolean function Standard form and Canonical form. Review of logic gates -Basic gates, Universal gates & Special Gates- Boolean functions simplifications using Algebraic functions & Karnaugh-map. SOP and POS simplifications & Implementations.		
Module 2	**Combinational Logic circuits**	**10 Sessions**
Digital logic circuit Introduction –Classification of circuit- Analysis and Design procedure of Combinational Logic circuits - Binary Adder and Subtractor- Magnitude comparator- , Multiplexers - DeMultiplexers, Decoders, Encoders and Priority Encoders-Real world scenario based application design using logic gates.		
Module 3	**Sequential logic circuits**	**10 Sessions**
Introduction to sequential circuits- Storage elements: latches and flip flops- Characteristic tables and equations, excitation table- Analysis of clocked sequential circuits- state diagram-state table and design of finite state machines - Registers & Counters.		

Targeted Application & Tools that can be used:

NA

Project Based Learning proposed for this course

NA

Text Books:

1. **T1:** Thomas L Flyod, *"Digital Fundamentals"*, Eleventh Edition, Pearson Education. ISBN-10: 132737965. (2014).eBook-[PDF] DIGITAL LOGIC DESIGN FOURTH EDITION FLOYD | abri.engenderhealth.org
2. **T2.**Mano, M. Morris and Ciletti Michael D., *"Digital Design"*, Pearson Education, 6th edition

., [PDF] Digital Design By M. Morris Mano, Michael D Ciletti Book Free Download (studymaterialz.in)

References:

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

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

E-Resources:

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)

Course Code: ECE2052	**Course Title: Digital Design Lab** **Type of Course: School Core [Lab Only]**	**L-T-P-C**	**0**	**0**	**2**	**1**
Version No.	1.0					
Course Pre-requisites	**NIL**					
Anti-requisites	**NIL**					
Course Description	This course gives knowledge through the simulation software for design and implementation of various digital logic circuit. The theoretical concepts and working principles of combinational and sequential logic circuits are verified through implementation using openCV. This course provides necessary learning for fundamental concepts of computer and communication engineering. The purpose of this course is to support the students to exhibit the design and analysis skills. The course enhances the Design and Implementation abilities through laboratory tasks. Additionally, it provides an opportunity to verify the theoretical knowledge.					
Course Objectives	The objective of the course is to familiarize the learners with the design concepts of digital logic circuit and attain Skill Development through Experiential Learning by simulation-based laboratory Assessments.					
Course Outcomess	On successful completion of the course the students shall be able to CO1. **Interpret** Boolean laws and the logic gates functionality using available circuit elements in simulator platforms. CO2. **Apply** the fundamental laws and rules of Boolean algebra for simplified circuit simulation using any simulator tool. CO3. **Develop** combinational logic circuits by connecting offered gates for basic arithmetic operation CO4. **Make use of** circuit elements for ensuring the digital logic circuit's truth table perfection. CO5. **Examine** the performance of sequential logic circuit simulation using flip-flop and latch elements CO6. **Demonstrate** the operation of clocked and edge-triggered sequential components in an open-source simulation environment					

LIST OF EXPERIMENTS **30 Sessions**

S. No	Name of the Experiment/Exercise
1	Logic Gates
2	Boolean Algebra
3	Adder
4	Subtractor
5	Magnitude Comparator
6	Design of Combinational Circuits

7	Multiplexer, Demultiplexer & Applications of MUX
8	Flip-Flops
9	Shift Registers
10	Counter Design

Targeted Application & Tools that can be used: Targeted Applications: This course is contributed for placement in core companies, research & development work and also useful to know the existing & developing communications.

Professionally Used Software: CIRCUIT VERSE, HDL,VHDL

Project Based Learning proposed for this course

NA

Text Books:

T1. Mano, M. Morris and Ciletti Michael D., "Digital Design", Pearson Education, 6th edition

T2. Thomas L. Floyd "DIGITAL LOGIC DESIGN" , Pearson Education, fourth edition.

References:

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

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

- Presidency Lab Manual- for ECE2052 DIGITAL DESIGN LAB

E-Resources:

1. eBook1: Mano, M. Morris and Ciletti Michael D., *"Digital Design"*, Pearson Education. Book Free Download (studymaterialz.in)

[PDF] Digital Design By M. Morris Mano, Michael D Ciletti Book Free Download

2. eBook2:Floyd "DIGITAL LOGIC DESIGN" fourth edition- ePub, eBook- [PDF] DIGITAL LOGIC DESIGN FOURTH EDITION FLOYD | abri.engenderhealth.org.
3. NPTEL Course- NPTEL :: Electrical Engineering - NOC:Digital Electronic Circuits
4. Digital Logic Design PPT Slide 1 (iare.ac.in)
5. 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

Course Code: EEE1200	**Course Title: Basics of Electrical and Electronics Engineering. Type of Course: Theory - ESC**	**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:** **CO1: Apply** basic law**s** of Electrical Engineering to compute voltage, currents and other parameters in the circuits. **CO2: Discuss** various fundamental parameters appearing in the characteristics of semiconductor devices and their applications. **CO3: Summarize** the operations of different biasing configurations of BJTs and amplifiers. **CO4: Discuss** the performance characteristics and applications of various electrical Machines.	
Course Content:		
Module 1	**Introduction to Electrical Circuits**	**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**	**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, Zener diode, characteristics and its applications like voltage regulator.		
Module 3	**Transistors and its Applications**	**10 Sessions**
Transistor characteristics, Current components, BJT Configurations (CB, CC, CE configurations) and their current gains. Operating point, Biasing, Fixed Bias, and load line analysis. Single Stage amplifier. 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**	**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.		
Self-Learning Topics:		

Clipping and clamping circuits, Stabilization Techniques, Voltage divider bias and its stability factor, Multistage amplifier, Darlington pair.

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):

3. Kothari D. P. & Nagrath I. J., "Basic Electrical and Electronics Engineering", Tata McGraw-Hill
4. Education
5. 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.
6. A.P.Malvino, Electronic Principles,7thEdition, Tata McGraw Hill,2007
7. J. Millman, C. C. Halkias and C. D. Parikh, "Millman's Integrated Electronics", McGraw Hill Education, 2nd Edition.
8. 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: **EEE1251**	**Course Title: Basics of Electrical and Electronics Engineering Laboratory** **Type of Course: ESC - 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, instruments and materials. 9) Locate faults in systems. 10) 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 Outcomes	On successful completion of the course, the students shall be able to: 1. **Demonstrate** the behavior of passive elements and fundamental electrical circuit laws 2. **Determine** the power factor, impedance, and phase relationship of R and RL loads under AC supply conditions. 3. **Develop** electrical circuits and three-phase supply systems using MATLAB/Simulink to study resistance, voltage, and current characteristics. 4. **Analyze** the performance characteristics of electrical machines, including single-phase transformers and three-phase induction motors, under different operating conditions. 5. **Examine** the output characteristics and performance parameters of half-wave and full-wave rectifiers. 6. **Illustrate** the performance characteristics of Zener diode and MOSFET under different operating conditions.
Course Content:	List of Laboratory Tasks: **Experiment No 1:**Familiarization of passive elements. Level 1: Identify and understand the function and ratings of basic passive elements. Level 2: Study the behavior of passive elements in simple electrical circuits by measuring voltage, current, and impedance. **Experiment No 2:**Experimental Verification of Kirchhoff's Voltage Law (KVL) and Kirchhoff's Current Law (KCL). Level 1: Study and Verify KVL and KCL for the given electrical Circuit. **Experiment No 3:**Determination of Power Factor, and Impedance for R and RL Loads. Level 1: Determine the power factor and impedance of R and RL loads under steady-state AC supply conditions. Level 2: Analyze the effect of inductance on power factor, impedance, and phase angle for RL loads under varying load conditions. **Experiment No 4:**Modeling and Simulation of effective resistances in different circuit connections Using Simulink. Level 1: Model and simulate series and parallel resistor circuits in Simulink and determine their effective resistance. Level 2: Analyze the effect of different resistor combinations on total resistance, current distribution, and voltage drop. **Experiment No 5:**Simulation and analysis of line and phase voltages in Star-connected and Delta- connected supply systems. Level 1: Model and simulate star-connected and delta-connected three-phase

	supply systems and observe line and phase voltages. Level 2: Analyze and compare the relationship between line and phase voltages in star and delta connections. **Experiment6:** Determination of transformation ratio of a Single-Phase Transformer Under different tapping conditions. Level 1: Determine the transformation ratio of a single-phase transformer under no-load condition for various tap settings. Level 2: Analyze the effect of different tapping conditions on secondary voltage, regulation, and transformer performance. **Experiment No. 7:** Study of the speed-torque characteristics of a Three-Phase Induction Motor. Level 1: Study and analyze the speed–torque characteristics of a three-phase induction motor. Level 2: Evaluate the effect of supply voltage and current on motor performance parameters such as slip, and torque. **Experiment 8:** Performance Analysis of Half-Wave and Full-Wave Rectifier Circuits Using Simulink. Level 1: Model and simulate half-wave and full-wave rectifier circuits in Simulink and analyze their output waveforms. Level 2: Evaluate and compare rectifier performance parameters such as ripple factor, efficiency, and output voltage. **Experiment 9:** Study of the characteristics of a Zener Diode. Level 1: Study the forward and reverse V–I characteristics of a Zener diode using DC analysis. Level 2: Analyze the Zener breakdown characteristics and determine the Zener voltage and regulation behavior. **Experiment 10:** Study of the Output and Transfer Characteristics of a MOSFET. Level 1: Study, analyze and implement the output characteristics of a MOSFET under different gate-source voltage conditions and observe their results. Level 2: Analyze the transfer characteristics of a MOSFET and evaluate key parameters such as threshold voltage and transconductance.

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: MATLAB SIMULINK

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:

1. D. P. Kothari and I. J. Nagrath, Basic Electrical Engineering, 4th Edition, McGraw Hill Education, New Delhi, 2019.
2. B. L. Theraja and A. K. Theraja, A Textbook of Electrical Technology, Volume-I: Basic Electrical Engineering in SI Units, 24th Edition, S. Chand Publishing, New Delhi, 2024.

References

1. John Hiley, Keith Brown and Ian McKenzie-Smith, Hughes Electrical and Electronic Technology,

12th Edition, Pearson Education, 2016.
2. S. K. Bhattacharya and D. Devaraj, Basic Electrical and Electronics Engineering, 2nd Edition, Pearson Education India, 2017.
3. T. K. Nagsarkar and M. S. Sukhija, Principles of Basic Electrical Engineering, 1st Edition, Oxford University Press, 2018.
4. Robert L. Boylestad and Louis Nashelsky, Electronic Devices and Circuit Theory, 12th Edition, Pearson Education, 2019.
5. A. K. Maini and Varsha Agrawal, Electronic Devices and Circuits, 2nd Edition, Wiley India, 2019.
6. Adel S. Sedra and Kenneth C. Smith, Microelectronic Circuits, 8th Edition, Oxford University Press, 2019.

Online Resources:

1. https://presidencyuniversity.linways.com
2. https://www.digimat.in/nptel/courses/video/108105112/L01 "Fundamentals of Electrical Engineering - Basic Concepts, Examples"
3. 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
4. NPTEL lectures on "Rectifiers", by Prof. G. Bhuvaneswari, IIT Delhi https://nptel.ac.in/courses/108102145
5. NPTEL Lectures on "Electrical Machines II", by Prof. G. Sridhara Rao, IIT Madras. https://nptel.ac.in/courses/108106072

Topics relevant to "EMPLOYABILITY SKILLS": 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.

Topics relevant to "HUMAN VALUES & PROFESSIONAL ETHICS": Standard test methods, safety procedure.

Course Code: MEC1006	**Course Title: Engineering Graphics Type of Course: School Core & Theory Only**	**L- T-P- C**	**2**	**0**	**0**	**2**
VersionNo.	1.2					
CoursePre-requisites	**NIL**					
Anti-requisites	**NIL**					
CourseDescription	Thecourseisdesignedwiththeobjectiveofgivinganoverviewofengineering graphics.Itisintroductoryinnatureandacquaintsthestudentswiththetechniquesusedtocreateengineeringdrawings. The course emphasizes on projection ofpoints,lines,planesandsolids and isometricprojections.					
CourseObjective	The objective of the course is to familiarize the learners with the concepts of "Engineering Graphics" and attain SKILL DEVELOPMENT through Problem solving methodologies.					

CourseOutcomes	Onsuccessfulcompletionofthiscoursethestudentsshallbeableto: 1. Demonstratecompetencyof Engineering Graphics asperBISconventionsandstandards. 2. ComprehendthetheoryofprojectionfordrawingprojectionsofPoints,LinesandPlanesunder differentconditions. 3. PreparemultivieworthographicprojectionsofSolidsbyvisualizingthemindifferent positions. 4. Preparepictorialdrawingsusingtheprinciplesofisometricprojectionstovisualizeobjectsinthreedimensions.	
CourseContent:		
Module1	**Introduction toDrawing**	**02Sessions**
Topics: Introduction,drawinginstrumentsandtheiruses,relevantBISconventionsandstandards,Lettering,Lineconventions,dimensioning,Selectionofdrawing sheetsizeandscale.		
Module2	**Orthographic projectionsof Points,Straight LinesandPlane Surfaces**	**10Sessions**
Topics: Introduction, Definitions – Elements of projection and methods of projection, Planes of projection,referencelineandconventionsadopted.Firstangleandthirdangleprojections.ProjectionofPointsinall4 quadrants. Projections of Straight Lines (located in first quadrant/first angle projection only): True and apparentlengths, true and apparent Inclinations to reference planes. (No application problems). Projection ofPlane 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 methodonly.		
Module3	**OrthographicProjectionsof Solids**	**10Sessions**
Topics: Introduction,Projectionofrightregularprisms,pyramids,cone,hexahedronandtetrahedronindifferentpositions(ProblemsrestingonHPonly andFirstangleprojection).		
Module4	**IsometricProjections ofSolids (Usingisometricscale only)**	**8Sessions**
Topics: Introduction,Isometricscale,Isometricprojectionsofrightregularprisms,cylinders,pyramids, cones and theirfrustums, spheres and hemispheres, hexahedron (cube), andcombination of 2 solids, conversion of orthographic view to isometric projection of simpleobjects.		
TextBook: 1.N.D.Bhatt,"EngineeringDrawing:PlaneandSolidGeometry,"CharotarPublishingHousePvt.Ltd.		

References: **1.** K.R.Gopalakrishna,"EngineeringGraphics",SubhashPublishers,Bangalore. **2.** D.M.Kulkarni,A.P.Rastogi,A.K.Sarkar, "EngineeringGraphicswithAutoCAD,"PrenticeHall. **3.** D.A.Jolhe,"EngineeringDrawingwithIntroductiontoAutoCAD,"TataMcGrawHill. **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.

List of Professional Core Courses (PCC)

Course Code: CSE2200	**Course Title: Program Solving Using C** **Type of Course: Theory Course**	**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 ableto develop logics which will help them to create programs and applications in C. ACAlso bylearningthebasicprogrammingconstructs theycaneasilyswitchover toanyotherlanguageinfuture.					
Course Objective	TheobjectiveofthecourseistofamiliarizethelearnerswiththeconceptsofProblemSolvingUsingC andattainEmployabilitythroughProblemSolvingMethodologies.					
Course Outcomes	**Onsuccessfulcompletion ofthiscoursethestudentsshallbeableto:** 1. Describe algorithms with flowcharts for solving problems [Understand] 2. Demonstrate the application of C programming constructs in developing simple programs [Apply] 3. Implement applications using arrays and strings [Apply] 4. Apply functional programming principles to construct modular and reusable software components for solving complex problem [Apply] 5. Develop applications in C using structures and unions [Apply] 6. Differentiate between Sequential and Random Access File Processing through applications. [Analyze]					
Course Content:						
Module 1	**IntroductiontoCLanguage**	**6 Sessions**				
Topics: Introduction to Programming – Algorithms – Pseudo Code -Flow Chart – Compilation – Execution – PreprocessorDirectives (#define, #include, #undef) - Overview of C – Constants, Variables and Data types – Operators andExpressions – Managing Input and Output Operations – Decision Making and Branching - Decision Making andLooping.						
Module 2	**IntroductiontoArrays andStrings**	**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 TwoDimensionalArrays.ExamplePrograms– Matrixoperations.**Strings:**Introduction-DeclaringandInitializingString Variables–ReadingStringsfrom Terminal–Writing StringtoScreen– StringHandlingFunctions.						

Module 3	FunctionsandPointers	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 PointerVariables–InitializationofVariables –PointerOperators–PointerArithmetic–ArraysandPointers–Parameter Passing:PassbyValue,PassbyReference.		
Module 4	**StructuresandUnion**	**6 Sessions**
Topics: **Structures:**Introduction–DefiningaStructure–DeclaringStructureVariable–AccessingStructureMembers–ArrayofStructures –Arrayswithin Structures–**Union:**Introduction–DefiningandDeclaringUnion–DifferenceBetween Unionand Structure.		
Module 5	**Filehandling**	**6 Sessions**
Topics: Files:DefiningandOpeningaFile –ClosingaFile–Input/OutputOperationsonFile –RandomAccessFiles		
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: Laboratory Course**	**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 ableto develop logics which will help them to create programs and applications in C. ACAlso bylearningthebasicprogrammingconstructs theycaneasilyswitchover toanyotherlanguageinfuture.					
Course Objective	TheobjectiveofthecourseistofamiliarizethelearnerswiththeconceptsofProblemSolvingUsing C andattainEmployabilitythroughProblemSolvingMethodologies.					
Course Outcomes	**Onsuccessfulcompletion ofthiscoursethestudentsshallbeableto:** 1. Describe algorithms with flowcharts for solving problems [Understand] 2. Demonstrate the application of C programming constructs in developing simple programs [Apply] 3. Implement applications using arrays and strings [Apply] 4. Apply functional programming principles to construct modular and reusable software components for solving complex problem [Apply]					

	5. Develop applications in C using structures and unions [Apply] 6. Differentiate between Sequential and Random Access File Processing through applications. [Analyze]
Course Content:	

List of Practicals: **60 Sessions**

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):

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: CSE2251	**Course Title: Data Communications and Computer Networks** **Type of Course: Theory Course (PCC)**	**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 Objectives	To develop a strong foundation in data communication and computer networks by enabling students to understand networking concepts, analyze network architectures and protocols, explore transmission media and switching techniques, and evaluate network performance and emerging networking technologies.					
Course Outcomes	On successful completion of the course, the students shall be able to: **CO1:** Explain the fundamental concepts of data communication and computer networks. [Understand] **CO2:** Analyze the functions and protocols of the Data Link Layer. [Analyze] **CO3:** Apply IP addressing schemes and routing techniques in computer networks. [Apply] **CO4:** Explain the working principles of the Transport and Application layers. [Understand] **CO5:** Analyze network protocols and evaluate their performance for efficient data communication. [Analyze]					
Course Content:						
Module 1	**Introduction and Physical Layer**					**12 Sessions**
Topics: 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**					**12 Sessions**

Topics: 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**	**12 Sessions**
Topics: 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**	**9 Sessions**
Topics: 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. 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. "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: CSE2252	**Course Title: Data Communications and Computer Networks Lab** **Type of Course: Laboratory Course (PCC)**	**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	To develop practical skills in computer networking by enabling students to gain hands-on experience with networking tools and protocols, simulate and analyze data communication processes, implement socket programming for network applications, and evaluate network performance through experimental studies.
Course Outcomes	On successful completion of this course, students will be able to: **CO1:** Design and configure basic network topologies using routers, switches, and end devices. [Apply] **CO2:** Analyze and troubleshoot network connectivity and performance issues using networking tools and simulators. [Analyze] **CO3:** Demonstrate the working of key networking protocols such as TCP/IP, ARP, ICMP, and DHCP. [Apply] **CO4:** Apply IP addressing and subnetting techniques for efficient network resource management. [Apply] **CO5:** Evaluate network performance and optimize configurations for improved reliability and efficiency. [Evaluate]

Course Content:

List of Laboratory Tasks: **30 Sessions**

WEEK 1: Introduction to Networks

Lab Sheet 1: Network Basics

Level 1: Identify different types of networks (LAN, WAN, MAN) and list their characteristics.

Level 2: Analyze a real-world network setup and classify its components and topology.

Lab Sheet 2: Network Topologies

Level 1: Draw and explain Bus, Star, Ring, and Mesh topologies.

Level 2: Compare topologies and suggest the best topology for a given scenario.

WEEK 2: Transmission Media & Models

Lab Sheet 3: Transmission Media

Level 1: Identify guided and unguided transmission media with examples.

Level 2: Compare different transmission media based on bandwidth, cost, and reliability.

Lab Sheet 4: OSI & TCP/IP Models

Level 1: Explain the layers of OSI and TCP/IP models.

Level 2: Map OSI layers with TCP/IP layers and justify their functions.

WEEK 3: Physical Layer Concepts

Lab Sheet 5: Signals & Transmission

Level 1: Differentiate between analog and digital signals.

Level 2: Analyze signal transmission in real-world communication systems.

Lab Sheet 6: Multiplexing

Level 1: Explain FDM, TDM, and WDM.

Level 2: Design a multiplexing scenario for multiple users.

WEEK 4: Spread Spectrum & Encoding

Lab Sheet 7: Spread Spectrum

Level 1: Explain FHSS and DSSS.

Level 2: Compare spread spectrum techniques in wireless communication.

Lab Sheet 8: Encoding Techniques

Level 1: Demonstrate digital encoding techniques.

Level 2: Analyze encoding schemes for efficient transmission.

WEEK 5: Error Detection & Correction

Lab Sheet 9: Error Detection

Level 1: Implement Parity and LRC.

Level 2: Implement CRC algorithm.

Lab Sheet 10: Error Correction

Level 1: Implement Hamming Code.

Level 2: Analyze error correction efficiency.

WEEK 6: Flow Control

Lab Sheet 11: Stop and Wait

Level 1: Simulate Stop-and-Wait protocol.

Level 2: Analyze efficiency and delay.

Lab Sheet 12: Sliding Window

Level 1: Implement Sliding Window protocol.

Level 2: Compare with Stop-and-Wait.

WEEK 7: Multiple Access Protocols

Lab Sheet 13: CSMA/CD

Level 1: Explain working of CSMA/CD.

Level 2: Simulate collision detection.

Lab Sheet 14: CSMA/CA

Level 1: Explain CSMA/CA.

Level 2: Compare CSMA/CD and CSMA/CA.

WEEK 8: IEEE Standards

Lab Sheet 15: Ethernet (802.3)

Level 1: Study Ethernet frame structure.

Level 2: Analyze Ethernet performance.

Lab Sheet 16: Wireless LAN (802.11)

Level 1: Study Wi-Fi architecture.

Level 2: Analyze wireless communication challenges.

WEEK 9: Network Layer Basics

Lab Sheet 17: IP Addressing

Level 1: Identify IPv4 address classes.

Level 2: Convert IPv4 to binary and vice versa.

Lab Sheet 18: IPv6

Level 1: Understand IPv6 format.

Level 2: Compare IPv4 and IPv6.

WEEK 10: Subnetting

Lab Sheet 19: Subnetting Basics

Level 1: Calculate subnet mask.

Level 2: Design subnet for given requirements.

Lab Sheet 20: Advanced Subnetting

Level 1: Perform CIDR calculations.

Level 2: Solve real-world subnetting problems.

WEEK 11: Routing

Lab Sheet 21: Distance Vector Routing

Level 1: Explain RIP algorithm.

Level 2: Simulate routing table updates.

Lab Sheet 22: Link State Routing

Level 1: Explain OSPF.

Level 2: Compare RIP and OSPF.

WEEK 12: Advanced Routing

Lab Sheet 23: Multicast Routing

Level 1: Explain multicast concepts.

Level 2: Analyze MOSPF/DVMRP.

Lab Sheet 24: Modern Networking

Level 1: Study VXLAN/EVPN.

Level 2: Analyze virtual networking concepts.

WEEK 13: Transport Layer

Lab Sheet 25: TCP & UDP

Level 1: Compare TCP and UDP.

