PRESIDENCY UNIVERSITY

GAIN MORE KNOWLEDGE
REACH GREATER HEIGHTS

PROGRAMME REGULATIONS & CURRICULUM

2024-28

PRESIDENCY SCHOOL OF COMPUTER SCIENCE & ENGINEERING

BACHELOR OF TECHNOLOGY (B.TECH.)

COMPUTER ENGINEERING

PRESIDENCY UNIVERSITY

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

PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

Program Regulations and Curriculum

2024-2028

BACHELOR OF TECHNOLOGY (B.Tech.) in

COMPUTER ENGINEERING

B.Tech. [COM]

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

(As amended up to the 24th Meeting of the Academic Council held on 3rd August 2024. This document supersedes all previous guidelines)

Regulations No: PU/AC-24.05/SOCSE04/COM/2024-2028

Resolution No. 05 of the 24th Meeting of the Academic Council held on 3rd August 2024, and ratified by the Board of Management in its 24th Meeting held on 5th August 2024.

AUGUST-2024

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 a contemporary Learning-pedagogy, integrating theory and practice.
- Attract and nurture world-class faculty to excel in Teaching and Research, in the field of Core Engineering.
- Establish state-of-the-art facilities for effective Teaching and Learning-experiences.
- Promote Interdisciplinary Studies to nurture talent and impart relevant skill-sets for global impact.
- Instil Entrepreneurial and Leadership Skills to address Social, Environmental, and Community-needs.

2. Preamble to the Program Regulations and Curriculum

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

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

In exercise of the powers conferred by and in discharge of duties assigned under the relevant provision(s) of the Act, Statutes and Academic Regulations, 2024 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 2024-2028.

b. These Regulations are subject to, and pursuant to the Academic Regulations.
c. These Regulations shall be applicable to the ongoing Bachelor of Technology Degree Programs of the 2024-2028 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 2024-2025.

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, 2024-2028;*

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, 2022;*

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 2024-2028 are subject to, and, pursuant to the Academic Regulations. These Program Regulations shall be applicable to the following ongoing Bachelor of Technology (B.Tech.) Degree Programs of 2024-2028 offered by the Presidency School of Computer Science and Engineering (PSCS):

1. B.Tech. Computer Science and Engineering
2. B. Tech. Computer Science and Technology (Big Data)
3. B. Tech. Computer Science and Engineering (Block Chain)
4. B. Tech. Computer Science and Technology (DevOps)
5. B. Tech. Computer Science and Engineering (Cyber Security)
6. B. Tech. Computer Science and Engineering (Internet of Things)

7. B. Tech. Computer Science and Engineering (Data Science)
8. B. Tech. Computer Science and Technology [Artificial Intelligence and Machine Learning]
9. B. Tech. Information Science and Technology [Artificial Intelligence and Data Science]
10. B. Tech. Computer Science and Information Technology
11. B. Tech. Computer Science and Engineering (Networks)
12. B. Tech. Computer Engineering
13. B. Tech. Information Science and Engineering [Artificial Intelligence and Robotics]
14. B. Tech. Computer Science and Engineering (Artificial Intelligence and Machine Learning)
15. B. Tech Artificial Intelligence and Robotics

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 16.1 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 19.0 of Academic Regulations) in the prescribed maximum duration (Sub-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:

PEO1. Demonstrate expertise as competent and ethical Computer Engineering professionals by leveraging foundational knowledge, technical skills, and innovative approaches to analyze, design, and develop cutting-edge solutions in the fields of Artificial Intelligence, Machine Learning, and related technologies.

PEO2. Become a teaching and research professional in the area of Computer Engineering through lifelong learning.

PEO3. Evolve as a consultant in the Computer Engineering Industry.

PEO4. Transform as an entrepreneur in the Computer Engineering and other related areas of Specialization.

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 the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2. Problem Analysis: Identify, formulate, review research literature, and analyse complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3. Design/Development of Solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4. Conduct Investigations of Complex Problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5. Modern Tool Usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO6. The Engineer and Society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7. Environment and Sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9. Individual and Team Work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10. Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11. Project Management and Finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12. Life-Long Learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning 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:

PSO 01: An ability to use and develop cloud software, administrative features Infrastructure services and architectural patterns: ethical hacking and forensic security technologies

PSO 02: An ability to gain knowledge on design and control strategy; techniques to secure information and adapt to the fast-changing world of information

PSO 03: An ability to gain working Knowledge on emerging software tools and technologies and apply the knowledge of secure computing tools and techniques in the field of Information science and technology for solving real world problems.

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, 2024-2028, 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. (Computer Engineering) 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 Computer Engineering 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 Presidency 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 10.1.1, 10.1.2, and 10.1.3.

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 Presidency 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 8.8 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 8.10 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**

S.No	Credit Structure [L-T-P-C]	Percentage/ Marks	CA		Mid-Term		End-term		Project	Total	Exam Conducted b
			Theory	Practical	Theory	Practical	Theory	Practical			
1	3-0-0-3	Percentage	25%	-	25%	-	50%	-	-	100%	Mid-Term & End Te by CoE
		Marks	50	-	50	-	100	-	-	200	
2	2-0-2-3	Percentage	12.50%	12.50%	12.50%	12.50%	25%	25%	-	100%	Mid-Term & End Te by CoE * Except for stack courses
		Marks	25	25	25	25	50	50	-	200	
3	1-0-4-3	Percentage	-	25%	10%	40%	5%	20%	-	100%	Mid-Term & End Te by School
		Marks	-	25	10	40	5	20	-	100	
4	2-0-4-4	Percentage	12.50%	12.50%	10%	15%	20%	30%	-	100%	*Mid-Term & End Term by CoE
		Marks	25	25	20	30	40	60	-	200	
5	0-0-4-2	Percentage	-	50%	-	-	-	-	50%	100%	Project evaluated b at School level
		Marks	-	50	-	-	-	-	50	100	
6	0-0-2-1	Percentage	-	100%	-	-	-	-	-	100%	Only CA at School Le
		Marks	-	100	-	-	-	-	-	100	
7	3-0-2-4	Percentage	12.50%	12.50%	15%	10%	30%	20%	-	100%	Mid-Term & End Te by CoE
		Marks	25	25	30	20	60	40	-	200	
8	2-0-0-2	Percentage	25%	-	25%	-	50%	-	-	100%	Mid-Term & End Term CoE
		Marks	50	-	50	-	100	-	-	200	

***CSE3150-Front End Full stack development**

CSE3151-Java Full Stack Development

CSE3152-.Net Full Stack development

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

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

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-Clauses 8.9.1 and 8.9.2 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.

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 ANNEXURE B 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 Clause 13.3 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 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 8.11 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 for the award of the concerned Degree.

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

13.4 The maximum number of credits that can be transferred by a student shall be limited to forty percent (40%) of the mandatory minimum credit requirements specified by the concerned Program Regulations and Curriculum for the award of the concerned Degree. However, the grades obtained in the Courses transferred from other Institutions/MOOCs, as mentioned in this Section (13.0), shall not be included in the calculation of the CGPA.

PART B: PROGRAM STRUCTURE

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

The B.Tech. (Computer Engineering) Program Structure (2024-2028) 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. (Computer Engineering) 2024-2028: Summary of Mandatory Courses and Minimum Credit Contribution from various Baskets

Sl. No.	Baskets	Credit Contribution
1	Humanities and Social Sciences including Management Courses (HSMC)	10
2	Basic Science Courses (BSC)	19
3	Engineering Science Courses (ESC)	23
4	Professional Core Courses (PCC)	68
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

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. (Computer Engineering) 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 19.2.1 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.

PART C CURRICULUM STRUCTURE

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 Name	L	T	P	C
1	Technical English	1	0	2	2
2	Introduction to soft skills	0	0	2	1
3	Introduction to Design Thinking	1	0	0	1
4	Advanced English / Foreign Language courses	1	0	2	2
5	Enhancing Personality Through Soft Skills	0	0	2	1
6	Managerial Economics and Financial Analysis	3	0	0	3
				Total No. of Credits	**10**

Table 3.2: List of Basic Science Courses (BSC)					
S.No	Course Name	L	T	P	C
1	Calculus and Linear Algebra	3	0	2	4
2	Optoelectronics and Device Physics	2	0	2	3
3	Applied Statistics	2	0	0	2
4	Integral Transforms and Partial Differential Equations	3	0	0	3
5	Numerical Computations	3	0	0	3
6	Discrete Mathematics	4	0	0	4
				Total No. of Credits	**19**

Table 3.3: List of Engineering Science Courses (ESC)					
S.No	Course Name	L	T	P	C
1	Engineering Graphics	2	0	0	2
2	Problem Solving Using C	1	0	4	3
3	Digital Design	2	0	2	3
4	Basic Engineering Sciences	2	0	0	2
5	Problem Solving using JAVA	1	0	4	3
6	Basics of Electrical and Electronics Engineering	3	0	2	4
7	Innovative Projects Using Arduino	-	-	-	1
8	Computational Thinking using Python	**2**	**0**	2	3
9	Algorithmic Competitive Programming	**0**	**0**	**4**	**2**
				Total No. of Credits	23

Table 3.4 : List of Professional Core Courses (PCC)					
S. No	Course Name	L	T	P	C
1	Data Communication and Computer Networks	**3**	**0**	**0**	3

2	Computer Organization and Architecture	3	0	0	3
3	Web Technologies	**2**	**0**	**0**	2
4	Data Communication and Computer Networks Lab	**0**	**0**	**2**	1
5	Web Technologies Lab	**0**	**0**	**2**	1
6	Analysis of Algorithms	3	1	0	4
7	Database Management Systems	**3**	**0**	**0**	3
8	Operating Systems	3	0	0	3
9	Operating Systems Lab	0	0	2	1
10	Machine Learning	3	**0**	**0**	3
11	Machine Learning Lab	**0**	**0**	4	2
12	Database Management Systems Lab	**0**	**0**	**2**	1
13	Analysis of Algorithms Lab	**0**	**0**	**2**	1
14	Theory of Computation	3	0	0	3
15	Software Design and Development	3	0	0	3
16	Cloud Computing	**2**	**0**	**0**	2
17	Natural Language Processing	3	0	0	3
18	Scalable Application Development using Java	**3**	**0**	**0**	3
19	Scalable Application Development using Java Lab	0	0	4	2
20	Cloud Computing Lab	**0**	**0**	**2**	1
21	Cryptography and Network Security	**3**	0	0	**3**
22	IoT: Architecture and Protocols	3	0	0	3
24	Computer Vision and LLMs	2	0	0	2
25	Applied Machine Learning	**2**	0	**0**	**2**
26	Applied Machine Learning Lab	**0**	0	**2**	1
27	Computer Vision and LLMs Lab	0	0	2	1
28	Reinforcement Learning Lab	0	0	2	1
29	Natural Language Processing Lab	0	0	2	1
30	Essentials of AI	**3**	**0**	**0**	3
31	Essentials of AI Lab	**0**	**0**	**4**	2
32	Data Structures	3	0	0	3
33	Data Structures Lab	0	0	4	2
				Total No. of Credits	**68**

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

S.No	Course Name	L	T	P	C
1	Capstone Project	0	0	0	10
2	Mini Project	0	0	0	4
3	Internship	0	0	0	2
				Total No. of Credits	**16**

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 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). The same shall be prescribed in the Course Handout.

18.1 Internship

A student may undergo an Internship for a period of 4-6 weeks in an industry / company or academic / research institution during the Semester Break between 4th and 5th Semesters or 6th and 7th Semesters, subject to the following conditions:

18.1.1 The Internship shall be in conducted in accordance with the Internship Policy prescribed by the University from time to time.

18.1.2 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 Internship to a student;

18.1.3 The number of Internships available for the concerned Academic Term. Further, the available number of internships 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 Internship, as stated in Sub-Clause 18.1.2 above.

18.1.4 A student may opt for Internship 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 Internship on her / his own. Provided further, that the Industry / Company or academic / research institution offering such Internship confirms to the University that the Internship shall be conducted in accordance with the Program Regulations and Internship Policy of the University.

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

*18.2 **Mini-Project***

A student may opt to do a Project Work for a period of 4-6 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 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 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 project work confirms to the University that the 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 I 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 Internship 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 3.6 : 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, Machine Learning and Data Science							
Sl. No.	**Course Code**	**Course Name**	**L**	**T**	**P**	**C**	**Prerequisite**
1	CSE3400	Intelligent Systems with Machine Learning	2	0	2	3	CSE1700
2	CSE3401	Advanced Deep Learning Techniques	3	0	0	3	CAI2502
3	CSE3402	Computational Optimization for Intelligent Systems.	3	0	0	3	CSE1700
4	CSE3403	Reinforcement Learning for AI Systems	2	0	2	3	CAI3409
6	CSE3405	Synergistic Neural Fuzzy Computing	2	0	2	3	CSN2508
7	CSE3409	Emerging Technologies in Big Data	2	0	2	3	CSE3156
8	CSE3410	Statistical Techniques of Data Science	2	0	2	3	MAT1003
9	CSE3411	Predictive Analytics and Applications	2	0	2	3	MAT1003
10	CSE2021	Data Mining	3	0	0	3	MAT1003
11	CSE3413	No SQL Data Management	2	0	2	3	CSE3156

12	CSE3414	Applied Data Intelligence	2	0	2	3	Nil

Track - 2 Cloud, Security & Systems

Sl. No.	Course Code	Course Name	L	T	P	C	Prerequisite
1	CSE3415	Cloud Data Engineering	2	0	2	3	CSE3155
2	CSE3416	Federated Learning	2	0	2	3	CSE3155
3	CSE3417	Edge Computing	2	0	2	3	CSE3155
4	CSE3418	Network Security and Firewall Management	2	0	2	3	CSE3155
5	CSE3419	Information Security and Management	3	0	0	3	CSE3155
6	CSE3420	Network Intrusion Detection and Prevention	3	0	0	3	CSE3155
7	CSE3421	Principles and Practices of Web Security	2	0	2	3	CSE3155
8	CSE3422	Penetration Testing and Risk Assessment	3	0	0	3	CSE3155

Track -3 Programming

Sl. No.	Course Code	Course Name	L	T	P	C	Prerequisite
1	CSE3423	Go Programming	3	0	0	3	CSE1004
2	CSE3424	Advanced Database Management Systems	2	0	2	3	CSE3156
3	CSE3425	Programming in C# and .NET	1	0	4	3	CSE1006
4	CSE3426	Front End Full Stack Development	2	0	2	3	CSE1006
5	CSE3427	Java Full Stack Development	2	0	2	3	CSE1006
6	CSE3428	.Net Full Stack Development	2	0	2	3	CSE1006
7	CAI3427	Language Models for Text Mining	2	0	2	3	
8	CAI3428	Practical Deep Learning with TensorFlow	2	0	2	3	
9	CAI3429	Deep Learning for Computer Vision	2	0	2	3	
10	COM3401	Advanced Computer Architecture	3	0	0	3	

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

Table 3.7 : Open Elective Courses Baskets: Minimum Credits to be earned from this Basket is 9

Sl. No.	Course Code	Course Name	L	T	P	C	Type of Skill/ Focus	Course Caters to	Prerequisites/ Corequisites	Antirequisites	Future Courses that need this as a Prerequisite
Chemistry Basket											
1	CHE1003	Fundamentals of Sensors	3	0	0	3	S	ES	-	-	-
2	CHE1004	Smart materials for IOT	3	0	0	3	S	ES	-	-	-
3	CHE1005	Computational Chemistry	2	0	0	2	S	ES	-	-	-
4	CHE1006	Introduction to Nano technology	3	0	0	3	S	ES	-	-	-
5	CHE1007	Biodegradable electronics	2	0	0	2	S	ES	-	-	-
6	CHE1008	Energy and Sustainability	2	0	0	2	S	ES	-	-	-
7	CHE1009	3D printing with Polymers	2	0	0	2	S	ES	-	-	-
8	CHE1010	Bioinformatics and Healthcare IT	2	0	0	2	S	ES	-	-	-
9	CHE1011	Chemical and Petrochemical catalysts	3	0	0	3	S	ES	-	-	-
10	CHE1012	Introduction to Composite materials	2	0	0	2	S	ES	-	-	-
11	CHE1013	Chemistry for Engineers	3	0	0	3	S	ES	-	-	-
12	CHE1014	Surface and Coatings technology	3	0	0	3	S	ES	-	-	-
13	CHE1015	Waste to Fuels	2	0	0	2	S	ES	-	-	-
14	CHE1016	Forensic Science	3	0	0	3	S	ES	-	-	-
Civil Engineering Basket											
1	CIV1001	Disaster mitigation and management	3	0	0	3	S	-	-	-	-
2	CIV1002	Environment Science and Disaster Management	3	0	0	3	FC	-	-	-	-
3	CIV2001	Sustainability Concepts in Engineering	3	0	0	3	S	-	-	-	-
4	CIV2002	Occupational Health and Safety	3	0	0	3	S	-	-	-	-
5	CIV2003	Sustainable Materials and Green Buildings	3	0	0	3	EM	-	-	-	-
6	CIV2004	Integrated Project Management	3	0	0	3	EN	-	-	-	-
7	CIV2005	Environmental Impact Assessment	3	0	0	3	EN	-	-	-	-
8	CIV2006	Infrastructure Systems for Smart Cities	3	0	0	3	EN	-	-	-	-
9	CIV2044	Geospatial Applications for Engineers	2	0	2	3	EM	-	-	-	-
10	CIV2045	Environmental Meteorology	3	0	0	3	S	-	-	-	-
11	CIV3046	Project Problem Based Learning	3	0	0	3	S	-	-	-	-
12	CIV3059	Sustainability for Professional Practice	3	0	0	3	EN	-	-	-	-
Commerce Basket											
1	COM2001	Introduction to Human Resource Management	2	0	0	2	F	HP/GS	-	-	-
2	COM2002	Finance for Non-Finance	2	0	0	2	S	-	-	-	-
3	COM2003	Contemporary Management	2	0	0	2	F	-	-	-	-
4	COM2004	Introduction to Banking	2	0	0	2	F	-	-	-	-
5	COM2005	Introduction to Insurance	2	0	0	2	F	-	-	-	-
6	COM2006	Fundamentals of Management	2	0	0	2	F	-	-	-	-
7	COM2007	Basics of Accounting	3	0	0	3	F	-	-	-	-
Computer Science Basket											
1	CSE2002	Programming in Java	2	0	2	3	S/EM	-	-	-	-
2	CSE2003	Social Network Analytics	3	0	0	3	S	GS	-	-	-
3	CSE2004	Python Application Programming	2	0	2	3	S/ EM	-	-	-	-
4	CSE2005	Web design fundamentals	2	0	2	3	S/ EM/EN	-	-	-	-
5	CSE3111	Artificial Intelligence: Search Methods For Problem Solving	3	0	0	3	S/ EM/EN	-	-	-	-

6	CSE3112	Privacy And Security In Online Social Media	3	0	0	3	S/ EM/EN	-	-	-	-
7	CSE3113	Computational Complexity	3	0	0	3	S/ EM/EN	-	-	-	-
8	CSE3114	Deep Learning for Computer Vision	3	0	0	3	S/ EM/EN	-	-	-	-
9	CSE3115	Learning Analytics Tools	3	0	0	3	S/ EM/EN	-	-	-	-
Design Basket											
1	DES1001	Sketching and Painting	0	0	2	1	S	-	-	-	-
2	DES1002	Innovation and Creativity	2	0	0	2	F	-	-	-	-
3	DES1121	Introduction to UX design	1	0	2	2	S	-	-	-	-
4	DES1122	Introduction to Jewellery Making	1	0	2	2	S	-	-	-	-
5	DES1124	Spatial Stories	1	0	2	2	S	-	-	-	-
6	DES1125	Polymer Clay	1	0	2	2	S	-	-	-	-
7	DES2001	Design Thinking	3	0	0	3	S	-	-	-	-
8	DES1003	Servicability of Fashion Products	1	0	2	2	F	ES	-	-	-
9	DES1004	Choices in Virtual Fashion	1	0	2	2	F	ES, GS, HP	-	-	-
10	DES1005	Fashion Lifestyle and Product Diversity	1	0	2	2	F	ES, GS, HP	-	-	-
11	DES1006	Colour in Everyday Life	1	0	2	2	F	ES	-	-	-
12	DES2080	Art of Design Language	3	0	0	3	S	-	-	-	-
13	DES2081	Brand Building in Design	3	0	0	3	S	-	-	-	-
14	DES2085	Web Design Techniques	3	0	0	3	S	-	-	-	-
15	DES2089	3D Modeling for Professionals	1	0	4	3	S	-	-	-	-
16	DES2090	Creative Thinking for Professionals	3	0	0	3	S	-	-	-	-
17	DES2091	Idea Formulation	3	0	0	3	S	-	-	-	-
Electrical and Electronics Basket											
1	EEE1002	IoT based Smart Building Technology	3	0	0	3	S	-	-	-	-
2	EEE1003	Basic Circuit Analysis	3	0	0	3	S	-	-	-	-
3	EEE1004	Fundamentals of Industrial Automation	3	0	0	3	S	-	-	-	-
4	EEE1005	Electric Vehicles & Battery Technology	3	0	0	3	S	-	-	-	-
5	EEE1006	Smart Sensors for Engineering Applications	3	0	0	3	S	-	-	-	-
Electronics and Communication Basket											
1	ECE1003	Fundamentals of Electronics	3	0	0	3	F	-	-	-	-
2	ECE1004	Microprocessor based systems	3	0	0	3	F	-	-	-	-
3	ECE3089	Artificial Neural Networks	3	0	0	3	S	-	-	-	-
4	ECE3097	Smart Electronics in Agriculture	3	0	0	3	F/EM	-	-	-	-
5	ECE3098	Environment Monitoring Systems	3	0	0	3	F/EM	-	-	-	-
6	ECE3102	Consumer Electronics	3	0	0	3	F/EM	-	-	-	-
7	ECE3103	Product Design of Electronic Equipment	3	0	0	3	S/F/ EM / EN	-	-	-	-
8	ECE3106	Introduction to Data Analytics	3	0	0	3	F/EM	-	-	-	-
9	ECE3107	Machine Vision for Robotics	3	0	0	3	F/EM	-	-	-	-
English Basket											
1	ENG1008	Indian Literature	2	0	0	2	-	GS/ HP	-	-	-
2	ENG1009	Reading Advertisement	3	0	0	3	S	-	-	-	-
3	ENG1010	Verbal Aptitude for Placement	2	0	2	3	S	-	-	-	-
4	ENG1011	English for Career Development	3	0	0	3	S	-	-	-	-
5	ENG1012	Gender and Society in India	2	0	0	2	-	GS/ HP	-	-	-
6	ENG1013	Indian English Drama	3	0	0	3	-	-	-	-	-
7	ENG1014	Logic and Art of Negotiation	2	0	2	3	-	-	-	-	-

8	ENG1015	Professional Communication Skills for Engineers	1	0	0	1	-	-	-	-	-
DSA Basket											
1	DSA2001	Spirituality for Health	2	0	0	2	F	HP	-	-	-
2	DSA2002	Yoga for Health	2	0	0	2	S	HP	-	-	-
3	DSA2003	Stress Management and Well Being	2	0	0	2	F	-	-	-	-
Kannada Basket											
1	KAN1001	Kali Kannada	1	0	0	1	S	-	-	-	-
2	KAN1003	Kannada Kaipidi	3	0	0	3	S	-	-	-	-
3	KAN2001	Thili Kannada	1	0	0	1	S	-	-	-	-
4	KAN2003	Pradharshana Kale	1	0	2	2	S	-	-	-	-
5	KAN2004	Sahithya Vimarshe	2	0	0	2	S	-	-	-	-
6	KAN2005	Anuvadha Kala Sahithya	3	0	0	3	S	-	-	-	-
7	KAN2006	Vichara Manthana	3	0	0	3	S	-	-	-	-
8	KAN2007	Katha Sahithya Sampada	3	0	0	3	S	-	-	-	-
9	KAN2008	Ranga Pradarshana Kala	3	0	0	3	S	-	-	-	-
Foreign Language Basket											
1	FRL1004	Introduction of French Language	2	0	0	2	S	S	-	-	-
2	FRL1005	Fundamentals of French	2	0	0	2	S	S	-	-	-
3	FRL1009	Mandarin Chinese for Beginners	3	0	0	3	S	S	-	-	-
Law Basket											
1	LAW1001	Introduction to Sociology	2	0	0	0	2	F	HP	-	-
2	LAW2001	Indian Heritage and Culture	2	0	0	0	2	F	HP/GS	-	-
3	LAW2002	Introdcution to Law of Succession	2	0	0	0	2	F	HP/GS	-	-
4	LAW2003	Introduction to Company Law	2	0	0	0	2	F	HP	-	-
5	LAW2004	Introduction to Contracts	2	0	0	2	F	HP	-	-	-
6	LAW2005	Introduction to Copy Rights Law	2	0	0	2	F	HP	-	-	-
7	LAW2006	Introduction to Criminal Law	2	0	0	2	F	HP	-	-	-
8	LAW2007	Introduction to Insurance Law	2	0	0	2	F	HP	-	-	-
9	LAW2008	Introduction to Labour Law	2	0	0	2	F	HP	-	-	-
10	LAW2009	Introduction to Law of Marriages	2	0	0	2	F	HP/GS	-	-	-
11	LAW2010	Introduction to Patent Law	2	0	0	2	F	HP	-	-	-
12	LAW2011	Introduction to Personal Income Tax	2	0	0	2	F	HP	-	-	-
13	LAW2012	Introduction to Real Estate Law	2	0	0	2	F	HP	-	-	-
14	LAW2013	Introduction to Trademark Law	2	0	0	2	F	HP	-	-	-
15	LAW2014	Introduction to Competition Law	3	0	0	3	F	HP	-	-	-
16	LAW2015	Cyber Law	3	0	0	3	F	HP	-	-	-
17	LAW2016	Law on Sexual Harrassment	2	0	0	2	F	HP/GS	-	-	-
18	LAW2017	Media Laws and Ethics	2	0	0	2	F	HP/GS	-	-	-
Mathematics Basket											
1	MAT2008	Mathematical Reasoning	3	0	0	3	S	-	-	-	-
2	MAT2014	Advanced Business Mathematics	3	0	0	3	S	-	-	-	-
3	MAT2041	Functions of Complex Variables	3	0	0	3	S	-	-	-	-
4	MAT2042	Probability and Random Processes	3	0	0	3	S	-	-	-	-
5	MAT2043	Elements of Number Theory	3	0	0	3	S	-	-	-	-
6	MAT2044	Mathematical Modelling and Applications	3	0	0	3	S	-	-	-	-
Mechanical Basket (not to be offered for Mechanical Department students)											
1	MEC1001	Fundamentals of Automobile Engineering	3	0	0	3	F	-	-	-	-
2	MEC1002	Introduction to Matlab and Simulink	3	0	0	3	S/EM	-	-	-	-
3	MEC1003	Engineering Drawing	1	0	4	3	S	-	-	-	-
4	MEC2001	Renewable Energy Systems	3	0	0	3	F	ES	-	-	-
5	MEC2002	Operations Research & Management	3	0	0	3	F	-	-	-	-

6	MEC2003	Supply Chain Management	3	0	0	3	S/ EM/ EN	-	-	-	-
7	MEC2004	Six Sigma for Professionals	3	0	0	3	S/EM	-	-	MEC 2008	-
8	MEC2005	Fundamentals of Aerospace Engineering	3	0	0	3	F	-	-	-	-
9	MEC2006	Safety Engineering	3	0	0	3	S/EM	ES	-	-	-
10	MEC2007	Additive Manufacturing	3	0	0	3	F/EM	-	-	-	-
11	MEC3069	Engineering Optimisation	3	0	0	3	S/EM	-	-	-	-
12	MEC3070	Electronics Waste Management	3	0	0	3	F/S	ES	-	-	-
13	MEC3071	Hybrid Electric Vehicle Design	3	0	0	3	S/EM	ES	-	-	-
14	MEC3072	Thermal Management of Electronic Appliances	3	0	0	3	S/EM	-	-	-	-
15	MEC3200	Sustainable Technologies and Practices	3	0	0	3	S/EM	-	-	-	-
16	MEC3201	Industry 4.0	3	0	0	3	S/EM	-	-	-	-
Petroleum Basket											
1	PET1011	Energy Industry Dynamics	3	0	0	3	FC	ES	-	NIL	-
2	PET1012	Energy Sustainability Practices	3	0	0	3	FC	ES	-	NIL	-
Physics Basket											
1	PHY1003	Mechanics and Physics of Materials	3	0	0	3	FC / SD				
2	PHY1004	Astronomy	3	0	0	3	FC				
3	PHY1005	Game Physics	2	0	2	3	FC / SD				
4	PHY1006	Statistical Mechanics	2	0	0	2	FC				
5	PHY1007	Physics of Nanomaterials	3	0	0	3	FC				
6	PHY1008	Adventures in nanoworld	2	0	0	2	FC				
7	PHY2001	Medical Physics	2	0	0	2	FC	ES			
8	PHY2002	Sensor Physics	1	0	2	2	FC / SD				
9	PHY2003	Computational Physics	1	0	2	2	FC				
10	PHY2004	Laser Physics	3	0	0	3	FC	ES			
11	PHY2005	Science and Technology of Energy	3	0	0	3	FC	ES			
12	PHY2009	Essentials of Physics	2	0	0	2	FC				
Management Basket- I											
1	MGT2007	Digital Entrepreneurship	3	0	0	3	S/EM/EN	-	-	-	-
2	MGT2015	Engineering Economics	3	0	0	3	S	-	-	-	-
3	MGT2023	People Management	3	0	0	3	S/EM/ EN	HP	-	-	-
Management Basket- II											
1	MGT1001	Introduction to Psychology	3	0	0	3	F	HP	-	-	-
2	MGT1002	Business Intelligence	3	0	0	3	EN	-	-	-	-
3	MGT1003	NGO Management	3	0	0	3	S	-	-	-	-
4	MGT1004	Essentials of Leadership	3	0	0	3	EM/ EN	GS/ HP	-	-	-
5	MGT1005	Cross Cultural Communication	3	0	0	3	S/EM/ EN	HP	-	-	-
6	MGT2001	Business Analytics	3	0	0	3	S/ EM/EN	-	-	-	-
7	MGT2002	Organizational Behaviour	3	0	0	3	F	HP	-	-	-
8	MGT2003	Competitive Intelligence	3	0	0	3	S	-	-	-	-
9	MGT2004	Development of Enterprises	3	0	0	3	S/EM/EN	-	-	-	-
10	MGT2005	Economics and Cost Estimation	3	0	0	3	S/EM	-	-	-	-
11	MGT2006	Decision Making Under Uncertainty	3	0	0	3	S	-	-	-	-
12	MGT2008	Econometrics for Managers	3	0	0	3	S	-	-	-	-
13	MGT2009	Management Consulting	3	0	0	3	S/EM/EN	-	-	-	-
14	MGT2010	Managing People and Performance	3	0	0	3	S/EM/EN	HP/GS	-	-	-

15	MGT2011	Personal Finance	3	0	0	3	F	-	-	-	-
16	MGT2012	E Business for Management	3	0	0	3	S/EM	-	-	-	-
17	MGT2013	Project Management	3	0	0	3	EN / EM	GS/HP/ ES	-	-	-
18	MGT2014	Project Finance	3	0	0	3	EN / EM	HP	-	-	-
19	MGT2016	Business of Entertainment	3	0	0	3	EM/ EN	-	-	-	-
20	MGT2017	Principles of Management	3	0	0	3	S/EM/ EN	-	-	-	-
21	MGT2018	Professional and Business Ethics	3	0	0	3	S/EM/ EN	HP	-	-	-
22	MGT2019	Sales Techniques	3	0	0	3	S/EM/ EN	HP	-	-	-
23	MGT2020	Marketing for Engineers	3	0	0	3	S/EM/ EN	HP	-	-	-
24	MGT2021	Finance for Engineers	3	0	0	3	S/EM/ EN	HP	-	-	-
25	MGT2022	Customer Relationship Management	3	0	0	3	S/EM/ EN	HP	-	-	-
Media Studies Basket											
1	BAJ3050	Corporate Filmmaking and Film Business	0	0	4	2	EM	HP	-	-	-
2	BAJ3051	Digital Photography	2	0	2	3	EM	HP	-	-	-
3	BAJ3055	Introduction to News Anchoring and News Management	0	0	2	1	EM	-	-	-	-

21.List of MOOC (NPTEL) Courses

21.1 NPTEL - Discipline Elective Courses for B. Tech. (Computer Science Engineering)

Sl. No.	Course ID	Course Name	Duration
1	noc25-cs22	Deep Learning for Natural Language Processing	12 Weeks
2	noc25-cs49	Machine Learning for Engineering and Science Applications	12 Weeks
3	noc25-cs06	Algorithms in Compuatational Biology and Sequence Analysis	12 Weeks
4	noc25-cs45	Introduction to Large Language Models (LLMs)	12 Weeks
5	noc25-cs61	Quantum Algorithms and Cryptography	12 Weeks

21.2 NPTEL - Open Elective Courses for B. Tech. (Computer Science and Engineering)

Sl. No.	Course ID	Course Name	Duration
1	BBA2022	Supply Chain digitization	12 Weeks
2	BBA2021	E Business	12 Weeks
3	BBB2016	Business Analytics for Management Decisions	12 Weeks

4	BBB2015	Artifcial Intelligence for Investments	12 Weeks

The following Open-Elective courses are not to be offered for Computer Science and Engineering students and Allied Branches.