Level 2: Analyze use cases.

Lab Sheet 26: Congestion Control

Level 1: Explain congestion control methods.

Level 2: Analyze RED/DECbit techniques.

WEEK 14: Application Layer

Lab Sheet 27: DNS & HTTP

Level 1: Explain DNS working.

Level 2: Analyze HTTP request-response.

Lab Sheet 28: Email Protocols

Level 1: Explain SMTP, POP3, IMAP.

Level 2: Compare email protocols.

WEEK 15: Network Applications

Lab Sheet 29: SNMP & Web Services

Level 1: Explain SNMP architecture.

Level 2: Analyze web services.

Lab Sheet 30: Virtual Networking

Level 1: Study virtual networking basics.

Level 2: Analyze real-world virtual network implementation.

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.
Problem Solving: Choose and appropriate devices and implement various network concepts.
Programming: Simulation of any network using NS2.

Text Book(s):

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. "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: CSE2254	**Course Title:** Data Structures Lab **Type of Course:Laboratory Course**	**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 Outcomes	**On successful completion of the course the students shall be able to:** CO1. Understand the concepts of basic data structures such as arrays, stacks, and queues and their operations. (Understand) CO2. Implement linear data structures like linked lists for solving real-world problems. (Apply)					

	CO3. Apply appropriate non-linear data structures such as trees and graphs for given scenarios. (Apply) CO4. Demonstrate and implement various searching and sorting algorithms. (Apply) **CO5.** Analyze the efficiency of different data structures and algorithms in terms of time and space complexity. (Analyze)
Course Content:	

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.
R2Programming 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: CSE2253	**Course Title: Data Structures** **Type of Course: Theory Course (PCC)**	**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 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	To develop a strong foundation in data structures by enabling students to understand fundamental data structures and their operations, analyze algorithm efficiency, implement linear and non-linear data structures and apply appropriate techniques to solve real-world problems while evaluating trade-offs between different data structures.
Course Outcomes	On successful completion of this course, students will be able to: **CO1:** Explain the concepts of basic data structures such as arrays, stacks, and queues and their operations. [Understand] **CO2:** Apply linked list data structures to solve real-world problems. [Apply] **CO3:** Utilize appropriate non-linear data structures such as trees and graphs for given problem scenarios. [Apply] **CO4:** Implement and demonstrate various searching and sorting algorithms. [Apply] **CO5:** Analyze the efficiency of data structures and algorithms in terms of time and space complexity. [Analyze]

Course Content:

Module 1	**Introduction to Data Structure and Linear Data Structure – Stacks and Queues**	**9 Sessions**

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**	**12 Sessions**

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**	**12 Sessions**

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	6 Sessions

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	6 Sessions

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

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.

R2Programming 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: CSE2254	**Course Title: Data Structures Lab** **Type of Course:Laboratory Course (PCC)**	**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	To develop programming and problem-solving skills by enabling students to implement various data structures, analyze their performance through experimentation and apply structured approaches to solve computational problems effectively.					
Course Outcomes	On successful completion of this course, students will be able to: **CO1:** Explain the concepts of fundamental data structures such as arrays, stacks, and queues and their operations. [Understand] **CO2:** Implement linear data structures such as linked lists to solve real-world problems. [Apply] **CO3:** Utilize appropriate non-linear data structures such as trees and graphs for given problem scenarios. [Apply] **CO4:** Implement and demonstrate searching and sorting algorithms. [Apply] **CO5:** Analyze the efficiency of different data structures and algorithms in terms of time and space complexity. [Analyze]					
Course Content:		**30 Sessions**				

List of Laboratory Tasks:30 Sessions

WEEK 1: Introduction & Arrays

Lab Sheet 1: Arrays Basics

Level 1: Write a program to create an array and perform insertion and deletion operations.

Level 2: Write a program to find largest, smallest, and average elements in an array.

Lab Sheet 2: Array Applications

Level 1: Write a program to reverse an array.

Level 2: Write a program to merge two arrays and remove duplicates.

WEEK 2: Stack

Lab Sheet 3: Stack Implementation

Level 1: Implement stack using array (push and pop operations).

Level 2: Write a program to check overflow and underflow conditions.

Lab Sheet 4: Stack Applications

Level 1: Write a program to convert infix expression to postfix.

Level 2: Write a program to evaluate postfix expression.

WEEK 3: Queue

Lab Sheet 5: Queue Implementation

Level 1: Implement queue using array (enqueue and dequeue).

Level 2: Implement circular queue.

Lab Sheet 6: Queue Applications

Level 1: Write a program to simulate a queue in real-life scenario (ticket booking).

Level 2: Implement priority queue.

WEEK 4: Linked List Basics

Lab Sheet 7: Singly Linked List

Level 1: Create a singly linked list and perform insertion.

Level 2: Perform deletion and display operations.

Lab Sheet 8: Linked List Applications

Level 1: Count number of nodes in a linked list.

Level 2: Reverse a linked list.

WEEK 5: Advanced Linked List

Lab Sheet 9: Circular Linked List

Level 1: Create and traverse a circular linked list.

Level 2: Perform insertion and deletion in circular linked list.

Lab Sheet 10: Linked List Problems

Level 1: Merge two linked lists.

Level 2: Detect loop in linked list.

WEEK 6: Recursion

Lab Sheet 11: Recursion Basics

Level 1: Write a recursive program for factorial.

Level 2: Write a recursive program for Fibonacci series.

Lab Sheet 12: Recursion Applications

Level 1: Solve Tower of Hanoi using recursion.

Level 2: Compare recursive and iterative approaches.

WEEK 7: Trees Basics

Lab Sheet 13: Binary Tree

Level 1: Create a binary tree and perform traversal.

Level 2: Implement preorder, inorder, and postorder traversal.

Lab Sheet 14: Tree Properties

Level 1: Count nodes and height of tree.

Level 2: Check if tree is balanced.

WEEK 8: Binary Search Tree

Lab Sheet 15: BST Implementation

Level 1: Insert elements into BST.

Level 2: Delete elements from BST.

Lab Sheet 16: BST Applications

Level 1: Search an element in BST.

Level 2: Find minimum and maximum elements.

WEEK 9: Advanced Trees

Lab Sheet 17: AVL Trees

Level 1: Implement AVL tree insertion.

Level 2: Perform rotations (LL, RR, LR, RL).

Lab Sheet 18: Heaps

Level 1: Implement max heap.

Level 2: Perform heap sort.

WEEK 10: Expression Trees & Red-Black Trees

Lab Sheet 19: Expression Tree

Level 1: Construct expression tree from postfix.

Level 2: Evaluate expression tree.

Lab Sheet 20: Red-Black Tree

Level 1: Understand insertion rules.

Level 2: Implement basic operations.

WEEK 11: Graph Basics

Lab Sheet 21: Graph Representation

Level 1: Represent graph using adjacency matrix.

Level 2: Represent graph using adjacency list.

Lab Sheet 22: Graph Traversal

Level 1: Implement BFS.

Level 2: Implement DFS.

WEEK 12: Graph Algorithms

Lab Sheet 23: Shortest Path

Level 1: Implement Dijkstra's algorithm.

Level 2: Analyze shortest path in weighted graph.

Lab Sheet 24: Minimum Spanning Tree

Level 1: Implement Prim's algorithm.

Level 2: Implement Kruskal's algorithm.

WEEK 13: Hashing

Lab Sheet 25: Hash Tables

Level 1: Implement hashing using division method.

Level 2: Handle collisions using chaining.

Lab Sheet 26: Advanced Hashing

Level 1: Implement linear probing.

Level 2: Compare static vs dynamic hashing.

WEEK 14: Searching

Lab Sheet 27: Searching Techniques

Level 1: Implement sequential search.

Level 2: Implement binary search.

Lab Sheet 28: Searching Applications

Level 1: Compare searching techniques.

Level 2: Analyze time complexity.

WEEK 15: Sorting

Lab Sheet 29: Basic Sorting

Level 1: Implement bubble sort and selection sort.

Level 2: Implement insertion sort.

Lab Sheet 30: Advanced Sorting

Level 1: Implement quick sort.

Level 2: Implement merge sort and compare performance

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.

R2Programming 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: **CSE2255**	**Course Title: Object Oriented Programming Using Java** **Type of Course: Theory Course (PCC)**	**L-T- P- C**	**3**	**0**	**0**	**3**
Version No.	2.0					
Course Pre-requisites	Nil					
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 Outcomes	On successful completion of the course, students shall be able to: **CO1:** Describe fundamental programming concepts. [Understand] **CO2:** Apply the concepts of classes, objects, and methods to solve problems. [Apply] **CO3:** Apply the concepts of arrays and strings. [Apply] **CO4:** Implement inheritance and polymorphism to develop secure applications. [Apply] **CO5:** Apply the concepts of interfaces and error-handling mechanisms. [Apply]	
Course Content:		
Module 1	**Basic Concepts of Programming and Java**	**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**	**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**	**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**	**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**	**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

1. Herbert Schildt, "The Complete Reference Java 2", Tata McGraw Hill Education, 11th Edition,2019.

References

1. Cay S Horstmann and Cary Gornell, "CORE JAVA volume I-Fundamentals", Tenth Edition, Pearson 2015.
2. James W. Cooper, "Java TM Design Patterns – A Tutorial", Addison-Wesley Publishers.4th Edition, 2000.
3. E. Balagurusamy, "Programming with Java", Tata McGraw Hill Education, 6th Edition, 2019.

E book link 1: http://rmi.yaht.net/bookz/core.java/9780134177373-Vol-1.pdf **E book link 2:**Java(tm) Design Patterns: A Tutorial([PDF] [7qmsenjl97t0] (vdoc.pub)

Web resources

https://youtube.com/playlist?list=PLu0W_9lII9agS67Uits0UnJyrYiXhDS6q

https://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.

This is attained through assessment component mentioned in course handout.

Course Code: CSE2256	**Course Title: Object Oriented Programming Using Java Lab Type of Course: Laboratory Course (PCC)**	**L-T- P- C**	**0**	**0**	**2**	**1**
Version No.	2.0					
Course Pre-requisites	NIL					
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 Outcomes	On successful completion of the course, students shall be able to: **CO1:** Demonstrate fundamental programming concepts. [Apply] **CO2:** Apply the concepts of classes, objects, and methods to solve problems. [Apply] **CO3:** Apply the concepts of arrays and strings. [Apply] **CO4:** Implement inheritance and polymorphism to develop secure applications. [Apply] **CO5:** Apply the concepts of interfaces and error-handling mechanisms. [Apply]

Course Content: 30 Sessions

List of Laboratory Tasks:

Week 1:

Lab Sheet 1: Java Program Structure & Basic I/O

Level 1:

Write a Java program to display "Hello, Java".

Demonstrate use of variables and data types.

Level 2:

Write a Java program to take user input (name, age) and display formatted output.

Week 2:

Lab Sheet 2: Operators and Expressions

Level 1:

Write a Java program to perform arithmetic operations (+, -, *, /, %).

Level 2:

Develop a program to evaluate a complex expression using relational and logical operators.

Week 3:

Lab Sheet 3: Control Statements – Branching

Level 1:

Write a Java program to check whether a number is even or odd using if-else.

Level 2:

Implement a menu-driven calculator using switch-case.

Week 4:

Lab Sheet 4: Control Statements – Looping

Level 1:

Write programs using loops:

1. Print numbers from 1 to N
2. Find factorial of a number

Level 2:

Generate Fibonacci series and check whether a number is prime.

Week 5:

Lab Sheet 5: Classes and Objects

Level 1:

Create a class Student with data members and display details using object.

Level 2:

Create multiple objects and demonstrate accessing members using reference variables.

Week 6:

Lab Sheet 6: Methods and Access Specifiers

Level 1:

Write a class with methods to perform addition and subtraction.

Level 2:

Demonstrate use of public, private, and protected access specifiers.

Week 7:

Lab Sheet 7: Constructors and Overloading

Level 1:

Implement a class with a default constructor and parameterized constructor.

Level 2:

Demonstrate constructor overloading and use of this keyword.

Week 8:

Lab Sheet 8: Static Keyword & Nested Classes

Level 1:

Write a program demonstrating static variables and static methods.

Level 2:

Create a nested class and access its members from the outer class.

Week 9:

Lab Sheet 9: Arrays (1D and 2D)

Level 1:

Write programs for:

1. Sum of array elements
2. Find largest element

Level 2:

Perform matrix addition and multiplication using 2D arrays.

Week 10:

Lab Sheet 10: Array of Objects

Level 1:

Create an array of objects (e.g., Employee class).

Level 2:

Perform operations like sorting objects based on a field.

Week 11:

Lab Sheet 11: Strings and StringBuffer

Level 1:

Perform basic string operations: Length, concatenation, comparison

Level 2:

Demonstrate mutable operations using StringBuffer and StringBuilder.

Week 12:

Lab Sheet 12: Inheritance

Level 1:

Create a base class and derived class using single inheritance.

Level 2:

Implement multilevel inheritance and demonstrate use of super keyword.

Week 13:

Lab Sheet 13: Polymorphism & Abstract Classes

Level 1:

Demonstrate method overriding (runtime polymorphism).

Level 2:

Create an abstract class and implement its methods in a subclass.

Week 14:

Lab Sheet 14: Exception Handling

Level 1:

Handle exceptions using try-catch-finally.

Level 2:

Create custom exceptions and demonstrate multiple catch blocks.

Week 15:

Lab Sheet 15: File Handling and Serialization

Level 1:

Write a Java program to Create a file & Write and read data using File I/O

Level 2:

Implement object serialization and deserialization.

Use File streams and buffered streams for efficient I/O.

Targeted Application & Tools that can be used : JDK /Eclipse IDE/Visual Studio Code / net Beans IDE.

Text Book

1. Herbert Schildt, "The Complete Reference Java 2", Tata McGraw Hill Education, 11th Edition,2019.

References

1. Cay S Horstmann and Cary Gornell, "CORE JAVA volume I-Fundamentals", Tenth Edition, Pearson 2015.

2. James W. Cooper, "Java TM Design Patterns – A Tutorial", Addison-Wesley Publishers.4th Edition, 2000.

3. 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

https://youtube.com/playlist?list=PLu0W_9lII9agS67Uits0UnJyrYiXhDS6q

https://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.

This is attained through assessment component mentioned in course handout.

Course Code: **CSE2257**	**Course Title: Computer Organization and Architecture** **Type of Course: Theory Course (PCC)**	**L-T-P- C**	**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	To develop a comprehensive understanding of computer system organization by enabling students to analyze processor architecture and instruction execution, explore memory hierarchy and cache mechanisms, examine input/output organization and interfacing and evaluate system performance and optimization techniques.					
Course Outcomes	On successful completion of this course, students will be able to: **CO1:** Describe the fundamental components of a computer system and their interconnections. [Remember] **CO2:** Explain the concepts of Instruction Set Architecture (ISA) and memory organization. [Understand] **CO3:** Apply techniques to perform arithmetic and logical operations in computer systems. [Apply] **CO4:** Explain the organization and functioning of processor and memory subsystems. [Understand] **CO5:** Analyze the design and performance aspects of computer architecture components for efficient system operation. [Analyze]					
Course Content:						
Module 1	**Basic Structure of Computer**					**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**	**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**	**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**	**11 Sessions**

Topics:
Basic Processing Unit: Fundamental Concepts, Single Bus organization, Control sequence, Execution of a Complete 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-Hill Higher 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.
2. Web References:
3. NPTEL Course on "Computer architecture and organization" IIT Kharagpur By Prof. Indranil Sengupta, Prof. Kamalika Datta. *https://nptel.ac.in/courses/106105163*
4. NPTEL Course on "Computer Organization", IIT Madras By Prof. S. Raman.
5. *https://nptel.ac.in/courses/106106092*
6. *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: **CSE2258**	**Course Title: Web Technologies** **Type of Course : Theory Course (PCC)**	**L-T-P- C**	**3 – 0 – 0 - 3**
Version No.	2.0		
Course Pre-requisites	NIL		
Anti-requisites	NIL		
Course Description	This course highlights the basic web design using Hypertext Markup Language and Cascading Style Sheets. Students will be trained in planning and designing effective web pages by writing code using current leading trends in the web domain, enhancing web pages with the use of page layout techniques, text formatting, graphics, images, and multimedia. The focus is on popular key technologies that will help students to build Internet- and web-based applications that interact with other applications and with databases.		
Course Objective	To develop a strong foundation in web technologies by enabling students to understand internet protocols, design and develop static and dynamic web applications, explore client-side scripting and frameworks, implement server-side programming with database integration, and evaluate modern web application architectures.		
Course Outcomes	On successful completion of this course, students will be able to: **CO1:** Explain the core concepts of web technologies including the Internet, web servers, and browsers. [Understand] **CO2:** Develop well-structured web pages using XHTML with text formatting, images, links, lists, tables, forms, and frames following web standards. [Apply] **CO3:** Design responsive and structured web page layouts using advanced CSS techniques such as positioning, floating, and multicolumn design. [Apply] **CO4:**		

	Apply JavaScript concepts to create dynamic and interactive web pages. [Apply] **CO5:** Implement dynamic server-side web applications using PHP, including control structures, arrays, superglobals, and file handling. [Apply] **CO6:** Develop modular web applications by integrating PHP with MySQL databases and applying object-oriented programming principles. [Create]

Course Content:

Module 1	**Introduction to XHTML**	**10 Sessions**

Topics:

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.

Module 2	**Advanced CSS**	**10 Sessions**

Topics:

CSS: Introduction to CSS, Defining & Applying a style, Creating style sheets, types of style sheet, selectors, CSS font properties, border properties, Box model, opacity, CSS pseudo class and pseudo-elements

Advanced CSS: Layout, Normal Flow, Positioning Elements, Floating Elements, Responsive Design, CSS Frameworks **XML:** Basics, demonstration of applications using XML

Module 3	**Fundamentals of JavaScript**	**10 Sessions**

Topics:

JavaScript: Introduction to JavaScript, Basic JavaScript Instructions, Functions, Methods & Objects, Decisions and Loops, Document Object Model, Event handling, handling window pop-ups, JavaScript validation.

Module 4	**PHP – Application Level**	**15 Sessions**

Topics:

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

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):

T1. Robert. W. Sebesta, "Programming the World Wide Web", Pearson Education, 8th Edition,2015.
T2. CSS Notes for Professionals, ebook available at https://books.goalkicker.com/CSSBook/ (Retrieved on Jan. 20, 2022)
T3. Deitel, Deitel, Goldberg,"Internet & World Wide Web How to Program", Fifth Edition, Pearson

Education,2021.

References:

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.

Topics related to development of "FOUNDATION": Web, WWW, Web browsers, Web servers, Internet.

CSS, PHP. Designing for healthcare.

For Skill Development through Experiential Learning techniques. This is attained through assessment component mentioned in course handout.

E-References:

1. W3schools.com
2. Developer.mozilla.org/en-US/docs/Learn
3. docs.microsoft.com
4. informit.com/articles/ The Relationship Between Web 2.0 and Social Networking
5. https://presiuniv.knimbus.com/user#/home

Course Code: **CSE2259**	**Course Title: Web Technologies Lab** **Type of Course: Laboratory Course (PCC)**	**L- T-P- C**	**0-0-2-1**
Version No.	2.0		
Course Pre-requisites	NIL		
Anti-requisites	NIL		
Course Description	This course highlights the basic web design using Hypertext Markup Language and Cascading Style Sheets. Students will be trained in planning and designing effective web pages by writing code using current leading trends in the web domain, enhancing web pages with the use of page layout techniques, text formatting, graphics, images, and multimedia. The focus is on popular key technologies that will help students to build Internet- and web-based applications that interact with other applications and with databases.		
Course Objective	To develop skills in designing and building interactive web applications by enabling students to create web pages using HTML, CSS, and JavaScript, implement server-side scripting techniques, integrate applications with databases, and develop complete web-based solutions.		
Course Outcomes	On successful completion of this course, students will be able to: **CO1:** Explain the core concepts of web technologies including the Internet, web servers, and browsers. [Understand] **CO2:** Develop well-structured web pages using XHTML with text formatting, images, links,		

	lists, tables, forms, and frames in accordance with web standards. [Apply] **CO3:** Design responsive and structured web page layouts using advanced CSS techniques such as positioning, floating, and multicolumn layouts. [Apply] **CO4:** Apply JavaScript concepts to develop dynamic and interactive web applications. [Apply] **CO5:** Implement dynamic server-side web applications using PHP, including control structures, arrays, superglobals, and file handling. [Apply] **CO6:** Develop modular web applications by integrating PHP with MySQL databases and applying object-oriented programming principles. [Create]

COURSE CONTENT:

List of Laboratory Tasks:30 Sessions

WEEK 1: Introduction to Web & XHTML

Lab Sheet 1: Web Basics

Level 1: Create a simple XHTML page explaining Web, WWW, browsers, and servers.

Level 2: Design a webpage illustrating Internet architecture using text and images.

Lab Sheet 2: XHTML Structure

Level 1: Create a basic XHTML document with proper syntax and structure.

Level 2: Design a webpage using headings, paragraphs, and formatting tags.

WEEK 2: XHTML Elements

Lab Sheet 3: Lists and Links

Level 1: Create ordered, unordered, and definition lists in XHTML.

Level 2: Create hyperlinks for navigation between multiple pages.

Lab Sheet 4: Images and Tables

Level 1: Insert images into a webpage.

Level 2: Design a table with rows, columns, and formatting.

WEEK 3: XHTML Forms & Frames

Lab Sheet 5: Forms

Level 1: Create a form with text fields and buttons.

Level 2: Design a registration form with validation requirements.

Lab Sheet 6: Frames

Level 1: Create a webpage using frames.

Level 2: Design a multi-frame layout for a website.

WEEK 4: Introduction to CSS

Lab Sheet 7: CSS Basics

Level 1: Apply inline and internal CSS styles.

Level 2: Create an external stylesheet and apply it to multiple pages.

Lab Sheet 8: Selectors

Level 1: Use element, class, and ID selectors.

Level 2: Apply multiple selectors for styling complex pages.

WEEK 5: CSS Properties

Lab Sheet 9: Font and Text

Level 1: Apply font properties (size, style, family).

Level 2: Design a styled webpage using text formatting.

Lab Sheet 10: Box Model

Level 1: Demonstrate margin, padding, and border.

Level 2: Design layout using box model concepts.

WEEK 6: Advanced CSS

Lab Sheet 11: Positioning

Level 1: Use static and relative positioning.

Level 2: Design layout using absolute and fixed positioning.

Lab Sheet 12: Floating & Layout

Level 1: Apply float properties.

Level 2: Create a multi-column layout using float.

WEEK 7: Responsive Design

Lab Sheet 13: Responsive Web Design

Level 1: Use media queries.

Level 2: Design a responsive webpage.

Lab Sheet 14: CSS Frameworks

Level 1: Use a CSS framework (Bootstrap basics).

Level 2: Design responsive webpage using Bootstrap.

WEEK 8: XML Basics

Lab Sheet 15: XML Structure

Level 1: Create a simple XML document.

Level 2: Display XML data in webpage.

Lab Sheet 16: XML Applications

Level 1: Validate XML structure.

Level 2: Use XML for data representation in web pages.

WEEK 9: JavaScript Basics

Lab Sheet 17: JavaScript Introduction

Level 1: Write basic JavaScript program for output.

Level 2: Perform arithmetic operations using JavaScript.

Lab Sheet 18: Control Statements

Level 1: Use if-else statements.

Level 2: Implement loops in JavaScript.

WEEK 10: Functions & DOM

Lab Sheet 19: Functions

Level 1: Create and call functions.

Level 2: Use functions with parameters and return values.

Lab Sheet 20: DOM Manipulation

Level 1: Access HTML elements using DOM.

Level 2: Modify content dynamically using JavaScript.

WEEK 11: Events & Validation

Lab Sheet 21: Event Handling

Level 1: Handle click events.

Level 2: Create interactive webpage using events.

Lab Sheet 22: Form Validation

Level 1: Validate input fields using JavaScript.

Level 2: Create complete form validation system.