Sl. No	Course Code	Course Name	Total Credits	L-T-P-C
1	CSE3111	Artificial Intelligence : Search Methods For Problem Solving	3	3-0-0-3
2	CSE3112	Privacy And Security In Online Social Media	3	3-0-0-3
3	CSE3113	Computational Complexity	3	3-0-0-3
4	CSE3114	Deep Learning for Computer Vision	3	3-0-0-3
5	CSE3115	Learning Analytics Tools	3	3-0-0-3
6	CSE502	Technical Skills in JAVA	3	0-0-6-3
7	CSE503	Technical Skills in Python	3	0-0-6-3
8	CSE504	Comprehensive Technical Skills	5	0-0-10-5
9	CSE505	The Joy Of Computing Using Python	3	3-0-0-3
10	CSE3119	Coding Skills in Python	3	3-0-0-3
11	CSE3121	Parallel Computer Architecture	3	3-0-0-3
12	CSE3124	Games and Information	3	3-0-0-3
13	CSE3140	Introduction To Industry 4.0 And Industrial Internet Of Things	3	3-0-0-3
14	CSE3142	Affective Computing	3	3-0-0-3
15	CSE3112	Privacy and Security in Online Social Media	3	3-0-0-3
16	CSE3196	Foundations of Cyber Physical Systems	3	3-0-0-3
17	CSE3197	Getting Started with Competitive Programming	3	3-0-0-3
18	CSE3198	GPU Architectures And Programming	3	3-0-0-3
19	CSE3199	Artificial Intelligence: Knowledge Representation And Reasoning	3	3-0-0-3
20	CSE3200	Programming in Modern C++	3	3-0-0-3
21	CSE3201	Circuit Complexity Theory	3	3-0-0-3
22	CSE3202	Basics of Computational Complexity	3	3-0-0-3
23	CSE3212	ion to Computer and Network Performance Analysis Using Queuing	1	1-0-0-1
24	CSE3213	C Programming And Assembly Language	1	1-0-0-1
25	CSE3214	Python For Data Science	1	1-0-0-1
26	CSE3215	Software Conceptual Design	1	1-0-0-1
27	CSE3117	Industrial Digital Transformation	3	3-0-0-3
28	CSE3118	Blockchain for Decision Makers	3	3-0-0-3
29	CSE3349	Technology for Lawyers	3	3-0-0-3
30	CSEXXXX	Deep Learning for Natural Language Processing	3	3-0-0-3
31	CSEXXXX	Machine Learning for Engineering and science applications	3	3-0-0-3
32	CSEXXXX	Algorithms in Computational Biology and Sequence Analysis	3	3-0-0-3
33	CSEXXXX	Introduction to Large Language Models (LLMs)	3	3-0-0-3
34	CSEXXXX	Quantum Algorithms and Cryptography	3	3-0-0-3

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

Semester I (Physics Cycle)										
Sl. No.	Course Code	Course Name	Credit Structure				Contact Hours	Type of Course	Type of Skills	Course Addresses To
			L	T	P	C				
1	MAT1001	Calculus and Linear Algebra	3	0	2	4	5	BSC	FC	
2	PHY1002	Optoelectronics and Device Physics	2	0	2	3	4	BSC	FC	

3	MEC1006	Engineering Graphics	2	0	0	2	2	ESC	SD	
4	ENG1002	Technical English	1	0	2	2	3	HSMC	SD	
5	PPS1001	Introduction to soft skills	0	0	2	1	2	HSMC	SD	HP
6	CSE1004	Problem Solving Using C	1	0	4	3	5	ESC	SD	
7	ECE2007	Digital Design	2	0	2	3	4	ESC	FC/SD	ES
8	DES1146	Introduction to Design Thinking	1	0	0	1	1	HSMC	FC	HP
		TOTAL	**12**	**00**	**14**	**19**	**26**			

Semester II (Basic Engineering Science Cycle)										
Sl. No.	**Course Code**	**Course Name**	**Credit Structure**				**Contact Hours**	**Type of Course**	**Type of Skills**	**Course Addresses To**
			L	**T**	**P**	**C**				
1	MAT1003	Applied Statistics	2	0	0	2	2	BSC	EM	
2	CHE1018	Environmental Science	1	0	2	0	3	MAC	FC	ES
3	CIV1008	Basic Engineering Sciences	2	0	0	2	2	ESC	SD	
4	CSE1006	Problem Solving using JAVA	1	0	4	3	5	ESC	SD	
5	ENG2001/FRLXXXX	Advanced English / Foreign Language courses	1	0	2	2	3	HSMC	SD	
6	PPS1012	Enhancing Personality Through Soft Skills	0	0	2	1	2	HSMC	SD/EM	HP
7	EEE1007	Basics of Electrical and Electronics Engineering	3	0	2	4	5	ESC	FC	
8	LAW1007	Indian Constitution and Professional Ethics for Engineers	1	0	0	0	1	MAC	FC	HP
9	ECE2010	Innovative Projects Using Arduino	-	-	-	1	0	ESC	SD	
		TOTAL	**11**	**00**	**12**	**15**	**23**			

Semester III										
Sl. No.	**Course Code**	**Course Name**	**Credit Structure**				**Contact Hours**	**Type of Course**	**Type of Skills**	**Course Addresses To**
			L	**T**	**P**	**C**				
1	MAT2501	Integral Transforms and Partial Differential Equations	3	0	0	3	3	BSC		
2	CSE1508	Data Structures	3	0	0	3	3	PCC	S	
3	CSE1506	Data Communications and Computer Networks	3	0	0	3	3	PCC	SD	
4	CSE2501	Computer Organization and Architecture	3	0	0	3	3	PCC	SD	
5	COMXXXX	Professional Elective -1	3	0	0	3	3	PEC	SD	
6	MAT2605	Discrete Mathematics	4	0	0	4	4	BSC	EM	
7	CSE1500	Computational Thinking using Python	**2**	**0**	**2**	3	4	ESC		
	Practical Subjects									
9	CSE1509	Data Structures Lab	0	0	4	2	4	PCC	S	
10	CSE1507	Data Communications and Computer Networks Lab	0	0	2	1	2	PCC	SD	
		TOTAL	**21**	**0**	**08**	**25**	**29**			

Semester IV										
Sl. No.	**Course Code**	**Course Name**	**Credit Structure**				**Contact Hours**	**Type of Course**	**Type of Skills**	**Course Addresses To**
			L	**T**	**P**	**C**				
1	MAT2602	Numerical Computations	3	0	0	3	3	BSC		
2	CSE1512	Analysis of Algorithms	3	1	0	4	4	PCC		
3	CSE1510	Database Management Systems	3	0	0	3	3	PCC		
4	CSE2502	Operating Systems	3	0	0	3	3	PCC		
5	CSE1700	Essentials of AI	3	0	0	3	3	PCC		
6	CAI2500	Machine Learning	3	0	0	3	3	PCC		
7	MGTXXXX	Managerial Economics and Financial Analysis	3	0	0	3	3	HSMC		

	Practical Subjects									
8	CSE1513	Analysis of Algorithms Lab	**0**	**0**	**2**	**1**	**2**	**PCC**		
9	CSE1511	Database Management Systems Lab	0	0	2	1	2	PCC		
10	CSE1701	Essentials of AI Lab	**0**	**0**	**4**	2	4	PCC		
11	CAI2501	Machine Learning Lab	0	0	4	2	4	PCC		
12	CSE2514	Operating Systems Lab	0	0	2	1	2	PCC		
		TOTAL	**21**	**01**	**14**	**29**	**35**			

Semester V										
Sl. No.	**Course Code**	**Course Name**	**Credit Structure**				**Contact Hours**	**Type of Course**	**Type of Skills**	**Course Addresses To**
			L	**T**	**P**	**C**				
1	CSE1504	Web Technologies	2	0	0	2	2	PCC	SD	
2	CAI2504	Natural Language Processing	3	0	0	3	3	PCC	SD	
3	CSEXXXX	Professional Elective – II	3	0	0	3	3	PEC	SD/ EM	
4	CSE2500	Theory of Computation	3	0	0	3	3	PCC	SD/ EM	
5	CSE2000	Software Design & Development	3	0	0	3	3	PCC	SD/ EM	
6	XXXXXXX	Open Elective – I	3	0	0	3	3	OEC	SD/ EM/ EN	
7	CSE2506	Cloud Computing	2	0	0	2	2	PCC	SD/ EM	
8	CSE7000	Internship	-	-	-	2	0	PRW		
	Practical Subjects									
9	CSE2507	Cloud Computing Lab	0	0	2	1	2	PCC	SD/ EM	
10	CSE1505	Web Technologies Lab	0	0	2	1	2	PCC	SD	
11	CAI2505	Natural Language Processing Lab	0	0	2	1	2	PCC	SD	
		TOTAL	**19**	**00**	**06**	**24**	**25**			

Semester VI										
Sl. No.	**Course Code**	**Course Name**	**Credit Structure**				**Contact Hours**	**Type of Course**	**Type of Skills**	**Course Addresses To**
			L	**T**	**P**	**C**				
1	CSE2503	Cryptography and Network Security	3	0	0	3	3	PCC		
2	ECE3075	IoT: Architecture and Protocols	3	0	0	3	3	PCC	SD	
3	COM2502	Applied Machine Learning	2	0	0	2	2	PCC	SD	
4	COM2500	Computer Vision and LLMs	2	0	0	2	1	PCC	SD	
5	CSEXXXX	Professional Elective – III	3	0	0	3	3	PEC	SD/ EM	
6	CSEXXXX	Professional Elective –IV	3	0	0	3	3	PEC	SD/ EM	HP
7	CSE2504	Scalable Application Development using Java	**3**	**0**	**0**	3	3	PCC		
8	CSE2505	Scalable Application Development using Java Lab	0	0	4	2	4	PCC		
9	PPSXXXX	Industry Preparedness Program	2	0	0	0	2	MAC		
10	CSE2510	Competitive Programming and Problem Solving	0	0	4	2	4	ESC		
	Practical Subjects									
11	COM2503	Applied Machine Learning Lab	0	0	2	1	2	PCC	SD	
13	COM2501	Computer Vision and LLMs Lab	**0**	0	**2**	**1**	2	PCC	SD	
		TOTAL	**19**	**0**	**14**	**25**	**32**			

Semester VII										
Sl. No.	**Course Code**	**Course Name**	**Credit Structure**				**Contact Hours**	**Type of Course**	**Type of Skills**	**Course Addresses To**
			L	**T**	**P**	**C**				
1	CSEXXXX	Professional Elective – V	3	0	0	3	3	PEC	SD/ EM	
2	CSEXXXX	Professional Elective – VI	3	0	0	3	3	PEC	SD/ EM	
3	XXXXXXX	Open Elective – II	3	0	0	3	3	OEC	SD/ EM /EN	HP

4	CSE7100	Mini Project				4	0	PRW		
		TOTAL	**09**	**0**	**0**	**13**	**09**			

Semester VIII

Sl. No.	**Course Code**	**Course Name**	**Credit Structure**				**Contact Hours**	**Type of Course**	**Type of Skills**	**Course Addresses To**
			L	**T**	**P**	**C**				
1	CSE7300	Capstone Project	-	-	-	10	0	PRW	PIP2004	Capstone Project
		TOTAL	**00**	**00**	**00**	**10**	**0**			

Course Catalogues:

<table>
<tr><td>Course Code:
MAT1001_v03</td><td colspan="3">Course Title: Calculus and Linear Algebra

Type of Course: School Core
Lab Integrated</td><td>L-T- P-C</td><td>2</td><td>1</td><td>2</td><td>4</td></tr>
<tr><td>Version No.</td><td colspan="8">3.0</td></tr>
<tr><td>Course Pre-requisites</td><td colspan="8">Basic Concepts of Limits, Differentiation, Integration</td></tr>
<tr><td>Anti-requisites</td><td colspan="8">NIL</td></tr>
<tr><td>Course Description</td><td colspan="8">The course focuses on the concepts of calculus and linear algebra with reference to specific engineering problems. The course is of both conceptual and analytical type in nature. The lab sessions associated with the course are concerned with acquiring an ability to use the MATLAB software.</td></tr>
<tr><td>Course Objective</td><td colspan="8">The objective of the course is Skill Development of student by using Problem Solving Techniques.</td></tr>
<tr><td>Course Out Comes</td><td colspan="8">On successful completion of the course the students shall be able to:

1) Comprehend the knowledge of applications of matrix principles.
2) Understand the concept of partial derivatives and their applications.
3) Apply the principles of integral calculus to evaluate integrals.
4) Adopt the various analytical methods to solve differential equations.
5) Demonstrate the use of MATLAB software to deal with a variety of mathematical problems.</td></tr>
<tr><td>Course Content:</td><td colspan="8"></td></tr>
<tr><td>Module 1</td><td colspan="2">Linear Algebra</td><td colspan="4"></td><td>10 Classes</td></tr>
<tr><td colspan="9">Review: Types of matrices, elementary transformations, rank of a matrix, normal form, Solution of systems of linear equations: (Homogenous and non-homogenous system) AX = O and AX = B using rank method.

Linear Algebra:
Eigenvalues and Eigenvectors of a real matrix – Characteristic equation – Properties of Eigenvalues and Eigenvectors – Cayley-Hamilton theorem – Diagonalization of matrices – Reduction of a quadratic form to canonical form by orthogonal transformation – Nature of quadratic forms.
Engineering Applications of Linear Algebra.</td></tr>
<tr><td>Module 2</td><td colspan="2">Partial Derivatives</td><td colspan="4"></td><td>10 CLASSES</td></tr>
<tr><td colspan="9">Review: Differential calculus with single variable.</td></tr>
</table>

Partial Derivatives:

Homogeneous functions and Euler's theorem, Total derivative, Change of variables, Jacobians, Partial differentiation of implicit functions, Taylor's series for functions of two variables, Maxima and minima of functions of two variables, Lagrange's method of undetermined multipliers.
Engineering Applications of partial derivatives.

Module 3	**Advanced Integral calculus**			**12 Classes**

Review: Integral calculus for single integrals.

Advanced Integral calculus:

Beta and Gamma functions–interrelation-evaluation of integrals using gamma and beta functions; error function-properties. Multiple Integrals- Double integrals – Change of order of integration – Double integrals in polar coordinates – Area enclosed by plane curves, evaluation of triple integrals-change of variables between Cartesian and cylindrical and spherical polar co-ordinates.
Engineering applications of partial derivatives.

Module 4	**Ordinary Differential Equations**	**Assignment**	**Programming**	**12 Classes**

Review: First order and first-degree Ordinary Differential Equations, Method of separation of variables, Homogeneous and Non- Homogeneous Equations reducible to Homogeneous form.

Linear Differential Equations, Bernoulli's Differential Equation, Exact and Non- Exact Differential Equations, Higher order Differential Equation with constant coefficients and with right hand side of the form e^{ax}, sinax, cosax, $e^{ax}f(x)$, $x^nf(x)$ etc., Linear equations with variable coefficients such as Cauchy Equation and Lagrange's Equation, D-operators and Inverse D- operators, Method of Variation of Parameters.
Engineering applications of differential equations.

List of Laboratory Tasks:

Introductory Task: Introduction to usage of the software and simple programming tasks. [3 Sessions]
Experiment N0 1: Solution of Simple differentiation with single variable and use of chain Rule.
Experiment No. 2: Solution based on application of Tailors' Series using software
Experiment No. 3: Application of Maxima and Minima condition using software.
Experiment No. 4 Computation of different functions for a specific problem
Experiment No. 5 Computation of Area under a curve.
Experiment No. 6 Solution of a set of simultaneous equations in matrix method
Experiment No. 7 Computation of Eigen Values and Eigen Vectors.
Experiment No. 8 Solution of Partial Differential equation

Experiment No. 9 solution using Cauchy Equation and Lagrange's Equation

Targeted Application & Tools that can be used:

The contents of this course has direct applications in most of the core engineering courses for problem formulations, Problem Solution and system Design.

Tools Used: MatLab, Zylink.

Assignment:

List at least 3 sets of Matrix Applications concerning the respective branch of Engineering and obtain the solution using MATLAB.

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

Sankara Rao, Introduction to Partial differential equations, Prentice Hall of India, edition, 2011

B. S. Grewal (2017), Higher Engineering Mathematics by, 44th Edition, Khanna Publishers.

References:

Victor Henner, Tatyana Belozerova, Mickhail Khenner, Ordinary and Partial Differential Equations, CRC Press, Edition, 2013.

Walter Ledermann, Multiple integrals, Springer, 1st edition

Lay, Linear Algebra ansd its applications, 3rd Ed., 2002, Pearson Education India.

Erwin Kreyzig, Advanced Engineering Mathematics, John Wiley and sons, Inc.10th Edition

MatLab usage manual

E-resources/ Web links:

1. https://nptel.ac.in/courses/109104124
2. https://nptel.ac.in/courses/111106051
3. https://nptel.ac.in/courses/111102137
4. https://www.cuemath.com/learn/mathematics/algebra-vs-calculus/
5. https://stanford.edu/~shervine/teaching/cs-229/refresher-algebra-calculus
6. https://math.hmc.edu/calculus/hmc-mathematics-calculus-online-tutorials/linear-algebra/
7. https://www.math.hkust.edu.hk/~maqian/ma006_0607F.html
8. https://www.scu.edu.au/study-at-scu/units/math1005/2022/

Topics relevant to the development of Foundation Skills: **All solution methods**

Topics relevant to development of Employability skills: **Use of Matlab software.**

Course Code: **PHY1002**	**Course Title: Optoelectronics and Device Physics** **Type of Course: 1] School Core & Laboratory integrated**	L-T-P-C	2-0-2-3
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, experimental and analytical skills. The associated laboratory provides an opportunity to validate the concepts taught and enhances the ability to use the concepts for technological applications. The laboratory tasks aim to develop following skills: **An attitude of enquiry, confidence and ability to tackle new problems, ability to interpret events and results, observe and measure physical phenomena, select suitable equipment, instrument and materials, locate faults in systems.**		
Course Out Comes	On successful completion of the course the students shall be able to: CO1: Describe the concepts of semiconductors, magnetic materials and superconductors. CO2: Apply the concept of materials in the working of optoelectronic and magnetic devices. CO3: Discuss the quantum concepts used in advanced microscopy and quantum computers. CO4: Explain the applications of lasers and optical fibers in various technological fields. CO5: Interpret the results of various experiments to verify the concepts used in optoelectronics and advanced devices. **[Lab oriented].**		
Course Objective	The objective of the course is to familiarize the learners with the concepts of "Optoelectronics and device physics "and attain **Skill Development** through **Experiential Learning** techniques		
Course Content:	**The course content are detailed as follows**		
Module 1	**Fundamentals of Materials.**	Assignment	Plotting of magnetization (M) v/s Magnetic field

			(H) for diamagnetic, paramagnetic and ferromagnetic materials using excel/ origin software.
Topics: Concept of energy bands, charge carriers, carrier concentration, concept of Fermi level, Hall effect, Magnetic materials, Superconductors:			
Module 2	**Advanced Devices and applications**	Assignment	Data collection on efficiency of solar cells.
Topics: p-n junctions, Zener diode, transistor characteristics, Optoelectronic devices:, Solar cells, I-V characteristics, and LEDs			
Module 3	**Quantum concepts and Applications**	Term paper	Seminar on quantum computers.
Topics: Planck's quantum theory, applications of Quantum theory: de-Broglie hypothesis, matter waves, properties. de-Broglie wavelength associated with an electron. Heisenberg's uncertainty principle. Schrodinger time independent wave equation. Particle in a box			
Module 4	**Lasers and Optical fibers**	Term paper	Case study on medical applications of Lasers.
Topics: Interactions of radiations with matter, Characteristics of laser, conditions and requisites of laser, Modern day applications of laser: LIDAR, LASIK, Cutting, Welding and Drilling. 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.			
List of Laboratory Tasks: Experiment No. 1: Experimental errors and uncertainty using excel Level 1: Calculation of accuracy and precision of a given data Level 2: propagation of errors in addition, subtraction, multiplication and division.			

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

Level 1: Determination of Wavelength of Laser

Level 2: Finding the particle size of lycopodium powder.

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

Level 1: To determine the proportionality of Hall Voltage and magnetic flux density

Level 2: To determine the polarity of Charge carrier.

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

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

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

Experiment No. 5: To study input and output characteristics of a given Transistor.

Level 1: To determine the input resistance of a given transistor.

Level 2: To determine current transfer characteristics and transistor parameters of a given transistor.

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

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

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

Experiment No. 7: To study the current vs voltage characteristics of CdS photo-resistor at constant irradiance and To measure the photo-current as a function of the irradiance at constant voltage.

Level 1 To study the current vs voltage characteristics of CdS photo-resistor at constant irradiance.

Level 2: To measure the photo-current as a function of the irradiance at constant voltage.

Experiment No. 8: 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. 9: 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. 10: To determine the magnetic susceptibility of a given diamagnetic and paramagnetic substances using Quincke's method.

Level 1: To determine the magnetic susceptibility of a given diamagnetic substance.

Level 2: To determine the magnetic susceptibility of a given paramagnetic substance.

Experiment No. 11: 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. 12: 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.

Targeted Application & Tools that can be used:

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.

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

Engineering Physics by Avadhanalu, Revised edition, S. Chand Publications,2018.

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:

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

Topics relevant to "SKILL DEVELOPMENT": **Fundamentals of materials, Lasers and optical fibers.**

for Skill Development through Participative Learning Techniques. This is attained through the Assignment/ Presentation as mentioned in the assessment component in course handout.

Course Code: **MEC1006**	**Course Title:** **Engineering Graphics** **Type of Course:** **School Core & Theory Only**	**L- T- P- C**	2	0	0	2
Version No.	1.2					
Course Pre-requisites	NIL					
Anti-requisites	NIL					
Course Description	The course is designed with the objective of giving an overview of engineering graphics. It is introductory in nature and acquaints the students with the techniques used to create engineering drawings. The course emphasizes on projection of points, lines, planes and solids and isometric projections.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of "Engineering Graphics" and attain SKILL DEVELOPMENT through Problem solving methodologies.					

Course Outcomes	On successful completion of this course the students shall be able to: Demonstrate competency of Engineering Graphics as per BIS conventions and standards. Comprehend the theory of projection for drawing projections of Points, Lines and Planes under different conditions. Prepare multiview orthographic projections of Solids by visualizing them in different positions. Prepare pictorial drawings using the principles of isometric projections to visualize objects in three dimensions.

Course Content:

Module 1	Introduction to Drawing	Assignment	Standard technical drawing	**02 Sessions**
Topics: **Introduction, drawing instruments and their uses, relevant BIS conventions and standards, Lettering, Line conventions, dimensioning, Selection of drawing sheet size and scale.** **[02 Hours: Comprehension Level]**				
Module 2	Orthographic projections of Points, Straight Lines and Plane Surfaces	Assignment	Projection methods Analysis	**10 Sessions**
Topics: **Introduction, Definitions – Elements of projection and methods of projection, Planes of projection, reference line and conventions adopted. First angle and third angle projections. Projection of Points in all 4 quadrants.** **Projections of Straight Lines (located in first quadrant/first angle projection only): True and apparent lengths, true and apparent Inclinations to reference planes. (No application problems). Projection of Plane surfaces (First angle projection): Regular plane surfaces – triangle, square, rectangle, pentagon, hexagon and circle – in different positions inclined to both the planes using change of position method only.** **[10 Hours: Application Level]**				

Module 3	Orthographic Projections of Solids	Assignment	Multi-view drawing Analysis	**10 Sessions**
Topics: **Introduction, Projection of right regular prisms, pyramids, cone, hexahedron and tetrahedron in different positions (Problems resting on HP only and First angle projection).** **[10 Hours: Application Level]**				
Module 4	Isometric Projections of Solids (Using isometric scale only)	Assignment	Spatial Visualization	**8 Sessions**
Topics: **Introduction, Isometric scale, Isometric projections of right regular prisms, cylinders, pyramids, cones and their frustums, spheres and hemispheres, hexahedron (cube), and combination of 2 solids, conversion of orthographic view to isometric projection of simple objects.** **[8 Hours: Application Level]**				
Text Book: **1.N. D. Bhatt, "Engineering Drawing: Plane and Solid Geometry," Charotar Publishing House Pvt. Ltd.**				
References: **K.R. Gopalakrishna, "Engineering Graphics", Subhash Publishers, Bangalore.** **D. M. Kulkarni, A. P. Rastogi, A. K. Sarkar, "Engineering Graphics with AutoCAD," Prentice Hall.** **D. A. Jolhe, "Engineering Drawing with Introduction to AutoCAD," Tata McGraw Hill.** **Web resources:** **https://nptel.ac.in/courses/112103019**				
Topics relevant to "SKILL DEVELOPMENT": Projection in first and third angle for **SKILL DEVELOPMENT** through **Problem Solving methodologies**. This is attained through the assessment component mentioned in the course handout.				

Course Code: **ENG1002**	**Course Title:** **Technical English** **Type of Course:** **1] School Core** **2] Laboratory integrated**	**L-T-P-C**	1-0-2-2
Version No.	V. 3		
Course Pre-	Intermediate Level English		

requisites	
Course Anti-requisites	NIL
Course Description	Technical English course is designed to equip students with the language skills necessary for effective communication in technical and scientific contexts. The course focuses on the specialized vocabulary, writing styles, and communication techniques used in various technical fields, including engineering and information technology.
Course Objectives	The objective of this course is to develop the learners' **EMPLOYABILITY SKILLS** by using **EXPERIENTIAL LEARNING** and **PARTICIPATIVE LEARNING TECHNIQUES.**
Course Outcomes	**On successful completion of the course, the students shall be able to:** Develop proficiency in using technical vocabulary and terminology. Apply language skills for better speaking skills in technical fields. Write technical descriptions Demonstrate writing skills in writing technical documents such as reports, manuals, and articles.
Course Content:	

Module 1	**Fundamentals of Technical Communication**	**Worksheets& Quiz**	**Vocabulary building**	**9 Classes**

Introduction to Technical English

Differences between Technical English and General English

Technical Writing Basics

Technical Vocabulary

Module 2	**Technical Presentation**	**Presentations**	**Speaking Skills**	**12 Classes**

Introduction

Planning the Presentation

Creating the Presentation

Giving the Presentation

Module 3	**Technical Description**	**Assignment**	**Group Presentation**	**12 Classes**

Product Description

Process Description

User Manuals

Transcoding: Diagrams, charts and images

Module 4	**Technical Writing**	**Assignment**	**Writing Skills**	**12 Classes**

Email Writing

Persuasive and Descriptive Language

Professional Email Etiquette

Writing clear and concise technical emails

Communicating technical information effectively

Technical Report Writing

Types of technical reports (Lab reports, research reports, etc.)

Components of technical reports

Writing an abstract and executive summary

Structure and content organization

Transcoding: diagrams, charts and images

List of Laboratory Tasks:

Module-1

Level 1: Worksheets

Level 2: Worksheets

Module 2

Level 1: Preparing Presentation

Level 2: Giving Presentation (Individual)

Module-3

Level 1: Product Description & User Manual

Level 2: Process Description & Transcoding

Module 4

Level 1: Email Writing

Level 2: Report Writing

Targeted Applications & Tools that can be used:

Flipgrid

Quizzes

Youtube Videos

Podcast

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

Bring out the essence of technical communication with reference to the conventions of technical communication, with examples

Prepare a technical presentation on the importance of Technical Communication and its relevance in a technical field, with real-life examples.

The following individual, as well as group Assignments, will be given to the students.
Presentation
Describing a product/process
Individual Reports

Text Books

Kumar, Sanjay; Pushpalatha. *English Language and Communication Skills for Engineers.* Oxford University Press. 2018.

Brieger, Nick and Alison Paul. *Technical English Vocabulary and Grammar*. https://nmetau.edu.ua/file/technical_english_vocabulary_and_grammar.pdf

Reference Book:

Chauhan, Gajendra Singh, and Kashmiramka, Smita, ***Technical Communication.*** Cengage Publication. 2018.

Sunder Jain. *Technical Report Writing*. Centrum Press, 2013.

John Bowden. "Writing a Report: How to Prepare, Write & Present Really Effective Reports?". 9th Edition 2011

Comfort, Jeremy et. al. 1984. *Business Reports in English*. Cambridge University Press.

Sharma, R.C. and K. Mohan. 2011. Business Correspondence and Report Writing, Fourth Edition. Tata McGraw Hill.

Web Resources:

1:https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=JSTOR1_3307.

2;https://puniversity.informaticsglobal.com:2282/ehost/detail/detail?vid=5&sid=3a77d69b-abe5-4681-b39d-32dfdcb8f4a5%40redis&bdata=JnNpdGU9ZWhvc3QtbGl2ZQ%3d%3d#AN=154223466&db=iih

3: Last,Suzan, et. al. ***Technical Writing Essentials.*** University of Victoria, British Columbia, 2019 (E- Book)

4 Wambui, Tabita Wangare, et al. ***Communication Skills- Volume 1***, LAP LAMBRET, USA, 2012 (E Book)

Topics Relevant to the Development of Employability Skills:

Speaking Skills, Writing Skills, Critical Thinking and Critical Analysis, and Group Communication.

Course Code: PPS 1001	**Course Title:** **Introduction to Soft Skills** **Type of Course:** **Practical Only Course**	L- T-P-C	0	0	2	1
Version No.	1.0					
Course Pre-requisites	Students are expected to understand Basic English. Students should have desire and enthusiasm to involve, participate and learn.					

Anti-requisites	NIL		
Course Description	This course is designed to enable students understand soft skills concepts and improve confidence, communication and professional skills to give the students a competitive advantage and increase chances of success in the professional world. 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 "Soft Skills" and attain SKILL DEVELOPMENT through PARTICIPATIVE LEARNING techniques.		
Course Out Comes	**On successful completion of this course the students shall be able to:** CO1: **Recognize** significance of soft skills CO2: **Illustrate** effective communication while introducing oneself and others CO3: **List** techniques of forming healthy habits CO4: **Apply** SMART technique to achieve goals and increase productivity		
Course Content:			
Module 1	INTRODUCTION TO SOFT SKILLS	**Classroom activity**	**04 Hours**
Topics: Setting Expectations, Ice Breaker, Significance of soft skills, Formal grooming, punctuality			
Module 2	EFFECTIVE COMMUNICATION	**Individual Assessment**	**10 Hours**
Topics: Different styles of communication, Difference between hearing and listening, Effective communication for success, Email etiquette, Self-introduction framework, Video introduction, email-writing, Resume Building- Digital, Video, Traditional.			
Module 3	HABIT FORMATION	**Worksheets & Assignment**	**4 Hours**

Topics: Professional and personal ethics for success, Identity based habits, Domino effect, Habit Loop, Unlearning, standing up for what is right			
Module 4	Goal setting & Time Management	**Goal sheet**	**8 Hours**
A session where students will be introduced to Time management, setting SMART Goals, Introduction to OKR Techniques, Time Management Matrix, steps to managing time through outbound group activity, making a schedule, Daily Plan and calendars (To Do List), Monitoring/charting daily activity			
Targeted Application & Tools that can be used: LMS			
Project work/Assignment: Mention the Type of Project /Assignment proposed for this course			
Individual Assessment LMS MCQ			
The topics related to Skill Development: Communication and professional grooming, Goal setting and presentation for skill development through participative learning techniques. This is attained through assessment component mentioned in course handout.			

Course Code: CSE1004	**Course Title: Problem Solving Using C** **Type of Course: School Core Lab Integrated.**		**L- T-P-C**	1	0	4	3
Version No.	1.0						
Course Pre-requisites	**NIL**						
Anti-requisites	**NIL**						
Course Description	The course is designed to provide complete knowledge of C language. Students will be able to develop logics which will help them to create programs and applications in C. ACAlso by learning the basic programming constructs they can easily switch over to any other language in future.						
Course Object	The objective of the course is to familiarize the learners with the concepts of Problem Solving Using C and attain Employability through Problem Solving Methodologies.						
Course Outcomes	**On successful completion of this course the students shall be able to:** Write algorithms and to draw flowcharts for solving problems Demonstrate knowledge and develop simple applications in C programming constructs Develop and implement applications using arrays and strings Decompose a problem into functions and develop modular reusable code Solve applications in C using structures and Union Design applications using Sequential and Random Access File Processing.						
Course Content:							
Module 1	Introduction to C Language	Quiz	Problem Solving	**9 Hrs.**			

Topics: Introduction to Programming – Algorithms – Pseudo Code - Flow Chart – Compilation – Execution – Preprocessor Directives (#define, #include, #undef) - Overview of C – Constants, Variables and Data types – Operators and Expressions – Managing Input and Output Operations – Decision Making and Branching - Decision Making and Looping.				
Module 2	Introduction to Arrays and Strings	Quiz	Problem Solving	**9 Hrs.**
Topics: **Arrays:** Introduction – One Dimensional Array – Initialization of One Dimensional Arrays – Example Programs – Sorting (Bubble Sort, Selection Sort) – Searching (Linear Search) - Two Dimensional Arrays – Initialization of Two Dimensional Arrays. Example Programs – Matrix operations. **Strings:** Introduction – Declaring and Initializing String Variables – Reading Strings from Terminal – Writing String to Screen – String Handling Functions.				
Module 3	Functions and Pointers	Quiz	Problem Solving	**9 Hrs.**
Topics: **Functions:** Introduction – Need for User-defined functions – Elements of User-Defined Functions: declaration, definition and function call–Categories of Functions – Recursion. **Pointers:** Introduction – Declaring Pointer Variables – Initialization of Variables – Pointer Operators – Pointer Arithmetic – Arrays and Pointers – Parameter Passing: Pass by Value, Pass by Reference.				

Module 4	Structures and Union	Quiz	Problem Solving	**9 Hrs.**
Topics: **Structures:** Introduction – Defining a Structure – Declaring Structure Variable – Accessing Structure Members – Array of Structures – Arrays within Structures – **Union:** Introduction – Defining and Declaring Union – Difference Between Union and Structure.				
Module 5	File handling	Case Study	Problem Solving	**9 Hrs.**
Topics: Files: Defining and Opening a File – Closing a File – Input / Output Operations on File – Random Access Files				
List of Practical **Tasks Lab Sheet 1** **(Module I)** Programs using IO Statements, Conditional Statements and Looping Statements **Lab Sheet 2 (Module II)** **Programs using Arrays and Strings** **Lab Sheet 3 (Module III)** **Programs using Functions and Pointers** **Lab Sheet 4 (Module IV)** **Programs using Structures and Unions** **Lab Sheet 5 (Module V)** **Programs using Files**				
Text Book(s): **1.E. Balaguruswamy, "Programming in ANSI C", 8th Edition, 2019, McGraw Hill Education, ISBN: 978-93-5316- 513-0.**				
Reference Book(s): **Yashwant Kanetkar, Let us C, 17th Edition, BPB Publications, 2020.** **ReemaThareja, "Programming in C", Oxford University Press, Second Edition, 2016.**				

Kernighan, B.W and Ritchie,D.M, "The C Programming language", Second Edition, Pearson Education, 2015 **Schildt Herbert, "C: The Complete Reference", Tata McGraw Hill Education, 4th Edition, 2014.** **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: **ECE2007**	**Course Title:** **Digital Design** **Type of Course:** **Theory &Integrated Laboratory**	**L- T-P- C**	**2**	**0**	**2**	**3**
Version No.	2.0					
Course Pre-requisites	**[1] Elements of Electronics/Electrical Engineering, 2] Basic concepts of number representation, Boolean Algebra**					
Anti-requisites	**NIL**					
Course Description	**The purpose of this course is to enable the students to appreciate the fundamentals of digital logic circuits and Boolean algebra focusing on both combinational and sequential logic circuits. The course emphasizes on minimization techniques for making canonical and low-cost digital circuit implementations. This course deals with analysis and design of digital electronic circuits. The course also creates a foundation for future courses which includes Computer Architecture, Microprocessors, Microcontrollers, and Embedded Systems etc.** The course enhances the Design, Implementation and Programming abilities through laboratory tasks. The associated laboratory provides an opportunity to verify the theoretical knowledge.					
Course Objective	**The objective of the course is to familiarize the learners with the concepts of Digital Design and attain the SKILL DEVELOPMENT through EXPERIENTIAL LEARNING.**					
Course Outcomes	**On successful completion of this course the students shall be able to:** **Describe** the concepts of number systems, Boolean algebra and logic gates. **Apply** minimization techniques to simplify Boolean expressions. **Demonstrate** the Combinational circuits for a given logic Demonstrate the Sequential and programmable logic circuits **Implement** various combinational and sequential logic circuits using gates.					
Course Content:						
Module 1	**Fundamentals of Number systems- Boolean algebra and digital logic**	Application Assignment	Data Analysis task		**06 classes**	
Topics: Review of Number systems and logic gates, Number base conversions, Overview of Boolean functions and simplifications, two, three, four variable K-Maps- Don't care conditions- Both SOP and POS- Universal Gates (NAND & NOR) Implementations. Introduction to HDL.						
Module 2	**Boolean function simplification**	Application Assignment	Data Analysis task		**08 Classes**	
Topics: Introduction to Combinational circuits, Analysis, Design procedure, Binary Adder and Subtractor, Magnitude comparator, Parity generator and checker, Multiplexers-Demultiplexers, Decoders, Encoders and Priority Encoders, HDL Models of combinational circuits.						

Module 3	**Combinational Logic circuits:**	Application Assignment	Programming Task & Data Analysis task	**08 Classes**

Topics:
Introduction to sequential circuits, Storage elements: latches and flip flops, Characteristic tables and equations, excitation table, Analysis of clocked sequential circuits, Mealy & Moore Models of finite state machines - Registers & Counters. HDL Models of Sequential circuits.

List of Laboratory Tasks:

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

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

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

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

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

Experiment No. 6: Study of Flip flops

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

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

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

Targeted Application & Tools that can be used:

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

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

Text Book(s):

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

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

Reference(s):

Reference Book(s):

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

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

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

eBook1: Mano, M. Morris and Ciletti Michael D., *"Digital Design"*, Pearson Education.

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

}

eBook2: Floyd "DIGITAL LOGIC DESIGN" fourth edition- ePub, eBook- [PDF] DIGITAL LOGIC DESIGN FOURTH EDITION FLOYD | abri.engenderhealth.org.

NPTEL Course- NPTEL :: Electrical Engineering - NOC:Digital Electronic Circuits

Digital Logic Design PPT Slide 1 (iare.ac.in)

Lab Tutorial: Multisim Tutorial for Digital Circuits - Bing video

CircuitVerse - Digital Circuit Simulator online

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

Digital Design 5: LOGISIM Tutorial & Demo

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

E-content:

Z. Xin-Li and W. Hong-Ying, "The Application of Digital Electronics in Networking Communication," 2016 Eighth International Conference on Measuring Technology and Mechatronics Automation (ICMTMA), 2016, pp. 684-687, doi: 10.1109/ICMTMA.2016.168.

An encoding technique for design and optimization of combinational logic circuit DipayanBhadra;Tanvir Ahmed Tarique;Sultan Uddin Ahmed;Md. Shahjahan;KazuyukiMurase2010 13th International Conference on Computer and Information Technology (ICCIT)

A. Matrosova and V. Provkin, "Applying Incompletely Specified Boolean Functions for Patch Circuit Generation," 2021 IEEE East-West Design & Test Symposium (EWDTS), 2021, pp. 1-4, doi: 10.1109/EWDTS52692.2021.9581029.

A. Matrosova, V. Provkin and E. Nikolaeva, "Masking Internal Node Faults and Trojan Circuits in Logical Circuits," *2019 IEEE East-West Design & Test Symposium (EWDTS)*, 2019, pp. 1-4, doi: 10.1109/EWDTS.2019.8884434.

Course Code: DES1146	**Course Title:** Introduction to Design Thinking **Type of Course:** Theory	**L-T-P- C**	1	0	0	1
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	NIL					

Course Description		The course aims to introduce students to the fundamental principles and processes of Design Thinking and will learn to apply Design Thinking methodologies to real-world challenges. The course emphasizes empathy, creativity, and collaboration, equipping students with essential skills for successful engineering practice.
Course Objective		This course is designed to develop and familiarize the learners with the concepts of creating thinking and attain Entrepreneurship by using Participative Learning techniques.
Course Outcomes		On successful completion of the course the students shall be able to: Understand the concept and importance of Design Thinking. Differentiate between traditional problem-solving and Design Thinking. Identify the core stages of the Design Thinking process.
Course Content:		All assignments and projects must be developed using the reference materials available from the PU e-resource database – JSTOR, EBSCO, Library OPAC, NPTEL Videos, etc.

Module 1	Introduction to Design Thinking	Visual journal, book of essays, context-specific assignment/project		Visual output generation, by Visual Journal and narrative development.	**3 hours**
	Topic Definition and Introduction to Design Thinking				

Understand the Design Thinking Process					
Module 2	Design Thinking in Action	Visual journal, book of essays, context-specific assignment/project		Visual output generation, by visual journal and narrative development.	**12 hours**
Topics: Introduction to the steps of Design Thinking Process Understand use cases of Design thinking Design Thinking and Research Tools pertaining to Consumer Tech. , Home Tech. , Personal Tech. , Auto Tech. or Extended Reality.					
Targeted Application & Tools that can be used: Design ideation tools like Miro , SCAMPER etc. Research Tools for Human Centric Design using forecasting tools like WGSN Feedback tools like Google Forms , etc. Expert Lectures					
Text Book Thinking Design by S Balaram. New Delhi [India]: Sage Publications Pvt. Ltd. 2010. eBook., Database: eBook Collection (EBSCOhost) https://puniversity.informaticsglobal.com:2284/ehost/detail/detail?vid=6&sid=18ab1f43-1f92-4d02-ae2e-a9c06dc06d8c%40redis&bdata=JnNpdGU9ZWhvc3QtbGl2ZQ%3d%3d#AN=354920&db=nlebk					

References

Design Thinking by Clarke, Rachel Ivy. Series: Library Futures, Vol. 4. Chicago: ALA Neal-Schuman. 2020. eBook., Database: eBook Collection (EBSCOhost)
https://puniversity.informaticsglobal.com:2282/ehost/detail/detail?vid=4&sid=c80a7d79-eda4-4b7e-a0d6-afafe437962b%40redis&bdata=JnNpdGU9ZWhvc3QtbGl2ZQ%3d%3d#AN=2433506&db=nlebk

The Pocket Universal Methods of Design: 100 Ways to Research Complex Problems, Develop Innovative Ideas, and Design Effective Solutions by Bruce Hanington; Bella Martin. Minneapolis: Rockport Publishers. 2017. eBook., Database: eBook Collection (EBSCOhost)
https://puniversity.informaticsglobal.com:2282/ehost/detail/detail?vid=11&sid=f086b8c2-260e-4caa-8c48-d732c21a7724%40redis&bdata=JnNpdGU9ZWhvc3QtbGl2ZQ%3d%3d#AN=1638693&db=nlebk

What Is Design Thinking and Why Is It Important? By Rim Razzouk and Valerie Shute - Review of Educational Research, Vol. 82, No. 3 (September 2012), pp. 330-348 (19 pages), Published by: American Educational Research Association
https://puniversity.informaticsglobal.com:2054/stable/23260048?Search=yes&resultItemClick=true&searchText=design+thinking&searchUri=%2Faction%2FdoBasicSearch%3FQuery%3Ddesign%2Bthinking%26so%3Drel&ab_segments=0%2FSYC-6168%2Ftest&refreqid=fastly-default%3Acb1be24976e25734cb5fc13a8af6fdfb&seq=1#metadata_info_tab_contents

Abductive Thinking and Sensemaking: The Drivers of Design Synthesis by John Kolko, Design Issues, Vol. 26, No. 1 (Winter, 2010), pp. 15-28 (14 pages), Published by: The MIT Press
https://puniversity.informaticsglobal.com:2054/stable/20627839?Search=yes&resultItemClick=true&searchText=design+thinking&searchUri=%2Faction%2FdoBasicSearch%3FQuery%3Ddesign%2Bthinking%26so%3Drel&ab_segments=0%2FSYC-6168%2Ftest&refreqid=fastly-default%3A0b89336ea274d63c010536b01316d7bb&seq=1#metadata_info_tab_contents

Designerly Ways of Knowing: Design Discipline versus Design Science by Nigel Cross, Design Issues, Vol. 17, No. 3 (Summer, 2001), pp. 49-55 (7 pages), Published by: The MIT Press
https://puniversity.informaticsglobal.com:2054/stable/1511801?Search=yes&resultItemClick=true&searchText=design+thinking&searchUri=%2Faction%2FdoBasicSearch%3FQuery%3Ddesign%2Bthinking%26so%3Drel&ab_segments=0%2FSYC-6168%2Ftest&refreqid=fastly-default%3A0d5b607b163f60876ca973ed90e22b1c&seq=1#metadata_info_tab_contents

Course Code: **MAT1003**	**Course Title:** **Applied Statistics** **Type of Course:** School Core	**L T P C**	1	0	2	2
Version No.	3.0					
Course Pre-requisites	None					
Anti-requisites	None					
Course Description	The goal of this course is to provide a firm understanding of probability and statistics by means of a thorough treatment of descriptive statistics, probability and probability distributions keeping in mind the future courses having statistical, quantitative and probabilistic components. The course covers topics such as descriptive statistics, probability, rules for probability, random variables and probability distributions, standard discrete and continuous probability distributions.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of "Applied Statistics" and attain Skill Development Through Problem Solving techniques.					
Expected Outcome:	At the end of this course, students will be in a position to apply the techniques of descriptive statistics effectively interpret the ideas of probability and conditional probability demonstrate the knowledge of probability distributions Compute statistical parameters, correlation and regression, probability and sampling distributions using R software.					
Module 1	**Descriptive Statistics**	Assignment	Coding needed			**10 classes**
Introduction to Statistics, Data and statistical thinking, review of basic statistical parameters, Covariance, Correlation, Types of Measures of Correlation - Karl Pearson's Correlation Coefficient, Spearman Rank Correlation, linear regression, Multi linear regression **.**						
Module 2	**Proba**					**6 classes**

	bility			
Introduction to Probability, Probability of an event, Addition Principle, Multiplication law, Conditional Probability, Total Probability and Baye's theorem with examples				
Module 3	**Random Variables and Probability Distributions**		Coding needed	**14 classes**
Introduction to Random variables, Discrete Random Variables and Continuous Random Variables, Probability Distributions, Probability Mass Function and Probability Density Function, Various Probability distributions, Binomial, **Negative Binominal (Self Study)**, Poisson, Normal and Exponential distributions				
Module 4	**Sampling Theory**		Coding needed	**15 classes**
Introduction to Sampling Theory, Population, Statistic, Parameter, Sampling Distribution, Standard Error. Testing of Hypothesis, Types of Errors, Critical Region, level of Significance. Difference between Parametric and Non-parametric Tests, Large Sample Tests: Z-Test for Single Mean and **Difference of Means (Self Study)**, Small Sample Tests: Student's t-Test for Single Mean and **Difference of Means,** F-Test, Chi-Square Test.				
Targeted Application & Tools that can be used: The objective of the course is to familiarize students with the theoretical concepts of probability and statistics and to equip them with basic statistical tools to tackle engineering and real-life problems.				

Tools used: R Software / MS-Excel

Text Book

Ronald E Walpole, Raymond H Myers, Sharon L Myers, and Keying E Ye, Probability and Statistics for Engineers and Scientists, Pearson Education, 2016.

References

James T. McClave, P. George Benson and Terry Sincich, Statistics for Business and Economics, 2018.

David R. Anderson, Dennis J. Sweeney, Thomas A. Williams, Essentials of Modern Business Statistics with Microsoft Excel, 2020.

David R. Anderson, Dennis J. Sweeney, Thomas A. Williams, Essentials of Statistics for Business and Economics, 2019.

Douglas C. Montgomery and George C. Runger, Applied Statistics and Probability for Engineers, John Wiley and Sons, 2018.

Richard A. Johnson, Miller and Freund's Probability and Statistics for Engineers, 2018.

Kishor S Trivedi, Probability and Statistics with reliability, Queuing and Computer Science Applications, John Wiley & Sons, 2008.

Topics relevant to SKILL DEVELOPMENT: The goal of this course is to provide a firm understanding of probability and statistics by means of a thorough treatment of descriptive statistics, probability and probability distributions keeping in mind the future courses having statistical, quantitative and probabilistic components. The course covers topics such as descriptive statistics, probability, rules for probability, random variables and probability distributions, standard discrete and continuous probability distributions for **Skill Development through Problem Solving methodologies.** This is attained through assessment component mentioned in course handout.

<table>
<tr><td>Course Code: CIV1008</td><td colspan="2">Course Title: Basic Engineering Sciences
Type of Course: Theory Only</td><td>L-T-P-C</td><td>2</td><td>0</td><td>0</td><td>2</td></tr>
<tr><td>Version No.</td><td colspan="7">1.0</td></tr>
<tr><td>Course Pre-requisites</td><td colspan="7">NIL</td></tr>
<tr><td>Anti-requisites</td><td colspan="7">NIL</td></tr>
<tr><td>Course Description</td><td colspan="7">This basic course on engineering science is designed to introduce students to the fields of civil, mechanical and petroleum engineering. Student will be exposed to various fields in civil engineering and different manufacturing techniques in addition to machinery for power production and consumption. Additionally, students will be getting an overview of various sectors of oil & gas industries. This course acquaints students to basics of Industry 4.0 and Construction 4.0. The course aims to enable students to appreciate the multidisciplinary nature of engineering design and operations in the current era with mechanization and digitization transforming every aspect of engineering.</td></tr>
<tr><td>Course Objective</td><td colspan="7">The objective of the course is skill development of student by using Participative Learning techniques.</td></tr>
<tr><td>Course Outcomes</td><td colspan="7">On successful completion of this course the students shall be able to:
1] Recognize the significance of various disciplines in Civil Engineering
2] Discuss the recent evolutions in Civil Engineering
3] Explain various energies, energy generating machineries and energy consumption machineries
4] Describe the fundamental concept and terminology associated with the Petroleum Industry
5] Distinguish between conventional and modern manufacturing techniques.</td></tr>
<tr><td>Course Content:</td><td colspan="7"></td></tr>
<tr><td>Module 1</td><td>Introduction to various fields in Civil Engineering</td><td>Assignment</td><td colspan="3">Case studies on different Civil Engineering Projects</td><td>6 Sessions</td></tr>
<tr><td colspan="8">Topics: Introduction to Civil Engineering: Definition, scope and branches of Civil Engineering, Role of Civil Engineer, Overview of Infrastructure.</td></tr>
</table>

Module 2	**Current Trends and Evolution in Civil Engineering**	Assignment	Article Review	**6 Sessions**
Topics: Mechanization in Construction, Application of Digital Technologies in Planning, Design, execution, monitoring and maintenance of Construction. Overview of Smart Cities.				
Module 3	**Power Production and Consumption Machinery**	Assignment & Quiz	Data Collection	**6 Sessions**
Topics: Energy and its types, Engines and their applications, Pumps-Compressors and their applications.				
Module 4	**Overview of Petroleum Engineering**	Assignment & Quiz	Article Review	**6 Sessions**
Overview of the Petroleum Industry, Importance of Petroleum Engineering, lifecycle of Petroleum products, Classifications of E&P activities: Key difference between Offshore and Onshore, Onshore facilities, offshore platforms, Digitization of petroleum engineering				
Module 5	**Industry 4.0**	Assignment & Quiz	Data Collection	**6 Sessions**
Topics: Conventional manufacturing process: Metal forming, metal removal and metal joining process. Modern Manufacturing process: 3D Printing / Additive Manufacturing.				
Targeted Application & Tools that can be used: Application Areas include design and implementation of Smart City projects, Infrastructure maintenance, Power production, IC engines, Electric vehicles, onshore and offshore exploration and production activities				
Project work/Assignment:				
Assignment 1: Collect data and prepare report on various Mega Projects in Civil Engineering Assignment 2: Review Articles on current evolutions in Civil Engineering. Assignment 3: Collect data related to renewable energy generation (Wind, Solar) Assignment 4: Prepare an energy consumption chart for a compressor or pumps. Assignment 5: Prepare a report on role of 3D printing across various industries. Assignment 6: Prepare an assignment on geopolitical influence on oil and gas industries.				

Text Book:

T1. Elements of Civil and Mechanical Engineering, L.S. Jayagopal & R Rudramoorthy, Vikas Publishers

T2. Elements of Mechanical Engineering, by VK Manglik

T3. Fundamentals of Oil & Gas Industry for Beginners by Samir Dalvi, Notion Press; 1st edition

References

K.P. Roy, S.K. Hajra Choudhury, Nirjhar Roy, "Elements of Mechanical Engineering", Media Promoters and Publishers Pvt Ltd, Mumbai.

Nontechnical Guide to Petroleum Geology, Exploration, Drilling & Production by Norman J. Hyne, PennWell Books; 3rd Revised edition

Web-resources:

Basic Civil Engineering
https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=2706932&site=ehost-live

Post-parametric Automation in Design and Construction
https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=1155197&site=ehost-live

Smart Cities : Introducing Digital Innovation to Cities
https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=1993146&site=ehost-live

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

Mechanical Engineering
https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO106_REDO_1705

Additive Manufacturing: Opportunities, Challenges, Implications
https://search.ebscohost.com/login.aspx?direct=true&db=nlebk&AN=1134464&site=ehost-live

Society of Petroleum Engineers (SPE)
https://www.spe.org/en/

PetroWiki: A comprehensive online resource created by the Society of Petroleum Engineers that provides information on various aspects of petroleum engineering.
https://petrowiki.spe.org/PetroWiki

Rigzone: A resource for news and information about the oil and gas industry, including job postings and industry trends.
https://www.rigzone.com/

Topics relevant to the development of SKILLS:

Engines-Turbines and their applications.

Mechanization in Construction.

Digitization in Petroleum Industries

Course Code: CSE1006	**Course Title:** Problem Solving using JAVA **Type of Course:** **Integrated**	**L- P-C**	1	4	3
Version No.	2.0				
Course Pre-requisites	CSE1004 – Problem Solving Using C				
Anti-requisites	Nil				

Course Description	This course introduces the core concepts of object-oriented programming. This course has theory and lab component which emphasizes on understanding the implementation and application of object-oriented programming paradigm. It helps the student to build real time secure applications by applying these concepts and also for effective problem solving. The students interpret and understand the need for object oriented programming to build applications.			
Course Objective	The objective of the course is to familiarize the learners with the concepts of Problem-Solving using JAVA and attain **SKILL DEVELOPMENT** through **EXPERIENTIAL LEARNING** techniques			
Course Out Comes	**On successful completion of the course the students shall be able to:** **C.O. 1:** Describe the basic programming concepts. [Knowledge] **C.O. 2:** Apply the concept of classes, objects and methods to solve problems. [Application] **C.O. 3:** Apply the concept of arrays and strings. [Application] **C.O. 4:** Implement inheritance and polymorphism building secure applications. [Application] **C.O. 5:** Apply the concepts of interface and error handling mechanism. [Application]			
Course Content:				
Module 1	**Basic Concepts of Programming and Java**	Assignment	Data Collection/Interpretation	**12 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**	Case studies / Case let	Case studies / Case let	**12 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**	Quiz	Case studies / Case let	**14 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**	Quiz	Case studies / Case let	**14 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**	Quiz	Case studies / Case let	**14 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.				
List of Laboratory Tasks:				

P1 - Problem Solving using Basic Concepts.
P2 - Problem Solving using Basic Concepts and Command Line Arguments.
P3 - Programming assignment with class, objects, methods and Constructors.
P4 - Programming assignment with method overloading.
P5 - Programming assignment with constructor overloading.
P6 - Programming assignment with Static members and static methods.
P7 - Programming assignment with Nested classes.

P8 - Programming assignment using Arrays.
P9 - Programming assignment using Strings.
P10 - Programming assignment using String Builder.

P11 - Programming assignment using Inheritance and super keyword.
P12 - Programming assignment using Method overriding and Dynamic method invocation.
P13 - Programming assignment using Final keywords.
P14 - Programming assignment using Abstract keywords.
P15 - Programming assignment using Interface.
P16 - Programming assignment using Interface.
P17 - Programming assignment CharacterStream Classes
P18 - Programming assignment Read/Write Operations with File Channel

Targeted Application & Tools that can be used : JDK /eclipse IDE/ net Beans IDE.

Text Book

T1 Herbert Schildt, "The Complete Reference Java 2", Tata McGraw Hill Education.

References

R1: Cay S Horstmann and Cary Gornell, "CORE JAVA volume I-Fundamentals", Pearson
R2: James W. Cooper, "Java TM Design Patterns – A Tutorial", Addison-Wesley Publishers.

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":

Static Polymorphism
Method overloading, constructors
constructor overloading
this keyword
static keyword and Inner classes
Inheritance and Polymorphism.

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

ENG2001	**Advanced English**	**L- T- P- C**	1	0	2	2
Version No.	1.3					

Course Pre-requisites	ENG1002 Technical English			
Anti-requisites	**NIL**			
Course Description	The course emphasizes on technical communication at advanced level by exploring critical reading, technical presentation and review writing. The purpose of the course is to enable learners to review literature in any form or any technical article and deliver technical presentations. Extensive activities in practical sessions equip to express themselves in various forms of technical communications. Technical presentations and the module on career setting focus on learners' area of interests and enhance their English language writing skills to communicate effectively.			
Course Out Come	On successful completion of the course the students shall be able to: Develop a critical and informed response reflectively, analytically, discursively, and creatively to their reading. Communicate effectively, creatively, accurately and appropriately in their writing. Deliver technical presentations Design resume and create professional portfolio to find a suitable career			
Course Content: Theory				
Module 1	**Critical Reasoning and Writing**	**Writing Essays**	**Critical Reading**	**4 Classes**
Topics: A Catalog of Reading Strategies The Myth of Multitasking A Guide to Writing Essays Speculating about Causes or Effects Is Google Making Us Stupid (Self Study)				
Module 2	**Technical Presentation**	**Presentation**	**Oral Skills**	**3 Classes**
Topics: Planning the presentation Creating the presentation Giving the presentation				
Module 3	**Writing Reviews**	**Prezi**	**Review Writing**	**4 Classes**

Topics:

Review Writing

Short film reviews

Advanced English Grammar (Self Study)

Module 4	**Starting your Career**	**Online Writing Lab**	**Writing Skills**	**4 Classes**

Topics:

Preparing a Resume

Writing Effective Application Letter

Creating a Professional Portfolio

Course Content: Practical Sessions

Module 1	**Critical Reasoning and Writing**	**8 Classes**

Reading and Analyzing

Level 1 – Annotation

Level 2 - Assumptions

Writing Narrative Essays

Level 1 – Draft 1

Level 2 – Draft 2

Module 2	**Technical Presentation**	**10 Classes**

Fishbowl

In Fishbowl, students form concentric circles with a small group inside and a larger group outside. Students in the inner circle engage in an in-depth discussion, while students in the outer circle listen and critique content, logic, and group interaction.

Level 1 – within group

Level 2 – Among 2 group

Technical Group Presentation

Module 3	**Writing Reviews**	**Classes**

Practice Worksheets

Level 1 – Eliminating the Passive Voice

Level 2 – Simple, compound and complex sentences

Writing Short Film Reviews

Module 4	**Starting your Career**	**Classes**

Collaborative Project Job search and writing report Writing Resume		
Module 1-4	**Academic Journal**	**2 Classes**
Academic Journal Writing Level 1- Mid Term Level 2 – End Term		
Targeted Application & Tools that can be used: Writing reports, Review writing, Group Discussion, Dyadic interviews, Grammarly.com		
Project work/Assignment:		
Academic Journal – Assignment In Academic Journal (CIJ), students compile task and activities completed in each module and submit to the instructor at the middle and end of the semester.		
References Hering, Heik. *How to Write Technical Reports: Understanding Structure, Good Design, Convincing Presentation*. Springer. Johnson, Richard. (2010) *Technical Communication Today.* Pearson, 2015 Rice B. Adelrod, Charles R. Cooper and Ellen C. Carillo. (2020) *Reading Critically Writing Well: A Reader and Guide*. Beford/St. Martin's Macmillan Learning, New York. The Princeton Review. (2010) *MCAT Verbal Reasoning & Writing.* The Princeton Review, Inc. https://www.hitbullseye.com/Strong-and-Weak-Arguments.php Accessed on 10 Dec 2021 https://www.inc.com/guides/how-to-improve-your-presentation-skills.html Accessed on 10 Dec 2021		

Course Code: **PPS 1012**	**Course Title: Enhancing Personality through Soft Skills** Type of Course: Practical Only Course	L- T - P- C	0	0	2	1
Version No.	1.0					
Course Pre-	Students are expected to understand Basic English. Students should have desire and enthusiasm to involve, participate and learn.					

requisites			
Anti-requisites	NIL		
Course Description	This course is designed to enable students understand soft skills concepts and improve confidence, communication and professional skills to give the students a competitive advantage and increase chances of success in the professional world. 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 "Personality Development through Soft Skills" and attain SKILL DEVELOPMENT through PARTICIPATIVE LEARNING techniques.		
Course Out Comes	**On successful completion of this course the students shall be able to:** CO 1 **Identify the stages of** team formation (Remember) CO 2 **Demonstrate** effective presentation skills (Apply) CO3 **Prepare** professional social media profile (Apply)		
Course Content:			
Module 1	Team Building	Classroom and outbound team building activities.	6 Hours
Topics: Importance of team, stages of Team Formation, Trust and collaboration, Virtual Team. **Activity: Team Building outbound activity**			

Module 2	Art of Questioning	Role plays	**4 Sessions**
Topics: Framing Questions, 5W1H Technique, Open-ended and Close-ended questions, Funnel technique, Probing questions, Leading questions			
Module 3	Presentation Skills	Practice and evaluation of individual / group presentation	**10 Sessions**
Topics: Content development, Delivery techniques, Audience Analysis, Timing and Pacing, handling questions and challenges. **Activity:** Individual presentations and team presentation			
Module 4	Professional Brand Building	Brand Framework Activity	4 Sessions
Topics: Personal brand definition, Crafting a compelling LinkedIn profile, Networking strategies. **Activity:** Create a basic online profile			
Module 5	Recap / Revision /Feedback Session		1 Session

Targeted Application & Tools that can be used:

TED Talks
You Tube Links
Activities

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

Presentation Evaluation

Targeted Application & Tools that can be used:

TED Talks
YouTube Links
Videos by L&D Team shared on Edhitch/YouTube.com
LMS

Assignments proposed for this course

Evaluation on Presentation
Assignment on LinkedIn Post

YouTube Links: https://youtu.be/z__jxoczNWc (Steve Jobs Introducing the iPhone 4 in June 2010)

References

"Talk Like TED - The 9 Public-Speaking Secrets of the World's Top Minds" By Carmine Gallo St. Martin's Press Copyright © 2014 Carmine Gallo All rights reserved. ISBN: 978-1-250-04112-8

"The Presentation Secrets of Steve Jobs: How to Be Insanely Great in Front of Any Audience" MP3 CD – Import, 22 April 2014

"The Definitive Book of Body Language: The Hidden Meaning Behind People's Gestures and Expressions" Hardcover – Illustrated, 25 July 2006

"Crucial Conversations: Tools for Talking When Stakes Are High" Paperback – Import, 1 July 2002

Web links:

https://www.wordstream.com/blog/ws/2014/11/19/how-to-improve-presentation-skills
https://www.cbs.de/en/blog/15-effective-presentation-tips-to-improve-presentation-skills/

https://hbr.org/2022/05/the-art-of-asking-great-questions

Topics relevant to development of "SKILL": Art of Presentation, Team building, Art of questioning and Personal Branding for Skill Development through Participative Learning Techniques. This is attained through assessment component mentioned in course handout.

Course Code: EEE1007	**Course Title: Basics of Electrical and Electronics Engineering.** **Type of Course: Engineering Science - Theory & Integrated Laboratory**	**L-T-P-C**	3	0	2	4
Version No.	1.0					
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 emphasis on the characteristics and applications of Electrical and Electronics devices, working, analysis and design of electrical circuits using both active & passive components, fundamentals of electrical machines and basics of transistors and its application. The associated laboratory provides an opportunity to validate the concepts taught and enhances the ability to visualize the 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.					

Course Outcomes	**On successful completion of this course the students shall be able to:** **Explain** basic law**s** of Electrical Engineering to compute voltage, currents and other parameters in the circuits. **Discuss** various fundamental parameters appearing in the characteristics of semiconductor devices and their applications. **Summarize** the operations of different biasing configurations of BJTs and amplifiers. **Summarize**the performance characteristics and applications of various electrical Machines. **Demonstrate** the working of electrical machines to observe performance characteristics **Demonstrate** the working of electronic circuits to obtain the V-I Characteristics of various semiconductor devices.			
Course Content:				
Module 1	**Introduction to Electrical Circuits**	Assignment/ Quiz	Numerical solving Task	**10 Sessions**
DC Circuits: Concept of Circuit and Network, Types of elements, Network Reduction Techniques- Series and parallel connections of resistive networks, Star–to-Delta Transformations, Mesh Analysis, Numerical examples. **AC Circuits:** Fundamentals of single phase circuits - Series RL, RC and R-L-C Circuits, Concept of active power, reactive power and Power factor, Numerical examples. Introduction to three phase system and relation between line and phase values in Star & Delta connection, Numerical examples.				
Module 2	**Semiconductor and Diode applications**	Assignment/ Quiz	Memory Recall based Quizzes	**11 Sessions**
Mass Action Law, Charge densities in a semiconductor, Types of SC, Junction diodes -Ideal and practical behaviour, Modelling the Diode Characteristic, and Diode applications like rectifiers, Clipping and clamping circuits. Zener diode, characteristics and its applications like voltage regulator.				
Module 3	**Fundamentals of Electrical Machines**	Assignment/ Quiz	Memory Recall-based Quizzes	**12 Sessions**
Electrical Machines: Single phase transformers: principle of operation and EMF equation, Numerical examples. DC Motor: principle of operation, Back EMF, torque equation, Numerical examples. AC Motor: Principle operation of Induction Motors and its Applications. **Special Machines:** Introduction to special electrical machines and its applications.				
Module 4	**Transistors and its Applications**	Assignment/ Quiz	Numerical solving Task	**12 Sessions**
Transistor characteristics, Current components, BJT Configurations (CB, CC, CE configurations) and their current gains. Operating point, Biasing & stabilization techniques: Fixed Bias, Voltage divider bias and its stability factor and load line analysis. Single and multistage amplifier, Darlington pair. JFET (Construction, principal of Operation and Volt –Ampere characteristics). Pinch- off voltage, Comparison of BJT and FET. MOSFET (Construction, principal of Operation and symbol), MOSFET characteristics in Enhancement and Depletion modes.				
List of Laboratory Tasks: **Experiment No 1:** Verification of KVL and KCL for a given DC circuit. **Level 1:** Study and Verify KVL and KCL for the given electrical Circuit. **Level 2:** For the same circuit considered in level 1, perform the simulation using NI LabVIEW/Multisim/MATLAB. **Experiment No 2:** Analyse AC series circuits – RL, RC and RLC . **Level 1:** Conduct an experiment to perform and verify the impedance, current and power of Series RL and RC circuits **Level 2:**				

Experiment No 3: Calculation of power and power factor of the given AC Circuit.
Level 1: Conduct an experiment to measure the power and power factor for given resistive load.
Level 2: Conduct an experiment to measure the power and power factor for given inductive load.

Experiment No 4: Perform the experiments on given Transformer.
Level 1: Verify the EMF equation of a transformer and compute the voltage transformation ratio.
Level 2: Study the effect of load on the secondary side of the transformer and verify the EMF equation under load conditions.

Experiment 5: Load test on DC shunt motor
Level 1: Conduct load test on DC shunt motor and find its efficiency at different loads
Level 2: Conduct load test on DC shunt motor and plot the performance characteristics.

Experiment 6: Study of PN-Junction Diode Characteristics in Forward and Reverse Bias Conditions.
Level 1: Carry out an experiment to plot VI Characteristics and hence find the cut-in voltage on forward characteristics for the Silicon P-N Junction diode.
Level 2: Carry out an experiment to plot VI Characteristics of Zener diode and hence find the zener voltage on reverse characteristics for the Silicon P-N Junction zener diode.

Experiment 7: To observe the output waveform of half wave and full wave rectifier circuit and compute ripple factor and efficiency
Level 1: Identify the components required for a rectifier circuit, rig up the circuit, and sketch the output waveforms without filter.
Level 2: Rig up the rectifier circuit with RC filter, observe the output waveforms, determine the efficiency and ripple factor.

Experiment 8: To construct clipping and clamping circuits for different reference voltages and to verify the responses.
Level 1: Identify the components required for building a Clipper / Clamper circuit. Rig up the circuit according to the circuit diagram given and sketch the output waveform.
Level 2: Given a sinusoidal input of 10 V p-p, implement a positive / negative clipper with output clipped at 2 V.