WEEK 12: Advanced JavaScript

Lab Sheet 23: Objects & Methods

Level 1: Create JavaScript objects.

Level 2: Use methods and object properties.

Lab Sheet 24: Pop-ups

Level 1: Implement alert and prompt.

Level 2: Create dynamic pop-up-based interaction.

WEEK 13: PHP Basics

Lab Sheet 25: PHP Introduction

Level 1: Write basic PHP script.

Level 2: Handle form input using PHP.

Lab Sheet 26: PHP Control Structures

Level 1: Use if-else in PHP.

Level 2: Implement loops in PHP.

WEEK 14: PHP Applications

Lab Sheet 27: PHP Forms

Level 1: Process form data using PHP.

Level 2: Create login system using PHP.

Lab Sheet 28: PHP File Handling

Level 1: Read/write files using PHP.

Level 2: Store and retrieve data dynamically.

WEEK 15: Integration & Mini Project

Lab Sheet 29: Web Integration

Level 1: Combine XHTML, CSS, JavaScript.

Level 2: Build dynamic webpage using all technologies.

Lab Sheet 30: Mini Project

Level 1: Design a complete website.

Level 2: Develop full web application using XHTML, CSS, JS, PHP.

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):

T1. Robert. W. Sebesta, "Programming the World Wide Web", Pearson Education, 8th Edition,2015.
T2. CSS Notes for Professionals, ebook available at https://books.goalkicker.com/CSSBook/ (Retrieved on Jan. 20, 2022)
T3. Deitel, Deitel, Goldberg,"Internet & World Wide Web How to Program", Fifth Edition, Pearson Education,2021.

References:

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.

E-References:

1. W3schools.com
2. Developer.mozilla.org/en-US/docs/Learn
3. docs.microsoft.com
4. informit.com/articles/ The Relationship Between Web 2.0 and Social Networking
5. https://presiuniv.knimbus.com/user#/home

Topics related to development of "FOUNDATION": Web, WWW, Web browsers, Web servers, Internet.

CSS, PHP. Designing for healthcare For Skill Development through Experiential Learning techniques. This is attained through assessment component mentioned in course handout.

Course Code: CSE2260	**Course Title: Database Management Systems** **Type of Course: Theory Course (PCC)**	**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 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	To develop a strong foundation in database management systems by enabling students to understand database concepts and architecture, design relational databases using ER					

	modeling, apply normalization techniques to eliminate redundancy, execute SQL queries and transactions, and analyze query optimization and database performance.
Course Outcomes	On successful completion of this course, students will be able to: **CO1:** Explain the fundamental concepts of database systems, data models, schemas, and their role in modern applications. [Understand] **CO2:** Apply relational algebra operations to retrieve and manipulate data in relational databases. [Apply] **CO3:** Apply SQL queries and analyze query processing and optimization techniques. [Apply] **CO4:** Design and develop database systems by applying normalization principles, and demonstrate transaction processing, recovery, and security mechanisms. [Apply] **CO5:** Explain advanced database concepts and their applications in real-world scenarios. [Understand]

Course Content:

Module 1	**Introduction to Database Modelling and Relational Algebra**	**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**	**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 of expression, choosing evaluation plans, linear and bushy plans, dynamic programming algorithms.

Module 3	**Relational Database Design & Transaction Management**	**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), Join Dependencies (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**	**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, Native XML 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.

Assignments:

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: Laboratory Course (PCC)**	**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	To develop practical skills in database systems by enabling students to design and implement relational database schemas, perform SQL queries and operations, implement constraints, triggers, and stored procedures, and develop database-driven applications.					
Course Outcomes	On successful completion of this course, students will be able to: **CO1:** Create databases, tables, and constraints using SQL Data Definition Language (DDL) commands. [Understand] **CO2:** Apply SQL Data Manipulation Language (DML) commands to insert, update, delete, and retrieve data from relational databases. [Apply] **CO3:** Implement advanced SQL queries using joins, subqueries, views, indexes, and aggregate functions to solve real-world problems. [Apply] **CO4:** Design and normalize relational schemas and enforce integrity constraints using keys, triggers, and stored procedures. [Apply] **CO5:** Perform database transaction management, including concurrency control, backup, recovery, and security mechanisms using DBMS tools. [Apply]					
Course Content:						

List of Laboratory Tasks:30 Sessions

WEEK 1: Introduction to Database Concepts

Lab Sheet 1: DBMS Basics

Level 1: Identify schema and instance for a given database scenario.

Level 2: Analyze differences between traditional file systems and DBMS with examples.

Lab Sheet 2: Database Architecture

Level 1: Explain 3-schema architecture with diagram.

Level 2: Illustrate physical and logical data independence using examples.

WEEK 2: ER Modeling

Lab Sheet 3: ER Diagram Basics

Level 1: Draw ER diagram for a student database.

Level 2: Design ER diagram for a real-world system (library/hospital).

Lab Sheet 4: ER to Relational Mapping

Level 1: Convert ER diagram into relational schema.

Level 2: Handle complex relationships (many-to-many, weak entities).

WEEK 3: Relational Algebra

Lab Sheet 5: Basic Operations

Level 1: Perform selection and projection operations.

Level 2: Apply rename and set operations.

Lab Sheet 6: Advanced Operations

Level 1: Perform joins (inner join).

Level 2: Solve queries using outer joins and division operator.

WEEK 4: SQL Basics

Lab Sheet 7: DDL Commands

Level 1: Create and modify tables using SQL.

Level 2: Apply constraints (primary key, foreign key).

Lab Sheet 8: DML Commands

Level 1: Insert, update, delete records.

Level 2: Perform queries with conditions and operators.

WEEK 5: SQL Queries

Lab Sheet 9: Joins

Level 1: Write SQL queries using joins.

Level 2: Solve complex queries using multiple joins.

Lab Sheet 10: Aggregate Functions

Level 1: Use COUNT, SUM, AVG.

Level 2: Solve grouping queries with HAVING clause.

WEEK 6: Advanced SQL

Lab Sheet 11: Views

Level 1: Create and use views.

Level 2: Update data using views.

Lab Sheet 12: Procedures & Functions

Level 1: Write stored procedures.

Level 2: Implement user-defined functions.

WEEK 7: Triggers & Embedded SQL

Lab Sheet 13: Triggers

Level 1: Create trigger for insert/update.

Level 2: Design trigger for auditing changes.

Lab Sheet 14: Embedded & Dynamic SQL

Level 1: Demonstrate embedded SQL.

Level 2: Implement dynamic SQL queries.

WEEK 8: Query Optimization

Lab Sheet 15: Query Optimization Basics

Level 1: Explain query transformation techniques.

Level 2: Estimate cost of queries.

Lab Sheet 16: Query Plans

Level 1: Analyze linear execution plans.

Level 2: Compare bushy vs linear plans.

WEEK 9: Normalization

Lab Sheet 17: Normal Forms

Level 1: Convert schema to 1NF and 2NF.

Level 2: Convert schema to 3NF and BCNF.

Lab Sheet 18: Advanced Normalization

Level 1: Apply 4NF and 5NF.

Level 2: Analyze lossy vs lossless decomposition.

WEEK 10: Transactions

Lab Sheet 19: ACID Properties

Level 1: Explain ACID with examples.

Level 2: Analyze transaction schedules.

Lab Sheet 20: Concurrency Control

Level 1: Implement locking mechanisms.

Level 2: Analyze serializability and recoverability.

WEEK 11: Recovery Techniques

Lab Sheet 21: Logging

Level 1: Explain write-ahead logging.

Level 2: Simulate crash recovery.

Lab Sheet 22: Checkpointing

Level 1: Explain checkpointing.

Level 2: Design recovery process after failure.

WEEK 12: Advanced DB Concepts

Lab Sheet 23: NoSQL

Level 1: Compare SQL and NoSQL.

Level 2: Design document-based database.

Lab Sheet 24: Data Warehousing

Level 1: Explain data warehouse concepts.

Level 2: Design simple data warehouse schema.

WEEK 13: Advanced Database Systems

Lab Sheet 25: Object-Oriented DBMS

Level 1: Explain OODBMS concepts.

Level 2: Compare relational vs object-oriented DBMS.

Lab Sheet 26: Spatial & Temporal DB

Level 1: Explain spatial databases.

Level 2: Design temporal database application.

WEEK 14: Specialized Databases

Lab Sheet 27: Multimedia DB

Level 1: Explain multimedia database concepts.

Level 2: Design multimedia database system.

Lab Sheet 28: XML Databases

Level 1: Explain XML databases.

Level 2: Design document-oriented database.

WEEK 15: Case Study & Integration

Lab Sheet 29: Case Study

Level 1: Design database for real-world application.

Level 2: Optimize queries and schema.

Lab Sheet 30: Mini Project

Level 1: Implement database application.

Level 2: Integrate SQL, optimization, and transactions.

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.

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: CSE2262	**Course Title: Analysis of Algorithms** **Type of Course: Theory Course (PCC)**	**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	To develop the ability to design and analyze efficient algorithms by enabling students to evaluate time and space complexity, apply various algorithmic paradigms such as divide-and-conquer, greedy, and dynamic programming, and compare different approaches for solving computational problems.
Course Outcomes	On successful completion of this course, students will be able to: **CO1:** Explain the efficiency of algorithms in terms of time and space complexity. [Understand] **CO2:** Apply divide-and-conquer and decrease-and-conquer techniques to solve searching and sorting problems. [Apply] **CO3:** Apply dynamic programming and greedy techniques to solve computational problems. [Apply] **CO4:** Apply time–space trade-off techniques and analyze the limitations of algorithms for problem solving. [Apply] **CO5:** Analyze problems using backtracking and branch-and-bound techniques to develop optimal solutions. [Analyze]

Course Content:

Module 1	**Introduction**	**14 Sessions**
Introduction: Notion of algorithm, Importance of algorithm, Important Problem Types, Fundamentals of analysis of algorithm efficiency: General equation, Order of growth, basic efficiency classes, asymptotic notations, mathematical analysis of recursive & non-recursive algorithms, Brute Force technique: Bubble sort, selection sort, linear search, brute force string matching, Exhaustive search: Travelling salesperson problem, knapsack problem.		
Module 2	**Divide-and-conquer**	**11 Sessions**
Divide & Conquer technique: General Method, General recursive equation, Masters Theorem, Binary search, Merge sort, quick sort. Decrease & Conquer technique: General method, insertion sort, topological sorting		
Module 3	**Dynamic programming**	**13 Sessions**
Dynamic Programming: General method, Warshall algorithm, Floyd algorithm, 0/1 knapsack, Greedy method: General method, prim's algorithm, Kruskal algorithm, Dijkstra's algorithm, Huffman coding		
Module 4	**Space and time tradeoff:**	**11 Sessions**
Space and time tradeoff: introduction, Input enhancement in string matching: Horspool algorithm and Boyer-Moore algorithm, Limitations of algorithm power: Introduction, lower bound arguments, decision trees, P, NP and NP-complete problems		
Module 5	**Coping with limitations of algorithm power:**	**11 Sessions**
Coping with limitations of algorithm power: Introduction, Backtracking: n-queens problem, subset-sum problem, Hamiltonian circuit problem, Branch & bound: Travelling salesperson problem, knapsack problem, Job assignment 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.

Reference Books:

R1. AV Aho, J Hopcroft, JD Ullman, *"The Design and Analysis of Algorithms"*, Addison-Wesley, 1974

R2. Tim Roughgarden, *"Algorithms Illuminated"* (books 1 through 3), "Operating Systems Design and Implementation", Soundlikeyourself Publishing, 2017-2019.

R3. J. Kleinberg and E. Tardos, "*Algorithm Design*", Addison-Wesley, 2005

Online Resources:

1. EBook: https://presiuniv.knimbus.com/user#/home
2. https://onlinecourses.nptel.ac.in/noc26_cs67/
3. www.youtube.com/watch?v=so4zT-u09LM&list=PL8GCet93Cu1iRF611mrUu-Rhy12fMwZOf

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: Laboratory Course (PCC)**	**L- T-P- C**	**0**	**0**	**2**	**1**
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	To develop practical skills in algorithm design and analysis by enabling students to implement algorithms using programming languages, evaluate their performance through experimentation, compare the efficiency of different approaches, and solve optimization problems using appropriate algorithmic techniques.					
Course Outcomes	On successful completion of this course, students will be able to: **CO1:** Compute the efficiency of algorithms in terms of time and space complexity. [Apply] **CO2:** Apply divide-and-conquer techniques to solve searching and sorting problems. [Apply]					

	CO3: Apply dynamic programming techniques to solve computational problems. [Apply] **CO4:** Apply greedy techniques for solving optimization problems. [Apply] **CO5:** Apply backtracking techniques and analyze the limitations of algorithms for problem solving. [Apply]

Course Content

List of Laboratory Tasks:30 Sessions

WEEK 1: Algorithm Basics & Analysis

Lab Sheet 1: Algorithm Fundamentals

Level 1: Write an algorithm and program to find minimum and maximum elements in a list.

Level 2: Analyze the number of comparisons and derive time complexity.

Lab Sheet 2: Asymptotic Notations

Level 1: Write programs to demonstrate linear and quadratic time complexity.

Level 2: Compare growth rates of different functions using Big-O, Ω, and Θ.

WEEK 2: Efficiency Analysis

Lab Sheet 3: Recursive Algorithms

Level 1: Write a recursive program for factorial.

Level 2: Solve recurrence relations for recursive algorithms.

Lab Sheet 4: Non-Recursive Analysis

Level 1: Write programs with nested loops and count operations.

Level 2: Derive general equations and order of growth.

WEEK 3: Brute Force Techniques

Lab Sheet 5: Sorting Algorithms

Level 1: Implement Bubble Sort and Selection Sort.

Level 2: Compare their time complexity experimentally.

Lab Sheet 6: Searching & String Matching

Level 1: Implement Linear Search.

Level 2: Implement brute-force string matching algorithm.

WEEK 4: Exhaustive Search

Lab Sheet 7: Knapsack Problem

Level 1: Solve 0/1 Knapsack using brute force.

Level 2: Analyze solution space complexity.

Lab Sheet 8: Travelling Salesperson Problem

Level 1: Solve TSP using brute force approach.

Level 2: Compare performance for different inputs.

WEEK 5: Divide and Conquer

Lab Sheet 9: Binary Search

Level 1: Implement Binary Search.

Level 2: Analyze recurrence relation.

Lab Sheet 10: Merge Sort

Level 1: Implement Merge Sort.

Level 2: Analyze using Master's Theorem.

WEEK 6: Advanced Divide & Conquer

Lab Sheet 11: Quick Sort

Level 1: Implement Quick Sort.

Level 2: Compare best and worst cases.

Lab Sheet 12: Master's Theorem

Level 1: Solve recurrence equations.

Level 2: Apply Master's Theorem to different cases.

WEEK 7: Decrease and Conquer

Lab Sheet 13: Insertion Sort

Level 1: Implement Insertion Sort.

Level 2: Compare with Bubble Sort.

Lab Sheet 14: Topological Sorting

Level 1: Implement Topological Sort.

Level 2: Apply on real-world dependency graph.

WEEK 8: Dynamic Programming

Lab Sheet 15: Warshall Algorithm

Level 1: Implement Warshall algorithm.

Level 2: Compute transitive closure.

Lab Sheet 16: Floyd Algorithm

Level 1: Implement Floyd's algorithm.

Level 2: Compare with Dijkstra.

WEEK 9: DP Applications

Lab Sheet 17: Knapsack DP

Level 1: Solve 0/1 Knapsack using DP.

Level 2: Compare with brute force approach.

Lab Sheet 18: DP Optimization

Level 1: Identify overlapping subproblems.

Level 2: Convert recursive solution to DP.

WEEK 10: Greedy Algorithms

Lab Sheet 19: MST Algorithms

Level 1: Implement Prim's algorithm.

Level 2: Implement Kruskal's algorithm.

Lab Sheet 20: Shortest Path

Level 1: Implement Dijkstra's algorithm.

Level 2: Compare greedy vs DP approach.

WEEK 11: Huffman Coding

Lab Sheet 21: Huffman Tree

Level 1: Construct Huffman Tree.

Level 2: Analyze compression efficiency.

Lab Sheet 22: Greedy Analysis

Level 1: Demonstrate greedy choice property.

Level 2: Prove correctness of greedy algorithms.

WEEK 12: String Matching Optimization

Lab Sheet 23: Horspool Algorithm

Level 1: Implement Horspool algorithm.

Level 2: Compare with brute-force method.

Lab Sheet 24: Boyer-Moore Algorithm

Level 1: Implement Boyer-Moore algorithm.

Level 2: Analyze efficiency improvements.

WEEK 13: Complexity Theory

Lab Sheet 25: Lower Bounds

Level 1: Analyze decision tree for sorting.

Level 2: Derive lower bounds.

Lab Sheet 26: NP Problems

Level 1: Classify problems as P or NP.

Level 2: Analyze NP-complete problems.

WEEK 14: Backtracking

Lab Sheet 27: N-Queens

Level 1: Solve N-Queens problem.

Level 2: Analyze search tree.

Lab Sheet 28: Subset Sum

Level 1: Solve subset sum problem.

Level 2: Compare with DP solution.

WEEK 15: Branch & Bound

Lab Sheet 29: TSP

Level 1: Solve TSP using Branch & Bound.

Level 2: Compare with brute-force approach.

Lab Sheet 30: Job Assignment

Level 1: Solve assignment problem.

Level 2: Analyze pruning efficiency.

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:CSE2266	**Course Title: Theory of Computation Type of Course: Theory Course (PCC)**	**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	To develop a strong theoretical foundation in formal languages and automata by enabling students to analyze finite and pushdown automata, explore Turing machines and computability, evaluate complexity and decidability of problems, and apply these concepts to language design.					
Course Outcomes	On successful completion of this course, students will be able to: **CO1:** Explain the fundamental components and concepts of automata. [Understand] **CO2:** Construct finite automata for given languages. [Apply] **CO3:** Differentiate between regular grammars and context-free grammars. [Understand] **CO4:** Construct pushdown automata for given languages. [Apply] **CO5:** Design Turing machines for given languages. [Apply]					

Course Content:

Module 1	**Introduction to automata theory**	**6 Sessions**

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**	**12 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**	**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**	**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**	**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 Course (PCC)**	**L-T-P-C**	**3**	**0**	**0**	**3**
Version No.	1					
Course Pre-requisites	CSE2251 Data Communication and Computer Networks					
Anti-requisites	NIL					
Course Description	The Course covers the principles and practice of cryptography and network security, focusing in particular on the security aspects of the web and Internet. The cryptographic tools such as shared key encryption, public key encryption, key exchange, and digital signature are explored. The use and utilization of the internet protocols and applications such as SSL/ TLS, IPSEC, Kerberos, PGP, and S/ MIME, SET are reviewed. System security issues such as viruses, intrusion and firewalls are also explored.					
Course Objective	To develop a comprehensive understanding of cryptography and network security by enabling students to analyze security threats and vulnerabilities, apply encryption, authentication, and hashing techniques, study security protocols and standards, and evaluate secure system design.					
Course Outcomes	On successful completion of this course, students will be able to: **CO1:** Explain the fundamentals of cryptography and security architecture. [Understand] **CO2:** Apply classical encryption techniques for securing data. [Apply] **CO3:** Analyze symmetric key cryptography algorithms and their applications. [Apply] **CO4:** Apply number theory concepts in cryptographic systems. [Apply] **CO5:** Demonstrate public key cryptography and authentication methods. [Apply] **CO6:** Analyze and compare network security protocols and their applications. [Analyze]					

Course Content:

Module 1	**Introduction to Cryptography and types of Ciphers**	**10 Sessions**

Topics:

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 techniques: Caesar cipher, Monoalphabetic cipher, Polyalphabetic cipher, Play-fair and Hill Cipher, Transposition techniques: Rail fence, Basics of Steganography.

Module 2	**Symmetric Key Cryptography and Number Theory**	**13 Sessions**

Topics:

Introduction to Block Cipher and Stream Cipher, Feistel Structure, Symmetric Encryption Algorithms: Data Encryption Standard, Introduction to Galois Field, Advanced Encryption Standard, Modular Arithmetic, Prime numbers, Fermat's little theorem, Primality testing: Miller-Rabin algorithm, Euclidean and Extended Euclidean Algorithm, Euler Totient Function, Chinese Remainder Theorem

Module 3	**Public Key Cryptography and its Applications**	**12 Sessions**

Topics:

Asymmetric Encryption and Decryption: Overview of Public Key Cryptography, Distribution of public keys, X.509 certificates, RSA, Diffie-Helman Key exchange, Man in the middle attack, Cryptographic Hash functions, Secure Hash Algorithm, Message Authentication Codes – HMAC, Digital Signature case studies.

Module 4	**Network Security**	**10 Sessions**

Topics:

Network Security applications: Authentication: Kerberos, PKI. E-mail security: PGP, S/MIME. Web Security: Secure Socket Layer (SSL), Transport Layer Security (TLS). IP Security: IPSec Policy, Encapsulating Security Payload (ESP).

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.

PEDAGOGY PLANNED WITH TOPICS:

1. Problem Solving: Playfair and Hill cipher
2. Problem Solving & Participative Learning: DES, AES, RSA & Diffie Hellman
3. Self-learning: Man in the Middle attack.
4. Flip Class: Network security and Email security

Textbooks:

1. William Stallings, "Cryptography and Network Security - Principles and Practices", Prentice Hall, 7th Edition, 2017.

Reference Books:

1. Behrouz A Forouzan, Debdeep Mukhopadhyay, “Cryptography and Network Security”, McGraw Hill, third edition, 2010.

2. R. Rajaram, “Network Security and Cryptography” SciTech Publication. 3rd Edition, 2014.

3. Atul Kahate, “Cryptography and Network Security”, Tata McGraw-Hill, 2nd Edition, 2019.

4. Bruce Schneier, “Applied Cryptography”, John Wiley and Sons Inc. Second Edition, 2015.

Web references:

1. https://onlinecourses.nptel.ac.in/noc22_cs90/preview

2. e-pgpathshala UGC lecture series: E-Series and Self learning Materials.

https://epgp.inflibnet.ac.in/Home/ViewSubject?catid=fBYckQKJvP3a/8Vd3L08tQ==

3. http://182.72.188.195/cgi-bin/koha/opacdetail.pl?biblionumber=10133

4. http://182.72.188.195/cgi-bin/koha/opacdetail.pl?biblionumber=5875

Topics relevant to “Skill Development”: Symmetric and Asymmetric Encryption Algorithms and its problems.

Course Code: **CSE2269**	**Course Title: Operating Systems** **Type of Course: Theory Course (PCC)**	**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 Systems operations, Operating System structure and its design and implementation. It covers the 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 Objective	To develop a comprehensive understanding of operating system principles by enabling students to analyze process management and scheduling algorithms, explore memory and virtual memory management, examine file systems and storage organization and evaluate synchronization mechanisms and deadlock handling techniques.					
Course Outcomes	On successful completion of this course, students will be able to: **CO1:** Explain the fundamental concepts of operating systems and relevant case studies. [Understand] **CO2:** Apply various CPU scheduling algorithms for process management. [Apply] **CO3:**					

	Apply synchronization techniques to handle concurrency problems. [Apply] **CO4:** Apply deadlock detection and recovery methods in operating systems. [Apply] **CO5:** Explain various memory management techniques used in operating systems. [Apply]

Course Content:

Module 1	**Introduction to Operating Systems**	**9 Sessions**

Topics:

Introduction to OS , Operating-System Operations, Operating Systems Services, , System Calls and its types, Operating Systems Structure, System Program and its types, Linkers and Loaders, Overview of OS design and implementation, Open-source Operating Systems

Module 2	**Process Management**	**12 Sessions**

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**	**12 Sessions**

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**	**12 Sessions**

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 :

1. Demonstrate process concepts in LINUX OS.
2. Simulation of CPU scheduling algorithms.
3. Develop program to demonstrate use of Semaphores in threads.
4. Develop program to demonstrate use of deadlock avoidance algorithms.
5. Develop program to demonstrate use of page replacement algorithms.
6. Simulation of memory allocation strategies [first fit, best fit and worst fit].