Experiment 9: To calculate various parameters of emitter follower circuit using BJT
Level 1: Identify the components required to implement an emitter follower circuit. Rig up the circuit and observe the variations in output waveform with respect to the variations in input waveform.
Level 2: Determine the values of Z_{in} input impedance and Z_{out} output impedance for Emitter Follower.

Experiment 10: To Implement RC Coupled amplifier using a BJT and sketch the frequency response.
Level 1: Identify the components required to implement an RC coupled amplifier circuit. Rig up the circuit and sketch the frequency response.
Level 2: From the frequency response curve determine the value of the mid band gain and the bandwidth.

Targeted Application & Tools that can be used:

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

Text Book(s):

Kothari D. P. &Nagrath I. J., "Basic Electrical and Electronics Engineering", Tata McGraw-HillEducation.
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.
A.P.Malvino, Electronic Principles,7thEdition, Tata McGraw Hill,2007

J. Millman, C. C. Halkias and C. D. Parikh, “Millman’s Integrated Electronics”, McGraw Hill Education, 2nd Edition.
Basics of Electrical & Electronics Laboratory Manual.

Reference Book (s):

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

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

https://presiuniv.knimbus.com/user#home
https://www.digimat.in/nptel/courses/video/108105112/L01 “Fundamentals of Electrical Engineering-Basic Concepts, Examples”
Seminar Topic: https://nptel.ac.in/courses/108/105/108105153/ “Electrical Measurements”
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
Video lectures on “Analog Electronics” by Prof. S.C. Dutta Roy, IIT Delhi
https://nptel.ac.in/courses/108/102/108102095/
Video lectures on “Diodes”, by Prof.ChitralekhaMahanta, IIT Guwahati,
https://nptel.ac.in/courses/117/103/117103063/
“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 CurrentVoltageModeling," 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
F. Bonet, O. Aviñó-Salvadó, M. Vellvehi, X. Jordà, P. Godignon and X. Perpiñà, "Carrier Concentration Analysis in 1.2 kV SiCSchottky Diodes Under Current Crowding," in IEEE Electron DeviceLetters, vol. 43, no. 6, pp. 938-941, June 2022, doi: 10.1109/LED.2022.3171112. https://ieeexplore-ieeeorg- presiuniv.knimbus.com/document/9764749
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
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 Electrical and electronics circuit parameters, performance operation of Machines, and semiconductor devices for Skill Development through Experiential Learning techniques. This is attained through assessment component mentioned in course plan.

Course Code: LAW1007	**Course Title: Indian Constitution and Professional Ethics for Engineers** **Type of Course: Theory**	**L-T-P-C**	1			0
Version No.						
Course Prerequisites						

Anti-requisites	NIL
Course Description	The purpose of this course is to introduce the students to the theory, concepts and practice of Constitution of India which is the law of the land. Further, the course aims at acquainting the students with basic approaches and methodologies to analyse and decide on the ethical dilemma in the field of engineering. The course is both conceptual and analytical. comprehend the conceptual and legal framework of Constitution of India. Ethics and values are very beautifully weaved into the tapestry of the Indian Constitution. Therefore, the course provides an introduction to the essential theoretical basis of engineering ethics and its application through a range of industry rele responsibility for safety and risks, responsibility of employers, rights of engineers etc.
Course Objective	To introduce the students to the conceptual framework of Constitution of India and engineering ethics. To enhance the practical knowledge on responsibility of engineering professionals as citizens of India. To acquaint the student with the relevant contemporary issues surrounding constitutional values and professional ethics. To orient the students about the ethical concepts and frameworks enabling them to identify the codes and moral values relevant to the professional world.
Course Outcomes	**On successful completion of this course the students shall be able:** To understand foundational Indian constitutional law concepts and values. To identify the different pillars of democracy and their functions. society and the employer.

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

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

References:

Durga Das Basu, Commentary on the Constitution of India, 9th Edition, Lexis Nexis, 2019.
Rowan, John, and Zinaich Jr., Ethics for the Professions, Wadsworth, 2003.
R.C. Sekhar, Ethical Choices in Business, Response Books, Sage Publications, 1997.

Course Code: ECE 2010	**Course Title: Innovative Projects using Arduino**	**L- T- P- C**	-	-	-	1
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	NIL					
Course Description	This course is designed to provide an in-depth understanding of Arduino microcontrollers and their application in various real time projects involving sensors. Throughout the course, students will learn the fundamentals of Arduino programming and gain hands-on experience with a wide range of sensors. Students will explore how to connect and interface sensors with Arduino boards, read sensor data, and use it to control various output devices This course is suitable for beginners who are interested in exploring the world of electronics and developing practical applications using Arduino and sensors.					
Course Objective	The objective of the course is Employability Skills of student by using PARTICIPATIVE LEARNING techniques.					
Course Outcomes	**On successful completion of the course the students shall be able to** Explain the main features of the Arduino prototype board Demonstrate the hardware interfacing of the peripherals to Arduino system. Understand the types of sensors and its functions					

	Demonstrate the functioning of live projects carried out using Arduino system.			
Course Content:				
Module 1	**Basic concepts of Arduino**	Hands-on	Interfacing Task and Analysis	**4 Sessions**
Topics: Introduction to Arduino, Pin configuration and architecture, Device and platform features, Concept of digital and analog ports, Familiarizing with Arduino Interfacing Board, API's , Introduction to Embedded C and Arduino platform, Arduino Datatypes and variables, Arduino i/o Functions, Arduino Communications, Arduino IDE, Various Cloud Platforms.				
Module 2	**Sensory Devices**	Hands-on	Interfacing Task and Analysis	**4 Sessions**
Arduino Sensors: Humidity Sensor, Temperature Sensor, Water Detector / Sensor, PIR Sensor, Ultrasonic Sensor, Connecting Switches and actuators, sensor interface with Arduino. Introduction to 3D Printer: 3D Printer technology and its working Principles, Applications. Introduction to online Simulators: Working with Tinkercad Simulator.				
Topics: Types of Arduino boards, sensors, 3D Printer				
Targeted Application & Tools that can be used: **Application Area:** Home Automation, Environmental Monitoring, Agriculture and Farming, Industrial Automation, Internet of Things (IoT), Robotics, Wearable Devices, Security Systems, Education and Learning. These are just a few examples of the many application areas where Arduino and sensors can be applied. The flexibility and affordability of Arduino, combined with the wide range of sensors available, allow for endless possibilities in creating innovative projects. **Professionally Used Software:** students can use open SOURCE Softwares Arduino IDE and Tincker CAD				
Project work/Assignment:				
1. Projects: At the end of the course students will be completing the project work on solving many real time issues. **2. Book/Article review: At the end of each module a book reference or an article topic will be given to an individual or a group of students. They need to refer the library resources and write a report on their understanding about the assigned article in appropriate format. Presidency University Library Link .**				

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

Textbook(s):

Monk Simon "Programming Arduino: Getting Started with Sketches", Mc Graw Hill Publications Second Edition

References

Reference Book(s)

1. **Neerparaj Rai "Arduino Projects for Engineers" BPB publishers,first edition, 2016.**

2. **Ryan Turner "Arduino Programming " Nelly B.L. International Consulting Ltd. first edition,2019.**

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

Arduino trending Projects < https://www. https://projecthub.arduino.cc/>
Introduction to Arduino < https://onlinecourses.swayam2.ac.in/aic20_sp04/preview>
Case studies on Wearable technology< https://www.hticiitm.org/wearables>

E-content:

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

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

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

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

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

Course Code: CSE1506 **Course Title: Data Communications and Computer Networks** **Type of Course: Program Core Theory–Laboratory integrated**	**L-T-P-C**	3	0	0	3

Version No.

1.0

Course Pre- requisites

Digital Design

Anti-requisites

NIL

Course Description

The objective of this course is to provide knowledge in data communications and computer networks, its organization and its implementation, and gain practical experience in the installation, monitoring, and troubleshooting of LAN systems. .

The associated laboratory is designed to implement and simulate various networks using

Cisco packet tracer, NS2. All the lab exercises will focus on the fundamentals of creating multiple networks, topologies and analyzing the network traffics.

Course Objective

The objective of the course is to familiarize the learners with the concepts of Data Communications and Computer Networks and attain **Employability** through **Problem Solving** Methodologies.

Course Out Comes

On successful completion of the course, the students shall be able to: 1] I

llustrate the Basic Concepts Of Data Communication and Computer Networks. 2] Analyze the functionalities of the Data Link Layer.

3] Apply the Knowledge of IP Addressing and Routing Mechanisms in Computer Networks. 4] Demonstrate the working principles of the Transport layer and Application Layer.

Course Content:

Module 1	Introduction and Physical Layer- CO1	Assignment	Problem Solving	07 Classes

Introduction to Computer Networks and Data communications, Network Components – Topologies, Transmission Media –Reference Models -OSI Model – TCP/IP Suite.

Physical Layer -Analog and Digital Signals – Digital and Analog Signals – Transmission - Multiplexing and Spread Spectrum.

Module 2	Reference Models and Data Link Layer – CO2	Assignment	Problem Solving	**7 Classes**

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 – CO 3	Assignment	Problem Solving	**10 Classes**

Network Layer Services - Network Layer Services, Switching Techniques, IP Addressing methods- IPv4 IPV6 – Subnetting. Routing, - Distance Vector Routing – RIP-BGP-Link State Routing –OSPF-Multi cast Routing-MOSPF-DVMRP – Broad Cast Routing. EVPN-VXLAN, VPLS, ELAN.

Module 4	Transport and Application Layer - CO3	Assignment	Problem Solving	**10 Classes**

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.

List of Laboratory Tasks:

Lab sheet -1, M-1, 3 [2 Hours]

Experiment No 1:

Level 1: Study of basic network commands and network configuration commands.

Lab sheet -2, M-1[2 Hours] Experiment No 1:

Level 1: Identify and explore Network devices, models and cables. Introduction to Cisco packet tracer. Experiment No. 2:

Level 2 – Create various network topologies using a cisco packet tracer.

Lab sheet -3, M-2,3 [2 Hours]

Experiment No. 1:

Level 2 - Basic Configuration of switch/router using Cisco packet tracer. Experiment No. 2:

Level 2 -Configure the privilege level password and user authentication in the switch/router.

Lab sheet – 4, M-3 [2 Hours] Experiment No. 1:

Level 2 - Configure the DHCP server and wireless router and check the connectivity

Lab sheet – 5, M-3 [2 Hours] Experiment No. 1:

Level 2 - Configure the static routing in the Cisco packet tracer. Experiment No. 2:

Level 2 - Configure the dynamic routing protocol in the Cisco packet tracer.

Lab sheet – 6, M-4 [2 Hours]

Experiment No. 1: Configuration of DNS Server with Recursive & Integrative approach in Cisco packet tracer.

Lab sheet – 7, M-4 [2 Hours] Experiment No. 1:

Configure the telnet protocol in the router using the Cisco packet tracer. Lab sheet – 8, M-4[2 Hours]

Experiment No. 1:

Level1- Introduction to NS2 and basic TCL program. Lab sheet – 9, M-4 [2 Hours]

Experiment No. 1:

Level 1: Simulate three node Point to point network using UDP in NS2. Experiment No. 2:

Simulate transmission of Ping message using NS2. Lab sheet – 10, M-4[2 Hours]

Experiment No. 1:

Simulate Ethernet LAN using N-node in NS2. Experiment No. 2:

Simulate Ethernet LAN using N-node using multiple traffic in NS2 Lab sheet –11, M-3,4 [2 Hours]

Experiment No. 1:

Level 1- Introduction to Wire Shark. Experiment No. 2:

Level 2- Demonstration of packet analysis using wire shark.

Lab sheet –12, M-1,2,3 [2 Hours]

Experiment No. 1:

Level 2- Demonstration of switch and router configuration using real devices

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

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

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

Text Book

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

References

uter Networking: A Top-Down Approach", Eighth Edition, James F. Kurose, Keith W. Ross, Pearson publication, 2021.
n Stallings, Data and Computer Communication, 8th Edition, Pearson Education, 2007.
. 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 https://puniversity.informaticsglobal.com/login

Course Code:	**Course Title:** Environmental Science	**L- T- P- C**	**1**	**0**	**2**	**0**
CHE1018	**Type of Course:** School Core- Theory and Lab	**Contact hours**	**1**	**0**	**2**	**3**
Course Pre-requisites	NIL					
Anti-requisites	**NIL**					
Course Description	This course emphasizes the need to conserve biodiversity and adopt a more sustainable lifestyle by utilizing resources in a responsible way. Topics covered include basic principles of ecosystem functions; biodiversity and its conservation; human population growth; water resources, pollution; climate change; energy resources, and sustainability; Sustaining human societies, policies, and education. **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 "Environmental Science" and attain SKILL DEVELOPMENT through EXPERIENTIAL LEARNING techniques.**					
Course Outcomes	On successful completion of this course the students shall be able to: Appreciate the historical context of human interactions with the environment and the need for eco-balance. Describe basic knowledge about global climate change with particular reference to the Indian context. Understand biodiversity and its conservation Develop an understanding on types of pollution and ways to protect the environment Learn about various strategies on Global environmental management systems					

Course Content:				
Module 1	**Humans and the Environment**	Assignment	Data Collection	**01 class**

Topics: The man-environment interaction: Mastery of fire; Origin of agriculture; Emergence of city-states; Great ancient civilizations and the environment.

Self-learning topics: Humans as hunter-gatherers; Industrial revolution and its impact on the environment; Environmental Ethics and emergence of environmentalism.

Module 2	**Natural Resources and Sustainable Development**	Assignment		**03 Classes**

Topics:

Overview of natural resources: Definition of resource; Classification of natural resources- biotic and abiotic, renewable and non-renewable. **Water resources**: Types of water resources- fresh water and marine resources;

Soil and mineral resources: Important minerals; Mineral exploitation Soil as a resource and its degradation.

Energy resources: Sources of energy and their classification, renewable and non-renewable sources of energy; Advantages and disadvantages.

Self- learning topics: Availability and use of water resources; Environmental impact of over-exploitation, issues and challenges.; Environmental problems due to extraction of minerals and use; Sustainable Development Goals (SDGs)- targets, indicators, and challenges for SDGs.

Module 3	**Environmental Issues: Local, Regional and Global**	Case study		**02 Classes**

Topics:

Environmental Pollution: Types of Pollution- air, noise, water, soil, municipal solid waste, hazardous waste; Trans-boundary air pollution; Acid rain; Smog.

Land use and Land cover change: land degradation, deforestation, desertification, urbanization. Global change: Ozone layer depletion; Climate change

Self -learning topics: Environmental issues and scales

Module 4	**Conservation of Biodiversity and Ecosystems**	Assignment		**02 Classes**

Topics:

Biodiversity-Introduction, types, Species interactions, Extinct, endemic, endangered and rare species, Threats to biodiversity: Natural and anthropogenic activities.

Self-learning topics: Mega-biodiversity, Hot-spots, Major conservation policies. Biodiversity loss: past and current trends, impact.

Module 5	**Environmental Pollution and Health**	Case study		**03 Classes**

Topics:

Pollution, Definition, point and nonpoint sources of pollution, **Air pollution**- sources, major air pollutants, health impacts of air pollution.

Water pollution– Pollution sources, adverse health impacts on human and aquatic life and mitigation, Water quality parameters and standards.

Soil pollution and solid waste- Soil pollutants and their sources, solid and hazardous waste, Impact on human health.

Self-learning topics: Noise pollution, Thermal and radioactive pollution.

Module 6	**Climate Change: Impacts, Adaptation and Mitigation**	Assignment/case		**02 Classes**

Topics:

Understanding climate change: Natural variations in climate; Projections of global climate change with special reference to temperature, rainfall and extreme events; Importance of 1.5 °C and 2.0 °C limits to global warming; Impacts

Vulnerability and adaptation to climate change: Observed impacts of climate change on ocean and land systems; Sea level rise, changes in marine and coastal ecosystems; Impacts on forests and natural ecosystems; Indigenous knowledge for adaptation to climate change.

Self-learning topics: Mitigation of climate change: Synergies between adaptation and mitigation measures; National and international policy instruments for mitigation.

Module 7	**Environmental Management**	Case study	Data analysis	**02 Classes**

Topics:

Environmental management system: ISO 14001; Environmental risk assessment Pollution control and management; Waste Management- Concept of 3R (Reduce, Recycle and Reuse) and sustainability.

Self-learning topics: Environmental audit and impact assessment; Eco labeling /Eco mark scheme

Module 8	**Environmental Treaties and Legislation**	Case study	Data analysis	**01 Classes**

Topics:

Major International Environmental Agreements: Convention on Biological Diversity (CBD), Major Indian Environmental Legislations: Environmental Protection Act, Forest Conservation Act, Public awareness.

Self-learning topics: Paris Agreement, Conference of the Parties (COP), India's status as a party to major conventions: Air (Prevention and Control of Pollution) Act, Water (Prevention and control of Pollution) Act, Wildlife Protection Act.

List of laboratory tasks : Any eight experiments will be conducted

Determination of total alkalinity of a water sample (knowledge)
Estimation of water hardness by EDTA method and its removal (by zeolite/ ion exchange method) (Comprehensive)
Estimation of copper from industrial effluents by colorimetric method (Comprehensive)
Estimation of iron from industrial effluents by titrimetric method/potentiometric method (Comprehensive)
Estimation of nickel from industrial effluents by titrimetric method (Comprehensive)
Estimation of chloride in drinking water by titrimetric method (Comprehensive)
Estimation of fluoride in ground water by colorimetric method (Comprehensive)
Determination of calcium in aqueous solution (Comprehensive)
Determination of Total Dissolved Salts, conductivity and pH of a water samples (Knowledge)
Determination of Chemical oxygen demand in the industrial effluent. (Comprehensive)
Biological oxygen demand of waste water sample (Comprehensive)
Determination of dissolved oxygen of an industrial effluent (Comprehensive)
Quality monitoring analysis of a soil sample (knowledge)
Flame photometric estimation of Sodium and potassium (Application)
Gas Chromatographic analysis of volatile organic compounds (Application)

Targeted Application & Tools that can be used:

Application areas are Energy, Environment and sustainability

Tools: Statistical analysis of environmental pollutants using excel, origin 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.)

Lab evaluation/Assignment

End Term Exam

Self-learning

Assignment 1: Write a Statement of Environment report of your town/city/state/country

Assignment 2: Individual students will carry out the analyses of polluted solid, liquid, and gaseous samples and propose suitable mitigation measures. A detailed and in-depth report needs to be submitted for each case. This may include preparation of reagents, sample preparation (extraction), chemical analysis carried out, instruments and tools used, data collected and processed, inferences made and conclusions arrived at. Necessary support is given in the form of lab manual and reference links to e-books.

Text Book

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

Krishnamurthy, K.V. (2003) Text book of Biodiversity, Science Publishers, Plymouth, UK.

Jackson, A.R. & Jackson, J.M. (2000), Environmental Science: The natural environment and human impact, Pearson Education.

Reference Books

Fisher, Michael H. (2018) An Environmental History of India- From Earliest Times to the Twenty-First Century, Cambridge University Press.

William P. Cunningham and Mary Ann Cunningham (2017), Principles of Environmental Science: Inquiry & Applications, 8th Edition, McGraw-Hill Education, USA.

Sinha N., (2020) Wild and Wilful. Harper Collins, India.

www.ipcc.org; https://www.ipcc.ch/report/sixth-assessment-report-cycle/

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

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

E-resources:

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DOAB_1_06082022_18126

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DOAB_1_06082022_8761

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DOAJ_1_02082022_3333

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DOAB_1_06082022_3063

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DOAB_1_06082022_20719

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DOAB_1_06082022_16824

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DOAB_1_06082022_3954

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DOAB_1_06082022_491

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=CUSTOM_PACKAGE_16012023_WORLD_BUSINESS_COUNCIL_SUSTAINABLE_488

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=CUSTOM_PACKAGE_16012023_WORLD_BUSINESS_COUNCIL_SUSTAINABLE_583

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=SPRINGER_INDEST_1_171

https://presiuniv.knimbus.com/user#/searchresult?searchId=3R%20principle&_t=1687427221129

https://presiuniv.knimbus.com/user#/searchresult?searchId=eco%20labelling&_t=1687427279979

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=TEXTBOOK_LIBRARY01_06082022_395&xIndex=4

https://www.ugc.gov.in/oldpdf/modelcurriculum/env.pdf

Topics relevant to Skill Development:

Industrial revolution and its impact on the environment, Environmental impact of over-exploitation of water resources, pollution and ill effects, lab experiments for Skills development through Problem solving Techniques. This is attained through assessment component mentioned in course handout.

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

Course Code: **CSE1507**	**Course Title: Data Communication and Computer Networks Lab** **Type of Course: Lab**
Course Pre-requisites	
Anti-requisites	
Course Description	
Course Objective	
Course Out Comes	
Course Content	
Module 1	**Introduction to Computer Networks**
Learn to use commands like tcpdump, netstat, ifconfig, nslookup, ARP, NbtStat-n, Route, GETMAC, SYSTEMINFO and traceroute – Capture ping and traceroute PDUs using a network protocol analyzer and examine - Configuration and logging to a CISCO Router and introduction to the basic user Interfaces. Introduction to the basic router configuration and basic commands.	
Module 2	**Physical And Data Link Layer**
Configuration of IP addressing for a given scenario for a given set of topologie – Connecting devise – Configuration of Hub, Router, Switch and Repeaters using cisco packet tracer- Configure the privilege level password and user authentication in switch.	
Module 3	**Network Layer** **Transport Layer**

	Configure the DHCP Server and wireless router and check the connectivity - Configure the static routing using cisco packet tracer- Configure the Dynamic Routing routing (RIP Routing) using cisco packet tracer
Module 4	**Application Layer and Security in Computer Networks**
	Configure the Static NAT using cisco packet tracer - Configure the Dynamic NAT using cisco packet tracer. - Configure the DNS Server using cisco packet tracer - Configure the telnet protocol using cisco packet tracer - Wireshark Tool - Three Node Point To Point Network Using NS2 Simulator - Transmission of Ping Message Using NS2 Simulator - Ethernet LAN Using N-Nodes Using NS2 Simulator - Ethernet LAN Using N-Nodes With Multiple Traffic
	Targeted Application & Tools that can be used: Cisco Packet Tracer, Wireshark, NS2
	Case Study/Assignment: Assignment proposed for this course in CO1-CO4 Assume that a computer sends a frame at the transport layer to another computer and the destination port address is not running. According to what you read from chapter 2, what will happen to that process? Determine the possible bit rate and the number of levels over a channel for these cases? a. B = 2.4K Hz, noiseless channel with L = 16. b. B= 2.4K Hz, SNR = 20 dB. c. B = 3.0K Hz, SNR = 40 db. Using CISCO Packet Tracer Configuring Static and Default Routes Objectives • Configure static routes on each router to allow communication between all clients. • Test connectivity to ensure that each device can fully communicate with all other devices. Getting familiar with Wireshark software by installing it I your system, and perform following task: List out the packets which are having DNS protocols List of IP address present in the cache along with its MAC addresses Display all the packets which are having the DNS or HTTP protocol
	Problem Solving: Choose and appropriate devices and implement various network concepts.
	Text Book CCNA Routing and Switching Study Guide – Todd Lammle, 2013, Sybex. Wireshark Network Analysis: The Official Wireshark Certified Network Analyst Study Guide – Laura Chappell, 2012, Wireshark University. Computer Network Simulation Using NS2 – Ajit Kumar Nayak, Rajlaxmi Rai, Rakesh Mall, 2020, Routledge.
	References **R1:** Alberto Leon-Garcia and IndraWidjaja: Communication Networks - Fundamental Concepts and Key architectures, 2nd Edition Tata McGraw-Hill, 2004. **R2:** William Stallings: Data and Computer Communication, 8th Edition, Pearson Education, 2007.

	R3: "Computer Networking: A Top-Down Approach" – James F. Kurose and Keith W. Ross, 7th Edition, 2016, Pearson. Web Based Resources and E-books: W1: https://gaia.cs.umass.edu/kurose_ross/wireshark.php W2: https://www.youtube.com/watch?v=x7EJSY0bOK4&ab_channel=ChrisGreer W3: https://tutorials.ptnetacad.net/
	Topics relevant to "SKILL DEVELOPMENT": Application Layer, Transport Layer, Network Larver for **Skill development** through **Participative Learning t**echniques. This is attained through the assessment component mentioned in the course handout.

Course Code: CSE1508	**Course Title:** Data Structures **Type of Course: Integrated**	**L-T- P-C**	3	0	0	3
Version No.	1.0					
Course Pre-requisites	Problem Solving Using C					
Anti-requisites	NIL					
Course Description	This course introduces the fundamental concepts of data structures and to emphasize the importance of choosing an appropriate data structure and technique for program development. This course has theory and lab component which emphasizes on understanding the implementation and applications of data structures using Java programming language. With a good knowledge in the fundamental concepts of data structures and practical experience in implementing them, the student can be an effective designer, developer for new software applications.					
Course Objective	The objective of the course is SKILL DEVELOPMENT of student by using EXPERIENTIAL LEARNING techniques					
Course Out Comes	**On successful completion of the course the students shall be able to:** **CO1:** Describe the concept of basic data structure, stacks, queues, and arrays and their operations. **[Understand]** **CO2:** Utilize linked lists for real-time scenarios. **[Apply]** **CO3:** Apply an appropriate non-linear data structure for a given scenario. **[Apply]** **CO4:** Demonstrate different searching and sorting techniques. **[Apply]**					
Course Content:						
Module 1	**Introduction to Data Structure and Linear Data Structure – Stacks and Queues**	Assignment	Program activity			**9 Hours**
Introduction – Introduction to Data Structures, Types and concept of Arrays.						

Stack - Concepts and representation, Stack operations, stack implementation using array and Applications of Stack.

Queues - Representation of queue, Queue Operations, Queue implementation using array, Types of Queue and Applications of Queue.

Module 2	**Linear Data Structure-Linked List**	Assignment	Program activity	**12 Hours**

Topics: Linked List - Singly Linked List, Operation on linear list using singly linked storage structures, Circular List, Applications of Linked list.

Recursion - Recursive Definition and Processes.

Module 3	**Non-linear Data Structures - Trees**	Assignment	Program activity	**12 Hours**

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

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

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

Hashing: Introduction, Static Hashing, Dynamic Hashing

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

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

List of Laboratory Tasks:

Lab sheet -1

Level 1: Prompt the user, read input and print messages.Programs using class, methods and objects

Level 2: Programming Exercises on fundamental Data structure - Arrays based on Scenario.

Lab sheet -2

Level 1: Programming Exercises on Stack and its operations

Level 2: Programming Exercises on Stack and its operations with condition

Lab sheet -3

Level 1: Programming on Stack application infix to postfix Conversion

Level 2: -

Lab sheet -4

Level 1: Programming on Stack application – Evaluation of postfix

Lab sheet -5

Level 1: Programming Exercises on Queues and its operations with conditions

Level 2: -

Lab sheet -6

Level 1: Programming Exercises on Linked list and its operations.

Level 2: Programming Exercises on Linked list and its operations with various positions

Lab sheet -7

Level 1: Programming Exercises on Circular Linked list and its operations.

Level 2: Programming Exercises on Circular Linked list and its operations with various positions

Lab sheet -8

Level 1: Programming Exercises on factorial of a number

Level 2: Programming the tower of Hanoi using recursion

Lab sheet -9

Level 1: -

Level 2: Programming the tower of Hanoi using recursion

Lab sheet -10

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

Level 2: -

Lab sheet -11

Level 1: Program to Construct Binary Search Tree and Graph

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

Lab sheet -12

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

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

Lab sheet -13

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

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

Lab sheet -14 (Beyond syllabus activity)

Level 1: Program to Construct AVL Tree

Level 2:

Lab sheet -15 (Beyond syllabus activity)

Level 1: Program to Construct RED BLACK Tree

Targeted Application & Tools that can be used

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

Project work/Assignment:
Assignment: Students should complete the lab programs by end of each practical session and module wise assignments before the deadline.
Text Book **T1** Ellis Horowitz and Sartaj Sahni, Fundamentals of Data Structures in C, 2nd Edition, Universities Press, reprint 2018. **T2** Seymour Lipschutz, Data Structures Schaum's Outlines, Revised 1st Ed, McGraw Hill, 2014.
References **R1** Data structures and program design in C by Robert Kruse, Tondo C L, Bruce Leung, Pearson education publishers, 2017. **R2** Programming and Data Structure by Jackulin C Salini etal., Ane books publishers, 2019. **Web resources:** For theory: https://onlinecourses.nptel.ac.in/noc20_cs85/preview 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: **CSE1509**	**Course Title:** **Data Structures Lab** **Type of Course: Integrated**	0	0	4	2
Version No.	1				
Course Pre-requisites	**Problem Solving Using C**				
Anti-requisites	NIL				
Course Description	This course introduces the fundamental concepts of data structures and to emphasize the importance of choosing an appropriate data structure and technique for program development. This course has theory and lab component which emphasizes on understanding the implementation and applications of data structures using Java programming language. With a good knowledge in the fundamental concepts of data structures and practical experience in implementing them, the student can be an effective designer, developer for new software applications.				
Course Objective	The objective of the course is SKILL DEVELOPMENT of student by using EXPERIENTIAL LEARNING techniques				
Course Out Comes	**On successful completion of the course the students shall be able to:** **CO1:** Implement the basic operations of stacks, queues, **[Understand]** **CO2:** Utilize linked lists for real-time scenarios. **[Apply]**				

	CO3: Apply an appropriate non-linear data structure for a given scenario. **[Apply]** **CO4:** Demonstrate different searching and sorting techniques. **[Apply]**
	List of Laboratory Tasks: **1. Implement the operations on Stack and Queue, Circular Queue** **2. Programming on Stack application** **a.Infix to postfix conversion.** **b.Infix to prefix conversion.** **3. Write a program using linked list to Simulate memory allocation and Garbage collectio.** **4. Programming Exercises on Circular Linked list and its operations** **5. Implement Doubly Linked List (DLL) of Professor Data with the fields: ID, Name, Branch, Area of specialization.** **6.Given an array of elements, construct a complete binary tree from given array.** **8.Construct a menu driven program for traversing a binary search tree.** **9. Construct a program for Red Black Tree.** **10.Develop a program for graph for implementing shortest path.** **11.Develop a program for static and dynamic hashing.** **12.Design and develop a program that uses Hash Function H:K->L as H(K)=K mod m(reminder method) and implement hashing technique to map a given key K to the address space L. Resolve the collision (if any) using linear probing.** **13.Construct a program to perform selection sort.** **14.Implement Sequential and Binary Search** **15.Create an array of N elements and perform quick sort and 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.
	Text Book **T1** Ellis Horowitz and Sartaj Sahni, Fundamentals of Data Structures in C, 2nd Edition, Universities Press, reprint 2018. **T2** Seymour Lipschutz, Data Structures Schaum's Outlines, Revised 1st Ed, McGraw Hill, 2014.
	References **R1** Data structures and program design in C by Robert Kruse, Tondo C L, Bruce Leung, Pearson education publishers, 2017.

	R2 Programming and Data Structure by Jackulin C Salini etal., Ane books publishers, 2019. **Web resources:** For theory: https://onlinecourses.nptel.ac.in/noc20_cs85/preview 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: COM1701	**Course Title: Introduction to Bioinformatics** **Type of Course: General CSE Basket, Theory based**		**L- T-P- C**	3	0	0	3
Version No.	2.0						
Course Pre-requisites	Basics of Biology, basics of Computers.						
Anti-requisites	NIL						
Course Description	This course is designed to provide the knowledge of the concepts related to bioinformatics. The course is aimed at understanding the DNA and Protein sequences and databases. It also deals with Pairwise comparison and calculating the scoring matrix. Further, it focuses on Sequence Alignment techniques, discovering the Motifs in the sequence. Students will also learn the overview of Structural Bioinformatics and Genome sequencing.						
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Introduction to Bioinformatics** and attain **Employability** through **Participative Learning** techniques.						
Course Outcomes	**C.O.1:** Understand the DNA Protein sequence and structures. (Bloom's Level: **Knowledge) C.O.2:** Explain the file formats and sequence alignments of DNA sequence. (Bloom's Level: **Comprehension)** **C.O.3:** Apply the techniques of the motifs discovery for the analysis of Protein Sequence. (Bloom's Level: **Application)**						
Course Content:							
Module 1	**Fundamentals of Bioinformatics**	Quiz	Comprehension based Quizzes and assignments;	**9 Classes**			

Topics:

Introduction to Bioinformatics: Introduction to molecular biology, Cell, DNA, RNA, Transcription, Translation, Folding, Gene Structure, Introduction to Bioinformatics, Components and fields of bioinformatics, Omics, basic principles of structural/functional analysis of biological molecules, Biological Data Acquisition, Types of DNA sequences,Genomic DNA ,Mitochondrial DNA,DNA Sequencing tools, Protein sequencing and structure determination methods, Finding Reverse complement of a sequence.

Module 2	**Genome databases and Sequence Similarity**	Quizzes and assignments	Comprehension based Quizzes and assignments	**8 Classes**

Topics:

Types and classification of genome databases, DNA sequence retrieval system, various DNA and protein sequence file formats, Common sequence file formats; Files for multiple sequence alignment; Files for structural data, Frequent words and k-mers in Text, String Reconstruction problem, Sequence Similarity searching, Sequence Similarity searching tools, NCBI BLAST, PSI BLAST, Significance of sequence alignments, Alignment scores and gap
penalties.

Module 3	**DNA sequence analysis and applications**	Quizzes and assignments	Comprehension based Quizzes and assignments	**10 Classes**

Sequence similarity searches and alignment tools, Finding alignment using Needleman-Wunsch and Smith-Waterman algorithm, Heuristic Methods of sequence alignment, Pair-wise and multiple sequence alignments, DNA sequence analysis, Motif in protein sequence ,Motif discovery using Gibbs sampling,Motif finding, Gene Prediction models: Hidden Markov model(HMM), Generalized Hidden Markov model(GHMM), Bayesian method.

Targeted Application & Tools that can be used:

BLAST, FastA, , ClustalW, MEGA

Project work/Assignment:

Each batch of students (self-selected batch mates – up to 4 in a batch) will be allocated case studies/assignments

Textbook(s):

Bioinformatics: Sequence and Genome Analysis, David W. Mount, Cold Spring Harbor Laboratory Press, 2004.

Introduction to Bioinformatics, Arthur Lesk, Fifth Edition, Oxford University Press, 2019

References

Bioinformatics Methods and Applications, S. C. Rastogi, N.Mendiratta, P.Rastogi, Fourth Edition, Prentice Hall India.

Bioinformatics Algorithms- An Active Learning Approach, Phillip Compeau & Pavel Pevzner, 2nd Edition, Vol. I & II, Active Learning Publishers, 2015

E-References

1. https://puniversity.informaticsglobal.com:2229/login.aspx

Topics related to development of "Employability skills": Batch wise presentations on selected topics
String Reconstruction problem
Sequence Similarity searching
Alignment scores and gap penalties
Protein sequencing
Gene Prediction models: Hidden Markov model(HMM)
Finding similarities by performing pairwise and multiple sequence alignment,
Evaluating phylogenetic trees.

for developing **Employability Skills** through **Participative Learning techniques**. This is attained through assessment component mentioned in course handout.

Course Code: MAT2004	**Course Title: Discrete Mathematical Structures** **Type of Course: Program Core& Theory Only**	**L-T-P-C**	4	0	0	4
Version No.		2.0				
Course Pre-requisites		**NIL**				
Anti-requisites		**NIL**				
Course Description		This course highlights the basics of discrete structures and develop ability to solve problems involving mathematical logic, sets, functions, relations, principles of counting, pigeonhole principles, recurrence relations, Principles of Inclusion and Exclusion. forces, and moments with their applications in allied subjects. It is a prerequisite for several Courses involving Compiler Design, Artificial Intelligence. This course is both conceptual and analytical in nature that would help the student to use the concepts of discrete structures to solve and prediction of data analytics. The students should have prior knowledge of basic mathematics pursue the Course. After successful completion of the Course, the students would acquire knowledge to solve problems involving mathematical logic, sets, functions, relations, principles of counting, pigeon hole principles, recurrence relations, Principles of Inclusion and Exclusion with an emphasis on real-world engineering applications and problem solving.				
Course Objective		The objective of the course is to familiarize the learners with the concepts of Discrete Mathematics and attain SKILL DEVELOPMENT through PROBLEM SOLVING Methodologies techniques.				
Course Out Comes		On successful completion of the course the students shall be able to: Describe a logic sentence in terms of predicates, quantifiers, and logical connectives. Solve problems on Functions and Relations using basic principles of Set Theory. Explain the concepts of Boolean Algebra.				

		Apply basic counting techniques to combinatorial problem.			
Course Content :					
Module 1	**Foundations of Logics and Proofs**	Assignment	Problem Solving		**10 Sessions**
	Topics: Propositional Logic, Propositional Logic Equivalences, Inference rules, Normal forms, Introduction to Proofs, Resolution by Refutation, Predicates and Quantifiers, Introduction to Proofs. **Assignment: Problems.**				
Module 2	**Basic Structures: Sets, Functions,**	Assignment	Problem Solving		**10 Sessions**

	Relations			
Topics: Sets and set-operations, Venn Diagram, Cardinality of Sets, Functions: Types, Invertible Functions, Composition, Sequences and Summations, Relations and their properties & representations, Equivalence Relations, Closure of Relations. **Assignment:** Problems and applications				
Module 3	**Posets, Lattices and Boolean Algebra**	Assignment	Problem Solving	**10 Sessions**
Topics: Partial ordering, Posset, Hasse Diagram, Lattices & Algebraic structures, Basic properties of algebraic systems by lattices, Distributive lattices, complement of an element in a lattice, Boolean lattice & Boolean algebra,Topological Sorting. Assignment: Problems and Applications				
Module 4	**Principles of Counting Techniques**	Assignment	Problem Solving	**12 Sessions**

Topics: Number Theory: Integers and Division, GCD, Chinese Remainder Theorem, Solving Congruences, Pigeon Hole Principle, Mathematical Induction, Generalized Permutations and Combinations, Recurrence Relations, Applications of Recurrence Relations, Generating Functions, Principle of Inclusion and Exclusion, Applications of Inclusion and Exclusion. Assignment: Problems and Applications
Targeted Application & Tools that can be used: NIL
Project work/Assignment:
Problems on all the topics and relevance with field of computer science
Text Book T1. Kenneth H. Rosen, "Discrete Mathematics and its Applications", McGraw-Hill,s 7th Edition,2018.
References R1: Susanna EPP, "Discrete Mathematics with Applications", Cengage Learning, 4th Edition, 2010 R2. Thomas Koshy, "Discrete Mathematics with Applications", Elsevier, India, 2009. R3: Discrete mathematics for Computer Scientists and Mathematicians, Paperback (Rs. 533), Joel Mott, Abraham Kandel, Theodore Baker; Pearson Education India; 2 edition (2015), **ISBN-13:** 978-9332550490 **Weblinks:** W1: https://puniversity.informaticsglobal.com:2229/login.aspx W2: https://www.youtube.com/playlist?list=PLBlnK6fEyqRhqJPDXcvYlLfXPh37L89g3
Topics relevant to development of "SKILL": Mathematical Logic, Permutation and Combinations for Skill Development through Problem Solving methodologies. This is attained through assessment component mentioned in course handout.

Course Code: CSE 1512	**Course Title:** Analysis of Algorithms **Type of Course: THEORY Only**	**L- T-P-C**	3	1	0	4
Version No.	1.0					
Course Pre-requisites	**CSE2001 - Data Structures and Algorithms.**					
Anti-requisites	Nil					
Course Description	This course introduces techniques for the design and analysis of efficient algorithms and methods of applications. This course discusses the classic approaches for algorithm design such as Divide and Conquer, Dynamic Programming, Greedy method. This course also describes other basic strategies searching solution space. The core concepts of analyzing algorithms and classifying them into various complexity classes is covered in the end.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Analysis of Algorithms** and attain **Skill Development** through **Problem Solving** Methodologies.					
Course Out Comes	On successful completion of the course the students shall be able to: 1. Compute efficiency of a given algorithm.[Applying] 2. Apply divide and conquer technique for searching and sorting Problems.[Applying] 3. Apply the Dynamic Programming technique for a given problem. [Applying] 4. Apply greedy technique for solving a Problem.[Applying]					

	5. Demonstrate Back tracking technique and limitations of Algorithms.[Applying]			
Course Content:				
Module 1	**Introduction**	Assignment	Simulation/Data Analysis	**10 Sessions**
Introduction, Asymptotic Notations and its properties, Best case, worst case and average case- Sequential search, Sorting; Mathematical analysis for Recursive and Non-recursive algorithms: Substitution method and Master's Theorem.				
Module 2	**Divide-and-conquer**	Assignment	Simulation/Data Analysis	**08 Sessions**
Introduction. Insertion Sort; Merge sort, Quick sort, Binary search.				
Module 3	**Dynamic programming**	Term paper/Assignment	Simulation/Data Analysis	**10 Sessions**
Introduction with examples, Principles of Memoization, 0-1 Knapsack Problem, Bellman-Ford algorithm, Floyd-Warshall's Algorithms. Chain Matrix Multiplication.				
Module 4	**Greedy technique**	Term paper/Assignment	Simulation/Data Analysis	**09 Sessions**
Introduction, Fractional Knapsack Problem, Minimal Spanning Tree: Prim's Algorithm and Kruskal's Algorithm, Single-source Shortest Path: Dijkstra's Algorithm				
Module 5	**Complexity Classes**	Term paper/Assignment	Simulation/Data Analysis	**08 Sessions**
Complexity Classes- P,NP- NP Hard and NP Complete - Boolean Satisfiability Problem (SAT). Branch and Bound: Knapsack problem; Backtracking, - N-Queens problem.				

Text Book

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

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

References

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

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

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

Donald E. Knuth, "*The Art of Computer Programming",* Volumes 1and 3 Pearson.

Web-Resources

NPTEL: https://onlinecourses.nptel.ac.in/noc19_cs47/preview

Coursera: Analysis of Algorithms by Princeton University

Algorithms Specialization in Coursera by Stanford University(Group of 4 courses).

Algorithms Coding Contest Links maintained by Prof Gerth Stølting Brodal of Aarhus University

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

Course Code: CSE1511	**Course Title: Database Management Systems Laboratory** **Type of Course: 1) Laboratory**	**L-T-P-C**	0	0	2	1
Version No.	1.0					
Course Pre-requisites	Foundational understanding of data types, basic programming knowledge, operating systems and file management.					
Anti-requisites	NIL					
Course Description	The Database Management Systems (DBMS) Laboratory is designed to provide students with hands-on experience in database design, implementation, and management using SQL and database management tools such as MySQL. The lab complements theoretical concepts learned in database courses by allowing students to practice database creation, querying, and optimization techniques. The DBMS Lab enables students to develop industry-relevant skills in database management, preparing them for careers in **software development, data engineering, and database administration**.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of Database Management Systems and attain **Employability** through **Problem Solving** Methodologies.					
Course Out Comes	On successful completion of the course the students shall be able to: Demonstrate the database concepts, practice, and SQL queries. [Apply] Design and implement database schemas while applying normalization techniques to optimize structure. [Apply]] Develop and implement stored procedures, triggers, and views for automation and efficiency. [Apply] To Design and build database applications for real world problems. [Apply]					

Course Content:

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

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

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

Experiment No. 4: [2 Session]
4. To study and implement different types of Set and Join Operations [2 Slots]
Level 1: Demonstrate different types of Set Operations (UNION, UNION ALL, INTERSECT, MINUS) and Join Operations (INNER JOINs, OUTER JOINS, CROSS JOIN, NATURAL JOIN) on two or more tables of Airline

Database. Level 2: Use Set and Join operations to retrieve the data from two or more relations(tables) as per the given
scenario. [Airline Database]

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

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

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

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

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

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

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

Targeted Application & Tools that can be used:
Application Area: Relational database systems for Business, Scientific and Engineering Applications.
Tools/Simulator used: MySQL DB for student practice.

Also demonstration of ORACLE DB on object-relational database creation and JDBC connection.
Percentage of changes in this version: 50% of changes from earlier version. New topics are highlighted in italic.
Problem Solving: Constructing ER-Diagrams for a given real time requirements, Normalizing the databases, querying the databases using relational algebra. 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: CSE2502	**Course Title: Operating System** **Type of Course: Discipline Elective in Information Science & Engineering Basket** **Theory & Integrated Laboratory**	**L- T-P-C**	2	0	2	3
Version No.	1.0					
Course Pre-requisites	**[1] C Programming[2] Unix shell programming[3] Data Structure**					
Anti-requisites	**NIL**					
Course Description	The purpose of this course is to enable the students to understand the need for Operating systems and to develop the basic concepts of process management, synchronization and memory management. The course will expose students to Linux OS internals, its design and features. The course is both conceptual and analytical in nature towards managing the process and memory and needs fair knowledge of programming fundamentals, C programming and data structures. The course develops the critical thinking and analytical skills on allocating and					
	managing resources. The course also enhances the problem solving and systems programming abilities through assignments The associated laboratory provides an opportunity to validate the concepts taught as well as enhances the ability to approach designing new OS level features with confidence.					

Course Objective	The objective of the course is to familiarize the learners with the concepts of Operating System with Linux Internals and attain SKILL DEVELOPMENT through EXPERIENTIAL LEARNING techniques.			
Course Outcomes	**On successful completion of this course the students shall be able to:** **Explain** the structure and functions of OS **Solve** problems on various CPU Scheduling Algorithms **Apply** different techniques to various synchronization problems **Discuss** various memory management techniques **Apply** appropriate Linux commands for memory management and directory management			
Course Content:				
Module 1	**Introduction**	Quiz	Programming	**09 Classes**
Topics: Introduction to OS – Computer System Architecture , Operating System Structure, Operations – Different management activities handled by the OS, Computing environments, Operating System Services, User and OS interface, System Calls and its types, System Programs[loaders, linkers...], Overview of OS design and implementation. **Linux Operating System:** Introduction to Linux OS, Basic Commands of Linux OS				
Module 2	**Process Management**	Quizzes and assignments	Pseudocode/Programming	**9 Classes**
Topics: Process Concept, Operations on Processes, Inter Process Communication, Introduction to threads - Multithreading Models, Process Scheduling– Basic concepts, Scheduling Criteria, Scheduling Algorithms: FCFS, SJF, SRTF, RR, Priority, Multilevel Queue, Multilevel Feedback Queue. **Linux Operating System:** Process Management Commands and System Calls.				
Module 3	**Process Synchronization and Deadlocks**	Coding Assignment/Case Study	Pseudocode/Programming	**9 Classes**
Topics: The Critical-Section Problem - Peterson's Solution, Synchronization hardware, Mutex locks, Semaphores, Classic Problems of Synchronization, Monitors. Introduction to Deadlocks, Deadlock Characterization, Methods for handling deadlock: Deadlock Prevention- Deadlock Avoidance- Deadlock detection & Recovery from Deadlock **Linux Operating System: P**ipe, semaphore and message queue				

List of Laboratory Tasks:

Experiment No. 1: Basic UNIX Commands

Level 1: Linux commands- PATH, man, echo, printf, script, passwd, uname, who, date, stty, pwd, cd, mkdir, rmdir, ls, cp, mv, rm, cat, more, wc, lp, od, tar, file handling utilities, security by file permissions, process utilities

Level 2: Text Processing utilities and backup utilities , tail, head, sort, nl, uniq, grep, egrep, fgrep, cut, paste, join, tee, pg, comm, cmp, diff, tr, awk, cpio

Experiment No. 2: Programs using system calls of UNIX operating system

Level 1 Programs using the following system calls of UNIX operating system fork, exec, getpid, exit, wait, close, stat, opendir, readdir

Level 2 Simulate UNIX commands like cp, ls, grep.

Experiment No. 3: Programs to demonstrate process creation and termination **Level 1:** Program to demonstrate creating new processes and waiting for a process **Level 2:** Program to demonstrate creation of zombie processes and orphan process

Experiment No. 4: Programs to demonstrate inter process communication using Pipe

Level 1: Programs to illustrate execution of two commands concurrently with a command pipe and communication between two unrelated processes

Level 2: Program to demonstrate inter process communication using mkfifo, open, read, write and close APIs

Experiment No. 5: Programs to demonstrate inter process communication using message queues

Level 1: Program to create a message queue with read and write permissions and to write messages with different priority numbers

Level 2: Program to receive messages of different priorities from the message queue and display them

Experiment No. 6: Programs to demonstrate process synchronization using Semaphores

Level 1: Program that illustrates suspending and resuming processes using signals

Level 2: Program that illustrates access of shared memory using counting semaphore

Experiment No. 7: Programs to demonstrate the event of a deadlock and its avoidance

Level 1: Using POSIX Semaphores demonstrate the scenario where in deadlock happens due to incorrect use of semaphores

Level 2: Program to implement a solution to the Dining Philosopher problem using Monitors
Targeted Application & Tools that can be used: **Targeted Application:** Real time Applications such as traffic management system, banking system, health care and many more systems where there are entities that use and manage the resources. **Software Tools:** Linux Environment
Project work/Assignment:
Each batch of students (self-selected batch mates) will identify projects and implement with the most suitable 2 or 3 antecedents.
Textbook(s): Silberschatz A, Galvin P B and Gagne G, "Operating System Concepts", 9th edition Wiley, 2013 Sumitabha Das, "Unix concept and Programming", McGraw Hill education, 4th Edition, 2015
References Ellen Siever, Stephen Figgins, Robert Love, Arnold Robbins, Linux in a Nutshell, O'Reilly Media, Inc, 2009 Operating Systems | Internals and Design Principles | Ninth Edition | By Pearson Paperback – 1 March 2018. by William Stallings (Author)
Topics relevant to "SKILL **DEVELOPMENT"**: Linux OS commands and programming for **SKILL DEVELOPMENT through EXPERIENTIAL LEARNING techniques.**. This is attained through assessment component mentioned in the course handout.

Course Code: CSE1505	**Course Title: Web Technologies Lab** **Type of Course: Program core lab course**	**L-T- P-C**	0	0	2	1
Version No.	1.0					
Course Pre-requisites	Database Management Systems-CSE3156					
Anti-requisites	**NIL**					
Course Description	This course highlights the comprehensive introduction to scripting languages that are used for creating web-based applications.					

		The associated laboratory provides an opportunity to implement the concepts and enhance critical thinking and analytical skills.
Course Objective		**The objective of the course is to familiarize the learners with the concepts of Web Technology and attain Skill Development through Experiential Learning techniques.**
Course Outcomes		**On successful completion of this course the students shall be able to:** **CO1:** Implement web-based application using client-side scripting languages. **(Apply)** **CO2**: Apply various constructs to enhance the appearance of a website. **(Apply)** **CO3:** Apply server-side scripting languages to develop a web page linked to a database. **(Apply)**
Course Content:		

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

XML: Basics, Demonstration of applications using XML with XSLT.				
Module 3	**PHP – Application Level**	Quizzes and assignments	Application of PHP in web designing	**12 Sessions**
PHP: Introduction to server-side Development with PHP, Arrays, Superglobal Arrays, $GET and $ POST, $_SERVER Array, $_Files Array, Reading/Writing Files, PHP Classes and Objects, Object Oriented Design, Working with Databases, SQL, Database APIs, Managing a MySQL Database. Accessing MySQL in PHP, Applications.				
List of Laboratory Tasks: **Experiment No. 1: Demonstration of XHTML features** Level 1: Demonstration of various XHTML Tags (Level 1)				

Level 2: Design and develop static web pages for an online Book store (Level 2).

Experiment No. 2: Application of CSS in web designing

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

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

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

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

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

Experiment No. 4: Building a website.

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

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

Project work/Assignment:

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

Textbook(s):

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

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

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

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

Reference Book(s):

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

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

Additional web-based resources

W1. W3schools.com **W2.** Developer.mozilla.org/en-US/docs/Learn **W3.** docs.microsoft.com **W4.** informit.com/articles/ The Relationship Between Web 2.0 and Social Networking https://presiuniv.knimbus.com/user#/home
Topics related to development of "FOUNDATION": 1. Web, WWW, Web browsers, Web servers, Internet. 2. CSS, PHP. 3. Designing the website for healthcare.

Course Code: CSE1504	**Course Title: Web Technology Type of Course: Program core** **Theory Only**	**L- P- C**	2	0	2
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	**The objective of the course is to familiarize the learners with the concepts of Web Technology and attain Skill Development through Experiential Learning techniques.**				
Course Outcomes	**On successful completion of this course the students shall be able to:** **CO1:** Implement web-based application using client-side scripting languages. **(Application level)** **CO2**: Apply various constructs to enhance the appearance of a website. **(Application level) CO3**: Illustrate java-script concepts to demonstration dynamic web site**(Application level) CO4:** Apply server-side scripting languages to develop a web page linked to a database. **(Application level)**				

Course Content:				
Module 1	**Introduction to XHTML**	Quizzes and Assignments	Quizzes on various features of XHTML, simple applications	**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**	Quizzes and assignments	Comprehension based Quizzes and assignments; Application of CSS in designing webpages	**8 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**	Quizzes and assignments	Application of JavaScript for dynamic web page designing	**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**	Quizzes and assignments	Application of PHP in web designing	**14 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.

Course Code: CSE 3015	**Course Title: ADVANCED NATURAL LANGUAGE PROCESSING** **Type of Course: Integrated**	**L-T-P-C**	2	0	2	3
Version No.	1.0					
Course Pre-requisites	CSE 3014 – Fundamentals of Natural Language Processing					
Anti-requisites						
Course Description	This course is an advanced course for Natural Language Processing. As a part of the course, students will be introduced to solving multiple problems in natural language processing, such as sentiment analysis, machine translation, cognitive natural language processing, etc. **Topics include:** Machine translation, Text summarization, Sentiment analysis, Cognitive NLP, Gaze behaviour, Evaluation Metrics, etc.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Advanced Natural Language Processing**and attain **Employability** through E**xperiential Learning** techniques.					
Course Out Comes	**On successful completion of the course the students shall be able to:** • **Understand** how to solve different problems in natural language processing.[Comprehension] • **Solve** natural language generation problems such as machine translation and textsummarization. [Application] • **Perform** sentiment analysis on reviews to discern the stance of the writer.[Application] • **Use** public gaze behaviour data to improve the performance of different NLP systems.[Application]					
Course Content:						
Module 1	**Pre-trained LanguageModels**				**4 Sessions**	
Topics: Introduction to Pre-Trained Language Models. BERT. Multi-lingual variants of BERT. Introduction to NLTKand Huggingface Transformers.						
Module 2	**Machine Translation and Text Summarization**				**7 Sessions**	

Topics: Introduction to machine translation – source and target languages. Pivot-based machine translation. Using Transformers for machine translation. Monolingual machine translation examples. Machine translation evaluation metrics – BLEU. Implementation of BLEU score calculation using NLTK in Python. Other MT metrics – METEOR, TER, etc. Text summarization – definition. Types of summarizations – Extractive and Abstractive Summarization.
Summarization evaluation metrics – ROUGE score.

Module 3	**Sentiment Analysis**			**6 Sessions**

Topics: Introduction to Sentiment Analysis. Solving sentiment analysis using text classification. Classification of sentiment analysis based on different levels – polarity-based and intensity-based. Challenges in sentiment analysis
– sarcasm, thwarting, negations. Case studies in sentiment analysis – Reviewer rating prediction, short-textclassifications, etc.

Module 4	**Cognitive NLP Using Gaze Behaviour**			**7 Sessions**

Topics: Eye-Mind Hypothesis and gaze behaviour terminology. Using gaze behaviour for prediction of translation complexity, sentiment analysis complexity, sarcasm understandability, text complexity, text quality prediction, etc. Challenges with recording gaze behaviour at run time. Comparison of gaze behaviour across different people – normalization and binning. Gaze behaviour datasets. Mitigation of recording gaze behaviour at run time using type aggregation.

List of Laboratory Tasks:

1. Familiarization with Python. Using Python to read text files, basic tokenization and other preprocessing.
2. Introduction to NLTK and Huggingface Transformers in Python.
3. Using Huggingface Transformers to create a simple MT application.
4. Implementation of pivot-based machine translation using Huggingface Transformers.
5. Calculation of BLEU using NLTK – difference between sentence_bleu and corpus_bleu methods.
6. Implementation of extractive summarization.

7. Polarity classification of text using VADER.
8. Intensity prediction of text using Weighted Normalized Polarity Intensity.
9. Estimating gaze behaviour for a user using normalization and binning
10. Calculating gaze behaviour for a text based on type aggregation in multiple languages.
11. Complex word identification using gaze behaviour.

Targeted Application & Tools that can be used:

1. **Google Colab**
2. **Python IDE (Eg. PyCharm)**
3. **Huggingface Transformers**
4. **NLTK**

Project work/Assignment:

Assignment: Students will have to do a **course group assignment** over the course of the semester. The assignmenttopics can be taken from Modules 2 or 3 as per the instructor-in-charge.

Text Books

T1 Daniel Jurafsky, and James Martin. "*Speech and Language Processing*" (3rd edition draft, 2022).

T2 Abhijit Mishra, and Pushpak Bhattacharyya. "*Cognitively Inspired Natural Language Processing: AnInvestigation Based on Eye Tracking*". Springer, Singapore. 2018.

References

R1 Steven Bird, Ewan Klein, and Edward Loper. "*Natural Language Processing with Python: Analyzing Text withthe Natural Language Toolkit*". O'Reilly Publishers. 2009.

R2 Chris Manning, and Heinrich Schutze. "*Foundations of Statistical Natural Language Processing*". MIT Press.1999.

E book link R1: https://www.nltk.org/book/
E book link R2: https://nlp.stanford.edu/fsnlp/
Web resources: http://pu.informatics.global

Topics relevant to "EMPLOYABILITY SKILLS": Calculation of BLEU and ROUGE scores using NLTK **,** Estimating gaze behaviour through type aggregation, Using Hugging face Transformers for machine translation **for developing Employability Skills through Experiential Learning techniques. This is attained through assessment component mentioned in course handout.**

Course Code: CSE2500	**Course Title: Theory of Computation** **Type of Course: Theory Only**	**L - T - P - C**	3	1	0	4
Version No.	2.0					
Course Pre-requisites	The students should have the Knowledge on Set Theory					
Anti-Requisites	Nil					
Course Description	The course deals with introduction of formal languages and the correspondence between language classes and the automata that recognize them. Topics include: Formal definitions of grammars and acceptors, Deterministic and Nondeterministic systems, Grammar ambiguity, finite state and push-down automata; normal forms; Turing machines and its relations with algorithms.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Theory ofComputation as mentioned above** and attain **Skill Development** through **Problem Solving** Methodologies.					

Course Out Comes	On successful completion of the course the students shall be able to: 1. Describe various components of Automata. (Knowledge) 2. Illustrate Finite Automata for the given Language. (Application) 3. Distinguish between Regular grammar and Context free grammar. (Comprehension) 4. Construct Push down Automata. (Application) 5. Construct Turing machine for a Language. (Application)			
Course Content:				
Module 1	**Introduction to automata theory**	Assig nment	Problems on Strings and Language operations	**06 Sessio ns**
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**	Assig nment	Problems on DFA, NFA's	**13 Sessio ns**
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**	Assig nment	Problems on RE, CFG, PT, PLand Ambiguity	**12 Sessio ns**
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**	Assig nment	Problems on pushdown Automaton	**08 Sessio ns**
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**	Assig nment	Problems on Turning Machine	**07 Sessio ns**
Topics: Definition of a Turing Machine, Turing Machines as Language Accepters, Example Languages to construct Turingmachine, 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 IntelligenceTools:

1. JFLAP (Java Formal Language and Automata Package) Software simulation tool. It's interactive educationalsoftware written in Java to experiment topics in automata theory.
2. Turing machine Online simulators.

Text Book

1. Peter Linz, "An introduction to Formal Languages and Automata", Jones and Bartlett Publications 6th Ed, 2018.

References

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

Topics relevant to "SKILL DEVELOPMENT": Deterministic and Non-Deterministic Automaton, Regular Expressions,CFGs, Turning Machine and Pushdown automaton for Skill Development through Problem Solving methodologies. This is attained through assessment component mentioned in course handout.

Course Code: CSE2506	**Course Title: Cloud Computing** **Type of Course: Theory**	**L-T- P- C**	2	0	2	3
Version No.	1					
Course Pre-requisites	Basics of Distributed Computing, Service Oriented Architecture					
Anti-requisites	Nil					
Course Description	This Course is designed to impart the knowledge of Cloud Computing as a new computing paradigm. The course explores various Cloud Computing terminology, principles andapplications. The course also demonstrates the different views of the Cloud Computing such astheoretical, technical and commercial aspects.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Cloud Computing** and attain **Employability** through **Participative Learning** techniques.					
Course Out Comes	**On successful completion of the course the students shall be able to:** • **Describe fundamentals of cloud computing, virtualization and cloudcomputing services.** • **Explain security and standards in cloud computing.** Discuss Cloud mechanisms to optimize the QoS parameters. Develop applications using Cloud services and VM instances.					
Course Content:						
Module 1						**10 Sessions**

Introduction to Cloud
Cloud Computing at a Glance, Historical Developments, Building Cloud Computing Environments, Computing Platforms and Technologies, Technology Examples, Cloud Computing Architecture, IaaS, PaaS, SaaS, Types of Clouds, Economics of Cloud

Module 2	**10 Sessions**

Virtualization Techniques

Basics of Virtualization - Types of Virtualizations, Taxonomy of Virtualization Techniques, Implementation Levels of Virtualization.

Module 3	**09 Sessions**

Cloud QoS and Management

Cloud Infrastructure Mechanisms, SLAs, Specialized Cloud Mechanisms, Cloud Management Mechanisms, Cloud Security Mechanisms.

Module 4	**09 Sessions**

Cloud Platforms, Advances in cloud: introduction to **Amazon Web Services**: Introduction to Google **App Engine,**
Introduction to **Microsoft Azure.**
Media Clouds - Security Clouds - Computing Clouds - Mobile Clouds – Federated Clouds – Hybrid Cloud

Text Book

1. John Rittinghouse and James Ransome, "Cloud Computing, Implementation, Management and Security", CRC Press.
2. Rajkumar Buyya, Christian Vecchiola, and Thamarai Selvi, "Mastering Cloud Computing", McGraw Hill Education.

References

1. David E.Y. Sarna, "Implementing and Developing Cloud Applications", CRC Press.
2. Anthony T Velte, Toby J Velte, Robert Elsenpeter, "Cloud Computing: A Practical Approach", Tata McGraw-Hill.

Web resources: https://puniversity.informaticsglobal.com:2229/login.aspx

Topics relevant to development of "Skill Development Aws, Azure, APIs, Aneka Cloud Platform, EC2, Installation of VM Workstation**,** Infrastructure Security Challenges for Skill Development through Participative Learning techniques. This is attained through assessment component mentioned in course handout.

Course Code: CSE 3189	**Course Title:Deep Learning** **Type of Course:Program Core Theory and Laboratory Integrated**	**L-T-P-C**	3	0	3	3
Version No.	1.0					
Course Pre-requisites	• Data Mining and Machine Learning fundamentals • Basic working knowledge of Statistics and Probability • Familiarity with programming languages and hands on coding					
Anti-requisites	NIL					

Course Description	The course introduces the core intuitions behind Deep Learning, an advanced branch of Machine Learning involved in the development and application of Artificial Neural Networks that function by simulating the working principle of human brain. Deep learning algorithms extract layered high-level representations of data in a way that maximizes performance on a given task. The course includes theory and lab components which emphasizes on understanding the implementation and application of deep neural networks in various prominent problem domains like speech recognition, sentiment analysis, recommendations, and computer vision etc. The course facilitates the students to interpret and appreciate the successful application of deep neural nets in various prediction and classification tasks of ML.			
Course Object	The objective of the course is to familiarize the learners with the concepts of Deep Learningand attain **Skill Development** through **Experiential Learning** techniques.			
Course Out Comes	On successful completion of the course the students shall be able to: 1. Apply basic concepts of Deep Learning to develop feed forward models 2. Apply Supervised and Unsupervised Deep Learning techniques to build effectivemodelsfor prediction or classification tasks 3. Identify the deep learning algorithms which are more appropriate for various types oflearning tasks in various domains of Machine Learning and Machine vision. 4. Analyze performance of implemented Deep Neural models			
Course Content:				
Module 1	Introduction to Deep Learning	Assignment	Programming	**No. of Classes:10**
Topics: Machine Learning in a nutshell, Fundamentals of deep learning and neural networks,Deep Neural Network,Feedforward Neural Network, , Perceptron, MLP Structures, Activation Functions, Loss Functions, Gradient Descent, Back-propagation, Training Neural Networks Building your Deep Neural Network: Step by Step, Deep Neural Network for Classification.				
Module 2	Improving Deep Neural Networks	Assignment	Programming	**No. of Classes:09**
Topics: Hyperparameter tuning, Initialization, Overfitting and Underfitting, Regularization and Optimization, Dropout,Batch Normalization				
Module 3	Deep Supervised Learning Models	Assignment	Programming	**No. of Classes:10**

Topics:

Convolutional neural network,Prediction of image using Convolutional Neural Networks,Deep learning inSequential Data, RNN & LSTM, GRU, Sentiment Analysis

Module 4	Deep Unsupervised Learning	Assignment	Programming	**No. of Classes:10**

Topics:

Basics of Deep unsupervised learning, Auto encoders,Restricted Boltzmann Machine, Recommender systems

Text Book

1. Ian Goodfellow, YoshuaBengio, Aaron Courville, “Deep Learning”, MIT Press, 2017

References

1. Duda, R.O., Hart, P.E., and Stork, D.G. Pattern Classification. Wiley-Inderscience, 2nd Edition. 2013

2. Theodoridis, S. and Koutroumbas, K. Pattern Recognition. Edition 4, Academic Press, 2015

3. Russell, S. and Norvig, N. Artificial Intelligence: A Modern Approach. Prentice Hall Series in ArtificialIntelligence, 2013

4. Bishop, C. M. Neural Networks for Pattern Recognition, Oxford University Press, 2008.

https://sm-nitk.vlabs.ac.in/

https://nptel.ac.in/courses/105105157

Topics relevant to “SKILL DEVELOPMENT”: Real time Data Analysis, Naming and coding for Skill Development through Experiential Learning techniques. This is attained through assessment component mentioned in course handout.

Course Code: CSE3082	**Course Title:** Object Oriented analysis and Design with UML **Type of Course: Integrated Only**		**L-T-P-C**	3	0	0	3
Version No.	2.0					
Course Pre-requisites	Object Oriented Programming fundamentals, Software Engineering					
Anti-requisites						
Course Description	This course deals with producing detailed object models and designs from system requirements; using the modeling concepts provided by UML; identifying use cases and expanding them into full behavioral designs; expanding the analyzing into a design ready for implementing and constructing designs that are reliable. The course begins with an overview of the object oriented analysis and design.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of **A** Object Oriented analysis and Design with UML and attain SKILL DEVELOPMENT through EXPERENTIAL LEARNING techniques					
Course Out Comes	CO1 : Ability to analyze and model software specifications. CO2 : Ability to abstract object-based views for generic software systems. CO3 : Ability to deliver robust software components.					
Course Content:						
Module 1	Introduction to Objectoriented system-Knowledge level	Assignment	SRS	**20 Sessions**		
Object Basics-Object Oriented System Development Life Cycle- Use case driven approach-Rumbaugh Object Model-Booch Methodology-Jacobson Methodology-Unified Approach, Framing problem statement and SRS document.						
Module 2	**Object oriented analysis-Comprehensive Level**	Assignment	Class diagram	**10 Sessions**		
Identifying use cases-Object Analysis-Classification: Theory-Approaches for Identifying Classes: Noun Phrase approach, Common Class pattern approach, Use case driven approach, Classes, Responsibilities and Collaborators- Identifying Object relationships: Associations, Super–sub class relationships, Aggregation.						
Module 3	**Object oriented design-Comprehensive Level**	Term paper/Assignment	Object Diagram	**11 Sessions**		
Object Oriented Design Axioms-Designing Classes -Class visibility -Redefining attributes -Designing methods and protocols -Packages and managing classes -Access Layer- Object Storage Persistence - Object oriented Database System-Designing view layer classes -Macro level process -Micro level process- Prototyping the user interface –Quality Assurance Tests-Testing Strategies.						