Text Book :

1. Silberschatz A, Galvin P B and Gagne G , "Silberschatz's Operating System Concepts", Paperback, Global Edition Wiley, 2019

References :

1. Silberschatz A, Galvin P B and Gagne G, "Operating System Concepts", 10th edition Wiley, 2018.
2. William Stallings, "Operating Systems",Ninth Edition, By Pearson Paperback ,1 March 2018.
3. Sundaram RMD, Shriram K V, Abhishek S N, B Chella Prabha, " Cracking the Operating System skills", Dreamtech, paperback, 2020
4. Remzi H. Arpaci-Dusseau Andrea C. Arpaci-dusseau , "Operating Systems: Three Easy Pieces, Amazon digital Services", September 2018.

E-resources/Weblinks :

1. https://www.os-book.com/OS9/
2. https://pages.cs.wisc.edu/~remzi/OSTEP/
3. https://codex.cs.yale.edu/avi/os-book/OS10/index.html

Course Code: **CSE2270**	**Course Title: Operating Systems Lab** **Type of Course: Laboratory Course (PCC)**	**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 Objectives	To develop practical skills in operating systems by enabling students to implement scheduling and memory management algorithms, simulate process synchronization and deadlock handling, and build system-level programming capabilities.
Course Outcomes	On successful completion of this course, students will be able to: **CO1:** Demonstrate system-level programming using system calls and operating system structures. [Apply] **CO2:** Simulate process scheduling and multithreading techniques. [Apply] **CO3:** Apply synchronization techniques using semaphores and shared memory to handle concurrency problems. [Apply] **CO4:** Demonstrate memory management and file system concepts using simulation or scripting tools. [Apply] **CO5:** Analyze the performance of operating system algorithms related to scheduling, synchronization and memory management. [Analyze]

Course Content:

List of Laboratory Tasks:30 Sessions

WEEK 1: Introduction to OS

Lab Sheet 1: OS Basics

Level 1: Write a program to demonstrate basic system calls (file operations like open, read, write).

Level 2: Analyze different OS services with real-time examples.

Lab Sheet 2: OS Structure

Level 1: Illustrate OS structures (monolithic, layered, microkernel).

Level 2: Compare open-source OS (Linux) with proprietary OS.

WEEK 2: System Programs

Lab Sheet 3: Linkers & Loaders

Level 1: Demonstrate compilation process (preprocessing, compiling, linking).

Level 2: Analyze object files and executable generation.

Lab Sheet 4: System Utilities

Level 1: Write simple shell commands using system calls.

Level 2: Implement a mini shell program.

WEEK 3: Process Concept

Lab Sheet 5: Process Creation

Level 1: Write a program using fork() system call.

Level 2: Demonstrate parent-child communication.

Lab Sheet 6: Process Operations

Level 1: Implement process states and transitions.

Level 2: Simulate process control block (PCB).

EEK 4: Inter Process Communication

Lab Sheet 7: Pipes

Level 1: Implement communication using pipes.

Level 2: Implement bidirectional communication using pipes.

Lab Sheet 8: Sockets/RPC

Level 1: Write a client-server program using sockets.

Level 2: Simulate RPC communication.

WEEK 5: Threads

Lab Sheet 9: Thread Creation

Level 1: Create threads using pthread library.

Level 2: Compare multi-threading vs multi-processing.

Lab Sheet 10: Thread Issues

Level 1: Demonstrate thread synchronization issues.

Level 2: Solve race condition problem.

WEEK 6: CPU Scheduling

Lab Sheet 11: FCFS & SJF

Level 1: Implement FCFS scheduling.

Level 2: Implement SJF and compare results.

Lab Sheet 12: Advanced Scheduling

Level 1: Implement Round Robin scheduling.

Level 2: Implement Priority scheduling and analyze performance.

WEEK 7: Synchronization

Lab Sheet 13: Critical Section

Level 1: Implement Peterson's solution.

Level 2: Analyze critical section problem.

Lab Sheet 14: Semaphores

Level 1: Implement semaphore operations.

Level 2: Solve producer-consumer problem using semaphores.

WEEK 8: Classical Problems

Lab Sheet 15: Reader-Writer

Level 1: Implement Reader-Writer problem.

Level 2: Analyze starvation issues.

Lab Sheet 16: Dining Philosophers

Level 1: Implement Dining Philosophers problem.

Level 2: Avoid deadlock using strategies.

WEEK 9: Deadlocks

Lab Sheet 17: Deadlock Detection

Level 1: Implement Resource Allocation Graph.

Level 2: Detect deadlock using algorithm.

Lab Sheet 18: Deadlock Avoidance

Level 1: Implement Banker's algorithm.

Level 2: Analyze safe and unsafe states.

WEEK 10: Memory Management Basics

Lab Sheet 19: Memory Allocation

Level 1: Implement First Fit, Best Fit, Worst Fit.

Level 2: Compare memory utilization.

Lab Sheet 20: Paging

Level 1: Simulate paging mechanism.

Level 2: Analyze page table structure.

WEEK 11: Virtual Memory

Lab Sheet 21: Page Replacement

Level 1: Implement FIFO page replacement.

Level 2: Implement LRU and Optimal algorithms.

Lab Sheet 22: Page Faults

Level 1: Calculate page faults.

Level 2: Analyze thrashing condition.

WEEK 12: Advanced Memory Management

Lab Sheet 23: Segmentation

Level 1: Implement segmentation.

Level 2: Compare segmentation vs paging.

Lab Sheet 24: Demand Paging

Level 1: Simulate demand paging.

Level 2: Analyze copy-on-write mechanism.

WEEK 13: File System

Lab Sheet 25: File Operations

Level 1: Implement file operations (create, read, write).

Level 2: Simulate directory structure.

Lab Sheet 26: File Allocation

Level 1: Implement file allocation methods.

Level 2: Compare allocation techniques.

WEEK 14: File System Implementation

Lab Sheet 27: File System Design

Level 1: Design simple file system.

Level 2: Analyze file system performance.

Lab Sheet 28: Access Methods

Level 1: Implement sequential access.

Level 2: Implement indexed access.

WEEK 15: Integration & Simulation

Lab Sheet 29: OS Simulation

Level 1: Simulate OS components (process + memory).

Level 2: Integrate scheduling and memory management.

Lab Sheet 30: Mini Project

Level 1: Implement OS concept-based project.

Level 2: Analyze performance and optimization.

Project work/Assignment :

1. Demonstrate process concepts in LINUX OS.
2. Simulation of CPU scheduling algorithms.
3. Develop program to demonstrate use of Semaphores in threads.
4. Develop program to demonstrate use of deadlock avoidance algorithms.
5. Develop program to demonstrate use of page replacement algorithms.
6. Simulation of memory allocation strategies [first fit, best fit and worst fit].

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:

1. 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.

2. 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.

Text Book :

1. Silberschatz A, Galvin P B and Gagne G , “Silberschatz's Operating System Concepts”, Paperback, Global Edition Wiley, 2019

References :

1. Silberschatz A, Galvin P B and Gagne G, “Operating System Concepts”, 10th edition Wiley, 2018.
2. William Stallings, “Operating Systems”,Ninth Edition, By Pearson Paperback ,1 March 2018.
3. Sundaram RMD, Shriram K V, Abhishek S N, B Chella Prabha, “ Cracking the Operating System skills”, Dreamtech, paperback, 2020
4. Remzi H. Arpaci-Dusseau Andrea C. Arpaci-dusseau , “Operating Systems: Three Easy Pieces, Amazon digital Services”, September 2018.

E-resources/Weblinks

1. https://www.os-book.com/OS9/
2. https://pages.cs.wisc.edu/~remzi/OSTEP/
3. https://codex.cs.yale.edu/avi/os-book/OS10/index.html

Course Code: **CSE2285**	**Course Title: Cloud computing** **Type of Course: Theory Course (PCC)**	**L- T-P- C**	**2**	**0**	**0**	**2**
Version No.	2.0					
Course Pre-requisites	NIL					

Anti-requisites	NIL
Course Description	This course introduces Cloud Computing as a modern computing paradigm that enables scalable, on-demand delivery of computing resources and services. The course covers the foundations of cloud computing, including its evolution, characteristics, terminology, benefits, and challenges. It further explores cloud architectures, service and deployment models, and the role of virtualization technologies in building cloud environments. The course also introduces Edge and Fog Computing paradigms to address the limitations of centralized cloud systems, along with fundamental cloud security concepts such as data protection, access control, threats, and compliance. The course equips students with a conceptual understanding of contemporary cloud and distributed computing ecosystems.
Course Objective	To develop a comprehensive understanding of cloud computing by enabling students to learn its fundamental concepts, evolution, and characteristics, analyze cloud architectures and service and deployment models, understand virtualization technologies and emerging paradigms such as edge and fog computing, and gain foundational knowledge of cloud security principles including data protection, access control, threats and compliance.
Course Outcomes	On successful completion of this course, students will be able to: **CO1:** Explain the fundamentals and evolution of cloud computing. [Understand] **CO2:** Explain cloud architectures, service models, and virtualization concepts. [Understand] **CO3:** Analyze edge and fog computing paradigms and their applications. [Analyze] **CO4:** Analyze cloud security fundamentals, including threats and protection mechanisms. [Analyze] **CO5:** Apply identity and access management (IAM), encryption, and compliance controls to secure cloud environments. [Apply]

Course Content:

Module 1	**Foundations of Cloud Computing**	**8 Sessions**

Topics:

Evolution of Computing Paradigms, Motivation and Need for Cloud Computing, Definition of Cloud Computing, Characteristics of Cloud Computing, Cloud Computing Design Principles, Cloud Computing Terminology, Benefits and Challenges of Cloud Computing, Cloud Computing Use Cases and Application Domains

Module 2	**Cloud Architecture, Service Models, and Virtualization**	**8 Sessions**

Topics:

Cloud Reference Architecture, Cloud Service Models (IaaS, PaaS, SaaS), Cloud Deployment Models, Multitenant Technology in Cloud Computing, Fundamentals of Virtualization, Hypervisors and Virtual Machines, Role of Virtualization in Cloud Computing, Containerization, Fundamental Container Architecture Elements

Module 3	Edge and Fog Computing Paradigms	7 Sessions
Topics: Limitations of Centralized Cloud Computing, Motivation for Edge Computing, Edge Computing Architecture, Fog Computing Concepts, Cloud–Fog–Edge Comparison, Edge and Fog Computing Deployment Models, Edge Intelligence and AI at the Edge, Application Scenarios of Edge and Fog.		
Module 4	**Cloud Security Fundamentals**	**7 Sessions**
Topics: Cloud Security Overview, Shared Responsibility Model, Data Security and Privacy, Identity and Access Management, Threats and Vulnerabilities, Compliance and Governance, Cloud Security Architecture and Control Layers, Encryption and Key Management Fundamentals.		
Targeted Application & Tools that can be used: Amazon Web Services (AWS), IBM Cloud, VMware		
Text Book(s): 1. Thomas Erl, *Cloud Computing: Concepts, Technology & Architecture*, Pearson 2. Rajkumar Buyya et al., *Mastering Cloud Computing*, McGraw-Hill Education 3. John W. Rittinghouse & James F. Ransome, *Cloud Computing: Implementation, Management, and Security*, CRC Press		
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. 3. Web resources: https://presiuniv.knimbus.com/user#/home		

Course Code: CSE2286	Course Title: Cloud Computing Lab Type of Course Laboratory Course (PCC)	L- T-P- C	0	0	2	1
Version No.	1.0					
Course Pre-requisites	NIL					
Anti- requisites	NIL					
Course Description	This course enables students to apply cloud computing concepts through hands-on laboratory experiments using a hyperscale cloud platform. Students will analyze cloud infrastructure, service models, and deployment architectures, and deploy scalable, resilient, and secure cloud applications. The course emphasizes on-demand resource provisioning, platform-managed execution, containerization, edge–fog–cloud integration, and identity-based security controls. By completing this course, students will design and evaluate cloud solutions with respect to scalability, performance, availability, and security, aligned with real-world computing environments.					

Course Objectives	To develop practical expertise in cloud computing by enabling students to understand hyperscale cloud infrastructure and resource provisioning, design and deploy cloud applications using appropriate service models and container-based technologies, analyze scalability, availability, resilience, and performance of cloud and edge-integrated systems, and implement cloud security mechanisms including identity and access management, data protection, and threat modeling.
Course Outcomes	On successful completion of this course, students will be able to: **CO1:** Explain hyperscale cloud infrastructure, including regions, services, and on-demand resource provisioning. [Understand] **CO2:** Deploy cloud-based applications using appropriate service models, platforms, and container technologies. [Apply] **CO3:** Compare scalability, performance, availability, and resilience of applications in cloud and edge–fog–cloud environments. [Apply] **CO4:** Analyze identity management, data security, and threat modeling techniques for securing cloud-based applications. [Analyze] **CO5:** Apply cloud services and tools to deploy, secure, and manage applications effectively. [Apply]

List of Laboratory tasks: **30 Sessions**

Lab Sheet 1: Exploring Hyperscale Cloud Infrastructure (Tool: Google Cloud Console)

1. Explore regions, zones, and global infrastructure of Google Cloud
2. Identify compute, storage, and networking services
3. Relate infrastructure layout to cloud scalability and availability

Lab Sheet 2: On-Demand Resource Access Using Cloud Shell (Tool: Google Cloud Shell)

1. Access a browser-based cloud-managed Linux environment
2. Execute commands without provisioning servers
3. Demonstrate on-demand availability of cloud resources

Lab Sheet 3: Cloud Application Domains Demonstration (Tool: Google Colab)

1. Execute compute-intensive programs in cloud notebooks
2. Compare cloud execution with local execution
3. Map workloads to cloud application domains

Lab Sheet 4: Cloud Reference Architecture Design (Tool: Draw.io (Google Cloud Icons))

1. Design a reference architecture for a cloud-based application

2. Identify frontend, backend, data, and security layers
3. Map components to cloud services

Lab Sheet 5: Demonstrating Cloud Service Models (Tool: Google Cloud Console)

1. Identify services under IaaS, PaaS, and SaaS
2. Compare user and provider responsibilities
3. Understand abstraction levels in service models

Lab Sheet 6: Deploying a Web Application Using PaaS (Tool: Google App Engine (Standard Environment))

1. Deploy a simple web application
2. Modify and redeploy application code
3. Observe platform-managed execution

Lab Sheet 7: Application Versioning and Auto-Scaling (Tool: Google App Engine)

1. Create multiple versions of an application
2. Route traffic between versions
3. Observe automatic scaling behavior

Lab Sheet 8: Container-Based Application Deployment (Tool: Google Cloud Run)

1. Deploy a containerized application
2. Invoke application endpoints
3. Observe serverless container execution

Lab Sheet 9: Limitations of Centralized Cloud Computing (Tool: Google Colab)

1. Execute latency-sensitive tasks
2. Observe network dependency
3. Identify limitations of centralized cloud processing

Lab Sheet 10: Edge Computing Workflow Simulation (Tool: Google Cloud Functions)

1. Perform data preprocessing before cloud execution
2. Reduce data transfer to the cloud
3. Analyze latency improvement

Lab Sheet 11: Designing Edge–Fog–Cloud Architecture (Tool: Draw.io)

1. Design Edge–Fog–Cloud architecture for a smart application

2. Assign processing roles to each layer
3. Compare with cloud-only architecture

Lab Sheet 12: Identity and Access Management Configuration (Tool: Google Cloud IAM)

1. Create users, roles, and permissions
2. Implement least-privilege access
3. Verify access control enforcement

Lab Sheet 13: Data Security and Privacy in Cloud Applications (Tool: Google App Engine)

1. Observe data isolation in managed services
2. Analyze application-level access controls
3. Relate observations to cloud data security

Lab Sheet 14: Cloud Threat Modeling and Attack Surface Analysis (Tool: Draw.io + GCP Security Overview)

1. Identify threats at compute, network, and application layers
2. Map vulnerabilities to cloud architecture
3. Propose mitigation strategies

Lab Sheet 15: Application Resilience and Recovery (Tool: Google App Engine)

1. Simulate application failure
2. Redeploy application
3. Observe fault tolerance and recovery

List of Tools:

1. Google Cloud Console
2. Google Cloud Shell
3. Google Colab
4. Draw.io (with Google Cloud Icons)
5. Google App Engine (Standard Environment)
6. Google Cloud Run
7. Google Cloud Functions
8. Google Cloud IAM

Textbooks

1. Thomas Erl, *Cloud Computing: Concepts, Technology & Architecture*, Pearson
2. Rajkumar Buyya et al., *Mastering Cloud Computing*, McGraw-Hill Education
3. John W. Rittinghouse & James F. Ransome, *Cloud Computing: Implementation, Management, and Security*, CRC Press

Reference Books

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

W1: Web resources: https://presiuniv.knimbus.com/user#/home

Course Code: CSE2287	**Course Title: Software Design and Development** **Type of Course: Theory Course (PCC)**	**L- T-P- C**	**3**	**0**	**0**	**3**
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	NIL					
Course Description	This course introduces foundational concepts of software engineering and emphasizes a practical, structured approach to software system development. It spans software requirement analysis, process modeling, system design, Agile principles, DevOps culture, testing strategies, quality assurance, and software maintenance. Hands-on assignments and problem-solving foster understanding of contemporary software practices and industry tools such as CASE tools, GitHub, and Selenium. AI-enhanced tools are introduced for documentation, automation, and intelligent testing workflows					
Course Objectives	To develop a comprehensive understanding of software engineering by enabling students to analyze software development life cycle models, apply design principles and patterns, develop modular and maintainable software systems, perform testing and quality assurance, and manage software projects effectively.					

Course Outcomes	On successful completion of this course, students will be able to: **CO1:** Explain software engineering principles, ethics, and various software development life cycle models. [Understand] **CO2:** Identify software requirements and develop design models, including Software Requirement Specification (SRS) documents. [Apply] **CO3:** Apply Agile development practices and DevOps principles in software development. [Apply] **CO4:** Apply software testing strategies, configuration management, and maintenance techniques. [Apply] **CO5:** Evaluate software design and development practices to ensure quality, reliability, and scalability. [Evaluate]	
Course Content:		
Module 1	**Introduction to Software Engineering and Process Models**	**10 Sessions**
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. **AI Perspective:** Use of AI-based modelling assistants (e.g., ChatGPT-based flow generation)		
Module 2	**Software Requirements, Analysis and Design**	**12 Sessions**
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. **AI Perspective:** Taskade AI SRS Generator/ Bit.ai AI SRS Generator		
Module 3	**Agile Principles & Devops**	**10 Sessions**
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.		

AI Perspective: AI assistants for sprint planning and backlog prioritization		
Module 4	**Software Testing and Maintenance**	**13 Sessions**
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. **AI Perspective:** LambdaTest AI Test Case Generator tool/ Testsigma, testRigor tool for QA.		
Targeted Application & Tools that can be used: **Git hub/Lambda/Taskade/Bit.ai/Testsigma/TestRogor**		
Text Books: T1. Roger S. Pressman, "Software Engineering – A Practitioner's Approach", VII Edition, McGraw-Hill, 2017. T2. Bob Hughes, Mike Cotterell, Rajib Mall, "Software Project Management", VI Edition, McGraw-Hill, 2018.		
References: R1. Rajib Mall, "Fundamentals of Software Engineering", VI Edition, PHI learning private limited, 2015. R2. Ian Sommerville, "Software Engineering", IX Edition, Pearson Education Asia, 2011. R3. Agile Software Development Principles, Patterns and Practices.1st Edition, Wiley, 2002 **Journals / Magazines** • IEEE Software • ACM Transactions on Software Engineering and Methodology • Journal of Systems and Software **SWAYAM / NPTEL / MOOCs** • NPTEL – Software Engineering (Prof. Rajib Mall, IIT KGP) • Coursera – Agile Development (University of Virginia) • Udemy – DevOps Masterclass • EdX – Software Testing Fundamentals (TUM)		

Topics Relevant to "Skill Development": Black box Testing, White box Testing, AI tools for design, modelling, testing and document Automated Testing for Skill development through Participative Learning Techniques. This is attained through assessment mentioned in the course handout

Course Code: CSE2505	**Course Title: Mobile Application Development** **Type of Course: Theory Course (PCC)**	**L- T-P- C**	**2**	**0**	**0**	**2**

Version No.	2.0
Course Pre-requisites	CSE2255 Object Oriented Programming Using Java
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	To develop a comprehensive understanding of mobile application development by enabling students to analyze mobile architectures and platforms, design and develop mobile applications, implement effective user interfaces and user experience principles, integrate applications with backend services, and evaluate performance and usability.
Course Outcomes	On successful completion of this course, students will be able to: **CO1:** Explain the fundamentals of mobile application development and its architecture. [Understand] **CO2:** Develop mobile applications using appropriate Android views and UI components. [Apply] **CO3:** Demonstrate the use of services, broadcast receivers, notifications, and content providers in mobile applications. [Apply] **CO4:** Apply data persistence techniques to perform CRUD operations in mobile applications. [Apply] **CO5:** Apply advanced concepts in mobile application development to build functional applications. [Apply]
Course Content:	

Module 1	**Introduction and Architecture of Android**	**5 Sessions**

Topics:

Android: History and features, Architecture, Development Tools, Android Debug Bridge (ADB), and Life cycle.

Module 2	**User Interfaces, Intent and Fragments**	**8 Sessions**

Topics:

Views, Layout, Menu, Intent and Fragments.

Module 3	**Components of Android**	**7 Sessions**

Topics:

Activities, Services, Broadcast receivers, Content providers, User Navigation

Module 4	**Notifications and Data Persistence**	**5 Sessions**

Topics:

Notification, Shared Preferences, SQLite database, Android Room with a View, Firebase.

Module 5	**Advance App Development**	**5 Sessions**

Topics:

Graphics and Animation, App Widgets, Sensors, Performance, Location, Places, Mapping, Custom Views, Canvas.

Targeted Application & Tools that can be used:

Applications:

1. Native Android Applications
2. Native iOS Applications
3. Cross Platform mobile Apps
4. Mobile web Applications

Text Book(s):

1. T1. Pradeep kothari "Android Application Development - Black Book", dreamtechpress
2. T2. Barry Burd (Author), "Android Application Development" ALL – IN – ONE FOR Dummies
3. T3. Jeff Mcherter (Author),Scott Gowell (Author), "Professional mobile Application Development" paperback, Wrox - Wiley India Private Limited
4. 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: **CSE2506**	**Course Title: Mobile Application Development Lab** **Type of Course: Laboratory Course (PCC)**	**L- T-P- C**	**0**	**0**	**4**	**2**
Version No.	2.0					
Course Pre-requisites	CSE2256 Object Oriented Programming Using Java Lab					
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	To develop practical skills in mobile application development by enabling students to build applications using modern development tools, implement user interface components and features, perform testing and debugging, and deploy mobile applications effectively.					
Course Outcomes	On successful completion of this course, students will be able to: **CO1:** Develop functional mobile applications using appropriate tools and frameworks. [Apply] **CO2:** Design and implement interactive user interfaces for mobile applications. [Apply] **CO3:** Integrate cloud services and APIs into mobile applications. [Apply] **CO4:** Integrate backend systems and manage data effectively within mobile applications. [Apply] **CO5:** Deploy, publish and maintain advanced mobile applications. [Create]					
Course Content:						
Module 1	**Introduction and Architecture of Android**					**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**	**10 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**	**10 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**	**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 feeconcession.

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	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:

1. Native Android Applications (Java/Kotlin)
a. Android Mobile Apps built for Android smartphones and tablets using Java or Kotlin programming languages.
b. Target audience: Android users.

2. Native iOS Applications (Swift)
a. iOS Mobile Apps designed for iPhone and iPad using Swift.
b. Target audience: iOS users (Apple ecosystem).

3. Cross-Platform Mobile Apps (Flutter, React Native)
a. Cross-platform apps designed to run on both Android and iOS from a single codebase using frameworks like Flutter or React Native.
b. Target audience: Users on both Android and iOS platforms.

4. Mobile Web Applications (Progressive Web Apps - PWA)
a. Mobile-optimized web applications using HTML5, CSS3, and JavaScript that run in a browser with native-like functionality (offline support, push notifications).
b. Target audience: Users accessing apps via mobile browsers.

Development Tools and Frameworks

Integrated Development Environments (IDEs)

a. Android Studio (for Android): The official IDE for Android development, supporting Java, Kotlin, and Android SDK.
b. 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.
c. Visual Studio Code (VS Code): Lightweight IDE for working with Flutter, React Native, and web development projects.

Cross-Platform Development Frameworks

a. Flutter: Open-source UI framework by Google for building natively compiled applications for mobile, web, and desktop from a single codebase.
b. React Native: Open-source framework developed by Facebook for building cross-platform apps with JavaScript and React.

Backend & Cloud Tools

a. Firebase: Google's backend-as-a-service (BaaS) platform offering authentication, real-time databases, cloud storage, and push notifications for mobile apps.
b. AWS Amplify: Cloud platform for backend services (API, storage, authentication) and mobile deployment.
c. SQLite / Realm: Local storage solutions for mobile apps to manage data storage and retrieval on-device.

Mobile App Testing and Debugging Tools

a. Android Emulator (for Android): A virtual device to run and test Android apps without needing physical devices.
b. Xcode Simulator (for iOS): A tool to simulate different iOS devices and test apps during development.
c. Appium: Open-source tool for automated testing across native, hybrid, and mobile web applications.

Version Control and Collaboration

a. Git: Version control system for managing code changes and collaborating with teams.
b. GitHub / GitLab / Bitbucket: Online platforms for hosting Git repositories, collaboration, and version control management.

Mobile App Deployment Tools

a. Google Play Console: For managing Android app publishing, distribution, and monitoring.
b. Apple App Store Connect: For managing iOS app submissions, reviews, and releases on the Apple App Store.

UI/UX Design Tools

a. Figma / Adobe XD: Tools for UI/UX design and wireframing to create the visual elements of mobile applications before development.
b. Sketch: Vector-based design tool for iOS UI design and prototyping

Text Book(s):

1. T1. Pradeep kothari “Android Application Development - Black Book”, dreamtechpress
2. T2. Barry Burd (Author), “Android Application Development” ALL – IN – ONE FOR Dummies
3. T3. Jeff Mcherter (Author),Scott Gowell (Author), “Professional mobile Application Development” paperback, Wrox - Wiley India Private Limited
4. 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: **CSE2501**	**Course Title: Data Analytics Lab** **Type of Course: Laboratory Course (PCC)**	**L-T-P- C**	**0**	**0**	**2**	**1**
Version No.	1.0					
Course Pre-requisites	MAT2402 Probability and Statistics					
Anti-requisites	NIL					
Course Description	Data Analytics Lab 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	To develop practical skills in data analytics by enabling students to perform data preprocessing and cleaning, apply analytical tools on real-world datasets, visualize data using appropriate techniques, and interpret results to support effective decision-making.					
Course Outcomes	On successful completion of this course, students will be able to: **CO1:** Explain different types of data and variables used in data analytics. [Understand] **CO2:** Apply appropriate statistical methods for analyzing datasets. [Apply] **CO3:**					

	Perform data collection, preprocessing, and analysis for real-world applications. [Apply] **CO4:** Visualize data using charts and graphical techniques to extract meaningful insights. [Apply] **CO5:** Implement data analysis techniques using R programming for practical problem-solving. [Apply]

Course Content:

List of Laboratory Tasks: **30 Sessions**

Module 1:

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:

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:

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:

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:

1. http://www.modernstatisticswithr.com/solutions.html#solutionsch3
2. https://johnmuschelli.com/intro_to_r/
3. 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

This is attained through assessment component mentioned in course handout.

Course Code: IST2505	**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	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					

Description	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 ofEthical Hacking and attain 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. 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:					
Module 1	**Introduction to Ethical Hacking**	Assignment		Programming activity	**8 Hours**
Topics: Introduction to Ethical Hacking Introduction to ethical hacking and its importance, Legal and ethical considerations in ethical hacking, Differentiating between black hat, white hat, and grey hat hacking, Basic cyber security concepts and terminology, Overview of penetration testing methodologies.					
Module 2	**Foot printing and Information Gathering**	Assignment		Programming activity	**8 Hours**
Topics: Passive and active information gathering techniques, Who is lookup, DNS enumeration, and social engineering, Tools and methodologies for foot printing, Google hacking and OSINT (Open Source Intelligence) techniques					
Module 3	**Scanning and Enumeration**	Assignment		Programming activity	**7 Hours**
Topics: Port scanning techniques: SYN,TCP,UDP scans; Service enumeration and version detection; NetBIOS, SNMP, and SMTP enumeration; Vulnerability scanning and assessment					
Module 4	**System Hacking and Exploitation**	Assignment		Programming activity	**7 Hours**

Topics:

Password cracking techniques and tools; Privilege escalation and maintaining access; Malware types and counter measures; Exploiting common vulnerabilities (e.g.,buffer overflow, SQL Injection)

Text Book

1.Rafay Baloch, 2015: "Ethical Hacking and Penetration Testing Guide" Apple Academic Press Inc.

2.Harper Allen, Gray HatHacking: The EthicalHackersHandBook, 3rdEdition, McGrawHill, 2011

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

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: IST2506	**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 ofEthical Hacking and attain 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: CO1.Utilize footprinting and information gathering techniques to collect information about target systems. CO2. Perform network scanning and enumeration using ethical hacking tools to identify active hosts, ports, and services. CO3. Execute vulnerability assessment techniques to identify security weaknesses in systems and networks. CO4. Implement password security analysis and basic system hacking techniques in a controlled laboratory environment. CO5. Demonstrate ethical hacking methodologies and security best practices to assess system security.
Course Content:	

List of Laboratory Tasks:

Experiments:

1. Installation and familiarization of Kali Linux and Ethical Hacking tools.

2. Study of cyber security policies, ethical hacking guidelines, and legal considerations.

3. Perform WHOIS lookup to gather domain information.

4. Perform DNS lookup and DNS enumeration using NSLookup and Dig.

5. Gather information about a target using Open Source Intelligence (OSINT)

techniques.

6. Use Google Dorks for information gathering and footprinting.

7. Analyze website information using online footprinting tools.

8. Perform network reconnaissance using Ping and Traceroute utilities.

9. Discover live hosts in a network using Nmap. 10. Perform TCP Connect Scan using Nmap. 11. Perform SYN Scan using Nmap and analyze results. 12. Perform UDP Port Scanning and identify active services. 13. Perform Service Enumeration and Version Detection using Nmap. 14. Enumerate NetBIOS information from a target system. 15. Perform SNMP Enumeration using appropriate tools. 16. Conduct SMTP Enumeration and identify mail server information. 17. Perform Vulnerability Scanning using OpenVAS or Nessus Essentials. 18. Analyze vulnerability scan reports and identify security risks. 19. Demonstrate Password Strength Analysis using sample password files. 20. Demonstrate SQL Injection vulnerabilities in a vulnerable web application and suggest preventive measures.

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: IST2002	**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 Outcomes	On successful completion of this course the students shall be able to: 2. Describe Numpy and Matrix Operations [Remember]					

	3. 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. Ofsessions: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. Ofsessions: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. Ofsessions: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**	**No. Ofsessions:11**	

			and implemented in real time application.	

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 Algorithmfor developing **Skills development** through **Experiential Learning techniques. This is attained through assessment component mentioned in course handout.**

Course Code: IST2003	**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					

CourseDescription	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 Outcomes	On successful completion of this course the students shall beable 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]	
CourseContent:	**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 Algorithmfor developing **Skills development** through **Experiential Learning techniques.** **This is attained through assessment component mentioned in course handout.**		

Course Code: **CSE2509**	**Course Title: Data Analytics** **Type of Course: Theory**	**L-T-P-C**	3	0	0	3
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 Outcomes	**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	**11 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	**12 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	**10 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	Case studies	Programming	**12**

	Analysis-CO4			**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

5. 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.

6. Introduction to statistics and Data analytics, Christian H, Michael S, Springer,2016

7. Introduction to R- Robert Parker, John Mushcelli and Andrew Jaffe, Johns Hopkins University, 2020 (E-resource)

8. Introduction to Time Series and Forecasting (Springer Texts in Statistics), Peter Brockwell, Richard A. Davis, Springer, 2016.

References

3. 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.

4. 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.

List of course in Project Work basket (PRW)

Course Code: CSE7000	**Course Title: Internship** **Type of Course:**	**L- T- P- C**	0	0	0	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 Learningtechniques.					
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 Learningtechniques.
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. 4. (Analyze) 5. Interpret the events and results for meaningful conclusions. (Evaluate) 6. 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 Learningtechniques.
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)

List of Mandatory Courses (MAC)

Course Code: **PPS1025**	**CourseTitle:Industry Readiness Program – I** **TypeofCourse: HSMC**	**L- T - P- C**	0	0	2	0
Version No.	1.0					
Course Pre-requisites	NIL					
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"EmployabilityforYoungProfessionals"andattainSKILLDEVELOPMENT through PARTICIPATIVE LEARNING techniques.**					
CourseOutcomes	**Onsuccessfulcompletionofthiscoursethestudentsshallbeableto:** CO1**Define**their career goals CO2**Practice ethical habits**forbetter career success CO3**Demonstrate**effectiveemail writing techniques					
Course Content						
Module1	**Goal Setting & Grooming**	Classroomactivities	**10 Hours**			
Topics:SMART Goals, formal grooming through self-introduction activity **Activity:**Realworld scenarios						
Module2	**Habit Formation**	Roleplays	**10 Hours**			
Topics: Professional and Personal ethics for success and activity-based practice **Activity:**Students to present 2 min video on building professional ethics						
Module3	**Email Etiquettes**	Individualandgroup presentation	**10Hours**			
Topics: Types of prompts to generate effective or desired results for email etiquettes **Activity:**Individualstudent presenting various search prompts						
TargetedApplication&Toolsthatcanbe used: 1. TED Talks 2. YouTube Links Activities						
Assignmentproposedforthis course Assignment 1: SMART Goal Assignment 2: AI tools for prompt search						
ContinuousIndividualAssessment Module 1: Presentation Module 2: Activity based assessment Module 3: Class assessment						
Thetopicsrelatedtoskilldevelopment: 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: PPS1026	**CourseTitle:Industry Readiness Program – II** TypeofCourse:PracticalOnlyCourse	L-T-P-C	0	0	2	0
Version No.	1.0					
Course Pre-requisites						
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 ReadinessforYoungProfessionals"andattain SKILLDEVELOPMENT through PARTICIPATIVE LEARNING techniques.**					
Course Outcomes	**Onsuccessfulcompletionofthiscoursethestudentsshallbeableto:** CO1**Apply**different communication skills for success in workplace CO2**Practice team building skills**forcareer success CO3**Demonstrate**ethical leadership skills in workplace					
Course Content						
Module1	**Effective Communication**	Classroomactivities	**10 Hours**			
Topics: Practice effective communication skills (Verbal, Non-verbal, Written and Visual) **Activity:**Use social media prompts to prepare self-introduction videos						
Module2	**Team Building**	Group Activity	**10 Hours**			
Topics: Skills of an effective team player **Activity:**Student group activity to build class networking						
Module3	**Leadership**	Case study	**10Hours**			
Topics: Types of leadership, using empathy in leadership **Activity:**Individualpresentation by students on corporate leaders.						
Faculty : L&D						
TargetedApplication&Toolsthatcanbe used: 1. TED Talks 2. YouTube Links Activities						
Assignmentproposedforthis course Assignment 1: One minute reel Assignment 2: Team building assignment						
ContinuousIndividualAssessment 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.

CourseCode: APT4002	**CourseTitle:Introductionto Aptitude**	**L-T-P-C**	0	0	2	0
Version No.	1.0					
CoursePre-equisites						
Anti-requisites	Nil					
Course Description	Theobjectiveofthiscourseistopreparethetraineestotackle thequestionson 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 higherorderthinkingquestions.Thefocusofthiscourseistoteachthestudents tonotonlygettothe correct answers,butto get there fasterthanever before, which will improve their employability factor.					
Course Objective	The objective of the course is to familiarize the learners with the concepts ofAptitude and attain Skill Development through Problem Solving techniques.					
Course Outcomes	Onsuccessfulcompletionofthecoursethestudentsshallbeableto: CO1]**Recall**allthebasicmathematicalconceptstheylearntinhighschool. CO2] **Identify** the principle concept needed in a question. CO3] **Solve** the quantitative and logical ability questions with the appropriate concept. CO4]**Analyze**thedatagivenincomplex problems. CO5]**Rearrange**theinformation tosimplify thequestion					
CourseContent:						
Module1	Quantitative Ability	Assignment	Bloom'sLevel: Application		**12Hours**	
Topics: IntroductiontoAptitude,workingofTables,Squares,Cubes						
Module2	Logical Reasoning	Assignment	Bloom'sLevel: Application		**18Hours**	
Topics: Linear&CircularArrangementPuzzle,Coding&Decoding,BloodRelations,Directions,Orderingand Ranking, Clocks and Calendars,Number Series, Wrong number series, Visual Reasoning						
TargetedApplication&Toolsthatcanbeused: Applicationarea:PlacementactivitiesandCompetitiveexaminations . Tools: LMS						
TextBook 1. QuantitativeAptitudebyRS Aggarwal 2. Verbal&Non-VerbalReasoningbyRSAggarwal						
References 1. www.indiabix.com 2. www.youtube.com/c/TheAptitudeGuy/videos						

Topics relevant to Skill development: Quantitative and reasoning aptitude for Skill DevelopmentthroughProblemsolvingTechniques.Thisisattainedthroughassessment componentmentionedincoursehandout.	

Course Code: **APT4004**	**CourseTitle:AptitudeTraining-Intermediate** TypeofCourse:PracticalOnlyCourse	**L- T - P- C**	0	0	2	0	
VersionNo.							
CoursePre-requisites	StudentsshouldhavethebasicconceptsofQuantitativeaptitudealongwithits applicationsinreallifeproblems.						
Anti-requisites	NIL						
Course Description	Thisisaskill-basedtrainingprogramforthestudents.Thiscourseisdesigned to enable the students to enhance their skills in Quantitative Aptitude.						
Course Objective	**Theobjectiveofthecourseistofamiliarizethelearnerswiththeconcept sof Aptitude and attain Skill Development through Problem Solving techniques.**						
CourseOutcomes	**Onsuccessfulcompletionofthiscoursethestudentsshallbeableto:** CO1: Recall allthebasicmathematicalconcepts. CO2:Identifytheprincipleconceptneededinaquestion. CO3: Solve the quantitative and logical ability questions with the appropriateconcept. CO4:Analyzethedatagivenincomplex problems.						
Course Content:							
Module1	**QuantitativeAbility1**	Assignment				**16Hours**	
Topics: Number System, Percentage, Ratio and Proportion, Average, Mixture and Allegation, Time and Work, Profit and Loss							
Module2	QuantitativeAbility2	Assignment				**14Hours**	
Topics: TimeSpeedandDistance,BoatsandStreams,SimpleInterest,CompoundInterest,Probability, Permutation and Combination							
Targeted Application & Tools that can be used: Applicationarea:PlacementactivitiesandCompetitiveexaminations. Tools: LMS							
ContinuousEvaluation:							
CA1–OnlineTest CA2–OnlineTest CA3–OnlineTest Assignment							

Text Book:

1. FastTrackObjectivebyRajesh Verma
2. RSAggarwal
3. RakeshYadav

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: APT4006	**Course Title: Logical and Critical Thinking** **Type of Course:** Audited	**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 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. CO3] **Analyze** and structure the reasoning techniques and spatial visualizationskills					
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 Internal Assessments					
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: **CHE7601**	**Environmental Studies** **Type of Course: MOOC course**	**L- T- P- C**	**1-**	**0**	**0**	**0**
		Contact 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.				
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: 1. An attitude of enquiry. 2. Write reports **The topics related to Environment and Sustainability :** All topics in theory component are relevant to Environment and Sustainability.					
Module 4	**Social Issues, Legislation, and Practical Applications**				

Course Code: CIV7601	Course Title: Universal Human Values and Ethics Type of Course: MAC course	L - T - P - C	1	0	0	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*:

1) UHV II - https://www.youtube.com/watch?v=NhFBzn5qKIM&list=PLWDeKF97v9SO8vvjC1KyqteziTbTjN1So&pp=0gcJCWMEOCosWNin
2) Lecture by Dr. Kumar Sambhav, NPTEL course: Universal Human Values, https://onlinecourses.swayam2.ac.in/aic22_ge23/preview
3) 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

1. 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
2. Human Values, A.N. Tripathi, New Age Intl. Publishers, New Delhi, 2019.
3. Premvir Kapoor, Professional Ethics and Human Values, Khanna Book Publishing, New Delhi, 2022.

Reference Books

1. E.F. Schumacher, 1973, Small is Beautiful: a study of economics as if people mattered, Blond & Briggs, Britain.
2. Sussan George, 1976, How the Other Half Dies, Penguin Press. Reprinted 1986.
3. Donella H. Meadows, Dennis L. Meadows, Jorgen Randers, William W. Behrens III, 1972, Limits to Growth – Club of Rome's report, Universe Books.
4. A Nagraj, 1998, Jeevan Vidya Ek Parichay, Divya Path Sansthan, Amarkantak.
5. P L Dhar, RR Gaur, 1990, Science and Humanism, Commonwealth Publishers.

6. A N Tripathy, 2003, Human Values, New Age International Publishers.
7. E G Seebauer& Robert L. Berry, 2000, Fundamentals of Ethics for Scientists & Engineers, Oxford University Press
8. M Govindrajran, S Natrajan& V.S. Senthil Kumar, Engineering Ethics (including Human Values), Eastern Economy Edition, Prentice Hall of India Ltd.
9. B P Banerjee, 2005, Foundations of Ethics and Management, Excel Books.
10. William P. Cunningham and Mary Ann Cunningham (2020), Principles of Environmental Science: Inquiry & Applications, 9th Edition, McGraw-Hill Education, USA.

Resources:

1. https://onlinecourses.swayam2.ac.in/imb25_mg195/preview
2. https://onlinecourses.nptel.ac.in/noc25_mg141/preview
3. https://onlinecourses.swayam2.ac.in/ini25_hs52/preview
4. https://onlinecourses.nptel.ac.in/noc25_hs219/preview
5. https://onlinecourses.swayam2.ac.in/cec25_mg14/preview
6. https://onlinecourses.swayam2.ac.in/imb25_mg195/preview
7. https://onlinecourses.swayam2.ac.in/imb25_mg196/preview

Topics relevant to Skill Development:

1. An attitude of enquiry.
2. Write reports

The topics related to Human values and Professional ethics:

All topics in are relevant to Human values and Professional ethics.

Course Code: **LAW7601**	**Indian Constitution** **Type of Course: MAC course**	**L- L-T-P- C**	**1**	**0**	**0**	**0**
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 fundamentals of Indian Constitution concepts and their relevance to 75+ Years of Republic of India (https://constitution75.com/) as well as **#AzaadiKaAmrutMahotsav / Azadi Ka Amrit Mahotsav** (https://amritmahotsav.nic.in). It is designed to equip students with the knowledge about the Constitution of India. This course aims to introduce the constitutional law of India to students from all walks of life and help them					

	understand the constitutional principles as applied and understood in everyday life. The objective of making the Constitution of India, familiar to all students, and not only to law students, this course aims and objectifies legal understanding in the simplest of forms. **This course is designed to cater to Constitutional Studies.**
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 basic understanding of the Indian Constitution and the concepts and issues relevant to day-to-day life of the nation and to equip the Citizen with the zeal of capacity building. Recognizing and identify the values of the Constitution of India. 2. Enabling the Citizen-centric Awareness of Rights and Responsibilities of the State 3. Explain the role of the State actors in building India. 4. Understanding the Gandhian vision over the power of the LSG (Local Self-Governance)
Course Content:	
Module 1	**Understanding the Making of the Constitution: The Constituent Assembly & The Constitution of India**
Topics: Historical Context of Constituent Assembly - Compositions & Functions of Constituent Assembly What is a Constitution? – Why have a Constitution? – Constitutional Change - Features of Indian Constitution – Preamble of Indian Constitution	
Module 2	**Citizen's Fundamental Rights and State's Responsibilities (Directive Principles)**
Topics: Introduction to Fundamental Rights - Right to Equality – Facets of Right to Equality - Right to Freedom - Constitutional Position of Some Democratic Rights - Right Against Exploitation - Right to Freedom of Religion - Right to Constitutional Remedies Directive Principles of the State Policy	
Module 3	**Organs Of the Government**
Topics: **Executive**: The President of India - Powers and Functions of President of India - Emergency Powers and the Position of the President **Legislature**: Union Council of Ministers - Prime Minister - The Rajya Sabha - The Lok Sabha - Relation between the Lok Sabha & Rajya Sabha - Office of the Speaker – Important Parliamentary Committees **Judiciary**: The Structure and Organization of the Judiciary & the High Court - The Supreme Court -	

Role of The Supreme Court - Judicial Activism in India - Basic Structure Doctrine & PIL

Module 4	Federalism & Decentralization		

Topics:

What is Federalism? - Centre-State Legislative Relations - Centre-State Administrative Relations - Centre-State Financial Relations

The 5th & 6th Schedules - Municipality- (History of Indian Municipality, Organization & Functions) – Panchayat 1 (Idea of Panchayat, Organization and Powers of Panchayats in India)

Targeted Application & Tools that can be used:

Application areas to familiarize students with fundamentals of Indian Constitutional concepts.

Tools: Online Tools – NPTEL and Swayam.

Project work/Assignment:

Assessment Type

- Online end term exam will be conducted as notified by the Presidency University.

Online Link*:

1) Prof. Amitabha Ray, SWAYAM Course: "Constitutional Government & Democracy in India"

https://onlinecourses.swayam2.ac.in/cec19_hs13/preview

* Other source links are available in below Resources link.

Text Book

1. Durga Das Basu --- Introduction to the Constitution of India, 23rd Edition (Gurgaon; LexisNexis, 2018).
2. MP Jain's Constitutional Law of India, Lexis Nexis
3. V.N Shukla's Indian Constitutional Law, M.P Singh 13th Edition
4. MV Pylee's Constitution of India
5. J.C.Johari -- The Constitution of India: A Politico-Legal Study (Greater Noida: Sterling Publishers Pvt. Ltd. 2013).
6. Himangshu Roy and M.P.Singh – Indian Political System, 4th Edition (Bengaluru; Pearson Education, 2018)
7. Vidya Bhushan & Vishnoo Bhagwan--- Indian Administration (S. Chand, 2011)
8. S.R.Maheswari --- Indian Administration (Orient Blackswan, 2001)
9. Dr. A.Avasthi & A.P. Avasthi --- Indian Administration (L.N. Agarwal Educational Publishing, 2017).
10. B. L. Fadia --- Indian Government and Politics (Sahitya a. Bhawan, 13th Revised Edition, 2017).
11. P.M.Bakshi – The Constitution of India (Prayagraj, UP; a. Universal Law Publishing, January, 2018)

Reference Books

1. HM Seervai, Constitutional Law of India, 4th Ed. Vol I, II, & III

2. Uday Raj Rai, Constitutional Law-I 3. Democracy and Constitutionalism in India, Oxford University Press 2009 **Resources:** 1. https://onlinecourses.nptel.ac.in/noc20_lw03/course?&force_user=true 2. https://onlinecourses.swayam2.ac.in/cec19_hs13/course?&force_user=true 3. https://nptel.ac.in/courses/129106003 4. https://nptel.ac.in/courses/129106411 5. https://nptel.ac.in/courses/129105608 6. https://nptel.ac.in/courses/129106002	
Topics relevant to Skill Development: 1. An attitude of inquiry. 2. Write reports **The topics related to Constitutional Studies and its application :** All topics in theory component are relevant to Indian Constitution.	