Module 4	**Object oriented UML Modeling-Applicationlevel**	Term paper/Assignment	Dynamic Diagrams	**9 Sessions**

Static and Dynamic Modeling-Unified Modeling Language -UML diagrams: Class Diagrams-Use case Diagram-UML Dynamic modeling: Interaction diagram, Sequence diagram, Collaboration diagram, State-chart diagram,
Activity diagram

Targeted Application & Tools that can be used: Star UML

Text Book

Object Oriented Modeling and Design using UML, Second Edition, Michael Blaha and James Rumbaugh, Pearson
Education, Second Edition, 2007

References

R1. Applying UML and Patterns, Third Edition, Craig Larman, Pearson Education, 2008 R2. Object Oriented Analysisand Design with Applications, Grady Booch, Addison-Wesly SecondEdition, 1994 R3. Object Oriented Systems Development using Unified Modeling Language, Ali Behrami, McGraw Hill International Edition, 1999 R4. Design Patterns, Gamma et. al., Pearson Education, 2006.

E-Resources

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

Topics relevant to the development of SKILLS:

1. Aggregation
2. Quality Assurance Tests
3. Responsibilities and Collaborators
4. Swimlane Diagram
5. Pattern Model

for **Skill Development** through **Experiential Learning** techniques. This is attained through assessment componentmentioned in course handout.

Course Code: CSE 3050	**Course Title:** Software Project Management **Type of Course: Theory Only Course**	**L-T- P-**	3	0	0	3
Version No.	1					
Course Pre-requisites	Basics of Programming					
Anti-requisites						

Course Description	Effective software project management is crucial to the success of any software development or maintenance project. The roles and responsibilities of the project manager is numerous and varied. However, at the broad level, these can be classified in to the project planning and monitoring and control activities. Project planning involves making cost, effort, and duration estimation and preparing various types of plans such as schedule, configuration management, risk management, quality management. Staffing plan etc. The monitoring and control activities encompass keeping track of progress and removing bottlenecks using techniques such as PERT, GANTT, and also effective risk management, team building etc.			
Course Objective	The objective of the course is to familiarize the learners with the concepts of Software Project Management and attain **Employability** through **Participative Learning** techniques.			
Course Out Comes	**On successful completion of the course the students shall be able to:** • Understand the different project contexts and appropriate management strategy. • Practice the role of professional ethics in successful software development. • Identify the key phases of project management. • Determine an appropriate project management approach through an evaluation of the business context and scope of the project.			
Course Content:				
Module 1	**Conventional & Modern Software Management**	Assignment	Case studies	**9 Sessions**
Topics: Waterfall Model, Conventional Software Management Performance; Evolution of Software Economics - Software economics, Pragmatic software cost estimation, Reducing software product size, Improving software processes. Principles of Conventional Software Engineering, Principles of Modern Software Management, Transitioning to an interactive Process.				
Module 2	**Software Management Process Framework**	Case studies / Case let	Case studies	**9 Sessions**
Topics: Life cycle phases, The artifact sets, Management artifacts, Engineering artifacts, Pragmatic artifacts; Model Based Software Architectures - A management perspective and A technical perspective.				
Module 3	**Project Organization and Planning**	Quiz	Case studies	**10 Sessions**
Topics: Work breakdown structures, Planning guidelines, The cost and schedule estimating process, The iteration planning process, Pragmatic planning, Line-of-Business organizations, Project organizations, Evolution of organizations; Process automation - Automation building blocks, The project environment.				
Module 4	**Project Control and Process Instrumentation**	Quiz	Case studies	**10 Sessi**

				ons

Topics:
PROJECT CONTROL AND PROCESS INSTRUMENTATION :The Seven-Core metrics, Management indicators, Qualityindicators, Life-Cycle expectations, Pragmatic software metrics, Metrics automation, Modern project profiles, Next generation software economics, Modern process transitions.

Targeted Application & Tools that can be used:

Project work/Assignment:

Assignment:

Text Book

T1. Walker Royce, "Software Project Management : A unified Framework", 1st Edition, Pearson Education, 2021

References

R1. Bob Hughes and Mike Cotterell, "Software Project Management", 3rd Edition, Tata McGraw Hill Edition,2005.

R2. Joel Henry, "Software Project Management", 1st Edition, Pearson Education, 2006.

E book link T1:
https://www.edutechlearners.com/download/Software%20Project%20Management.pdf

Web resources: https://onlinecourses.nptel.ac.in/noc19_cs70/preview
brary resources:
https://presiuniv.knimbus.com/user#/searchresult?searchId=eBook&curPage=0&layout=grid&sortFieldId=doc_title_str&topresult=false&content=*software%20project%20management*&sub_category_name=Computer%20Science%20and%20IT

Topics relevant to development of "EMPLOYABILITY SKILLS": Life cycle Phases, Seven Core Metrics, fordevelopment of Employability Skills through the Participative Learning Techniques. This is attained through the assessment components mentioned in the course handout.

Course Code: CSE3011	**Course Title: Reinforcement Learning** **Type of Course: Theory Only**	**L-T-P-C**	2	0	0	2
Version No.	1.0					
Course Pre-requisites	• Knowledge of programming in Python is required. • Knowledge of probabilities/statistics, calculus and linear algebra is required. • Machine learning background, as provided for example by COMP-551 or COMP-652 isrequired.					
Anti-requisites	NIL					

Course Description	The goal of this class is to provide an introduction to reinforcement learning, a very active research sub-field of machine learning. Reinforcement learning is concerned with building programs that learn how to predict and act in a stochastic environment, based on past experience. Applications of reinforcement learning range from classical control problems, such as power plant optimization or dynamical system control, to game playing, inventory control, and many other fields. Notably, reinforcement learning has also produced very compelling models of animal and human learning. During this course, we will study theoretical properties and practical applications of reinforcement learning. We will follow the second edition of the classic textbook by Sutton & Barto (available online for free, or from MIT Press), and supplement it as needed with papers and other materials.			
Course Objective	The objective of the course is to familiarize the learners with the concepts of **ReinforcementLearning** and attain **Skill Development** through **Problem Solving** Methodologies.			
Course Out Comes	On successful completion of the course the students shall be able to: 1. Knowledge of basic and advanced reinforcement learning techniques. 2. Identification of suitable learning tasks to which these learning techniques can beapplied. 3. Appreciation of some of the current limitations of reinforcement learning techniques. 4. Formulation of decision problems, set up and run computational experiments, evaluation of results from experiments.			
Course Content:				
Module 1	Introduction	Assignment	Programming	**No. of Classes:10**
Topics: Course logistics and overview. Origin and history of Reinforcement Learning research. Its connections with other related fields and with different branches of machine learning. **Probability Primer** Brush up of Probability concepts - Axioms of probability, concepts of random variables, PMF, PDFs, CDFs, Expectation. Concepts of joint and multiple random variables, joint, conditional and marginal distributions.Correlation and independence.				
Module 2	Markov Decision Process	Assignment	Programming	**No. of Classes:10**
Topics: Introduction to RL terminology, Markov property, Markov chains, Markov reward process (MRP). Introduction to and proof of Bellman equations for MRPs along with proof of existence of solution to Bellman equations in MRP. Introduction to Markov decision process (MDP), state and action value functions, Bellman expectation equations, optimality of value functions and policies, Bellman optimality equations.				
Module 3	Prediction and Control by Dynamic Programing	Assignment	Programming	**No. of Classes:10**
Topics: Overview of dynamic programing for MDP, definition and formulation of planning in MDPs, principle of optimality, iterative policy evaluation, policy iteration, value iteration, Banach fixed point theorem, proof of contraction mapping property of Bellman expectation and optimality operators, proof of convergence of policy evaluation and value iteration algorithms, DP extensions **Monte Carlo Methods for Model Free Prediction and Control** Overview of Monte Carlo methods for model free RL, First visit and every visit Monte Carlo, Monte Carlo control, On policy and off policy learning, Importance sampling.				
Module 4	TD Methods and Policy Gradients	Assignment	Programming	**No. of Classe**

				s:10

Topics:

Incremental Monte Carlo Methods for Model Free Prediction, Overview TD(0), TD(1) and TD(λ), k-step estimators, unified view of DP, MC and TD evaluation methods, TD Control methods - SARSA, Q-Learning and their variants.

Getting started with policy gradient methods, Log-derivative trick, Naive REINFORCE algorithm, bias and variance in Reinforcement Learning, Reducing variance in policy gradient estimates, baselines, advantage function, actor-critic methods.

Targeted Application & Tools that can be used:

While Convolution Neural Network (CNN) and Recurrent Neural Network (RNN) are becoming more important for businesses due to their applications in Computer Vision (CV) and Natural Language Processing (NLP), Reinforcement Learning (RL) as a framework for computational neuroscience to model decision making process seems to be undervalued. Besides, there seems to be very little resources detailing how RL is applied in different industries. Despite the criticisms about RL's weaknesses, RL should never be neglected in the space of corporate research given its huge potentials in assisting decision making.

Tools: Torch, Google Colaboratory, Spider, Jupiter Notebook

Project work/Assignment:

This part is written for general readers. At the same time, it will be of greater value for readers with some knowledge about RL.

- **Resources management in computer clusters**

Designing algorithms to allocate limited resources to different tasks is challenging and requires human-generated heuristics. The paper "Resource Management with Deep Reinforcement Learning" [2] showed how to use RL to automatically learn to allocate and schedule computer resources to waiting jobs, with the objective to minimize the average job slowdown.

State space was formulated as the current resources allocation and the resources profile of jobs. For action space, they used a trick to allow the agent to choose more than one action at each time step. Reward was the sum of (- 1/duration of the job) over all the jobs in the system. Then they combined REINFORCE algorithm and baseline value to calculate the policy gradients and find the best policy parameters that give the probability distribution of actions to minimize the objective.

- **Traffic Light Control**

Researchers tried to design a traffic light controller to solve the congestion problem. Tested only on simulated environment though, their methods showed superior results than traditional methods and shed a light on the potential uses of multi-agent RL in designing traffic system.

Five agents were put in the five-intersection traffic network, with a RL agent at the central intersection to control traffic signalling. The state was defined as eight-dimensional vector with each element representing the relative traffic flow of each lane. Eight choices were available to the agent, each representing a phase combination, and the reward function was defined as reduction in delay compared with previous time step. The authors used DQN to learn the Q value of the {state, action} pairs.

- **Robotics**

There are tremendous works on applying RL in Robotics. Readers are referred to for a survey of RL in Robotics. In particular, trained a robot to learn policies to map raw video images to robot's actions. The RGB images were fed to a CNN and outputs were the motor torques. The RL component was the guided policy search to generate training data that came from its own state distribution.

- **Web System Configuration**

There are more than 100 configurable parameters in a web system and the process of tuning the parameters requires a skilled operator and numerous trail-and-error tests. The paper "A Reinforcement Learning Approach to Online Web System Auto-configuration" showed the first attempt in the domain on how to do autonomic reconfiguration of parameters in multi-tier web systems in VM-based dynamic environments.

The reconfiguration process can be formulated as a finite MDP. The state space was the system configuration, action space was {increase, decrease, keep} for each parameter, and reward was defined as the difference between the given targeted response time and measured response time. The authors used the model-free Q-learning algorithm to do the task.

Text Book

1. "Reinforcement Learning: An Introduction", Richard S. Sutton and Andrew G. Barto, 2nd Edition
2. "Probability, Statistics, and Random Processes for Electrical Engineering", 3rd Edition, Alberto Leon-Garcia
3. "Machine Learning: A Probabilistic Perspective", Kevin P. Murphy

References

1. Richard S. Sutton and Andrew G. Barto, "Reinforcement learning: An introduction", Second Edition, MITPress, 2019.
2. Li, Yuxi. "Deep reinforcement learning." arXiv preprint arXiv:1810.06339 (2018).
3. Wiering, Marco, and Martijn Van Otterlo. "Reinforcement learning." Adaptation, learning, and optimization 12 (2012):

E-Resources

NPTEL course – https://onlinecourses.nptel.ac.in/noc19_cs55/preview
https://archive.nptel.ac.in/courses/106/106/106106143/
https://www.digimat.in/nptel/courses/video/106106143/L35.html

Topics relevant to "SKILL DEVELOPMENT": Real time Data Analysis using Reinforcement learning **for Skill Development through Problem Solving techniques. This is attained through assessment component mentioned in course handout.**

Course Code: COM1700	Course Title: Introduction to Bioinformatics Type of Course: Theory only course	L- T-P- C	3	0	0	3
Version No.	1					
Course Pre-requisites	NIL					
Anti-requisites	NIL					
Course Description	The purpose of this course is to introduce students to this interdisciplinary field of science that combines computer science and mathematics to analyze and interpret biological data. This course is designed to give students both a theoretical background and a working knowledge of the techniques employed in bioinformatics. Topics include: DNA and Protein sequences, bioinformatics databases, sequence similarity measures, sequence alignment motif finding and phylogenetic. Emphasis will be placed on biological sequence analysis and its applications.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of Introduction to Bioinformatics and attain **Skill Development** through **Participative Learning** techniques.					
Course Outcomes	On successful completion of this course the students shall be able to: CO1: Describe the structure of DNA, RNA and protein sequences [Understand] CO2: Identify the results obtained from the bioinformatics database [Understand] CO3: Interpret biological sequences and sequence similarity [Apply] CO4: Solve phylogenetic trees based on biological sequence data [Apply]					
Course Content:						
Module 1	Fundamentals of Molecular Biology and Bioinformatics	Assignment	Comprehension based Quizzes and assignments;	12 sessions		
Topics: Introduction to molecular biology: Cell, DNA, RNA, Transcription, Translation, Folding, Gene Structure, Introduction to Bioinformatics: Components and fields of bioinformatics, Omics, basic principles of structural/functional analysis of biological molecules, Biological Data Acquisition, Types of DNA sequences, Protein sequencing and structure determination methods, Finding Reverse complement of a sequence.						
Module 2	Genome databases and Sequence Similarity	Assignments	Comprehension based Quizzes and assignments	11 sessions		
Topics: Bioinformatic Resources, Types and classification of genome databases, file formats, Frequent words and k-mers in Text, Substitution matrices, PAM, BLOSUM, Gap penalties, Similarity search, BLAST, Significance of sequence alignments, Alignment scores and gap penalties						
Module 3	DNA sequence analysis	Quizzes and assignments	Comprehension based Quizzes and assignments	12 sessions		
Topics: Dynamic Programming Algorithms for Sequence Alignment - Needleman-Wunsch and Smith-Waterman, Local Alignment, Global Alignment, FastA , Multiple Sequence Alignment, Common multiple alignment methods, Practical aspects of multiple alignments, ClustalW.						
Module 4	Motifs and Phylogenetics	Case study	Comprehension based Quizzes and assignments	10 sessions		

Topics:
Motifs and patterns, PROSITE, Motif discovery using Gibbs sampling, Elements of phylogenetic models, Determining the substitution model tree, Evaluating phylogenetic trees, Distance based Methods – UPGMA, Neighbour Joining, Distance Measures

Course Code: COM250	Course Title: Computer Vision and LLMs Type of Course: Program Core -Theory	L-T-P-C	2	0	0	2
Version No.	1.0					
Course Pre-requisites	CSE3157 – Artificial Intelligence and Machine Learning					
Anti-requisites	NIL					
Course Description	This course combines computer vision with large language models. It provides an introduction to computer vision, deep learning techniques for computer vision, followed by large language models, and finally multimodality in the form of text and vision. Topics: Convolution, Convolutional Neural Networks, Image Representation, Softmax Function, Attention, Transformers, Multimodality					
Course Objectives	The objective of the course is EMPLOYBILITY of student by using EXPERIENTIAL LEARNING techniques.					
Course Out Comes	On successful completion of this course the students shall be able to: 1. **Summarize** the concepts of digital image processing (Understand). 2. **Describe** the various Image formation techniques and its difficulties (Understand). 3. **Apply** the linear filter to images (Apply). 4. **Use** EM Algorithm for identifying the segments in an image (Apply).					

Topics relevant to "Skill Development": Finding similarities by performing pairwise and multiple sequence alignment, Evaluating phylogenetic trees.

Textbook:
1. N. C. Jones and P. A. Pevzner, Bioinformatics and Functional Genomics, 3rd Edition, Wiley-Blackwell, 2015
2. D.W Mount, Bioinformatics: Sequence and Genome analysis, 2ndEdn, Cold Spring Harbor Laboratory Press, 2004.
3. D. E. Krane and M. L. Raymer, Fundamental Concepts of Bioinformatics, Pearson Education, 2003

References
1. Kappelmann-Fenzl, Melanie, ed. *Web Scraping in Social Networking and Data Analysis*. Heidelberg, 2021.
2. S. Balamurugan, Anand T. Krishnan, Dinesh Goyal, Balakumar Chandrasekaran, Computation in Bioinformatics, Multidisciplinary Applications, Wiley,2021

Web References:
1. https://www.coursera.org/specializations/bioinformatics
2. https://onlinecourses.nptel.ac.in/noc21_bt06/preview

Catalogue prepared by

Recommended by the Board of Studies on

Date of Approval by the Academic Council

Course Content:

Module 1	Introduction to Computer Vision	Adversarial Quiz Tests	Module Tests	No. of Sessions: 09

Basic Image Processing Operations – Sampling and quantization, Image Resizing, Aliasing and image enhancement, Spatial domain filtering. **Advanced Image processing Operations** – Color Image Processing, Image Restoration and reconstruction, Image Compression, Image Segmentation. **Image Formation** – Sources, Shadows and Shading. **Image Models** – Geometric Image Features, Analytical Image Features

Module 2	Deep Learning for Computer Vision	Adversarial Quiz Tests	Module Tests	No. of sessions: 06

Feedforward Neural Networks, Gradient Descent, Regularization. Convolutional Neural Networks for Image Classification. Advanced CNN Architectures – VGG, ResNet. CNNs for Object Detection and Segmentation.

Module 3	Deep Generative Models	Adversarial Quiz Tests	Module Tests	No. of sessions: 09

Attention and Transformers. Soft and Hard Attention – Image Captioning. Vision Transformers. Generative Adversarial Networks. Autoencoders and Variational Autoencoders. Diffusion Models. Applications beyond Image Captioning – Visual QA and Visual Dialogs.

Module 4	Vision Language Models	Adversarial Quiz Tests	Module Tests	No. of Sessions: 06

Diffusion Models. Classifier and Classifier-Free Diffusion Guidance. Text-conditioned Diffusion Models. Sampling, Prediction Space, Noise Schedules, Architectures of Diffusion Models. Self-Supervised Learning – SimCLR. Contrastive Learning. Vision Language Models. Examples of Vision Language Models – Dall-E,

Course Code:	Course Title: Computer Vision and LLMs Lab	L-T-P-C	0	0	2	1
CSM2501	Type of Course: Program Core Laboratory					
Version No.	1.0					
Course Pre-requisites	CSE3157 – Artificial Intelligence and Machine Learning					
Anti-requisites	NIL					
Course Description	This course combines computer vision with large language models. It provides an introduction to computer vision, deep learning techniques for computer vision, followed by large language models, and finally multimodality in the form of text and vision. Topics: Convolution, Convolutional Neural Networks, Image Representation, Softmax Function, Attention, Transformers, Multimodality					
Course Objectives	The objective of the course is EMPLOYABILITY of students by using EXPERIENTIAL LEARNING techniques.					
Course Out Comes:	On successful completion of this course the students shall be able to: 5. **Summarize** the concepts of digital image processing (Understand). 6. **Describe** the various Image formation techniques and its difficulties (Understand). 7. **Apply** the linear filter to images (Apply). 8. **Use** EM Algorithm for identifying the segments in an image (Apply).					
Course Content:	No. of Sessions: 15 (30 hours)					

Imagen, StyleGAN.

Targeted Application & Tools that can be used:

1. Google Colab
2. Python IDEs like PyCharm

Project Work/Assignment: Group project on a Computer Vision task like image captioning, sentiment analysis, etc.

Textbook(s):

1. Ian Goodfellow, Yoshua Bengio, Aaron Courville. *Deep Learning*, (1st Edition). The MIT Press, 2016.
2. David Forsyth, Jean Ponce. *Computer Vision: A Modern Approach* (2nd Edition). Pearson Education India, 2015.
3. David Foster. *Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play* (2nd Edition). O'Reilly, 2023.

References:

R1. Vineeth Balasubramanian. *Deep Learning for Computer Vision* (1st Edition). NPTEL. 2020.

Weblinks

W1. NPTEL Course: https://nptel.ac.in/courses/106106224

Date of Approval by the Academic Council

Recommended by the Board of Studies on

Catalogue prepared by

Experiment No. 1: Image File Handling

Level 1: Read image files using Python.

Level 2: Save image files using Python.

Experiment No. 2: Introduction to Digital Image Processing

Level 1: Perform image enhancement operations

Level 2: Perform filtering with morphological operators

Experiment No. 3: Image Denoising

Level 1: Perform image denoising operation.

Level 2: Noise removal using a Weiner filter.

Experiment No. 4: Image Segmentation

Level 1: Perform edge-based and region-based image segmentation

Level 2: Perform image labelling.

Experiment No. 5 & 6: Image Classification

Level 1: Perform image classification using Logistic Regression and Support Vector Machine

Level 2: Perform image classification using Multilayer Perceptron and CNN.

Experiment No. 7 & 8: Object Detection

Level 1: Detect objects in a scene using HOG

Level 2: Detect objects in a scene using CNN

Experiment No. 9 & 10: Optical Character Recognition
Level 1: Implement a CNN to detect printed characters in various fonts.
Level 2: Implement a CNN to detect and decipher handwritten characters.

Experiment No. 11: Image Generation Using DALL-E
Level 1: Using GPT Vision model for text to image generation
Level 2: Creating an image by first creating a prompt and then an image.

Experiment No. 12: Generative Adversarial Network
Level 1: Implement a GAN for neural style transfer.
Level 2: Use a GAN to generate a Ghiblified image of an event.

Experiment No. 13: Image to Image Generation
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. 14 & 15: Subtitle Generation
Level 1: Generate subtitles for a video in English.
Level 2: Generate ***English*** subtitles for an anime (Japanese animated) video. NOTE: The audio here will be in Japanese!

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

Students will perform a shared task in the semester.

Textbook(s):

1. Ian Goodfellow, Yoshua Bengio, Aaron Courville. *Deep Learning*, (1st Edition). The MIT Press, 2016.
2. David Forsyth, Jean Ponce. *Computer Vision: A Modern Approach* (2nd Edition). Pearson Education India, 2015.
3. David Foster. *Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play* (2nd Edition). O'Reilly, 2023.

References:

R1. Vineeth Balasubramaniam. *Deep Learning for Computer Vision* (1st Edition). NPTEL. 2020.

Weblinks

W1. NPTEL Course: https://nptel.ac.in/courses/106106224

Catalogue prepared by	Dr. Sandeep Albert Mathias
Recommended by the Board of Studies on	BOS NO: SOCSE 2nd BOS held on 17/03/25
Date of Approval by the Academic Council	Academic Council Meeting No 21, Dated 17/03/25

Course Code: COM3400	Course Title: Digital Signal Processing Type of Course: Lab Integrated course		L- T-P- C	2	0	2	3
Version No.	1						
Course Pre-requisites	NIL						
Anti-requisites	NIL						
Course Description	This course introduces the fundamental principles and techniques of Digital Signal Processing (DSP). It focuses on discrete-time signals and systems, frequency domain analysis, digital filter design, and applications of DSP in areas such as audio processing, communications, and biomedical signal analysis. Emphasis is placed on algorithm development, implementation aspects, and real-world applications using tools like MATLAB and Python.						
Course Objective	To equip students with both theoretical understanding and practical skills in processing digital signals, enhancing their problem-solving capabilities for real-time applications in signal-intensive systems.						
Course Outcomes	On successful completion of this course the students shall be able to: CO1: Understand the mathematical foundations of digital signal processing. (Knowledge) CO2: Analyze and design discrete-time systems using time and frequency domain techniques. (Comprehension) CO3: Apply digital filters to real-world signal processing problems. (Application) CO4: Simulate and implement DSP algorithms using programming tools. (Application)						
Course Content:							
Module 1	Introduction to DSP and Discrete-Time Signals	Assignment	Comprehension based Quizzes and assignments;	9 sessions			
Topics: Basic elements of DSP, classification of signals, discrete-time signals and systems, LTI systems, convolution, correlation, difference equations, properties of systems, stability, and causality.							
Module 2	Z-Transform and Frequency Analysis	Assignments	Comprehension based Quizzes and assignments	9 sessions			
Topics: Z-transform and its properties, inverse Z-transform, pole-zero analysis, frequency response, discrete Fourier transform (DFT), FFT algorithms and their implementation.							
Module 3	Digital Filter Design	Quizzes and assignments	Comprehension based Quizzes and assignments	12 sessions			

Topics: Design of FIR and IIR filters, windowing techniques, Butterworth and Chebyshev filters, filter realization structures, quantization effects in filter design, stability and performance analysis.				
Module 4	Applications of DSP and Real-Time Processing	Case study	Comprehension based Quizzes and assignments	10 sessions
Topics: DSP in audio and speech processing, image and video signal processing, DSP in biomedical signals (EEG, ECG), embedded DSP, introduction to DSP processors, case study on real-time signal processing with MATLAB/Simulink or Python.				

Textbook:

1. Oppenheim, A. V., Schafer, R. W., & Buck, J. R. (1999). Discrete-Time Signal Processing. Pearson Education.
2. Proakis, J. G., & Manolakis, D. G. (2007). Digital Signal Processing: Principles, Algorithms, and Applications. Prentice Hall.
3. Smith, Steven W. (1997). The Scientist and Engineer's Guide to Digital Signal Processing. California Technical Publishing.

References

3. Kappelmann-Fenzl, Melanie, ed. *Next Generation Sequencing and Data Analysis*. Heidelberg, 2021.
4. S. Balamurugan, Anand T. Krishnan, Dinesh Goyal, Balakumar Chandrasekaran, Computation in Bioinformatics, Multidisciplinary Applications, Wiley,2021

Web References:

3. https://www.coursera.org/specializations/bioinformatics
4. https://onlinecourses.nptel.ac.in/noc21_bt06/preview

Topics relevant to "Skill Development": MATLAB/Simulink-based implementation of filters, Python-based simulation of DSP algorithms, real-time signal analysis projects.

Catalogue prepared by	Dr. Pamela Vinitha Eric
Recommended by the Board of Studies on	
Date of Approval by the Academic Council	

Course Code: COM3403	**Course Title: Edge and Fog Computing** **Type of Course: Theory Only Course Discipline Elective**	**L-P-C**	**3**	**0**	**3**
Version No.	1.0				
Course Pre-requisites	Distributed Systems and Algorithms				
Anti-requisites	**Nil**				
Course Description	In this course, the students will study significant tools and applications that comprise today's cloud computing platform, with a special focus on using the cloud for big data applications. The course covers various topics such as the evolution of the computing industry, cloud computing basics, and edge computing. The course provides information on the different types of edge compute deployments, and different types of edge compute services (such as CDN Edge, IOT Edge, and Multi-access Edge (MEC)). The course also educates the students on the different vendor platforms, software services, standard bodies, and open source communities available for edge computing. Students will also create a research project of their choosing.				
Course Objective	The objective of the course is the skill development of students by using Participative Learning techniques				
Course Out Comes	On successful completion of the course, the students shall be able to: CO1 Understand the principles, and architectures of edge computing (Knowledge) CO2 Describe IoT Architecture and Core IoT Modules (Comprehension) CO3 Summarize Edge to Cloud Protocols (Comprehension) CO4 Demonstrate Edge computing with RaspberryPi (Comprehension)				
Course Content:					
Module 1	Introduction to Edge and Fog Computing	Term paper/Assignment/Case Study	Programming/Simulation/Data Collection/any other such associated activity	**9 Sessions**	
Topics: Fundamentals of Distributed Computing, Evolution from Cloud to Edge and Fog Computing, Need for Edge and Fog Computing in IoT and AI Applications, Key Differences: Cloud vs. Edge vs. Fog Computing, Architecture and Components of Edge and Fog Computing, Real-World Use Cases: Smart Cities, Healthcare, and Industry 4.0					
Module 2	Edge Computing: Architecture	Term paper/Assignment/**Case Study**	Programming/Simulation/Data Collection/any other such associated activity	**9 Sessions**	

	, Platforms, and Technologies			
Topics: Edge Devices and Edge Nodes, Edge Computing Infrastructure and Middleware, Edge AI: Running AI/ML Models on Edge Devices, Communication Protocols: MQTT, CoAP, and 5G in Edge Computing, Edge-Oriented Frameworks: AWS Greengrass, Azure IoT Edge, Google Edge TPU, Case Study: Edge Computing in Autonomous Vehicles				
Module 3	Fog Computing: Concepts, Architectures, and Security	Term paper/Assignment/Case Study	Programming/Simulation/Data Collection/any other such associated activity	**7 Sessions**
Topics: Fog Computing vs. Edge Computing: Key Differences, Fog Node Deployment and Network Considerations, Fog Computing Architecture and Middleware, Security and Privacy Challenges in Fog Computing, Resource Management and Orchestration in Fog Environments, Case Study: Fog Computing in Industrial Automation and Smart Grids				
Module 4	Integration, Applications, and Future Trends	Term paper/Assignment/Case Study	**Programming/Simulation**/ Data Collection/any other such associated activity	**7 Sessions**
Topics: Integration of Fog, Edge, and Cloud Computing, Role of Edge and Fog in 5G and Beyond Energy Efficiency and Sustainability in Edge and Fog Computing, Blockchain and Fog Computing for Secure Transactions, Challenges, Open Research Areas, and Future Trends, Hands-on Project: Deploying a Fog-Enabled IoT System				

Targeted Applications & Tools that can be used:

- **Application** : Smart Surveillance Video Stream Processing at the Edge for Real-Time Human Objects Tracking.
- **Tools** :Eclipse ioFog: An integrated development environment built by the Eclipse Foundation, backed by IBM. Eclipse ioFog is the organization's open-source edge computing platform.

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

Exploring topics such as developing scalable architectures, moving from closed systems to open systems, and ethical issues rising from data sensing, addresses both the challenges and opportunities of Edge computing presents. Students can harness federating Edge resources, middleware design issues, data management and predictive analysis, smart transportation and surveillance applications, and more. A coordinated and integrated solutions can be provided by thorough knowledge of the foundations, applications, and issues that are central to Edge computing.