Professional Electives Courses (PEC)

Course Code: CSE3426	**Course Title: Front-end Full Stack Development** **Course Type : Lab Integrated**	**L- T-P-C**	2		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:	**Course Title: Java Full Stack Development**	**L-T-P-**	2	0	2	3

CSE3427	**Course Type: Lab Integrated**	**C**				

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: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					
CourseDescription	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 Outcomes	Onsuccessfulcompletionofthiscoursethestudentsshallbeableto: 1. **Process** text data to derive information from text. [Apply] 2. **Apply** insights from textual information to real-world business. [Apply] 3. **Develop** solutions for a particular NLP problem using different machine learning and deep learning techniques. [Apply] 4. **Utilize** different NLP tools and packages. [Apply]					
CourseContent:						
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:

3. Google Colab
4. Python IDEs like PyCharm

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

2. 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):

4. 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).

5. 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)

Course Code: CAI3428	**Course Title: Practical Deep Learning with TensorFlow** **Type of Course: Discipline Elective - Theory & Integrated Laboratory**	**L- T- P- C**	**2**	**0**	**2**	**3**
Version No.	1.0					
Course Pre-requisites	CSE2264					
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.					
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. Implement backpropagation and gradient descent techniques to train neural networks effectively. (Apply) 2. Build and train deep learning models using Python libraries such as TensorFlow					

	and Keras for real-world applications. (Apply) 3. Utilize deep learning techniques for image classification, object detection, sentiment analysis, and language modeling. (Apply)			
Course Content:				
Module 1	**Basics of Neural Networks**	Assignment		**18[8L+10P]** **Sessions**
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.				
Module 2	**TensorFlow Basics**	Assignment		**14[7L+7P]** **Sessions**
Topics: Introduction to TensorFlow, TensorFlow dataset, Machine Learning with TensorFlow				
Module 3	**Deep Learning methods with Tensor Flow and Keras**	Assignment		**14[6L+8P]** **Sessions**
Topics: Main Features of TensorFlow, Keras basics, AI with Keras.				
Project work/Assignment:				
1. Assignment 1 on (Module 1 and Module 2) **2. 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

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

<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 Outcomes</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</td><td>Assignme</td><td colspan="4">Practical</td><td>No. of</td></tr>
</table>

	Learning for Vision	nt		**Classes:8**
Introduction to Deep Learning & Neural Networks, Convolutional Neural Networks (CNNs) Architecture 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: IST2501	**Course Title:** **Optimization Methods in Machine Learning** **Type of Course:** **Discipline Elective in Artificial Intelligence and Machine Learning Basket** **Theory**	**L-T-P-C**	3-0-0-3
Version No.	1.0		
Course Pre-requisites	**CSE2264 Machine Learning Techniques**		
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 **Employability** through **Problem Solving**Methodologies.			
Course Outcomes	On successful completion of this course the students shall be able to: 1. **Describe** fundamentals of Machine learning [**Knowledge].** 2. **Explain**Machine learning models **[Comprehension].** 3. **Discuss**Convex optimization models **[Comprehension].** 4. **Apply** Methods for convex optimization **[Application].**			
Course Content:				
Module 1:	**Fundamentals of Machine learning**	**Quiz**	Knowledge based Quiz	**8 Sessions**
Topics: Machine learning paradigm, empirical risk minimization, structural risk minimization, learning guarantees, introduction of VC-dimension.				
Module 2:	**Machine learning models**	**Quiz**	Comprehension based Quiz	**10 Sessions**
Topics: logistic regression, support vector machines, sparse regression, low dimensional embedding, low rank matrix factorization, sparse PCA, multiple kernel learning.				
Module 3	**Convex optimization models**	**Assignment**	Batch-wise Assignments	**9 Sessions**
Topics: linear optimization, convex quadratic optimization, second order cone optimization, semidefinite optimization, convex composite optimization				
Module 4:	**Methods for convex optimization**	Assignment and Presentation	Batch-wise Assignment and Presentations	**11 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 Matlab tool**				
Project work/Assignment: **Survey on Methods for convex 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 related to development of "EMPLOYABILITY SKILL": Convex optimization models and Methods for convex optimization, for development of Employability Skills through Participative Learning Techniques. This is attained through assessment components mentioned in course handout.

Course Code: IST3410	Course Name: Data Handling and Visualization Techniques (With AI Enhancements) Type of Course: Theory Course	L- T-P- C	2-0-2-3
Version No.	2.0		
NCrF Level-	5.5		
Course Pre-requisites	CSD1710		
Co-requisites	NIL		
Anti-requisites	NIL		
Course Description	This course builds practical knowledge in data collection, preparation, analysis, and visualization. It emphasizes working with large datasets, using Python libraries, and developing visually effective and insightful data visualizations. The lab-integrated component strengthens proficiency in handling real-world data with Python-based tools. AI enhancements include machine learning for pattern detection, data summarization, and intelligent chart selection through AI-driven visualization assistants.		
Course Objective			
Course Outcomes	**On successful completion of the course the students shall be able to:** CO1: Understand the data handling pipeline, including acquisition, cleaning, transformation, and storage. [Understand] CO2: Use structured and unstructured data at scale using appropriate tools. [Apply] CO3: Apply fundamental principles of data visualization and storytelling. [Apply] CO4: Implement visualization techniques using Python libraries. [Apply] CO5:Analyze different visualization strategies for diverse data types using AI-based recommendations. [Analyse]		
Course Content:			
Module 1	**Introduction to Data Handling**	Assessment : Quiz / Conceptual Case Study etc	12 Sessions
Data acquisition and basic models: Web scraping, APIs, binary formats, Data cleaning, handling missing data, data transformation, String operations and data preparation, **Python Libraries**: NumPy, Pandas, Matplotlib, GGPlot			

- **AI Perspective**:
  - AI-assisted data cleaning using anomaly detection
  - Automatic imputation strategy selection using ML
  - Polars and DuckDB for columnar, out-of-core processing
  - Schema inference using LLMs (eg. Hugginface transformers + lightweight models)

Module 2	**Data Wrangling and Analysis**	Assessment : Quiz / Conceptual Case Study etc	11 Sessions

Hierarchical indexing, merging, reshaping datasets, Working with large datasets: memory optimization and processing, Predictive analysis and recommender systems (case studies)

- **AI Perspective**:
  - Use of AI to infer data schemas and automate joins/merges
  - Data profiling using tools like Pandas Profiling, DataPrep
  - *Vector search pipelines*—store embeddings in **FAISS** or **Chroma** for similarity-based joins or record-linkage tasks.

Module 3	**Data Visualization Techniques**	Assessment : Quiz / Conceptual Case Study etc	11 Sessions

Visualization design principles: abstraction, perception, color theory, Scalar/point visualizations: height maps, color maps, contouring, Vector/matrix visualizations: glyphs, vector fields

- **AI Perspective**:
  - AI-based chart recommendation systems (e.g., ChartGPT, VizGPT)
  - Explainable AI for visual diagnostics
  - *Generative-AI dashboards*—autogenerate dashboards from natural-language specs using Pandasai or Open Interpreter.

Module 4	**Diverse Types of Visual Analysis**	Assessment : Quiz / Conceptual Case Study etc	11 Sessions

Visualization of time-series, multivariate, and textual data, Web analytics, dashboarding for business insights

- **AI Perspective**:
  - Sentiment visualization using NLP
  - Time-series prediction overlays using Facebook Prophet/ARIMA models
  - RAG-powered storyline charts

List of Laboratory Tasks

Lab sheet 1 (3 Practical Sessions)

- NumPy & Pandas functions
- Web scraping & interacting with APIs

Lab sheet 2 (2 Practical Sessions)

- Data cleaning and wrangling
- Handling missing values

Lab sheet 3 (4 Practical Sessions)

- Data visualization using Matplotlib, Seaborn
- Massive datasets from finance, healthcare, census

Lab sheet 4 (4 Practical Sessions)

- Time-series stock market data
- Market basket visualization

• Text analysis using word clouds, frequency graphs
Targeted Applications & Tools:
• Anaconda • Google Colab • Python (Pandas, NumPy, Matplotlib, Seaborn, Plotly) • Google Data Studio • Tableau Public
Text Book • **McKinney, W. Python for Data Analysis, O'Reilly Media, 2nd Edition, 2017** • **Munzner, T., Visualization Analysis and Design, CRC Press, 2015** • **Ossama Embarak, Data Analysis and Visualization Using Python, Apress, 2018**
References • **Salvador García, Julián Luengo, Francisco Herrera, Data Preprocessing in Data Mining, Springer, 2015** • **Belorkar, A., Interactive Data Visualization with Python, Packt, 2nd Edition, 2018**
E-Resources/Web Resources **Making data visual : a practical guide to using visualization for insight, Shroff Publishers and Distributors, 2018** **http://puniversity.informaticsglobal.com:2232/cgi-bin/koha/opac-detail.pl?biblionumber=17611** **Python for Data Science, IIT Madras** **https://nptel.ac.in/courses/106106212**

Course Code: IST3406	Course Name: **Human Computer Interaction** Type of Course: Theory Course	L-T-P- C	3-0-0-3
Version No.	2.0		
NCrF Level-	5.5		
Course Pre-requisites	NIL		
Co-requisites	NIL		
Anti-requisites	NIL		
Course Description	This course introduces the principles, methods, and emerging trends in Human-Computer Interaction (HCI). It focuses on designing intuitive, user-centered interfaces for web, mobile, and emerging platforms like AR/VR. The course integrates cognitive psychology, design thinking, accessibility standards, and modern prototyping tools to build effective, inclusive, and interactive digital experiences.		
Course Objective	The objective of the course is SKILL DEVELOPMENT of students by using PROBLEM SOLVING METHODOLOGIES		

Course Outcomes	**On successful completion of the course the students shall be able to:** **CO1: Recall the fundamental concepts, models, and principles of Human-Computer Interaction. [Remember]** **CO2: Explain cognitive factors, ergonomics, and design principles influencing user interface development. [Understand]** **CO3: Interpret emerging trends and challenges in HCI, including multimodal interaction, mobile interfaces, and inclusive design. [Apply]** **CO4: Apply standard guidelines and methodologies to design user-centered interactive systems. [Apply]** **CO5: Implement evaluation techniques such as usability testing and expert reviews to assess interface effectiveness. [Apply]**		
Course Content:			
Module 1	**Introduction to HCI and Emerging Trends**	Assessment : Quiz / Conceptual Case Study etc	11 Sessions
Introduction to HCI: Definition, Scope, and Importance, Human Perception, Input-Output Channels, Human Memory, Cognition, and Emotion, Cognitive frameworks and models of interaction, Psychology and the design of interactive systems, Ergonomics and Universal Usability, Introduction to AI in HCI: Conversational interfaces, intelligent agents, Basics of AR/VR interfaces and immersive interaction			
Module 2	**Interface Design Principles and Prototyping**	Assessment : Quiz / Conceptual Case Study etc	11 Sessions
Good and Bad Design: Case Studies, Interaction Design: Guidelines, Principles, Theories, Design Process: Conceptual and Physical Design, Prototyping and Construction: Wireframes, Mockups, Interactive Prototypes, Four Pillars of Design and Development Methodologies, Participatory Design and Scenario Development, Legal and Ethical Issues in Interface Design, UX Design Fundamentals and Design Thinking Approach, Accessibility Standards (WCAG) and Inclusive Design			
Module 3	**Evaluating Interface Design and User Testing**	Assessment : Quiz / Conceptual Case Study etc	11 Sessions
Importance of Interface Evaluation, Evaluation Goals and Strategies, Expert Reviews, Usability Testing, and Usability Labs, Surveys, Acceptance Testing, and Real-world Evaluation, Controlled Experiments and Choosing Evaluation Methods, Introduction to A/B Testing and Remote Usability Testing, Natural Language Processing in HCI Applications			
Module 4	**Information Presentation, Multimodal and Ubiquitous Interfaces**	Assessment : Quiz / Conceptual Case Study etc	12 Sessions
Data Type by Task Taxonomy for Information Presentation, Challenges in Information Visualization, Groupware: Asynchronous and Synchronous Distributed Interfaces, Face-to-Face Interfaces, Speech and Auditory Interfaces, Multimodal Interaction and Design for Diversity, Graphical User Interfaces for Mobile, Web, and Wearables, AR/VR Interfaces for Immersive Experiences, Ubiquitous Computing and Context-Aware Interaction, Designing for Smart Environments (IoT-based Interaction)			
Targeted Applications & Tools:			
Figma, Adobe XD, Sketch	Design and prototyping of interactive user interfaces for web and mobile applications.		
HTML, CSS, JavaScript	Implementation of interactive front-end interfaces following HCI principles.		
UsabilityHub, Maze, Google Forms	Usability testing and gathering user feedback for interface evaluation.		
Eye-tracking Tools (Optional)	Studying user attention patterns to improve interface design.		
Accessibility Evaluation Tools (e.g., WAVE)	Ensuring inclusive and accessible interface design for diverse user groups.		
Skill-Building Activities through Experiential Learning:			

- User Interface Redesign Challenge
  Select a poorly designed website or app interface. Identify usability flaws using HCI principles. Redesign the interface using Figma or Adobe XD, focusing on accessibility, visual hierarchy, and interaction flow. Present before-and-after designs and justify improvements based on design guidelines and cognitive ergonomics.

- Usability Testing Simulation
  Conduct a usability test on an interactive prototype (web or mobile) developed using HTML and CSS. Recruit peers as users to complete specific tasks. Observe interaction patterns, record difficulties, and suggest improvements based on usability metrics such as effectiveness, efficiency, and user satisfaction.

- Multimodal Interaction Prototype
  Develop a simple multimodal interface combining at least two input/output modes (e.g., voice and touch) using prototyping tools or simulations. Explore how combining different modalities enhances usability, accessibility, and user experience in complex environments like smart homes or mobile applications.

Text Book

- Ben Shneiderman and Catherine Plaisant, "Designing the User Interface: Strategies for Effective Human-Computer Interaction", 6th Edition, Pearson Addison Wesley, 2016.
- Dix A. et al. "Human-Computer Interaction", 3rd Edition, Pearson Prentice Hall, 2004.

References

- Yvonne Rogers, Helen sharp, Jenny Preece, "Interaction Design: Beyond Human Computer Interaction", 5th Edition, Wiley, 2019.
- The Essentials of Interaction Design, Fourth Edition by Cooper, Reimann, Cronin, & Noessel (2014).

E-Resources/Web Resources

- Human Computer Interaction https://nptel.ac.in/courses/106103115
- HCI Interactions https://onlinecourses.nptel.ac.in/noc19_cs86/preview

Course Code: IST3401	Course Name: UI UX Design Type of Course: Theory Course	L-T-P- C	3-0-0-3
Version No.	2.0		
NCrF Level-	5.5		
Course Pre-requisites	NIL		
Co-requisites	NIL		
Anti-requisites	NIL		
Course Description	This course provides an in-depth understanding of User Experience (UX) Design principles, methods, and tools for creating intuitive, efficient, and engaging digital products. It emphasizes user research, interaction design, usability testing, and contemporary practices such as UX for mobile, inclusive design, and AI-integrated user experiences across web and mobile platforms.		
Course Objective	The course aims to equip learners with the skills to create user-centered digital products by combining theoretical knowledge with practical application of design thinking methodologies. Objectives include understanding user needs and		

	behavior, applying visual design principles to create appealing interfaces, developing wireframes and interactive prototypes, conducting user research and usability testing, and mastering industry-standard design tools to deliver seamless, intuitive user experiences.		
Course Outcomes	**On successful completion of the course the students shall be able to:** **CO1: Recall key concepts, principles, and terminology related to User Experience design. [Remember]** **CO2: Explain the process of user research, persona development, and usability evaluation. [Understand]** **CO3: Apply design principles to create wireframes, prototypes, and interactive mock-ups. [Apply]** **CO4: Explore inclusive, mobile-friendly, user interfaces for digital products. [Apply]** **CO5: Analyze the user experience using usability testing and iterative design. [Analyze]**		
Course Content:			
Module 1	**Introduction to UI**	Assessment : Quiz / Conceptual Case Study etc	08 Sessions
Topics: What is User Interface Design (UI) -The Relationship Between UI and UX , Roles in UI/UX, A Brief Historical Overview of Interface Design, Interface Conventions, Approaches to Screen Based UI, Template vs Content, Formal Elements of Interface Design, Active Elements of Interface Design, Composing the Elements of Interface Design, UI Design Process, Visual Communication design component in Interface Design.			
Module 2	**Introduction to UX**	Assessment : Quiz / Conceptual Case Study etc	10 Sessions
Topics: UX Basics- Foundation of UX design, Good and poor design, Understanding Your Users, Designing the Experience Elements of user Experience, Visual Design Principles, Functional Layout, Interaction design, Introduction to the Interface, Navigation Design, User Testing, Developing and Releasing Your Design.			
Module 3	**UI UX Design Tools**	Assessment : Quiz / Conceptual Case Study etc	10 Sessions
Topics: User Study- Interviews, writing personas: user and device personas, User Context, Building Low Fidelity Wireframe and High-Fidelity Polished Wireframe Using wireframing Tools, Creating the working Prototype using Prototyping tools, Sharing and Exporting Design.			
Module 4	**Visual Design and UI Prototyping**	Assessment : Quiz / Conceptual Case Study etc	12 Sessions
Topics: Fundamentals of Visual Design, Color theory, typography, and layout, Creating visually appealing interfaces, UI Prototyping Tools, Rapid prototyping techniques. Mobile-First Design :Design considerations for mobile devices, Responsive web design principles Adaptive and Responsive Prototyping, Building prototypes for various screen sizes, Testing on multiple devices			
Targeted Applications & Tools:			
Tools/Software/Platform		Application	
Figma, Adobe XD, Axure		Wireframing, prototyping, and interactive mock-up development	
Optimal Workshop, Dovetail		User research, surveys, and usability testing	
Google Analytics, Hotjar		Analyzing user behavior and identifying UX improvements	
WAVE, Axe Accessibility Tools		Accessibility evaluation for web and mobile	

	interfaces

Skill-Building Activities through Experiential Learning:

- Persona Development Workshop:
  Conduct user interviews or secondary research to create detailed personas representing target users. Map user goals, frustrations, and behaviors, helping build empathy and inform design decisions for real-world digital product scenarios.

- Mobile App Prototype Design:
  Using Figma or Adobe XD, design a mobile app prototype for a service of your choice, applying mobile-first principles, accessibility guidelines, and incorporating AI-driven features such as smart suggestions or chatbots.

- Usability Testing and Iterative Improvement:
  Perform usability testing on an existing website or prototype using heuristic evaluation and user testing tools. Identify design flaws, propose improvements, and document an iterative design process based on user feedback.

Text Book

- Unger, R., & Chandler, C. (2012). A Project Guide to UX Design (2nd ed.). New Riders Publishing.
- Garrett, J. J. (2011). The Elements of User Experience. Pearson Education.

References

- Galitz, W. O. (2007). The Essential Guide to User Interface Design (3rd ed.). Wiley Publishing.
- Hartson, R., & Pyla, P. S. (2012). The UX Book: Process and Guidelines for Ensuring a Quality User Experience. Elsevier.
- Lidwell, W., Holden, K., & Butler, J. (2010). Universal Principles of Design. Rockport Publishers.

E-Resources/Web Resources

- IEEE: UI-UX design using user centred design method
- ARRUS Journal: UI/UX Design for Web-Based Learning Apps
- SWAYAM NPTEL: User Interface Design – IIT Roorkee
- Coursera: Introduction to UX Principles and Processes

Course Code : IST3411	Course Name: Search Engine Optimizationn Type of Course: Theory	L-T-P-C	3-0-0-3
Version No.	2.0		
NCrF Level-	5.5		
Course Pre-requisites	NIL		
Co-requisites	NIL		
Anti-requisites	NIL		
Course Description	This course introduces foundational and advanced concepts of Search Engine Optimization (SEO), covering search engine functionality, SEO-friendly web design, on-page and off-page optimization, technical SEO, analytics integration, and content structuring. It emphasizes keyword strategy, link building, and search algorithm awareness. AI integration includes NLP-based keyword clustering, AI content scoring, and predictive SEO ranking analysis using tools like SurferSEO and Clearscope.		
Course	The course aims to:		

Objective	1. Introduce the principles and working of search engines. 2. Develop skills to design SEO-friendly websites and optimize technical foundations. 3. Equip students with knowledge of both On-page and Off-page SEO strategies. 4. Familiarize students with SEO tools, analytics, and performance tracking. 5. Prepare students to create data-driven SEO strategies for real-world applications.		
Course Outcomes	**On successful completion of the course the students shall be able to:** CO1: Explain how search engines operate and influence website visibility. [Understand] CO2: Follow SEO principles in website development. [Understand] CO3: Implement technical and content-based optimizations effectively. [Apply] CO4: Differentiate and apply on-page and off-page SEO strategies. [Analyse] CO5: Use analytics and AI-based SEO tools for continuous performance tracking and optimization. [Apply]		
Course Content:			
Module 1	How Search Engines Work	Assessment : Quiz/Assignment	12 Sessions
Topics: Putting Search Engines in Context, Meeting the Search Engines, Recognizing and Reading Search Results,Getting Your Site in the Right Results,Knowing What Drives Search Results,Spam Issues: When Search Engines Get Fooled, Indexing, ranking algorithms, and search result structures, Factors influencing rankings, Spam and black-hat SEO practices • AI Perspective: o Use of Google BERT and RankBrain in search ranking o AI-based search behavior analysis and click-through prediction			
Module 2	SEO Web Design	Assessment : Assignment	11 Sessions
Topics: Basics of SEO Web Design,Building an SEO-Friendly Site,Making Pages Search Engine-Compatible,Perfecting Navigation and Linking Techniques • AI Perspective: o AI-aided UX analysis for SEO (e.g., heatmaps, dwell time tracking) o Semantic HTML generation using generative AI for accessibility and indexing			
Module 3	Optimizing the Foundations & Analyzing Results	Assessment : Quiz/Case Study	11 Sessions
Topics: Server Issues: Why Your Server Matters,Using Redirects for SEO (301 Redirects),Backend Issues: Content Management Systems,Solving SEO Roadblocks,Employing Site Analytics,Tracking User Behavior with Web Analytics • AI Perspective:			

- Predictive SEO traffic modeling using time-series forecasting
- AI content audit tools (e.g., SurferSEO, MarketMuse) for optimization gaps
- LLM-powered log-file clustering

Module 4	On-page SEO vs Off-page SEO	Assessment : Case Study/Presentation	11 Sessions

Topics:

On-page SEO: Website Content, URL Structure, Pictures, Title Tags, Meta Tags, Headline Tags, Internal Linking, **Off-page SEO:** Link Building, Social Media for Content Sharing, Email Campaigns

- AI Perspective:
  - NLP-based keyword clustering and LSI keyword suggestions
  - AI-based link quality scoring and spam detection in backlinks
  - Entity SEO with knowledge graph embeddings (fine tune hugging face or openAI models)

Targeted Applications & Tools:

Applications	• Website optimization for improved search ranking • Content strategy development for digital visibility
Tools/Software	• Google Analytics • SEMrush / Ahrefs / Moz • WordPress SEO (Yoast) • Keyword research tools

Skill-Building Activities through Experiential Learning:

- Keyword research and competitor analysis.
- Webpage content optimization with metadata and tags.
- Analytics-based performance tracking (traffic & conversions).
- Case study analysis of successful SEO campaigns.
- Developing a complete SEO strategy for a given research article or webpage.

Text Book

T1: Bruce Clay, Susan Esparza, Search Engine Optimization All-in-One For Dummies, John Wiley Distributor, 2nd Edition, 2012.

T2: HubSpot, Introduction to SEO: Getting Started With SEO to Achieve Business Goals (eBook). https://www.hubspot.com/hs-fs/hub/53/file-13221845-pdf/docs/ebooks/introduction-to-seo-ebook.pdf

References

R1: Eric Enge, Stephan M. Spencer, Jessie Stricchiola, *The Art of SEO: Mastering Search Engine Optimization*, O'Reilly Media, 2015.

R2: David Amerland, *Google Semantic Search*, Pearson Education India, 2014.

E-Resources/Web Resources

1. Search Engine Journal https://www.searchenginejournal.com
2. International Journal of Technology Marketing

Course Code: IST3403	Course Name: Service Oriented Architecture Type of Course: Theory Course	L-T-P- C	3-0-0-3
Version No.	2.0		
NCrF Level-	5.5		

Course Pre-requisites	CSE2258
Co-requisites	NIL
Anti-requisites	NIL
Course Description	This course covers architectural paradigms for designing scalable, loosely coupled, and reusable software services. It examines the foundation and standards of Service Oriented Architecture (SOA), including technologies such as XML, SOAP, WSDL, UDDI, REST, and WS-* protocols. The course also explores service composition, orchestration, and integration in enterprise platforms like J2EE. Emphasis is placed on analyzing real-world system interactions and developing interoperable, secure, and distributed service-based applications.
Course Objective	This course is designed to improve the learners' EMPLOYABILITY SKILLS by using EXPERIENTIAL LEARNING techniques.
Course Outcomes	**On successful completion of the course the students shall be able to:** **CO1: Describe the fundamentals of XML and data structuring. [Understand]** **CO2: Demonstrate XML-based applications for structured data exchange. [Apply]** **CO3: Explain SOA principles and compare them with other architectures. [Understand]** **CO4: Apply service description and discovery techniques using SOAP, WSDL, and UDDI. [Apply]** **CO5: Implement secure SOA-based applications using WS-standards and J2EE API's [Apply]**

Course Content:

Module 1	**XML and Structured Data Interchang**	Assessment : Assignment – XML Validation and Schema Modeling	10 Sessions
XML document modelling, Namespaces, DTD, XML Schema (XSD), XML validation workflows, XPath and XQuery for structured querying, XML-based APIs in programming environments			
Module 2	**Building XML-Based Applications**	Assessment : Assignment – Real-Time Data Parsing and Formatting	09 Sessions
DOM and SAX parsing, XSLT transformations and templating, XML storage in relational systems, REST-style data interchange using XML/JSON, Case: Generating data reports from structured XML sources			
Module 3	**Introduction to SOA and Microservices Alignment**	Assessment : Assignment – Service Layer Analysis	08 Sessions
SOA architecture and its evolution, Characteristics and layers of SOA, Service contracts and service granularity, Microservices vs SOA comparison, Statelessness, service autonomy, and scalability			
Module 4	**Web Services and WS-Specifications**	Assessment : Assignment – Secure Web Service Deployment	08 Sessions
Service discovery: UDDI registry, Messaging with SOAP and WS-Addressing, Service description with WSDL, WS-* protocols: WS-Security, WS-Reliable Messaging, Integration using Java EE, Spring WS			
Module 5	**Service Composition and Enterprise Integration**	Assessment : Assignment – Business Process Composition	08 Sessions
Service orchestration with BPEL, Choreography concepts, Service lifecycle and versioning, Policy specification with WS-Policy, SOA in enterprise middleware (e.g., MuleSoft, WSO2, Oracle SOA Suite)			

Targeted Applications & Tools:

Tools/Software/Platform	Application
Postman / SOAP UI	API and web service testing, SOAP and REST protocol inspection

Oxygen XML / Altova XMLSpy	XML schema design, validation, and transformation workflows
Apache Axis2 / CXF	Web service deployment and WS-* compliance
J2EE / Spring Boot	Service component development and containerized deployment
WSO2 / MuleSoft / BPEL Designer	SOA orchestration and choreography modeling

Skill-Building Activities through Experiential Learning:

- XML Data Contracts: Learners will define enterprise-grade XML schemas and apply transformations using XSLT and XQuery for reporting and presentation.
- Web Service Prototyping: Students will publish secure SOAP/REST services using Java and test them with live clients.
- SOA Layered Modeling: Based on business scenarios, students will define service layers and draft contracts using UML and BPMN.
- Service Orchestration: Develop BPEL workflows using middleware platforms, focusing on orchestration logic and exception handling.
- Security and Policy Compliance: Implement authentication tokens, message encryption, and WS-Policy for service interactions.

Text Book

- Schmelzer, R., et al. (2002). XML and Web Services. Pearson Education.
- Erl, T. (2005). Service-Oriented Architecture: Concepts, Technology, and Design. Pearson Education.
- Josuttis, N. (2007). SOA in Practice: The Art of Distributed System Design. O'Reilly Media.

References

- Newcomer, E., & Lomow, G. (2005). Understanding SOA with Web Services. Addison-Wesley.
- Chatterjee, S., & Webber, J. (2004). Developing Enterprise Web Services: An Architect's Guide. Prentice Hall.
- Papazoglou, M. (2008). Web Services: Principles and Technology. Pearson.
- Krafzig, D., Banke, K., & Slama, D. (2005). Enterprise SOA: Service-Oriented Architecture Best Practices. Prentice Hall.

E-Resources/Web Resources

- https://nptel.ac.in/courses/soa
- https://www.udemy.com/topic/soa/Engineering Mechanics, R K Bansal, Sanjay Bansal, Lakshmi Publications, 2016
- https://in.coursera.org/learn/service-oriented-architecture
- https://sm-nitk.vlabs.ac.in/

Course Code: IST3404	Course Name: Information System Audit Type of Course: Theory Course	L- T-P-C	3-0-0-3
Version No.	2.0		
NCrF Level-	5.5		
Course Pre-requisites	CCS2503		
Co-requisites	NIL		

Anti-requisites	NIL		
Course Description	This course provides a comprehensive understanding of auditing Information Systems, emphasizing risk identification, control assessment, and security incident evaluation. It covers IS audit processes, management controls, and risk management aligned with industry standards. Contemporary topics such as ERP audits, cloud security, digital forensics, and emerging risks are integrated to build future-ready IS audit capabilities.		
Course Objective	The objective of the course is the Skill development of students by using Participative Learning techniques		
Course Outcomes	**On successful completion of the course the students shall be able to:** **CO1: Explain fundamental concepts of information systems and their role in organizational decision-making. [Understand]** **CO2: Understand IS audit processes, including management, operational, and environmental controls, across the system development lifecycle. [Understand]** **CO3: Demonstrate awareness of cyber forensic tools, emerging audit challenges, and regulatory compliance in modern IS audit practices. [Apply]** **CO4: Apply IS audit procedures to ERP systems, cloud environments, and network infrastructures to identify risks and vulnerabilities. [Apply]** **CO5: Apply risk management strategies, including change management, disaster recovery, and business continuity planning, in IS environments [Apply]**		
Course Content:			
Module 1	Introduction to Information Systems	Assessment : Quiz / Conceptual Case Study etc	11 Sessions
Topics: Information and Data: Definition and Distinctions, Meaning of Information and its characteristics, Difference between Data and Information, Features and Qualities of Information, Types of Information, Process of Generating Information, Value and Cost of Information, Information as a Corporate Resource, Information needs of top or Strategic Management, Information needs of Middle-Level Management/Tactical Management, Information needs of Low Level or Operational Management, Factors affecting Information Needs			
Module 2	Management Information Systems (MIS) and Enterprise Resource Management (ERM)	Assessment : Quiz / Conceptual Case Study etc	12 Sessions
Topics: MIS - Concepts, Requirements, Comprehensive Structure of MIS, Conceptual Structure of MIS, Issues of MIS Structure, Characteristics of MIS, Myths About MIS, Top-down approach to MIS Development, Bottom-up approach to MIS Development. Enterprise Resource Planning(ERP)- Objectives, Genesis of ERP, Modules of ERP, ERP basic features, General Features of ERP, Characteristics of ERP, Critical Success Factors for ERP Implementation, Benefits of ERP, Limitations of ERP, Criterion for choosing the ERP system/ERP vendor, Popular ERP service providers, SAP ERP Key Strengths, Oracle ERP Highlights, Microsoft ERP Highlights. Assignment: Enterprise Resource Management (ERM) and Real-time applications of ERP.			
Module 3	System Audit	Assessment : Quiz / Conceptual Case Study etc	11 Sessions
Topics: What is a System Audit? Information System Audit Plan, Foundations of Information Systems Auditing, Foundation of Information Systems Auditing, Nature of System Audit, Significance of System Audit, Scope of Information System Audit, Information System Audit Process, Audit of Management Controls, Audit of Operational Controls, Audit of Environmental Controls, Systems Audit & Management Functions, Effects of Computers on Auditing Impact of Computers on Evidence Collection, Impact of Computers on Evidence Evaluation.			

Assignment: E-Commerce Security and Audit.

Module 4	Risk Assessment in Audit planning	Assessment : Quiz / Conceptual Case Study etc	11 Sessions

Topics: Systems Audit of Computerized Secretarial Functions, System Audit of Computerised Secretarial Function, Verification of Input Controls, Use of Test Pack, Illustrative Test Pack Norms and Procedures For Computerization, Change Management, Documentation Standards, Technology Renewal, Computers, Controls & Security, Fraud, error, vandalism, excessive costs, competitive disadvantage, business, interruption, social costs, statutory sanctions, Risk Assessment and Risk Management, Preventive/detective/corrective strategies.

Assignment: Disaster Recovery Planning (DRP)

Targeted Applications & Tools:

Tools/Software/Platform	Application
MS Excel, MS Word, Google Docs	Documentation of audit reports, risk assessment charts, system audit checklists
SAP ERP, Oracle ERP, Microsoft Dynamics	ERP system audit exposure, analysis of real-time ERP data and system vulnerabilities
Nessus, OpenVAS, Nmap	Network vulnerability scanning, risk identification during audit
Wireshark	Packet analysis to detect anomalies and potential system compromise during audits
Kali Linux (Optional)	Exposure to penetration testing tools for understanding system weaknesses (demo-only)
AWS, Azure (Free Tier)	Basic audit exposure to cloud infrastructure, user access reviews, and resource monitoring
ISACA Audit Templates & Standards	Industry-accepted audit checklists, templates, and frameworks for conducting structured IS audits
Google Workspace/ Microsoft 365 Security Tools	Familiarity with auditing collaboration platforms for compliance and security
Cyber Forensic Tools (Autopsy, FTK Imager)	Basic exposure to digital evidence acquisition and forensic analysis

Skill-Building Activities through Experiential Learning:

- **ERP System Security Gap Identification (50 words)**
  Explore a real-time ERP demo environment (SAP, Oracle, or simulated ERP). Identify potential access control issues, role-based permission gaps, and common vulnerabilities. Document findings using an audit checklist template. This activity builds practical skills in identifying risks in enterprise resource management platforms during an audit.
- N**etwork Vulnerability Scanning Simulation (50 words)**
  Use tools like Nessus or OpenVAS in a controlled lab setup to simulate scanning a network for common vulnerabilities. Interpret the scan report, classify risks, and suggest mitigation strategies. This fosters hands-on understanding of vulnerability detection and risk assessment crucial for modern IS audit practices.
- **Cloud System Access Review and Risk Assessment (50 words)**
  Leverage a basic cloud account (AWS/Azure Free Tier) to explore user access configurations. Review IAM roles, permissions, and security settings. Identify potential misconfigurations and recommend corrective measures. This enhances skills in assessing cloud security, a critical component of contemporary information system audits.

Text Book

- Richard E. Cascarino: Auditor's Guide to Information Systems Auditing,2020.
- Mohit Agarwal : Information Technology And System Audit, Agarwal Publications,2020.
- Gerardus Blokdyk: Information Systems Audit A Complete Guide,Emereo Publishing,2020.
- Kim, D., Solomon, M. G. (2021). Fundamentals of Information Systems Security. (n.p.): Jones & Bartlett Learning.

References

- Abhishek Mittal : Information Technology & Systems Audit, Pooja Law Publishing,2017.
- Peret, P. (2022). Information System Audit: How to Control the Digital Disruption. United States: CRC Press LLC.

E-Resources/Web Resources

- **https://www.coursera.org/learn/information-systems-audit**
- **Information System Audit (DISA 3.0) E-Learning (icai.org)**
- **2_2_Full_Book_Information_Security_and (1).pdf (sharepoint.com)**

Course Code: **IST3407**	**Course Title: Infrastructure Management** **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	This comprehensive course provides an in-depth exploration of modern **Information Technology (IT) systems**, covering essential concepts in **IT infrastructure, service management, security, and ethical practices**. Students will gain both theoretical knowledge and practical insights into managing IT resources effectively while addressing real-world challenges in technology deployment and maintenance.				
Course Objectives	This course is designed to improve the learners 'EMPLOYABILITY SKILLS' by using EXPERIENTIAL LEARNING techniques.				
Course Outcomes	On successful completion of the course the students shall be able to: **CO 1:Recall** fundamental concepts of IT infrastructure, including hardware, software, networks, and computing resources. **List** key components of IT service management processes such as incident management, problem management, and change management. **CO 2: Explain** the role of IT infrastructure in business operations, including design issues and system management. **Describe** security management principles, including access control, identity management, and intrusion detection. **CO 3:Implement** basic IT service support processes such as configuration management and incident resolution in a simulated environment. **Use** disaster recovery and backup strategies to ensure data retention and system availability. **CO 4:Compare** different IT service delivery models, including service level management and financial management. **Evaluate** ethical and legal issues in IT, such as cybercrimes, intellectual property rights, and privacy laws.				

Course Content:				
Module 1	**Introduction**	Assignment	Quiz	**L – 15-Sessions**
INTRODUCTION–Information Technology, Computer Hardware, Computer Software, Network and Internet, Computing Resources, IT INFRASTRUCTURE–Design Issues, Requirements, IT System Management Process, Service Management Process, Information System Design, IT Infrastructure Library. **SERVICE DELIVERY PROCESS**–Service Delivery Process, Service Level Management, Financial Management, Service Management, Capacity Management, Availability Management.				
Module 2	**Service Support Process**	Assignment	Project	**L-12-Sessions**
SERVICE SUPPORT PROCESS–Service Support Process, Configuration Management, Incident Management, Problem Management, Change Management, Release Management STORAGE MANAGEMENT–Backup & Storage, Archive & Retrieve, Disaster Recovery, Space Management, Database & Application Protection, Bare Machine Recovery, Data Retention.				
Module 3	**Security Management**	Assignment	Project	**L-9-Sessions**
SECURITY MANAGEMENT–Security, Computer and Internet Security, Physical Security, Identity Management, Access Management. Intrusion Detection, Security Information Management.				
Module 4	**IT Ethics**	Assignment	Project	**L-9-Sessions**
IT ETHICS–Introduction to Cyber Ethics, Intellectual Property, Privacy and Law, Computer Forensics, Ethics and Internet, Cyber Crimes EMERGING TRENDS in IT–Electronics Commerce, Electronic Data Interchange, Mobile Communication Development, Smart Card, Expert Systems.				
Targeted Application & Tools that can be used : 1. Execution of an IT Infrastructure Management will be done using "Colab", available at https://colab.research.google.com/ or Jupyter Notebook. 2. Laboratory tasks will be implemented using the necessary libraries available in Python				
Project work/Assignment: Mention the Type of Project /Assignment proposed for this course				
Students can be given group assignments to develop and implement an IT Infrastructure Management				
Text Book 1. IT Infrastructure & Management, Authors: Surendra Keshari, Narendra Kumar, DreamTech Press, Distributed by WILEY, 2020 Edition.				
References 1. Zero To Mastery In IT Infrastructure And It's Management- No.1 Book To Become Zero To Hero In				

Infrastructure Management, This Amazing Book Covers A-Z IT Infrastructure t Concepts, 2024 Latest Edition (Paperback, Dr. R.K. Jain)

Course Code: **CSE3450**	**Course Title: Introduction to NVIDIA Accelerated Computing and AI** **Type of Course: Lab Integrated Course**	**L - T - P - C**	**2**	**0**	**2**	**3**
Version No.	**1.0**					
Course Pre-requisites	**NIL**					
Anti-requisites	**NIL**					
Course Description	This course introduces the fundamentals of Artificial Intelligence and NVIDIA accelerated computing, focusing on GPU architecture, parallel processing, and the NVIDIA software ecosystem. It provides hands-on experience in GPU-accelerated data science, deep learning, and Generative AI using Python and NVIDIA tools, with attention to ethical and real-world applications.					
Course Objective	• To introduce students to the fundamentals of Artificial Intelligence and accelerated computing • To familiarize students with GPU architecture and NVIDIA ecosystem • To provide hands-on exposure to GPU-accelerated data science and deep learning • To develop problem-solving skills using NVIDIA tools and ethical AI practices					
Course Outcomes	**On successful completion of this course the students shall be able to:** CO1. Explain the fundamentals of Artificial Intelligence and NVIDIA accelerated computing concepts. (Understand) CO2. Apply Python programming and GPU-accelerated libraries for data processing and analysis tasks. (Apply) CO3. Implement basic deep learning models using GPU-based frameworks for improved computational performance. (Apply) CO4. Analyze the performance benefits of GPU acceleration in AI and data science workflows. (Analyze) CO5. Design simple AI and Generative AI solutions while considering ethical, societal, and responsible AI aspects. (Create)					
Course Content:						
Module 1	**Foundations of AI and NVIDIA Ecosystem**	Assignment			**9 Sessions**	

	Introduction to Artificial Intelligence: AI, Machine Learning, and Deep Learning Applications of AI in healthcare, finance, cybersecurity, smart cities, Introduction to GPUs and accelerated computing, CPU vs GPU architecture and parallel processing, Overview of NVIDIA ecosystem: GPUs, CUDA, Tensor Cores, Introduction to NVIDIA Deep Learning Institute (DLI) Hands-on: Exploring GPU environment and executing basic GPU programs.				
Module 2	**Python and Data Science with GPU Acceleration**	Assignment			**9 Sessions**
	Python programming fundamentals for AI, Python libraries: NumPy, Pandas, Matplotlib Introduction to Data Science workflow, Data preprocessing and exploratory data analysis, Introduction to RAPIDS for GPU-accelerated data processing, Performance comparison: CPU vs GPU computing, Hands-on: Data analysis using GPU-accelerated libraries.				
Module 3	**Fundamentals of Deep Learning using NVIDIA Tools**	Assignment			**9 Sessions**
	Basics of neural networks and deep learning, Activation functions, loss functions, and optimization techniques, Introduction to deep learning frameworks: PyTorch and TensorFlow, Role of CUDA and cuDNN in deep learning, training deep learning models on GPUs, Model evaluation and performance metrics, Hands-on: Training a basic neural network on GPU				
Module 4	**Generative AI and Real-World Applications**	Assignment			**9 Sessions**
	Introduction to Generative AI and foundation models, Basics of Large Language Models (LLMs), Image and text generation concepts, Introduction to NVIDIA NeMo, Retrieval Augmented Generation (RAG) concepts, Ethical, legal, and social issues in AI, Mini Project / Case Study: Design and demonstrate a basic AI or Generative AI application				
	Targeted Application & Tools that can be used: Tools: Python, TensorFlow, PyTorch, CUDA Toolkit, cuDNN, NVIDIA RAPIDS, TensorRT, Jupyter Notebook / Google Colab, NVIDIA NGC (GPU-optimized containers) Applications such as Image Classification and Object Detection Natural Language Processing (Text Classification, Chatbots) Generative AI (Text/Image Generation) Recommendation Systems Healthcare Analytics (medical imaging, diagnosis support)				
	List of Laboratory tasks: 1. Setting up the NVIDIA GPU environment and executing basic CUDA programs to verify the				

installation using nvidia-smi and a "Hello World" script.

2. Performance comparison of matrix multiplication between CPU (NumPy) and GPU (CuPy) to analyze computational speedup.
3. Exploring the NVIDIA Deep Learning Institute (DLI) ecosystem by executing hands-on tasks in cloud-based GPU notebooks.
4. Performing data preprocessing and Exploratory Data Analysis (EDA) on large-scale datasets using the NVIDIA RAPIDS cuDF library.
5. Implementing GPU-accelerated Linear and Logistic Regression models using RAPIDS cuML for efficient predictive modeling.
6. Implementing K-Means Clustering using cuML to perform customer segmentation on high-dimensional data.
7. Creating interactive data visualizations for large-scale datasets using GPU-accelerated libraries like cuXfilter.
8. Building and training a Multi-Layer Perceptron (MLP) on a GPU to classify handwritten digits from the MNIST dataset.
9. Conducting a comparative study of different Activation Functions and Optimizers (SGD vs. Adam) to evaluate their impact on model convergence.
10. Implementing an Image Classification system using Convolutional Neural Networks (CNNs) accelerated by PyTorch or TensorFlow on a GPU.
11. Fine-tuning a pre-trained deep learning model using Transfer Learning techniques for a custom image dataset.
12. Developing a real-time Object Detection system using pre-trained YOLO models optimized with cuDNN acceleration.
13. Building a Retrieval Augmented Generation (RAG) pipeline to enable context-aware Question and Answering (Q&A) from local documents.
14. Implementing a basic Generative AI model for automated text generation using the NVIDIA NeMo framework.
15. Conducting an AI Ethics Audit to analyze and document bias and security vulnerabilities in a deployed AI model.

Text Book(s):

T1. Ian Goodfellow, Yoshua Bengio, Aaron Courville, *Deep Learning*, MIT Press

T2. Aurélien Géron, *Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow*, O'Reilly

Reference(s):

R1. NVIDIA Deep Learning Institute (DLI)

R2. NVIDIA Developer Documentation

R3. PyTorch and TensorFlow official documentation

<table>
<tr><td>Course Code:

CSE3451</td><td colspan="2">Course Title: NVIDIA Applied Artificial Intelligence and Generative AI

Type of Course: Lab Integrated Course</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">CSE1500</td></tr>
<tr><td>Anti-requisites</td><td colspan="7">NIL</td></tr>
<tr><td>Course Description</td><td colspan="7">This course focuses on applied Artificial Intelligence and Generative AI using NVIDIA GPU platforms, enabling students to design, optimize, and deploy AI systems with accelerated computing. It emphasizes hands-on development of GPU-accelerated machine learning, deep learning, and LLM-based applications, while addressing responsible, scalable, and sustainable AI practices.</td></tr>
<tr><td>Course Objective</td><td colspan="7">• To strengthen applied AI skills using NVIDIA platforms
• To build efficient AI pipelines using GPU acceleration
• To understand Generative AI, LLMs, and deployment workflows
• To introduce responsible AI and real-world AI system design</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">On successful completion of this course the students shall be able to:
CO1. Analyze different AI workloads and select appropriate NVIDIA GPU-based solutions. (Analyze)
CO2. Build and optimize machine learning pipelines using GPU acceleration techniques. (Apply)
CO3. Train and fine-tune deep learning and Generative AI models for various applications. (Apply)
CO4. Evaluate the performance and scalability of AI models deployed on GPU-based systems. (Evaluate)
CO5. Design and deploy AI applications while considering ethical, scalability, and sustainability aspects. (Create)</td></tr>
<tr><td>Course Content:</td><td colspan="7"></td></tr>
<tr><td>Module 1</td><td colspan="2">NVIDIA AI Platforms and Workflows</td><td>Assignment</td><td colspan="2"></td><td colspan="2">9 Sessions</td></tr>
<tr><td colspan="8">Review of AI system architecture, AI workloads: training vs inference, NVIDIA AI software stack overview: CUDA (conceptual), cuDNN, TensorRT (overview), NVIDIA AI Enterprise ecosystem, Cloud vs On-prem GPU AI workflows, Hands-on: Exploring NVIDIA GPU environment & AI workflow</td></tr>
</table>

Module 2	**GPU-Accelerated Machine Learning & Data Pipelines**	Assignment		**9 Sessions**

End-to-end ML pipeline design, Data preprocessing and feature engineering on GPU, RAPIDS ecosystem: cuDF for dataframes, cuML for ML algorithms, Model evaluation and optimization, Performance comparison: CPU vs GPU ML pipelines, **Hands-on:** Build and optimize an ML pipeline using RAPIDS

Module 3	**Deep Learning and Model Optimization**	Assignment		**9 Sessions**

Review of deep learning architectures (CNNs, RNNs, Transformers), Training deep learning models on GPU, Transfer learning and fine-tuning, Mixed precision training and performance tuning, Model evaluation, overfitting, and generalization, **Hands-on:** Fine-tune a deep learning model on GPU

Module 4	**Generative AI, Deployment, and Responsible AI**	Assignment		**9 Sessions**

Fundamentals of Generative AI, Large Language Models (LLMs): concepts and use cases, NVIDIA NeMo framework overview, Retrieval Augmented Generation (RAG) pipeline, Model deployment basics: Inference optimization, NVIDIA Triton (conceptual), Responsible AI: Bias, fairness, privacy, Sustainability and green AI, **Mini Project:** Design and deploy a simple AI / GenAI application

Targeted Application & Tools that can be used:

Tools: Python, TensorFlow, PyTorch, CUDA Toolkit, cuDNN, NVIDIA RAPIDS, TensorRT, Jupyter Notebook / Google Colab, NVIDIA NGC (GPU-optimized containers)

Applications such as

Image Classification and Object Detection

Natural Language Processing (Text Classification, Chatbots)

Generative AI (Text/Image Generation)

Recommendation Systems

Healthcare Analytics (medical imaging, diagnosis support)

List of laboratory tasks:

1. Analyze and profile the hardware specifications of an NVIDIA GPU using nvidia-smi and NVML to document the correlation between core count, memory bandwidth, and compute capacity.
2. Design and execute a workflow to measure and compare the computational latency and throughput of a neural network during its training and inference phases on a GPU.
3. Implement an end-to-end data preprocessing pipeline using NVIDIA RAPIDS cuDF to perform complex operations on a large-scale dataset and benchmark its performance against CPU-based Pandas.
4. Develop a machine learning classification model using cuML and evaluate the reduction in search

time achieved through GPU-accelerated hyperparameter tuning.