Text Book

1. **Buyya, R., & Srirama, S. N. (Eds.). (2019).** *Fog and edge computing: Principles and paradigms.*Wiley. I**SBN:** 978-1-119-52498-4, DOI: 10.1002/9781119525085,

2. **Satyanarayanan, M. (2019).** *Edge computing: A primer*. Carnegie Mellon University.	
Topics relevant to development of "Skill Development": Implementation of Microcomputer RaspberryPi and device Interfacing	
Catalogue prepared by	
Recommended by the Board of Studies on	
Date of Approval by the Academic Council	

Course Code: **COM3404**	**Course Title:** **Cloud Security and Governance** **Type of Course:** **Discipline Elective in Cloud Computing Basket** **Theory**	**L-T-P- C**	3	0	0	3
Version No.	1.0					
Course Pre-requisites	**Cloud Computing**					
Anti-requisites	**NIL**					
Course Description	This course provides ground-up coverage on the high-level concepts of cloud landscape, architectural principles, and techniques. It describes the Cloud security architecture and explores the guiding security for Infrastructure and Software.					
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. **Explain** the fundamentals of cloud security and governance frameworks. **[Knowledge]** 2. **Analyze** cloud security architectures and apply identity and access management principles. **[Comprehension]** 3. **Evaluate** cloud data security, encryption techniques, and compliance standards. **[Evaluation]** 4. **Apply** network security, virtualization security, and threat management techniques in cloud environments. **[Application]** 5. **Assess** cloud security risks and propose mitigation strategies for different cloud service models. **[Analysis]** 6. **Develop** security policies and disaster recovery plans for cloud-based systems. **[Synthesis]**					
Course Content:						

Module 1:	**Introduction to Cloud Security and Governance**	**Quiz**		Knowledge-based Quiz	**10 Sessions**
	Topics: Fundamentals of Cloud Security, Cloud Governance Frameworks and Compliance, Shared Responsibility Model in Cloud Security, Security Considerations in Cloud Deployment Models, Risk Management and Threat Landscape in Cloud, Cloud Service Models and Security Implications (SaaS, PaaS, IaaS), Compliance Standards and Regulations (GDPR, HIPAA, ISO 27001)				
Module 2:	**Cloud Security Architecture and Identity Management**	**Quiz**		Comprehension based Quiz	**10 Sessions**
	Topics: Cloud Security Architecture and Design Principles, Identity and Access Management (IAM) in Cloud, Zero Trust Security Model for Cloud Environments, Role-Based and Attribute-Based Access Control, Authentication and Authorization in Cloud, Security Policy and Governance Best Practices, Cloud Security Automation and Orchestration				
Module 3	**Data Security, Privacy, and Compliance in Cloud**	**Assignment**		Batch-wise Assignments	**9 Sessions**
	Topics: Cloud Data Security Challenges and Best Practices, Data Encryption and Key Management in Cloud, Secure Data Storage and Transmission Techniques, Data Loss Prevention (DLP) in Cloud, Privacy-Preserving Techniques in Cloud Computing, Regulatory Compliance for Data Protection (CCPA, GDPR), Cloud Forensics and Incident Response				
Module 4:	**Cloud Infrastructure Security and Threat Management**	Assignment and Presentation		Batch-wise Assignment and Presentations	**9 Sessions**
	Topics: Cloud Network Security and Secure Configurations, Virtualization and Container Security in Cloud, Securing APIs and Microservices in Cloud Environments, Cloud Security Threats: DDoS, Malware, Insider Threats, Cloud Security Monitoring, Logging, and Threat Intelligence, Incident Response and Disaster Recovery in Cloud, Future Trends in Cloud Security and Governance				
	Targeted Application & Tools that can be used: **Use of CloudSim simulator.**				
	Project work/Assignment: **Survey on Cloud Service Providers**				
	Text Book 1. Tim Mather, Subra Kumaraswamy, and Shahed Latif **, "Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance", Publisher:** O'Reilly Media, **ISBN:** 978-0596802769 2. Roland L Krutz and Russell Dean Vines, "Cloud Security - A Comprehensive Guide to Secure Cloud Computing", Wiley Publishing, Inc. 2019.				
	References **1.** Sushil Jajodia, Krishna Kant, Pierangela Samarati, Anoop Singhal, Vipin Swarup, Cliff Wang, "*Secure Cloud Computing*", Springer, ISBN 978-1-4614-9278-8 (eBook). **2.** John Rittinghouse and James Ransome, "*Cloud Computing, Implementation, Management and Security*", CRC Press, 2010. **3.** Tim Mather, Subra Kumaraswamy, and Shahed Latif", "Cloud Security and Privacy – An Enterprise Perspective on Risks and Compliance", Oreily Publication, 2009.				
	Topics related to the development of "FOUNDATION": Cloud computing architecture, Security policy implementation. Topics related to the development of "EMPLOYABILITY": Infrastructure security and Data security.				
Catalogue prepared by					

Recommended by the Board of Studies on		
Date of Approval by the Academic Council		

Course Code: CAI3427	**Course Title: Language Models for Text Mining** **Type of Course: Discipline Elective - Theory & Integrated Laboratory**	**L-T-P-C**	2	0	0	2
Version No.	1.0					
Course Pre-requisites	CSE3001 – Artificial Intelligence and Machine Learning					
Anti-requisites	NIL					
Course Description	This course introduces the basics of Text Mining and Natural Language Processing. The course will teach students different concepts such as text mining, NLP, Sequence Labeling, etc. Topics: Text Mining, NLP, Tokenization, Lemmatization, Stemming, One-hot encoding, Language modelling, Bag-of-words, Term-document Matrix, Cosine similarity, Viterbi Algorithm, etc.					
Course Objectives	The objective of the course is EMPLOYBILITY of student by using EXPERIENTIAL LEARNING techniques.					
Course Out Comes	On successful completion of this course the students shall be able to: 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]

Course Content:

Module 1	**Text Mining**	**Adversarial Quiz Tests**	**Module Tests**	**No. of Sessions: 09**
Introduction to Text Mining. Text Mining vs. NLP. Text Mining Algorithms. Steps in Text Mining - Extraction, Preprocessing, Analysis and Evaluation. Lexical Resource Creation (NEW). Data collection. String Manipulation to Clean Data. Natural Language Processing. Research Paradigms in NLP. Sequential Data. Sequence Labeling (NEW). Viterbi Algorithm (NEW). Corpus. Building a HMM using a Corpus (NEW). Unknown word handling (NEW).				
Module 2	**Text Preprocessing**	**Adversarial Quiz Tests**	**Module Tests**	**No. of sessions: 06**
Introduction to Preprocessing. Tokenization. Stop Words Removal. Lemmatization and Stemming. PoS Tagging. Integer Encoding. Padding. One-Hot Encoding.				
Module 3	**Text Representations**	**Adversarial Quiz Tests**	**Module Tests**	**No. of sessions: 08**
Language Modeling. N-Gram Language Model. Bag-of-Words Model. Term-Document Matrix. Term Frequency. Inverse Document Frequency. TF-IDF. Cosine Similarity. Naive Bayes Classifier using Bag-of-Words. Topic Modeling. Latent Semantic Analysis. Singular Value Decomposition. Truncated SVD and Topic Vector. LDA Algorithm.				
Module 4	**Natural Language Processing with Keras**	**Adversarial Quiz Tests**	**Module Tests**	**No. of Sessions: 06**
Word Embeddings vs. One-Hot Encoding. Contextual Bag of Words (CBOW). Skipgram. Deep Learning for Document Classification.				

List of Laboratory Tasks:

Experiment No. 1: File Handling

Level 1: Read text files using Python and extract meaningful content.

Level 2: Parse text files using Python to preprocess the data for NLP tasks.

Experiment No. 2: Introduction to NLP Tools

Level 1: Install and use NLTK for basic text processing.

Level 2: Install and use SpaCy for tokenization, PoS tagging, and Named Entity Recognition.

Experiment No. 3: Corpus Cleaning Techniques

Level 1: Use NLTK for corpus cleaning techniques such as tokenization, stopword removal, and stemming.

Level 2: Prepare cleaned text data for downstream NLP tasks like classification or translation.

Experiment No. 4: Word Vector Usage

Level 1: Download and use pre-trained word vectors (e.g., Word2Vec, GloVe, or FastText).

Level 2: Compute similarity between two words, find the most similar word, and complete word analogies (e.g., king - man + woman = queen).

Experiment No. 5 & 6: Language Identification

Level 1: Build a simple language identifier using Bag-of-Words (BoW) features.

Level 2: Predict the language of a given text using the trained model.

Experiment No. 7 & 8: Lexical Simplification

Level 1: Implement a lexical simplifier to replace complex words with simpler alternatives.

Level 2: Generate a simplified version of a given word or sentence while preserving meaning.

Experiment No. 9 & 10: Sentiment Analysis

Level 1: Implement a basic sentiment classifier using a lexicon-based or machine learning approach.

Level 2: Compare the performance of an existing sentiment classifier (e.g., VADER, TextBlob, or a pre-trained Transformer model).

Experiment No. 11: Named Entity Recognition (NER)

Level 1: Extract named entities from a text using NLTK.

Level 2: Extract named entities using SpaCy and compare results.

Experiment No. 12 & 13: Implement a Hidden Markov Model (HMM)

Level 1: Implement a generic HMM for sequence prediction.

Level 2: Calculate the forward probability of a given sequence using HMM.

Experiment No. 14: Linguistic HMM

Level 1: Develop a Hidden Markov Model (HMM) for NLP tasks such as PoS tagging.

Level 2: Evaluate the performance of the HMM on a specific NLP task (e.g., Named Entity Recognition or Chunking).

Experiment No. 15: Machine Translation

Level 1: Implement Machine Translation (MT) using a pre-trained model from Hugging Face Transformers.

Level 2: Evaluate the quality of MT output via Round-Trip Translation (translate text to another language and back to check accuracy).

Targeted Application & Tools that can be used:

1. Google Colab
2. Python IDEs like PyCharm

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

1. 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.

Course Code: CAI3428	**Course Title:** **Practical Deep Learning with TensorFlow** **Type of Course: Theory & Integrated Laboratory**	**L- T-P- C**	**2**	**0**	**2**		**3**
Version No.	1.0						
Course Pre requisites	CSE 3001 Artificial Intelligence and						
Anti-requisites	NIL						

Textbook(s):

1. Daniel Jurafsky, James H. Martin. "Speech and Language Processing: An Introduction to Natural Language Processing", Computational Linguistics and Speech, Pearson Publication, 2025 (3rd Edition Draft).
2. Aditya Joshi, Pushpak Bhattacharyya. "Natural Language Processing", Wiley Publication, 2023 (1st Edition).

References:

R1. Chris Manning and Hinrich Schutze, "Foundations of Statistical Natural Language Processing", 1st Edition, MIT Press. 1999.

R2. Pawan Goyal, "Natural Language Processing", 1st Edition, 2016.

Weblinks

W1. E-Book link or R2: https://drive.google.com/file/d/10nbwAJd-dv6htOOZVBgAvLd1WscI0RqC/view

W2. Web Resource for T1: https://web.stanford.edu/~jurafsky/slp3/ - VERY VERY IMPORTANT!!!

W3. NPTEL Courses: https://nptel.ac.in/courses/106106211 CMI), https://nptel.ac.in/courses/106105158 (IIT Kgp), https://nptel.ac.in/courses/106101007 (IITB), https://nptel.ac.in/courses/106105572 (IIT Kgp - NEW)

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

COURSE TITLE & CODE : Artificial Intelligence and Machine Learning & CSE3157

COURSE CREDIT STRUCTURE : 3-0-2-4

CONTACT SESSIONS : 45 + 30 = 75

Asad Mohammed Khan, SAPTARSI SANYAL, Dr. Zafar Ali Khan N

COURSE PREREQUISITES:CSE1005 – Innovative Project – Python Programming

COURSE DESCRIPTION:

This course introduces the basic concepts of artificial intelligence. It introduces students to the basic concepts and techniques of Machine Learning (ML), a subset of Artificial Intelligence (AI), is an important set of techniques and algorithms used for solving several business and social problems. The objective of this course is to discuss machine learning model development using Python.

Topics include: Working with Collections and Data Frames; Regression algorithms; Classification algorithms; Optimization techniques – Gradient Descent algorithm, Gradient Descent for simple Linear Regression; Ensemble Learning – Random Forest, Boosting techniques – AdaBoost and Gradient Boosting; Grid Search for optimal parameters; Clustering algorithms; Forecasting with Time-Series data : Auto-Regressive Integrated Moving Average Models, Recommender Systems : Association Rule Mining, Collaborative Filtering, Text Analytics – Sentiment Classification using Naïve Bayesian model.

COURSE OBJECTIVES: The objective of the course is to familiarize the learners with the concepts of

COURSE CONTENT (SYLLABUS):

Module 1: Introduction to Artificial Intelligence and Searching
[L-10 P-6 Total: 16 sessions] [Understand]
Introduction to Artificial Intelligence, Definitions, foundation, History and Applications; Agents: Types of Agent, Structure of Intelligent agent and its functions, Agents and Environment; Indexing and Heuristic functions -Hill Climbing-Depth first and Breath first; A* - SMA* algorithms.

Module 2: Knowledge Representation. [L-12, P-6 Total: 18 sessions] [Apply]

Introduction to Knowledge representation, approaches and issues in knowledge representation, Knowledge-based agent and its Structure, Knowledge-Based Systems; Knowledge representation using Propositional logic and Predicate Logic- First-Order Logic - Syntax and Semantics, Knowledge Engineering - Unification and lifting, Forward chaining, Backward chaining.

Module: 3: Introduction to Machine Learning and Supervised & Unsupervised Learning [L-12 P-10 Total: 22 sessions] [Apply]
Introduction to the Machine Learning (ML) Framework, types of ML, types of variables/features used in ML algorithms, Concept Learning: Concept learning task, Concept learning as search, Find-S algorithm, Candidate Elimination Algorithm.

Supervised Learning – Classification & Regression - Decision Tree Learning, Random Forest - Support Vector Machines ; Simple Linear Regression Algorithm, Multivariate Regression Algorithm

Module 4: Machine Learning & Neural Network [L-11 P-8 Total: 19 sessions][Apply]

Neural and Belief networks - Perceptron - Multi-layer feed forward networks - Bayesian belief networks, Back propagation algorithm.

Unsupervised Learning – Clustering & Association - K-Means Clustering algorithm , Mean-shift algorithm , Apriori Algorithm, FP-growth algorithm

REFERENCE MATERIALS:

Textbook(s):

T1 Stuart J. Russell and Peter Norvig, Artificial intelligence: A Modern Approach, 3rd edition, Upper Saddle River, Prentice Hall 2021.

T2 Tom Mitchell, "Machine Learning", First Edition, Tata McGraw Hill India, 2017.

Course Code: CSE3216	**Course Title:** Mastering Object- Oriented Concepts in Python **Type of Course:** Lab	**L- T- P- C**	0	0		2
Version No.	1					
Course Pre-requisites	CSE1005 – Programming in Python					
Anti-requisites	NIL					
Course Description	This course covers mastering object-oriented concepts in Python, including classes, inheritance, polymorphism, and encapsulation. Students will learn to design and implement robust, reusable code using real-world examples. Ideal for those with basic Python knowledge, it enhances problem-solving skills and software development proficiency.					
Course Objective	**The objective of the course is to familiarize the learners with the concepts of Mastering Object Oriented Concepts in Python and attain Skill Development through Experiential Learning.**					
Course Out Comes	**CO1:** Explain features of Oops along with creation of Python classes and objects to represent real world Objects. [Understand] CO2: Demonstrate inheritance, polymorphism, and abstraction in Python to build maintainable and extendable software systems.[Apply] CO3: Demonstrate exception handling in Python to build robust error-handling mechanisms and debugging tool and Assess various file handling techniques in Python. [Apply]					
Course Content:						
Module 1	**Introduction to OOPS, Classes and Objects**	MCQ	Assignment			**10 Sessions**
Topics: **Introduction to OOPs:** Problems in Procedure Oriented Approach, Specialty of Python Language, Features of OOPS - Classes and Objects, Encapsulation, Abstraction, Inheritance and Polymorphism. **Classes and Objects:** Creating a Class, The Self Variable, Constructor, Destructors, Types of Variables, Namespaces, Types of Methods - Instance Methods, Class Methods, Static Methods, Passing Members of One Class to Another Class, Inner Classes.						
Module 2	**Inheritance and Polymorphism**	MCQ	Assignment			**10 Sessions**

Constructors in Inheritance, Overriding Super Class Constructors and Methods, The Super() Method, Types of Inheritance – Single Inheritance, Multiple Inheritance, Method Resolution Order(MRO), Polymorphism, Duck Typing Philosophy of Python, Operator Overloading, Method Overloading, Method Overriding.

Abstract Classes and Interfaces: Abstract Method and Abstract Class, Interfaces in Python, Abstract Classes vs. Interfaces.

Module 3	**Exceptions and Files in Python**	MCQ	Assignment	**10 Sessions**

Exceptions: Errors in a Python Program – Compile-Time Errors, Runtime Errors, Logical Errors. Exceptions, Exception Handling, Types of Exceptions, The Except Block, The assert Statement, User-Defined Exceptions, Logging the Exceptions.

Files in Python: Files, Types of Files in Python, Opening a File, Closing a File, Working with Text Files Containing Strings, Knowing whether a File Exists or Not, Working with Binary Files, The with Statement, Pickle in Python, The seek() and tell() Methods.

Targeted Application & Tools that can be used:

Python, PyCharm

Project work/Assignment:

Assignment:

Module 1 Assignment: Design and implement a Python application that simulates a banking system using classes and methods for customers and accounts.

Module 2 Assignment: Develop a Python application that simulates Library management system that demonstrates inheritance, polymorphism and abstraction concepts.

Module 3 Assignment: Develop a Python program that handles different types of exceptions while processing user input for a movie ticket booking system showcasing exception handling and File handling concepts.

Text Book

1. Dr. R Nageshwara Rao, "Core Python Programming", Dreamtech Press, 3rd Edition, 2021.

References

1. Alex Martelli, Anna Ravenscroft & Steve Holden, "Python in a Nutshell The Definitive Reference", O'Reilly Media, 3rd edition, 2017.
2. Luciano Ramalho, "Fluent Python Clear, Concise, and Effective Programming", O'Reilly Media, 2nd edition, 2022.
3. Mark Lutz, "Learning Python: Powerful Object-Oriented Programming", O'Reilly Media, 5th edition, 2013.
4. David Beazley, Brian K. Jones, "Python Cookbook: Recipes for Mastering Python 3", O'Reilly Media, 3rd edition, 2013.

Weblinks:

1. www.learnpython.org
2. https://realpython.com/python3-object-oriented
3. https://www.tutorialspoint.com/python/python_oops_concepts.htm

Topics relevant to "SKILL DEVELOPMENT":

Building Real-World Applications Using OOPS Concepts, Error Handling and Debugging Techniques, Concurrency in Python, Advanced File Handling Techniques, Creating and Managing Python Packages and Modules, Designing and Implementing Python Interfaces

This is attained through assessment component mentioned in course handout.

Catalogue prepared by							
Recommended by the Board of Studies on							
Date of Approval by the Academic Council							
Course Code: COM3401	Course Title: Advanced Computer Architecture	L-T-P-C	3	0	0		3
	Type of Course: Discipline Elective -Theory						
Version No	1.0						
Course Pre-requisites	Computer Organization and Architecture						
Anti-requisites	NIL						

Course Description	This course explores the design and analysis of modern computer architectures beyond the basics of conventional systems. It emphasizes high-performance architectures such as superscalar, VLIW, and multicore processors. Topics include instruction-level parallelism, memory hierarchy optimization, cache coherence protocols, interconnection networks, and parallel programming models. The course also covers emerging trends like heterogeneous computing, GPU architectures, and domain-specific accelerators. Through analytical techniques and simulation tools, students gain insights into performance evaluation and architectural trade-offs critical for designing next-generation computing systems.
Course Objectives	The objective of the course is to familiarize students with sophisticated memory subsystems including multi-level caches, virtual memory, memory consistency models, and optimization strategies.
Course Out Comes	On successful completion of this course the students shall be able to: 1: Recall fundamental concepts of computer organization and architecture. *(Knowledge)* 2: Evaluate processor performance using quantitative metrics such as CPI, MIPS, and Amdahl's Law. *(Evaluation)* 3: Explain the basics of instruction pipelines and the RISC architecture principles. *(Comprehension)* 4: Analyze the impact of pipelining on instruction throughput and performance. *(Analysis)*

Course Content:

Module 1	Review of Basic Computer Organization	Assignment		No. of sessions:10
Review of Basic Computer Organization, Performance Evaluation Methods, Introduction to RISC Instruction Pipeline, Instruction Pipeline and Performance.				
Module 2	Pipeline Hazards	Assignment		No. of sessions:10

Pipeline Hazards and Analysis, Branch Prediction, MIPS Pipeline for Multi-Cycle Operations.				
Module 3	Compiler Techniques	Assignment		No. of sessions:10
Compiler Techniques to Explore Instruction Level Parallelism, Dynamic Scheduling with Tomasulo's Algorithm and Speculative Execution.				
Module 4	Advanced Pipelining	Assignment		No. of Sessions: 10
Advanced Pipelining and Superscalar Processors, Exploiting Data Level Parallelism: Vector and GPU Architectures, Architectural Simulation using gem5.				

Targeted Application & Tools that can be used:

3. **OpenMP / MPI libraries** – For writing and analyzing parallel code
4. **CUDA Toolkit** – To explore GPU architecture and GPGPU computing

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):

3. Computer Architecture - A Quantitative Approach,5th edition, John L. Hennessy, David A. Patterson.
4. Computer Systems Design and Architecture, 2nd Edition, Vincent P. Heuring
5. Computer Organization and Architecture, 6th Edition, William Stallings
6. Advanced Computer Architectures-A Design Space Approach, Dezsosima, Terence Fountain, Peter Kacsuk.

Catalogue prepared by	
Recommended by the Board of Studies on	
Date of Approval by the Academic Council	

Course	Course Title: **Environmental Science**	L- T- P- C	1	0	2	0

Code: CHE1018	Type of Course: School Core- Theory and Lab	Contact hours	1	0	2	3
Version No.	2.0					
Course Pre-requisites	NIL					
Anti-requisites	**NIL**					
Course Description	This course emphasizes the need to conserve biodiversity and adopt a more sustainable lifestyle by utilizing resources in a responsible way. Topics covered include basic principles of ecosystem functions; biodiversity and its conservation; human population growth; water resources, pollution; climate change; energy resources, and sustainability; Sustaining human societies, policies, and education. **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 "Environmental Science" and attain SKILL DEVELOPMENT through EXPERIENTIAL LEARNING techniques.**					
Course Outcomes	On successful completion of this course the students shall be able to: 1) Appreciate the historical context of human interactions with the environment and the need for eco-balance. 2) Describe basic knowledge about global climate change with particular reference to the Indian context. 3) Understand biodiversity and its conservation 4) Develop an understanding on types of pollution and ways to protect the environment 5) Learn about various strategies on Global environmental management systems					
Course Content:						
Module 1	**Humans and the Environment**	Assignment	Data Collection		**01 class**	
Topics: The man-environment interaction: Mastery of fire; Origin of agriculture; Emergence of city-states; Great ancient civilizations and the environment. ***Self-learning topics:*** Humans as hunter-gatherers; Industrial revolution and its impact on the environment; Environmental Ethics and emergence of environmentalism.						
Module 2	**Natural Resources and Sustainable Development**	Assignment			**03 Classes**	

Topics:
Overview of natural resources: Definition of resource; Classification of natural resources- biotic and abiotic, renewable and non-renewable. **Water resources**: Types of water resources- fresh water and marine resources;

Soil and mineral resources: Important minerals; Mineral exploitation Soil as a resource and its degradation.

Energy resources: Sources of energy and their classification, renewable and non-renewable sources of energy; Advantages and disadvantages.

Self- learning topics: Availability and use of water resources; Environmental impact of over-exploitation, issues and challenges.; Environmental problems due to extraction of minerals and use; Sustainable Development Goals (SDGs)- targets, indicators, and challenges for SDGs.

Module 3	**Environmental Issues: Local, Regional and Global**	Case study		**02 Classes**

Topics:
Environmental Pollution: Types of Pollution- air, noise, water, soil, municipal solid waste, hazardous waste; Trans- boundary air pollution; Acid rain; Smog.

Land use and Land cover change: land degradation, deforestation, desertification, urbanization. Global change: Ozone layer depletion; Climate change

Self -learning topics: Environmental issues and scales

	Module 4	**Conservation of Biodiversity and Ecosystems**	Assignment		**02 Classes**

Topics:
Biodiversity-Introduction, types, Species interactions, Extinct, endemic, endangered and rare species, Threats to biodiversity: Natural and anthropogenic activities.

Self-learning topics: Mega-biodiversity, Hot-spots, Major conservation policies. Biodiversity loss: past and current trends, impact.

	Module 5	**Environmental Pollution and Health**	Case study		**03 Classes**

Topics:
Pollution, Definition, point and nonpoint sources of pollution, **Air pollution**- sources, major air pollutants, health impacts of air pollution.

Water pollution– Pollution sources, adverse health impacts on human and aquatic life and mitigation, Water quality parameters and standards.

Soil pollution and solid waste- Soil pollutants and their sources, solid and hazardous waste, Impact on human health.

Self-learning topics: Noise pollution, Thermal and radioactive pollution.

	Module 6	**Climate Change: Impacts, Adaptation and Mitigation**	Assignment/case		**02 Classes**

Topics:
Understanding climate change: Natural variations in climate; Projections of global climate change with special reference to temperature, rainfall and extreme events; Importance of 1.5 °C and 2.0 °C limits to global warming; Impacts

Vulnerability and adaptation to climate change: Observed impacts of climate change on ocean and land systems; Sea level rise, changes in marine and coastal ecosystems; Impacts on forests and natural ecosystems; Indigenous knowledge for adaptation to climate change.

Self-learning topics: Mitigation of climate change: Synergies between adaptation and mitigation measures; National and international policy instruments for mitigation.

	Module 7	**Environmental Management**	Case study	Data analysis	**02 Classes**

Topics:
Environmental management system: ISO 14001; Environmental risk assessment Pollution control and management; Waste Management- Concept of 3R (Reduce, Recycle and Reuse) and sustainability.

Self-learning topics: Environmental audit and impact assessment; Eco labeling /Eco mark scheme

	Module 8	**Environmental Treaties and Legislation**	Case study	Data analysis	**01 Classes**

Topics:
Major International Environmental Agreements: Convention on Biological Diversity (CBD), Major Indian Environmental Legislations: Environmental Protection Act, Forest Conservation Act, Public awareness.

Self-learning topics: Paris Agreement, Conference of the Parties (COP), India's status as a party to major conventions: Air (Prevention and Control of Pollution) Act, Water (Prevention and control of Pollution) Act, Wildlife Protection Act.

List of laboratory tasks : Any eight experiments will be conducted

1. Determination of total alkalinity of a water sample (knowledge)
2. Estimation of water hardness by EDTA method and its removal (by zeolite/ ion exchange method) (Comprehensive)
3. Estimation of copper from industrial effluents by colorimetric method (Comprehensive)
4. Estimation of iron from industrial effluents by titrimetric method/potentiometric method (Comprehensive)
5. Estimation of nickel from industrial effluents by titrimetric method (Comprehensive)
6. Estimation of chloride in drinking water by titrimetric method (Comprehensive)
7. Estimation of fluoride in ground water by colorimetric method (Comprehensive)
8. Determination of calcium in aqueous solution (Comprehensive)
9. Determination of Total Dissolved Salts, conductivity and pH of a water samples (Knowledge)
10. Determination of Chemical oxygen demand in the industrial effluent. (Comprehensive)
11. Biological oxygen demand of waste water sample (Comprehensive)
12. Determination of dissolved oxygen of an industrial effluent (Comprehensive)
13. Quality monitoring analysis of a soil sample (knowledge)
14. Flame photometric estimation of Sodium and potassium (Application)
15. Gas Chromatographic analysis of volatile organic compounds (Application)

Targeted Application & Tools that can be used:
Application areas are Energy, Environment and sustainability
Tools: Statistical analysis of environmental pollutants using excel, origin 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.)**
- **Lab evaluation/Assignment**
- **End Term Exam**
- **Self-learning**

Assignment 1: Write a Statement of Environment report of your town/city/state/country

Assignment 2: Individual students will carry out the analyses of polluted solid, liquid, and gaseous samples and propose suitable mitigation measures. A detailed and in-depth report needs to be submitted for each case. This may include preparation of reagents, sample preparation (extraction), chemical analysis carried out, instruments and tools used, data collected and processed, inferences made and conclusions arrived at. Necessary support is given in the form of
lab manual and reference links to e-books.

Text Book

1. G. Tyler Miller and Scott Spoolman (2020), Living in the Environment, 20th Edition, Cengage Learning, USA
2. Krishnamurthy, K.V. (2003) Text book of Biodiversity, Science Publishers, Plymouth, UK.
3. Jackson, A.R. & Jackson, J.M. (2000), Environmental Science: The natural environment and human impact, Pearson Education.

Reference Books

1. Fisher, Michael H. (2018) An Environmental History of India- From Earliest Times to the Twenty-First Century, Cambridge University Press.
2. William P. Cunningham and Mary Ann Cunningham (2017), Principles of Environmental Science: Inquiry & Applications, 8th Edition, McGraw-Hill Education, USA.
3. Sinha N., (2020) Wild and Wilful. Harper Collins, India.
4. www.ipcc.org; https://www.ipcc.ch/report/sixth-assessment-report-cycle/
5. Theodore, M. K. and Theodore, Louis (2021) Introduction to Environmental Management, 2nd Edition. CRC Press.
6. Richard A. Marcantonio, Marc Lame (2022). Environmental Management: Concepts and Practical Skills. Cambridge University Press.

E-resources:

1. **https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DO AB_1_06082022_18126**
2. **https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DO AB_1_06082022_8761**
3. **https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DO AJ_1_02082022_3333**
4. **https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DO AB_1_06082022_3063**
5. **https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DO AB_1_06082022_20719**
6. **https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DO AB_1_06082022_16824**
7. **https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DO AB_1_06082022_3954**
8. **https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DO AB_1_06082022_491**
9. **https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=CU STOM_PACKAGE_16012023_WORLD_BUSINESS_COUNCIL_SUSTAINABLE_488**
10. **https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=CU STOM_PACKAGE_16012023_WORLD_BUSINESS_COUNCIL_SUSTAINABLE_583**
11. **https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=SP RINGER_INDEST_1_171**
12. **https://presiuniv.knimbus.com/user#/searchresult?searchId=3R%20principle&_t=1687427221129**
13. **https://presiuniv.knimbus.com/user#/searchresult?searchId=eco%20labelling&_t=1687427279979**
14. **https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=TE XTBOOK_LIBRARY01_06082022_395&xIndex=4**
15. **https://www.ugc.gov.in/oldpdf/modelcurriculum/env.pdf**

Topics relevant to Skill Development: Industrial revolution and its impact on the environment, Environmental impact of over-exploitation of water resources, pollution and ill effects, lab experiments for Skills development through Problem solving Techniques. This is attained through assessment component mentioned in course handout. All topics in theory component are relevant to Environment and Sustainability.	
Catalog prepared by	Faculties of Department of Chemistry
Recommended by the Board of Studies on	PU/SOE/CHE/BOS-07/2022-23 9th BOS held on 10/07/23
Date of Approval by the Academic Council	21st Academic council dated: 6th September 2023

Course Code: **CSN2508**	**Course Title:CSN2508 Neural Networks and Fuzzy Logic** **Type of Course: Discipline Elective in AI & ML Basket** **Theory Course**	**L-T-P-C**	3	0	0	3
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	NIL					
Course Description	This course aims to introduce the basic concepts of Neural Networks and Fuzzy Logic. Neural networks reflect the behavior of the human brain, allowing computer programs to recognize patterns and solve common problems in the fields of AI, machine learning, and deep learning. Fuzzy Logic is a method of reasoning that resembles human reasoning. The approach of Fuzzy Logic imitates the way of decision-making in humans that involves all intermediate possibilities between digital values YES and NO. This course introduces fundamental concepts in Neural Networks and Fuzzy Logic Theory.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Neural Networks and Fuzzy Logic** and attain **Skill Development** through **Participative Learning** techniques.					
Course Outcomes	**On successful completion of this course the students shall be able to:** 1. Define the concept of Neural Networks. [Knowledge] 2. Define the ideas behind most common learning algorithms in Neural Network.[Knowledge] 3. Discuss the concepts of Fuzzy Sets and Relations. [Comprehension] 4. Demonstrate the Fuzzy logic concepts and its applications.[Application]					
Course Content:						
Module 1	**Introduction to Neural Network**	Quiz	Single Layer Perceptron	**9Classes**		
Topics: Introduction to NN: History, Artificial and biological neural networks, Artificial intelligence and neural networks.						

Neurons and Neural Networks: Biological neurons, Models of single neurons, Different neural network models.
Single Layer Perceptron: Least mean square algorithm, Learning curves, Learning rates, Perceptron.

Module 2	**Multilayer Perceptron**	Quiz	Multilayer Perceptron	**10 Classes**

Topics:
Multilayer Perceptron: The XOR problem, Back-propagation algorithm, Heuristic for improving the back-propagation algorithm, Some examples.
Radial-Basis Function Networks: Interpolation, Regularization, Learning strategies.
Kohonen Self-Organising Maps: Self-organizing map, The SOM algorithm, Learning vector quantization.

Module 3	**Fuzzy Sets, Operations and Relations**	Quiz	Fuzzy Operations	**10Classes**

Topics:
Fuzzy Sets: Crisp Sets - an Overview, Fuzzy Sets - Definition and Examples, α - Cuts and its Properties, Representations of Fuzzy Sets, Extension Principles of Fuzzy Sets.
Fuzzy Operations: Operations on Fuzzy Sets - Fuzzy Complements, Fuzzy Intersections, Fuzzy Unions, Combinations of Operations, Aggregation Operations.
Fuzzy Relations: Binary Fuzzy relations, Fuzzy Equivalence Relations, Fuzzy Compatibility Relations.

Module 4	**Fuzzy Logic and Fuzzy Logic Controller**	Assignment	Developing Fuzzy Logic Controller	**10Classes**

Fuzzy Logic: Classical Logic, Multivalued Logic, Fuzzy Propositions, Fuzzy Quantifiers, Linguistic Hedges, Inference from Conditional Fuzzy Propositions, Conditional and Qualified Propositions and Quantified Propositions.
Fuzzy Controllers: An Overview, Fuzzification Module, Fuzzy Rule Base, Fuzzy Inference Engine, Defuzzification Module, An Example.

Targeted Application & Tools that can be used:

1. Python Libraries and Software (Eg.,Tensorflow, Scikit-Learn etc.)
2. Matlab (Neural Network Toolbox, Fuzzy Logic Toolbox)

Project work/Assignment:

Students will have to do group assignments for Modules 2 & 4. As a part of their assignments, they will have to implement the solution to particular problems.

Textbook(s):

1. Haykin, Simon. "*Neural networks and learning machines*", 3/E. Pearson Education India, 2011. https://www.pearson.com/en-us/subject-catalog/p/Haykin-Neural-Networks-and-Learning-Machines-3rd-Edition/P200000003278/9780133002553
2. George J. Klir and Bo Yuan, "*Fuzzy Sets and Fuzzy Logic- Theory and Applications*", Prentice Hall of India, 2015.
https://www.worldcat.org/title/fuzzy-sets-and-fuzzy-logic-theory-and-applications/oclc/505215200

References:

1. Shivanandam, Deepa S, "*Principles of Soft computing*", N Wiley India, 3rd Edition, 2018.https://www.wileyindia.com/principles-of-soft-computing-3ed.html
2. Timothy J. Ross, "*Fuzzy Logic with Engineering Applications*", Third Edition, Wiley, 2011. https://onlinelibrary.wiley.com/doi/book/10.1002/9781119994374
3. Kumar S., "*Neural Networks - A Classroom Approach*", Tata McGraw Hill, 2nd Edition 2017.https://www.worldcat.org/title/neural-networks-a-classroom-approach/oclc/56955342
4. Fakhreddine O. Karray, and Clarence W. De Silva. "*Soft computing and intelligent systems design: theory, tools, and applications*". Pearson Education, 2009.

Weblinks

https://www.pearson.com/en-gb/search.html?q=Karray%20Soft-Computing-and-Intelligent-Systems-Design-Theory-Tools-and-Applications

Topics relevant to "Skill Development ": Assignment implementations in software, batch wise presentations are used for Skill Development through Participative Learning techniques. This is attained through assessment component mentioned in course handout.

Course Code: **COM2504**	**Course Title: Applied Machine Learning** **Type of Course: 1] Program Core** **2] Laboratory integrated**	**L - T - P - C**	2	0	2	3
Version No.	1.0					
Course Pre-requisites	**CSE3001 Artificial Intelligence and Machine Learning**					
Anti-requisites	**NIL**					
Course Description	Machine Learning algorithms are the key to develop intelligent systems such as Apple's Siri, Google's self-driving cars etc. This course introduces the concepts of the core machine learning techniques such as Regression learning, Bayesian learning, Ensemble learning, Perceptron learning, Unsupervised learning, Competitive learning, learning from Gaussian mixture models and learning to detect outliers. Course lectures covers both the theoretical foundations as well as the essential algorithms for the various learning methods. Lab sessions complement the lectures and enable the students in developing intelligent systems for real life problems.					
Course Objectives	This course is designed to improve the learners 'EMPLOYABILITY SKILLS' by using EXPERIENTIAL LEARNING techniques. The supervised hands-on laboratory exercises, assessments and the group projects facilitate this learning process.					