5. Construct a high-performance, GPU-resident ETL pipeline designed to minimize data transfer overhead between host and device memory for real-time analytics.
6. Implement transfer learning by fine-tuning a pre-trained Vision Transformer (ViT) or ResNet model from the NVIDIA NGC catalog for a domain-specific image classification task.
7. Configure and execute Automatic Mixed Precision (AMP) training to evaluate the trade-offs between FP32 and half-precision formats in terms of training speed and numerical stability.
8. Conduct a comparative study of deep learning optimization by implementing GPU-accelerated regularization techniques and monitoring convergence behaviors via TensorBoard.
9. Perform model pruning and quantization on a deep learning architecture to analyze the impact of model size reduction on inference speed and prediction accuracy.
10. Deploy a Large Language Model using the NVIDIA NeMo framework to evaluate zero-shot and few-shot learning capabilities through systematic prompt engineering.
11. Design and implement a Retrieval Augmented Generation (RAG) pipeline integrating a vector database with an LLM to facilitate context-aware document-based question answering.
12. Optimize a generative AI model for production by converting it into a TensorRT engine and measuring the resulting increase in inference throughput and queries per second.
13. Simulate a multi-model deployment environment using the NVIDIA Triton Inference Server framework to manage and serve concurrent AI workloads efficiently.
14. Conduct a comprehensive responsible AI audit on a generative model to detect and document algorithmic bias related to protected attributes like gender, race, or profession.
15. Design and deploy a complete AI or Generative AI application that solves a real-world problem while adhering to "Green AI" sustainability principles and ethical safeguards.

Text Book(s):

T1. Ian Goodfellow, Yoshua Bengio, Aaron Courville, *Deep Learning*, MIT Press – Core reference for deep learning concepts and architectures

T2. Aurélien Géron, *Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow*, O'Reilly – Practical ML & DL workflows with model tuning and evaluation

T3. Kirk & Hwu, *Programming Massively Parallel Processors*, Morgan Kaufmann – Conceptual understanding of GPU and parallel computing

T4. Francois Chollet, *Deep Learning with Python*, Manning Publications – Model design, transfer learning, and optimization

T5. Chip Huyen, *Designing Machine Learning Systems*, O'Reilly – Industry-oriented ML system design and deployment concepts

Reference(s):

R1. NVIDIA Deep Learning Institute (DLI)

- GPU-Accelerated ML & DL
- Generative AI Explained
- Building RAG Applications
- Model Optimization & Deployment

R2. NVIDIA Developer Documentation

- NVIDIA AI software stack
- TensorRT, Triton Inference Server (overview)
- RAPIDS documentation

Course Code: **CSE3452**	**Course Title: Advanced AI Systems and Generative Intelligence using NVIDIA Platforms** **Type of Course: Lab Integrated Course**	**L-T-P-C**	**2**	**0**	**2**	**3**
Version No.	**1.0**					
Course Pre-requisites	CSE1500					
Anti-requisites	**NIL**					
Course Description	This advanced course explores the design, optimization, and deployment of large-scale AI and Generative Intelligence systems using NVIDIA platforms and software stacks. It combines theory, hands-on labs, and a capstone project to develop high-performance, ethical, and sustainable AI solutions for real-world, production-grade environments.					
Course Objective	• To design high-performance AI systems using NVIDIA platforms • To optimize deep learning and Generative AI models for scale and deployment • To build end-to-end AI pipelines including training, inference, and monitoring • To evaluate AI systems with respect to ethics, sustainability, and real-world constraints					
Course Outcomes	**On successful completion of this course the students shall be able to:** CO1. Architect and evaluate large-scale AI systems using the NVIDIA AI software stack. (Analyze) CO2. Optimize deep learning and large language models (LLMs) for performance, memory efficiency, and inference speed. (Apply) CO3. Design and deploy Generative AI systems using advanced NVIDIA frameworks and tools. (Create)					

	CO4. Analyze system-level challenges in scaling AI applications, including resource management and distributed computing. (Analyze) CO5. Evaluate ethical, sustainable, and responsible AI practices in real-world deployments. (Evaluate)			
Course Content:				
Module 1	**Advanced NVIDIA AI Architecture & Systems**	Assignment		**9 Sessions**
NVIDIA AI system architecture (training, inference, orchestration), Multi-GPU and distributed AI concepts, NVIDIA AI software stack deep dive: CUDA (advanced concepts – streams, concurrency), cuDNN, NCCL, TensorRT, AI workloads on Cloud vs On-Prem vs Edge, **Hands-on:** Profiling AI workloads using NVIDIA tools.				
Module 2	**Model Optimization and High-Performance AI**	Assignment		**9 Sessions**
Performance bottlenecks in deep learning systems, Mixed precision & quantization techniques, Model pruning, distillation, and compression, Efficient data pipelines for large-scale training, Benchmarking and optimization strategies, **Hands-on:** Optimizing DL models using TensorRT / AMP				
Module 3	**Advanced Generative AI & LLM Systems**	Assignment		**9 Sessions**
Foundation models and LLM architectures, Fine-tuning strategies (LoRA, PEFT, adapters), NVIDIA NeMo deep dive, Retrieval Augmented Generation (RAG) at scale, Multimodal Generative AI (text, image, video), **Hands-on:** Fine-tuning and serving LLMs on GPU.				
Module 4	**AI Deployment, MLOps, and Responsible AI**	Assignment		**9 Sessions**
End-to-end AI deployment pipeline, NVIDIA Triton Inference Server, Model serving, monitoring, and versioning, AI security, privacy, and adversarial threats, Green AI and sustainable computing, Ethical and legal considerations in AI systems, **Capstone Project:** Design and deploy an advanced AI / GenAI system.				
Targeted Application & Tools that can be used: Tools: Python, TensorFlow, PyTorch, CUDA Toolkit, cuDNN, NVIDIA RAPIDS, TensorRT, Jupyter Notebook / Google Colab, NVIDIA NGC (GPU-optimized containers) Applications such as				

Image Classification and Object Detection

Natural Language Processing (Text Classification, Chatbots)

Generative AI (Text/Image Generation)

Recommendation Systems

Healthcare Analytics (medical imaging, diagnosis support)

List of laboratory tasks:

1. Profile the execution of a ResNet-50 training script using NVIDIA Nsight Systems to identify hardware bottlenecks and document GPU utilization across various batch sizes.
2. Implement a script using CUDA Streams to overlap Host-to-Device data transfers with kernel computations and evaluate the total execution time against a sequential baseline.
3. Establish a multi-GPU training environment using the NCCL backend to measure scaling efficiency and "Time-per-Epoch" metrics across multiple GPU configurations.
4. Deploy a pre-trained model on both high-performance Cloud GPUs and NVIDIA Jetson edge devices to compare latency, power consumption, and memory footprints.
5. Implement Automatic Mixed Precision (AMP) in a training pipeline to analyze its impact on training speed and VRAM usage while maintaining model accuracy.
6. Convert a FP32 PyTorch model into an optimized INT8 TensorRT engine using post-training calibration to benchmark inference throughput improvements.
7. Design a Knowledge Distillation pipeline where a heavy teacher model trains a lightweight student model and evaluate performance retention relative to parameter reduction.
8. Build a high-performance input pipeline using the NVIDIA DALI library and compare CPU utilization and GPU starvation rates against standard CPU-based data loaders.
9. Apply LoRA (Low-Rank Adaptation) fine-tuning to a foundation Large Language Model using the NVIDIA NeMo framework for a domain-specific task.
10. Construct a scalable Retrieval Augmented Generation (RAG) pipeline using a vector database to implement a retrieval-to-generation workflow for large document corpora.
11. Develop an end-to-end application integrating a Vision-Language Model with a text-to-image generator, optimized for efficient execution on a single 24GB VRAM GPU.
12. Compile an LLM using the TensorRT-LLM library and implement in-flight batching and KV-caching to maximize token-per-second output during high-concurrency testing.
13. Deploy a model ensemble on an NVIDIA Triton Inference Server and configure dynamic batching to manage over 100 concurrent inference requests.
14. Perform a "Red Teaming" exercise on a deployed LLM using adversarial prompts and implement NVIDIA NeMo Guardrails to mitigate identified safety vulnerabilities.
15. Design a sustainability dashboard for an AI project using GPU power-monitoring tools to calculate carbon footprints and propose energy reduction strategies.

Text Book(s):

T1. **Ian Goodfellow, Yoshua Bengio, Aaron Courville**, *Deep Learning*, MIT Press – Advanced deep learning theory and architectures

T2. Chip Huyen, *Designing Machine Learning Systems*, O'Reilly – End-to-end ML/AI system design,

deployment, and monitoring

T3. Kirk & Hwu, *Programming Massively Parallel Processors*, Morgan Kaufmann – GPU computing and parallel processing fundamentals

T4. Francois Chollet, *Deep Learning with Python*, Manning – Advanced model design, optimization, and fine-tuning

T5. Boris Ginsburg et al., *Deep Learning Performance Optimization*, NVIDIA Whitepapers – Performance tuning and optimization strategies

Reference(s):

R1. NVIDIA Deep Learning Institute (DLI) – Advanced

- Advanced GPU-Accelerated Deep Learning
- Building and Optimizing LLMs with NVIDIA NeMo
- Efficient AI Inference with TensorRT
- Deploying AI Models with NVIDIA Triton Inference Server

R2. NVIDIA Developer Resources

- NVIDIA AI Software Stack Documentation
- CUDA Advanced Programming Guide
- TensorRT and Triton official documentation
- NVIDIA NeMo framework documentation

3-0-0-3	**L-T-P-C**	**Course Title: Business Intelligence and Analytics** **Type of Course:1] Theory**	**Course Code:** IST3412
1.0			**Version No.**
CSE1002: Programming using Python **CSE2012: Database Management Systems**			**Course Pre-requisites**
NIL			**Anti-requisites**
The purpose of the course is to instill a strong foundation of scientific process orientation that is the cornerstone of effective. Business Intelligence (BI) is a set of architectures, theories, methodologies and technologies that transform structured, semi-structured and unstructured data into meaningful and useful information. Students will analyze enterprise data requirements to develop queries, reports and build OLAP cubes that use business analytics to answer complex business questions.			**Course Description**
This course is designed to improve the learners' EMPLOYABILITY SKILLS by using PROBLEM SOLVING Methodologies.			**Course Objective**
On successful completion of this course the students shall be able to: 1.Understand the basic concepts and applications of Business Intelligence and Analytics.[Understand] 2. Apply data warehousing and ETL techniques for managing business data.[Apply]. 3.Analyze business data using analytical and data mining techniques.[Analyze]. 4. Create dashboards and reports using Business Intelligence tools.[Apply] 5. Analyze business problems and propose data-driven solutions using analytics.[Analyze]			**Course Out Comes**
			Course Content:

9 Hours		Assignment	Unit-1 - Introduction: Business Intelligence, Analytics and Decision Support	**Module 1**
Information Systems Support for Decision Making - An Early Framework for Computerized Decision Support - The Concept of Decision Support Systems - A Framework for Business Intelligence - Business Analytics Overview - Brief Introduction to Big Data Analytics - Clickstream Analysis – Metrics - Clickstream Analysis - Practical Solutions - Competitive Intelligence Analysis				
9 Hours		Assignment	Decision Making	**Module 2**
Decision Making - Introduction and Definitions - Phases of the Decision - Making Process - The Intelligence Phase - Design Phase - Choice Phase - Implementation Phase - Decision Support Systems Capabilities - Decision Support Systems Classification - Decision Support Systems Components				
9 Hours		Assignment	Predictive Modeling and Sentiment Analysis	**Module 3**
Basic Concepts of Neural Networks - Developing Neural Network - -Based Systems - Illuminating the Black Box of ANN with Sensitivity - Support Vector Machines - A Process Based Approach to the Use of SVM - Nearest Neighbor Method for Prediction -Sentiment Analysis Overview - Sentiment Analysis Applications - Sentiment Analysis Process - Sentiment Analysis - Speech Analytics				
9 Hours		Assignment	Multi-Criteria Decision-Making Systems	**Module 4**
Decision Support Systems modeling - Structure of mathematical models for decision support - Decision making under certainty - Uncertainty and Risk - Decision modeling with spreadsheets - Mathematical programming optimization - Decision analysis introduction - Decision tables -				

Decision Trees - Multi-criteria decision making - Pairwise comparisons.					
	9 Hours		Assignment	Automated Decision Systems	**Module 5**

Topics:

Automated Decision Systems - The Artificial Intelligence field - Basic concepts of Expert Systems - Applications of Expert Systems - Structure of Expert Systems - Knowledge Engineering - Development of Expert Systems - Location based Analytics - Cloud Computing - Business Intelligence.

Targeted Application & Tools that can be used: Anaconda/Google Colab, Google Data Studio, Deep Note

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

1. Gain an immersive understanding of the practices and processes used by a junior or associate data analyst in their day-to-day job
2. Learn key analytical skills (data cleaning, analysis, & visualization) and tools (spread sheets, SQL, R programming, Tableau)

Text Book

1. Business Intelligence, Analytics, Data Science, and AI – Ramesh Sharda, Dursun Delen, and Efraim Turban, Pearson Education, 5th Edition, 2024.
2. Data Mining for Business Analytics – Galit Shmueli, Peter C. Bruce, Nitin R. Patel, Wiley, 4th Edition, 2023.

References

1. Data Warehousing, Data Mining and OLAP – Alex Berson and Stephen J. Smith, McGraw-Hill, **1st Edition, 2004**.
2. Business Intelligence Guidebook – Rick Sherman, Morgan Kaufmann, **1st Edition, 2014**.
3. The Big Book of Dashboards – Steve Wexler, Jeffrey Shaffer, Andy Cotgreave, Wiley, **1st Edition, 2017**.
4. Practical Statistics for Data Scientists – Peter Bruce, Andrew Bruce, Peter Gedeck, O'Reilly Media, **2nd Edition, 2020**.
5. Python for Data Analysis – Wes McKinney, O'Reilly Media, **3rd Edition, 2022**

Web links

R1. http://owl.english.purdue.edu/owl/resource/560/01/

R2. http://myregisapp.regis.edu/Citrix/StoreWeb/

R3. https://in.coursera.org/courses?query=business%20intelligence

R4. https://www.coursera.org/learn/business-intelligence-data-analytics

R5.https://www.udemy.com/course/business-intelligence-and-data-analytics/
Topics relevant to development of "Employability": Business Intelligence, Big Data Analytics, Data Scientist.

3	0	0	3	L- P- T-C	Course Title: Cyber Security and Data Privacy Type of Course: Theory	Course Code: IST3413
1.0						Version No.
NIL						Course Pre-requisites
NIL						Anti-requisites
This course provides a foundational yet comprehensive understanding of modern cyber security practices and data privacy principles. Students will explore real-world cyber threats, security frameworks, cryptographic techniques, and regulatory frameworks for privacy. Emphasis is placed on practical approaches to securing networks, ethical hacking, and forensic analysis to prepare learners for dynamic challenges in digital security.						Course Description
On successful completion of this course the students shall be able to: Assess cyber threats across various domains and evaluate potential security risks. (Bloom's Level: Evaluate) Apply cryptographic methods to secure data and communications. (Bloom's Level: Apply) Implement security tools for protecting networks and systems. (Bloom's Level: Apply) Understand data privacy laws, including frameworks like GDPR. (Bloom's Level: Understand) Perform basic penetration testing and digital forensics investigations to detect and analyze security breaches. (Bloom's Level: Analyze)						Course Outcomes
Course Content:						
	Assignment				Foundations of Cyber Security	Module 1
Topics: Introduction to Cyber Security – Cyber threat landscape: malware, ransomware, phishing –						

Risk assessment and management – Security policies and standards (e.g., ISO 27001) – Security protocols and layers of defense.			
13 Classes	Assignment	Cryptography and Network Security	Module 2
Topics: Symmetric and asymmetric encryption – Hash functions and digital signatures – Public Key Infrastructure (PKI) – Secure communication protocols (TLS, SSH) – Network security fundamentals: firewalls, IDS/IPS, VPNs. Networking basics (home network and large-scale business networks), Networking concepts (CIDR, subnets), and protocols (DNS, DHCP, IP). Security of protocols, sample application hosted on-premises.			
10 Classes	Assignment	Data Privacy and Legal Frameworks	Module 3
Topics: Principles of data privacy and data governance – GDPR, HIPAA, and Indian IT Act – Data classification and anonymization techniques – Privacy-by-design in systems – Consent, rights, and responsibilities of data subjects.			
12 Classes	Assignment	Ethical Hacking and Digital Forensics	Module-4
Topics: Introduction to ethical hacking & penetration testing – Footprinting, scanning, and enumeration techniques – Common tools: Nmap, Metasploit, Wireshark – Incident response and log analysis – Basics of digital forensics and evidence handling.			
Project work/Assignment:			
Assignment 1 on (Module 1 and Module 2) Assignment 2 on (Module 3 and Module 4)			
REFERENCE MATERIALS: TEXTBOOKS Engebretson, P. The Basics of Hacking and Penetration Testing. Syngress, 2nd Edition, 2013. ISBN: 978-0124116443. Singer, P. W., & Friedman, A. Cybersecurity and Cyberwar: What Everyone Needs to Know. Oxford University Press, 1st Edition, 2014. ISBN: 978-0199918119. Kolah, A. The GDPR Handbook: A Guide to Implementing the EU General Data Protection Regulation. Kogan Page, 1st Edition, 2018. ISBN: 978-0749474942. REFERENCES William Stallings. Cryptography and Network Security: Principles and Practice. Pearson, 8th Edition, 2024. ISBN: 978-1292742366.			

William Stallings and Lawrie Brown. Computer Security: Principles and Practice. Pearson, 5th Edition, 2023. ISBN: 978-0138091712. William Stallings. Network Security Essentials: Applications and Standards. Pearson, 6th Edition, 2017. ISBN: 978-0134527338. JOURNALS/MAGAZINES IEEE Security & Privacy Magazine IEEE Transactions on Information Forensics and Security IEEE Transactions on Dependable and Secure Computing IEEE Internet Computing IEEE Communications Magazine SWAYAM/NPTEL/MOOCs: NPTEL on SWAYAM – Cyber Security and Privacy Weblink: https://onlinecourses.nptel.ac.in/noc24_cs121/preview

Dr.K.Marimuthu	Catalogue prepared by
	Recommended by the Board of Studies on
	Date of Approval by the Academic Council

3-0-0-3	L-T-P-C	Course Name: Quantum Computing and AI	Course Code : IST3414
1.0			Version No.
6			NCrF Level-
CSE1500			Course Pre-requisites
NIL			Co-requisites
NIL			Anti-requisites
This course provides an introduction to the theory and practice of quantum computation. Topics covered include: quantum mechanics to understand quantum computation. Quantum algorithms. The Shor's factorization algorithm Grover's search algorithm Mathematical models of quantum computation, Quantum Machine Learning, and to physical systems.			Course Description

The course aims to: 1. Adapt basic knowledge on various techniques and areas of applications in Quantum computing. 2. Analyze common problem in quantum computing, alignment techniques, ethical issues, public data sources, and evolutionary modelling 3. Discover the practical use of tools for specific Quantum Computing areas and attain skills.	Course Objective
On successful completion of the course the students shall be able to: CO1: Identify various theoretical foundations of quantum computing .[Understand] CO2: Develop the ability for critical assessment of scientific approach in quantum computing[Apply] CO3: Design quantum circuits using quantum gates [Apply] CO4: Analyse the behaviour of basic quantum algorithms. [Evaluate] CO5: Understand the difference between classical and quantum machine learning approach [Apply]	Course Outcomes

Course Content:

10 Sessions	Assessment: Quiz/Assignment	Introduction	Module 1
Topics: Introduction to quantum computing. Qubits, Bloch sphere, multiple qubits, quantum states and measurements, Quantum Entanglement, Postulates of quantum mechanics, Duality theorem and quantum system. Classical computation vs quantum computation.			
12 Sessions	Assessment: Assignment	Quantum Model of Computation	Module 2
Topics: The model of quantum computation, Vector space representation of Bloch sphear, Quantum circuits: single qubit gates, multiple qubit gates, design of quantum circuits.			
13 Sessions	Assessment: Quiz/Case Study	Quantum Algorithms	Module 3
Topics: Deutsch-Jozsa algorithm and Grover's search algorithm. Shor's algorithm for factoring, Quantum Fourier transform			
10 Sessions	Assessment: Case Study/Presentation	Quantum Information theory and Quantum machine learning	Module 4
Comparison between classical and quantum information theory, Applications of quantum information, Bell states, Quantum Machine Learning, no cloning theorem. Introduction to quantum communication and post quantum cryptography.			
Targeted Applications & Tools:			

• Quantum Circuits • Quantum Gates • Quantum Machine Learning Algorithms	Applications
• Framework- Qiskit , PennyLane • Language- Python	Tools/Software

Suitable Skill-Building Activities through Experiential Learning:

- Create quantum circuit functions that can compute the XOR, AND, NAND and OR gates using the NOT gate (expressed as x in Qiskit), the CNOT gate (expressed as cx in Qiskit) and the Toffoli gate (expressed as ccx in Qiskit) .
- Measure the Bloch sphere coordinates of a qubit using the Aer simulator and plot the vector on the Bloch sphere
- Investigate the relationship between the number of qubits required for the desired accuracy of the phase estimation with high probability.

Project Work:

- Create a program that builds an oracle for a given string (e.g. given 01101, will return a QuantumCircuit that inverts the phase of the state and leaves all other states unchanged.

Text Book

T1: Nielsen, M., & Chuang, I. (2010). Quantum Computation and Quantum Information: 10th Anniversary Edition. Cambridge: Cambridge University Press. doi:10.1017/CBO9780511976667
T2: McMahon D. Quantum Computing Explained. Hoboken N.J: Wiley-Interscience : IEEE Computer Society; 2008.

References

R1:Benenti G., Casati G. and Strini G., Principles of Quantum Computation and Information, Vol. I: Basic Concepts, Vol II: Basic Tools and Special Topics, World Scientific. (2004)
R2: Pittenger A. O., An Introduction to Quantum Computing Algorithms (2000).
E book link R1:
http://community.qiskit.org/textbook
E book link R2
https://github.com/Qiskit
R3 Web resources:
Abraham Asfaw and Antonio Corcoles & et al. “Learn Quantum Computation Using Qiskit”, 2020, http://community.qiskit.org/textbook
IBM Qiskit Global Summer School 2021: Quantum Machine Learning, https://qiskit.org/events/summer-school/
https://quantum-computing.ibm.com/
https://qiskit.org/
https://presiuniv.knimbus.com/u