Course Out Comes	On successful completion of the course the students shall be able to: 1] Apply advanced supervised machine learning methods for predictive modeling.[Application] 2] Produce machine learning models with better predictive performance using metalearning algorithms [Application] 3] Create predictive models using Perceptron learning algorithms[Application] 4] Employ advanced unsupervised learning algorithms for clustering, competitivelearning and outlier detection[Application] 5] Implement machine learning based intelligent models using Python libraries.[Application]			
Course Content:				
Module 1	Supervised Learning	Assignment	Programming using Keras/Sklearn	**No. of Classes L – 7 P – 12**
Topics: An overview of Machine Learning(ML); ML workflow; types of ML; Types of features, Feature Engineering -Data Imputation Methods; Regression – introduction; simple linear regression, loss functions; Polynomial Regression; Logistic Regression; Softmax Regression with cross entropy as cost function**; Bayesian Learning** – Bayes Theorem, estimating conditional probabilities for categorical and continuous features, Naïve Bayes for supervised learning; Bayesian Belief networks; Support Vector Machines – soft margin and kernel tricks.				
Module 2	Ensemble Learning	Assignment	Programming using Keras/Sklearn	**No. of Classes L-3 P-4**
Topics: **Ensemble Learning** – using subset of instances – Bagging, Pasting, using subset of features –random patches and random subspaces method; Voting Classifier, Random Forest; Boosting – AdaBoost, Gradient Boosting, Extremely Randomized Trees, Stacking.				
Module 3	Perceptron Learning	Assignment /Quiz	Programming using Keras/Sklearn	**No. of Classes L-7 P-2**

Topics: **Perceptron Learning** – from biological to artificial neurons, Perceptrons, Linear Threshold Units,logical computations with Perceptrons, common activation functions – sigmoid, tanh, relu and softmax, common loss functions, multi-layer Perceptrons and the Backpropagation algorithm using Gradient Descent.				
Module 4	Unsupervised Learning	Assignment	Programming using Keras/Sklearn	**No. of Classes L-6 P-6**
Topics: **Unsupervised Learning** – simple k Means clustering- simple and mini-batch**;** updating centroidsincrementally; finding the optimal number of clusters using Elbow method ; Silhoutte coefficient, drawbacksof kMeans, kMeans++ **;** Divisive hierarchical clustering – bisecting k-means, clustering using Minimum Spanning Tree (MST) **Competitive Learning** - Clustering using Kohenen's Self Organising Maps (SOM), **Density Based Spatial Clustering – DBSCAN**; clustering using Gaussian Mixture Models (GMM) with EM algorithm ; Outlier Detection methods – **Isolation Forest, Local Outlier Factor(LOF)**				

List of Laboratory Tasks:

Experiment N0 1: Methods for handling missing values
Level 1: Given a data set from UCI repository, implement the different ways of handling missing values in itusing Scikit-learn library of Python
Level 2: Implement one of these methods using a custom defined function in Python.

Experiment No. 2: Data Visualization
Level 1 Perform Exploratory Data Analysis for a given data set by creating Scatter Plot, Pair Plot, Count Plotusing Matplotlib and Seaborn
Level 2 Create Heat Maps, WordCloud

Experiment No. 3: Regression learning
Level 1 Given a data set from UCI repository, implement the simple linear regression algorithm and estimatethe models parameters and the performance metrics. Plot the learning curves.
Level 2 Implement the polynomial regression algorithm. Compare the learning curves of Polynomial andLinear Regression.

Experiment No.4: Logistic regression
Level 1 Write custom code for generating the logistic/sigmoid plot for a given input
Level 2 Given a data set from UCI repository, implement the Logistic regression algorithm. Estimate the classprobabilities for a given test data set. Plot and analyze the decision boundaries.

Experiment No.5: Bayesian Learning
Level 1 Given a data set from UCI repository, implement a classification model using the Bayesian algorithm

Experiment No.6: Support Vector Machine(SVM)
Level 1 Given data sets from UCI repository, implement a linear SVM and a non-linear SVM based classificationmodel.

Experiment No. 7: Ensemble Learning
Level 1 : Implement Ensemble Learning algorithms such as Bagging, Pasting and Out-of Bag Evaluation
Level 2 : Random Patches and Random Subspace Method

Experiment No. 8: Ensemble Learning
Level 1 : AdaBoost and Gradient Boosting, Stacking

Experiment No. 9: Perceptron Learning
Level 1 : Implement the Perceptron Classifier
Level 2 : – An Image Classifier Using the Sequential API of Keras

Experiment No. 10: Unsupervised Learning
Level 1 : K-means – simple and mini-batch. Finding the optimal number of clusters using Elbow method and Silhoutte Coefficient . Compare the inertia of both as k increases. Tuning the hyperparameter 'k' usingGridSearchCV.
Level 2 : – Using clustering for Image segmentation and Preprocessing. Kmeans++

Experiment No. 11: Density Based Clustering
Level 1 Implement DBSCAN – clustering using the local density estimation. Perform hard and soft clustering for new instances.
Experiment No. 12: Outlier Detection
Level 1 Outlier Detection using Isolation Forest and Local Outlier Factor

Targeted Application & Tools that can be used :

1. Execution of the ML algorithms will be done using the Google's cloud service namely "Colab", available at https://colab.research.google.com/ or Jupyter Notebook.
2. The data sets will be from the bench marking repositories such as UCI machine learning repository available at : https://archive.ics.uci.edu/ml/index.php
3. Laboratory tasks will be implemented using the libraries available in Python such as Scikit learn, matplotlib, seaborn, perceptron and the deep learning framework namely Keras.

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

Students can be assigned a mini project to develop a machine learning application for real-life problems in various domains such as health care, business intelligence, environmental modeling, etc.

Text Book

There are a number of useful textbooks for the course, but each cover only a part of the course syllabus. Following is an indicative list of textbooks.

1. Aurélien Géron, "Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow", Oreilly, Second Edition, 2019.
2. Andreas C Muller, Sarah Guido, "Introduction to Machine Learning with Python :A Guide for Data Scientists", Oreilly, First Edition, 2018
3. Giuseppe Bonaccorso, "Machine Learning Algorithms: A reference guide to popular algorithms from data science and machine learning", Packt Publishing, 2017.

References In references apart from the books and web links, mention a few standards &Hand books relevant to the Laboratory tasks used by the professionals.

1. Tan P. N., Steinbach M & Kumar V. "***Introduction to Data Mining"*, Pearson Education, 2016.
2. **https://towardsdatascience.com/machine-learning/home**
3. MIT Opencourseware: https://ocw.mit.edu/courses/6-0002-introduction-to-computational-thinking- and-data-science-fall-2016/resources/lecture-11-introduction-to-machine-learning/
4. https://onlinecourses.nptel.ac.in/noc21_cs85/preview

Course Code: COM2503	Course Title: Applied Machine Learning Lab Type of Course: Program Core -Laboratory	L-T-P-C	0	0	2	1
Version No.	1.0					
Course Pre-requisites	CSEXXXX – Computational Thinking with Python Lab CSE3157 – Artificial Intelligence and Machine Learning					
Anti-requisites	NIL					
Course Description	Machine Learning algorithms are the key to develop intelligent systems such as Apple's Siri, Google's self-driving cars etc. This course introduces the concepts of the core machine learning techniques such as Regression learning, Bayesian learning, Ensemble learning, Perceptron learning, Unsupervised learning, Competitive learning, learning from Gaussian mixture models and learning to detect outliers. Course lectures covers both the theoretical foundations as well as the essential algorithms for the various learning methods. Lab sessions complement the lectures and enable the students in					

	developing intelligent systems for real life problems.
Course Objectives	The objective of the course is EMPLOYBILITY of student by using EXPERIENTIAL LEARNING techniques.
Course Out Comes	On successful completion of this course the students shall be able to: 5. **Apply** advanced supervised machine learning methods for predictive modeling. [Apply] 6. **Produce** machine learning models with better predictive performance using meta learning algorithms [Apply] 7. **Create** predictive models using Perceptron learning algorithms [Apply] 8. **Employ** advanced unsupervised learning algorithms for clustering, competitive learning and outlier detection [Apply] 9. **Implement** machine learning based intelligent models using Python libraries. [Apply]

Course Content: No. of Sessions: 15 (30 hours)

Experiment No. 1: File Handling Using Python
Level 1: Read a CSV file using Python
Level 2: Read a text file using Python

Experiment No. 2: Methods for handling missing values
Level 1: Given a data set from UCI repository, implement the different ways of handling missing values in it using Scikit-learn library of Python
Level 2: Implement one of these methods using a custom defined function in Python.

Experiment No. 3: Data Visualization
Level 1: Perform Exploratory Data Analysis for a given data set by creating Scatter Plot, Pair Plot, Count Plot using Matplotlib and Seaborn
Level 2: Create Heat Maps, WordCloud

Experiment No. 4: Regression learning
Level 1: Given a data set from UCI repository, implement the simple linear regression algorithm and estimate the models parameters and the performance metrics. Plot the learning curves.
Level 2: Implement the polynomial regression algorithm. Compare the learning curves of Polynomial and Linear Regression.

Experiment No. 5: Logistic Regression
Level 1: Write custom code for generating the logistic/sigmoid plot for a given input
Level 2: Given a data set from UCI repository, implement the Logistic regression algorithm. Estimate the class probabilities for a given test data set. Plot and analyze the decision boundaries.

Experiment No. 6: Bayesian Learning
Level 1: Given a data set from UCI repository, implement a classification model using the Bayesian algorithm.
Level 2: Implement a Naïve Bayes classifier using 5-fold cross-validation

Experiment No. 7: Support Vector Machine (SVM)
Level 1: Given data sets from UCI repository, implement a linear SVM and a non-linear SVM based classification model.
Level 2: Construct kernels with 5-fold cross-validation for SVM.

Experiment No. 8 & 9: Ensemble Learning
Level 1: Implement Ensemble Learning algorithms such as Bagging, Pasting and Out-of Bag Evaluation
Level 2: Random Patches and Random Subspace Method, Adaboost and Gradient Boosting, Stacking.

Experiment No. 10: Perceptron Learning
Level 1: Implement the Perceptron Classifier
Level 2: An Image Classifier Using the Sequential API of Keras

Experiment No. 11 & 12: Unsupervised Learning
Level 1: K-means – simple and mini-batch. Finding the optimal number of clusters using Elbow method and Silhoutte Coefficient . Compare the inertia of both as k increases. Tuning the hyperparameter 'k' using GridSearchCV.
Level 2: Using clustering for Image segmentation and Preprocessing. Kmeans++

Experiment No. 13: Density Based Clustering
Level 1: Implement DBSCAN – clustering using the local density estimation. Perform hard and soft clustering for new instances.
Level 2: Outlier Detection using Isolation Forest and Local Outlier Factor

Experiment No. 14: Association Rule Mining
Level 1: Implement the Apriori Algorithm for Association Rule Mining
Level 2: Implement the Dynamic Itemset Counting Algorithm for Association Rule Mining.

Experiment No. 15: Collaborative Filtering
Level 1: Implement Collaborative Filtering using Item-Based Filtering
Level 2: Implement Collaborative Filtering using User-Based Filtering

Targeted Application & Tools that can be used:

5. Google Colab
6. Python IDEs like PyCharm

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

Students can be assigned a mini project to develop a machine learning application for real-life problems in various domains such as health care, business intelligence, environmental modeling, etc.

Textbook(s):

4. Aurélien Géron. *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow*, Oreilly, 3rd Edition, 2022.

References:

R1. Andreas C Muller, and Sarah Guido. *Introduction to Machine Learning with Python: A Guide for Data Scientists*, O'Reilly, 1st Edition, 2016.

Weblinks

W1. NPTEL Courses: https://nptel.ac.in/courses/106106139 (IIT M), https://nptel.ac.in/courses/106105152 (IIT Kgp)

Catalogue prepared by	Dr. Sandeep Albert Mathias
Recommended by the Board of Studies on	BOS NO: SOCSE 2nd BOS held on 17/03/25
Date of Approval by the Academic Council	Academic Council Meeting No 21, Dated 17/03/25

Course Code: CSE7000	**Course Title: Internship** **Type of Course:**	**L- T-P- C**	-	-	-	2
Version No.	**1.0**					
Course Pre-requisites	**Knowledge and Skills related to all the courses studied in previous semesters.**					
Anti-requisites	**NIL**					
Course Description	Students observe science and technology in action, develop an awareness of the method of scientific experimentation, and often get an opportunity to see, study and operate sophisticated and costly equipment. They also learn about the implementation of the principles of management they have learnt in class, when they observe multidisciplinary teams of experts from engineering, science, economics, operations research, and management deal with techno-economic problems at the micro and macro levels. Finally, it enables them to develop and refine their language, communication and inter-personal skills, both by its very nature, and by the various evaluation components, such as seminar, group discussion, project report preparation, etc. The broad-based core education, strong in mathematics and science and rich in analytical tools, provides the foundation necessary for the student to understand properly the nature of real-life problems.						
Course Objectives	The objective of the course is to familiarize the learners with the concepts of Professional Practice and attain Employability Skills through Experiential Learning techniques.					
Course Outcomes	On successful completion of this course the students shall be able to: 1. Identify the engineering problems related to local, regional, national or global needs. (Understand) 2. Apply appropriate techniques or modern tools for solving the intended problem. (Apply) 3. Design the experiments as per the standards and specifications. (Analyze) 4. Interpret the events and results for meaningful conclusions. (Evaluate)					
Catalogue prepared by	Mr. Md Ziaur Rahman					
Recommended by the Board of Studies on						

Date of Approval bythe Academic Council	

Course Code:CSE 7100	**Course Title: Mini Project** **Type of Course:**	**L- T-P- C**	0	0	0	4
Version No.	**1.0**					
Course Pre-requisites	**Knowledge and Skills related to all the courses studied in previous semesters.**					
Anti-requisites	**NIL**					
Course Description	Students observe science and technology in action, develop an awareness of the method of scientific experimentation, and often get an opportunity to see, study and operate sophisticated and costly equipment. They also learn about the implementation of the principles of management they have learnt in class, when they observe multidisciplinary teams of experts from engineering, science, economics, operations research, and management deal with techno-economic problems at the micro and macro levels. Finally, it enables them to develop and refine their language, communication and inter-personal skills, both by its very nature, and by the various evaluation components, such as seminar, group discussion, project report preparation, etc. The broad-based core education, strong in mathematics and science and rich in analytical tools, provides the foundation necessary for the student to understand properly the nature of real-life problems. The students have options to pursue this course as either Project Work and Dissertation at the university, or Project Work in an Industry/ Company/ Research Laboratory, or Internship Program in an Industry/Company.					
Course Objectives	The objective of the course is to familiarize the learners with the concepts of Professional Practice and attain Employability Skills through Experiential Learning techniques.					
Course Outcomes	On successful completion of this course the students shall be able to: 1. Identify the engineering problems related to local, regional, national or global needs. (Understand) 2. Apply appropriate techniques or modern tools for solving the intended problem. (Apply) 3. Design the experiments as per the standards and specifications. (Analyze) 4. Interpret the events and results for meaningful conclusions. (Evaluate) 5. Appraise project findings and communicate effectively through scholarly publications. (Create)					
Catalogue prepared by	Dr. Sampath A K					
Recommended by the Board of Studies						

on	
Date of Approval bythe Academic Council	

Course Code:CSE 7300	**Course Title:** **Capstone Project** **Type of Course:**	**L- T-P- C**	0	0	0	10
Version No.	**1.0**					
Course Pre-requisites	**Knowledge and Skills related to all the courses studied in previous semesters.**					
Anti-requisites	**NIL**					
Course Description	Students observe science and technology in action, develop an awareness of the method of scientific experimentation, and often get an opportunity to see, study and operate sophisticated and costly equipment. They also learn about the implementation of the principles of management they have learnt in class, when they observe multidisciplinary teams of experts from engineering, science, economics, operations research, and management deal with techno-economic problems at the micro and macro levels. Finally, it enables them to develop and refine their language, communication and inter-personal skills, both by its very nature, and by the various evaluation components, such as seminar, group discussion, project report preparation, etc. The broad-based core education, strong in mathematics and science and rich in analytical tools, provides the foundation necessary for the student to understand properly the nature of real-life problems. The students have options to pursue this course as either Project Work and Dissertation at the university, or Project Work in an Industry/ Company/ Research Laboratory, or Internship Program in an Industry/Company.					
Course Objectives	The objective of the course is to familiarize the learners with the concepts of Professional Practice and attain Employability Skills through Experiential Learning techniques.					
Course Outcomes	On successful completion of this course the students shall be able to: 1. Identify problems based on societal /research needs. (Understand) 2. Apply Knowledge and skill to solve societal problems in a group. (Apply) 3. Develop interpersonal skills to work as member of a group or leader. (Apply) 4. Analyze the inferences from available results through theoretical / Experimental / Simulations. (Analyze) 5. Analyze the impact of solutions in societal and environmental context for sustainable development. (Analyze) 6. Improve in written and oral communication. (Create) 7. Demonstrate capabilities of self-learning in a group, which leads to lifelong learning. (Understand)					

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

Course Code: CAI2505	Course Title: Natural Language Processing Type of Course: Program Core -Laboratory	L-T-P-C	0	0	2	1
Version No.	1.0					
Course Pre-requisites	Artificial Intelligence and Machine Learning					
Anti-requisites	NIL					
Course Description	This course introduces the basics of Natural Language Processing methods with specific emphasis on modern applications. The course will teach students different concepts of natural language processing, such as word representations, text representations, part-of-speech tagging, word sense disambiguation, parsing, etc. Topics: Word representations, Part-of-Speech tagging, chunking, parsing, text classification, sentiment analysis, named entity recognition, and machine translation.					
Course Objectives	The objective of the course is EMPLOYBILITY of student by using EXPERIENTIAL LEARNING techniques.					
Course Out Comes	On successful completion of this course the students shall be able to: 10. Define different problems related to natural language processing. [Understand] 11. Discuss using NLP techniques for different applications. [Apply] 12. Propose solutions for a particular NLP problem using different machine learning and deep learning techniques. [Apply] 13. Learn to use different NLP tools and packages. [Apply]					
Course Content:	No. of Sessions: 15 (30 hours)					
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:

7. Execution of the NLP task will be done using the Google's cloud service namely "Colab", available at https://colab.research.google.com/, Anaconda Navigator.
8. Laboratory tasks will be implemented using the libraries available in Python such as NLTK, Gensim, Spacy and Huggingface Transformers.

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

To enhance their understanding and gain practical exposure to NLP concepts, students are encouraged to complete a certification related to Natural Language Processing (NLP).

- Natural Language Processing - NPTEL
- Deep Learning for NLP - NPTEL
- Applied Natural Language Processing - NPTEL

Textbook(s):

5. Daniel Jurafsky, James H. Martin. "Speech and Language Processing: An Introduction to Natural Language Processing", Computational Linguistics and Speech, Pearson Publication, 2024 (3rd Edition Draft).
6. 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)

Catalogue prepared by	Dr. Sandeep Albert Mathias Ms. Devi.S
Recommended by the Board of Studies on	BOS NO: SOCSE 2nd BOS held on 17/03/25
Date of Approval by the Academic Council	Academic Council Meeting No 21, Dated 17/03/25

Course Code: CAI2505	Course Title: Natural Language Processing Type of Course: Program Core -Laboratory	L-T-P-C	0	0	2	1
Version No.	1.0					
Course Pre-requisites	Artificial Intelligence and Machine Learning					
Anti-requisites	NIL					

Course Description	This course introduces the basics of Natural Language Processing methods with specific emphasis on modern applications. The course will teach students different concepts of natural language processing, such as word representations, text representations, part-of-speech tagging, word sense disambiguation, parsing, etc. Topics: Word representations, Part-of-Speech tagging, chunking, parsing, text classification, sentiment analysis, named entity recognition, and machine translation.
Course Objectives	The objective of the course is EMPLOYBILITY of student by using EXPERIENTIAL LEARNING techniques.
Course Out Comes	On successful completion of this course the students shall be able to: 14. Define different problems related to natural language processing. [Understand] 15. Discuss using NLP techniques for different applications. [Apply] 16. Propose solutions for a particular NLP problem using different machine learning and deep learning techniques. [Apply] 17. Learn to use different NLP tools and packages. [Apply]

Course Content: No. of Sessions: 15 (30 hours)

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:

9. Execution of the NLP task will be done using the Google's cloud service namely "Colab", available at https://colab.research.google.com/, Anaconda Navigator.
10. Laboratory tasks will be implemented using the libraries available in Python such as NLTK, Gensim, Spacy and Huggingface Transformers.

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

To enhance their understanding and gain practical exposure to NLP concepts, students are encouraged to complete a certification related to Natural Language Processing (NLP).
Natural Language Processing - NPTEL
Deep Learning for NLP - NPTEL
Applied Natural Language Processing - NPTEL

Textbook(s):

7. Daniel Jurafsky, James H. Martin. "Speech and Language Processing: An Introduction to Natural Language Processing", Computational Linguistics and Speech, Pearson Publication, 2024 (3rd Edition Draft).
8. 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)	
Catalogue prepared by	Dr. Sandeep Albert Mathias Ms. Devi.S
Recommended by the Board of Studies on	BOS NO: SOCSE 2nd BOS held on 17/03/25
Date of Approval by the Academic Council	Academic Council Meeting No 21, Dated 17/03/25

Course Code: **CSE2506**	**Course Title:** Cloud computing **Type of Course:** Theory	**L- T-P- C**	3	**0**	0	**3**
Version No.	2.0					
Course Pre-requisites						
Anti-requisites	**NIL**					
Course Description	This Course is designed to introduce the concepts of Cloud Computing as a new computing paradigm. Cloud Computing has emerged in recent years as a new paradigm for hosting and delivering services over the Internet. The students can explore various Cloud Computing terminology, principles and applications. Understanding different views of the Cloud Computing such as theoretical, technical and commercial aspects. Topics include: Evolution of cloud computing and its services					

	available today, Introduction, Architecture of cloud computing, Infrastructure, platform, software, Types of cloud, Business models, cloud services, Collaborating using cloud services, Virtualization for cloud, Security, Standards and Applications.			
Course Objective	The objective of the course is to familiarize the learners with the concepts of Could computing and Virtualization and attain Employability through Participative Learning techniques.			
Course Outcomes	On successful completion of the course the students shall be able to: · Describe fundamentals of cloud computing, virtualization and cloud computing services. · Discuss high-throughput and data-intensive computing. · Explain security and standards in cloud computing. · Demonstrate the installation and configuration of virtual machine.			
Course Content:				
Module 1	Introduction to Cloud and Virtualization	Assignment	Virtualization	**10 Sessions**
Topics: Introduction to Cloud and Virtualization Cloud Computing at a Glance, Historical Developments, Building Cloud Computing Environments, Computing Platforms and Technologies, Virtualization, Characteristics of Virtualized Environments Taxonomy of Virtualization Techniques, Virtualization and Cloud Computing, Technology Examples, Cloud Computing Architecture, IaaS, PaaS, SaaS, Types of Clouds, Economics of Cloud				
Module 2	High Throughput and Data Intensive Computing	Assignment	Virtualization	**10 Sessions**
Topics:				

High Throughput and Data Intensive Computing: Task computing, MPI applications, Task based programming, Introduction to DIC, Technologies for DIC, Aneka Map Reduce Programming.

Module 3	Cloud Security and Standards	Assignment	Virtualization	**9 Sessions**

Topics:

Cloud Security and Standards: Cloud Security Challenges, Software-as-a-Service Security, Application standards, Client standards, Infrastructure and Service standards.

Module 4	Cloud Platforms	Assignment	Virtualization	**9 Sessions**

Cloud Platforms, Advances in cloud: introduction to Amazon Web Services: Introduction to Google App Engine, Introduction to Microsoft Azure. Media Clouds - Security Clouds - Computing Clouds - Mobile Clouds – Federated Clouds – Hybrid Cloud

Targeted Application & Tools that can be used:

Text Book(s):

1. John Rittinghouse and James Ransome, “Cloud Computing, Implementation, Management and

Security”, CRC Press.

2. Rajkumar Buyya, Christian Vecchiola, and Thamarai Selvi, “Mastering Cloud Computing”, McGraw

Hill Education.

Reference(s):

1. David E.Y. Sarna, “Implementing and Developing Cloud Applications”, CRC Press.

2. Anthony T Velte, Toby J Velte, Robert Elsenpeter, “Cloud Computing: A Practical Approach”, Tata McGraw-Hill. Web resources: https://presiuniv.knimbus.com/user#/home

Catalogue prepared by	
Recommended by the Board of Studies on	
Date of Approval by the Academic Council	

Course Code: **CSE2507**	**Course Title:** Cloud computing **Type of Course :** Lab	**L- T-P- C**	0	**0**	2	**1**
Version No.	1.0					
Course Pre-requisites						
Anti-requisites	**NIL**					

<table>
<tr><td>Course Description</td><td colspan="4">This course is designed to give hands-on experience with cloud platforms, services, and deployment models. Students will learn to set up, configure, and manage cloud environments using platforms like AWS, Microsoft Azure, and Google Cloud. The course covers virtualization, containerization, serverless computing, cloud storage, security, and scalability. Through practical assignments, students will develop skills in deploying cloud applications, managing cloud resources, automating cloud workflows, and implementing cost-effective cloud solutions.</td></tr>
<tr><td>Course Objective</td><td colspan="4">The objective of the course is to Understand Cloud Infrastructure, deploy and manage virtual machines, implement cloud storage, develop and deploy cloud applications, optimize cost and performance.</td></tr>
<tr><td>Course Outcomes</td><td colspan="4">On successful completion of the course the students shall be able to:
CO1· Deploy and Manage Cloud Resources.
CO2. Develop and Deploy Cloud-based Applications
CO3. Optimize Performance and Cost in the Cloud
CO4. Implement Security and Automation in Cloud Environments</td></tr>
<tr><td>Course Content:</td><td colspan="4"></td></tr>
<tr><td>Module 1</td><td>Introduction to Cloud and Virtualization</td><td>Assignment</td><td>Virtualization</td><td>10 Sessions</td></tr>
<tr><td colspan="5">Lab Assignment 1: Setting Up Virtual Machines on Cloud
• Create a Virtual Machine (VM) on AWS/Azure/GCP
Configure OS, storage, and network settings
• Connect to the VM using SSH/RDP
Install web server (Apache/Nginx) and deploy a static webpage
Lab Assignment 2: Containerization Using Docker
• Install Docker on a local or cloud VM
• Create and run a Docker container
• Build a custom Docker image with a simple Python/Node.js application
• Push the image to Docker Hub and deploy it on a new VM</td></tr>
</table>

Module 2	High Throughput and Data Intensive Computing	Assignment	Virtualization	**10 Sessions**

Lab Assignment 1: Setting Up a Distributed Computing Environment

- Launch a **Hadoop or Spark cluster** on AWS EMR / Azure HDInsight / Google Dataproc
- Configure **HDFS (Hadoop Distributed File System)** for big data storage
- Run a **basic MapReduce job** on sample data

Lab Assignment 2: Data Preprocessing with Cloud Storage

- Store large datasets in **Amazon S3 / Azure Blob Storage / Google Cloud Storage**
- Use **Apache Spark or Hadoop** to read, clean, and process data
- Convert datasets into **Parquet or Avro formats** for efficient storage

Lab Assignment 3: Batch Processing with Apache Spark

- Load large datasets (e.g., logs, tweets, transaction data) into **Spark DataFrame**
- Perform **ETL (Extract, Transform, Load) operations** on the data
- Use **SparkSQL** for querying large datasets

Lab Assignment 4: Real-Time Data Processing with Spark Streaming

- Set up **Kafka / AWS Kinesis / Google Pub/Sub** for real-time data ingestion
- Process streaming data using **Spark Streaming**
- Perform windowed aggregations and visualize real-time trends

Lab Assignment 5: Cloud-Based Machine Learning with Big Data

- Use **Google BigQuery ML / AWS SageMaker / Azure Machine Learning** for model training
- Train a **linear regression or classification model** on a large dataset
- Deploy the trained model as an API for real-time predictions

Lab Assignment 6: Running Parallel Machine Learning Workloads

- Implement **distributed ML training** using **Spark MLlib or TensorFlow on Cloud TPUs**
- Train models on a large dataset and optimize performance using distributed execution

Lab Assignment 7: Auto-Scaling and Load Balancing for Data Processing

- Deploy a **serverless Spark job** using **AWS Glue / Azure Synapse**
- Implement **auto-scaling for high-throughput jobs**
- Measure performance improvements using cloud monitoring tools

Lab Assignment 8: Cost Optimization for High-Throughput Data Processing

- Analyze **cloud cost reports** for data-intensive workloads
- Optimize cloud storage and compute resources for cost-efficiency
- Compare **on-demand vs. reserved vs. spot instances** for cost savings

Module 3	Cloud Security and Standards	Assignment	Virtualization	**9 Sessions**

Lab Assignment 9: Configuring Identity and Access Management (IAM)

- Set up **IAM roles and policies** in AWS / Azure / Google Cloud
- Create and assign users with **least privilege access**
- Implement **Multi-Factor Authentication (MFA)** for added security
- Audit IAM policies using **AWS IAM Access Analyzer / Azure Security Center**

Lab Assignment 10: Setting Up Single Sign-On (SSO) and Role-Based Access Control (RBAC)

- Configure **AWS Cognito / Azure Active Directory / Google IAM** for authentication
- Implement **Role-Based Access Control (RBAC)** for users and groups
- Integrate **OAuth 2.0 / OpenID Connect (OIDC) / SAML** for secure authentication

Lab Assignment 11: Encrypting Data at Rest and in Transit

- Encrypt **cloud storage (S3, Blob, Cloud Storage)** using **KMS (Key Management Service)**
- Set up **TLS/SSL certificates** for secure web traffic encryption
- Enable **database encryption (AWS RDS, Azure SQL, GCP Cloud SQL)**

Lab Assignment 12: Implementing Compliance & Governance in Cloud

- Enable **GDPR, HIPAA, ISO 27001 compliance tools** in cloud platforms
- Use **AWS Config / Azure Policy / GCP Security Command Center** to enforce compliance
- Conduct **security audits and generate compliance reports**

Lab Assignment 13: Implementing Cloud Monitoring & Threat Detection

- Configure **AWS CloudTrail / Azure Monitor / GCP Operations Suite** for activity logging
- Set up **intrusion detection systems (IDS) & anomaly detection**
- Analyze security logs using **Amazon GuardDuty / Azure Sentinel / Chronicle Security**

Lab Assignment 14: Automating Security Incident Response

- Deploy a **Serverless Lambda / Azure Logic App** to automatically respond to security incidents
- Implement **automated alerts** for suspicious activity
- Test a **denial-of-service (DDoS) simulation** and implement mitigation strategies

Module 4	Cloud Platforms	Assignment	Virtualization	**9 Sessions**

Lab Assignment 15: Getting Started with Cloud Platforms

- Create a **free-tier account** on AWS, Azure, or Google Cloud
- Navigate the **Cloud Console, CLI, and SDKs**
- Explore and configure **dashboard, billing, and IAM settings**

Lab Assignment 16: Launching a Virtual Machine (VM) on Cloud

- Deploy a **VM instance** using **AWS EC2, Azure Virtual Machines, or Google Compute Engine**
  Configure **OS, storage, networking, and security groups**
  Connect to the instance using **SSH (Linux) or RDP (Windows)**

Lab Assignment 17: Cloud Storage and File Management

- Create **Object Storage (AWS S3 / Azure Blob Storage / Google Cloud Storage)**
  Upload, download, and set access permissions for files
  Implement **Lifecycle Policies and Versioning**

Lab Assignment 18: Cloud Database Management

- Deploy a **Relational Database (AWS RDS / Azure SQL Database / Cloud SQL)**

Connect and query the database using **MySQL/PostgreSQL clients**
Set up database **backups and automatic scaling**

Lab Assignment 19: Configuring Virtual Networks in Cloud

- Set up a **Virtual Private Cloud (VPC) / Azure Virtual Network / GCP VPC**
  Configure **subnets, firewalls, and security groups**
  Test **network communication between two VMs**

Lab Assignment 20: Deploying a Web Application on Cloud

Deploy a Python/Node.js/Java web app using:

- **AWS Elastic Beanstalk**
- **Azure App Service**
- **Google App Engine**
  Connect the app to Cloud Database (RDS, CosmosDB, Firestore)
  Monitor application performance and logs

Targeted Application & Tools that can be used:

Text Book(s):

1. John Rittinghouse and James Ransome, "Cloud Computing, Implementation, Management and Security", CRC Press.

2. Rajkumar Buyya, Christian Vecchiola, and Thamarai Selvi, "Mastering Cloud Computing", McGraw Hill Education.

Reference(s):

1. David E.Y. Sarna, "Implementing and Developing Cloud Applications", CRC Press.

2. Anthony T Velte, Toby J Velte, Robert Elsenpeter, "Cloud Computing: A Practical Approach", Tata McGraw-Hill. Web resources: https://presiuniv.knimbus.com/user#/home

Catalogue prepared by	
Recommended by the	

Board of Studies on	
Date of Approval by the Academic Council	

Course Code: CSE2510	**Course Title:** Competitive Programming and Problem Solving **Type of Course: Program Core**	L-T-P-C	0	0	4	2
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	**NIL**					
Course Description	The **Competitive Programming and Problem Solving** course equips students with efficient problem-solving skills for coding competitions and real-world challenges. Starting with brute-force solutions, students learn to optimize time and space complexity using advanced techniques like dynamic programming, greedy algorithms, and backtracking. Hands-on practice on platforms like CodeChef and Codeforces helps tackle problems involving number theory, data structures, and algorithmic paradigms. By understanding CP constraints and fostering a strategic mindset, students gain the confidence to excel in competitions, technical interviews, and practical applications.					
Course Out Comes	On successful completion of the course the students shall be able to: CO1 : Understanding the issues of online platforms and Competitive Programming (CP) and developing brute force coding for commonly asked CP problems. CO2 : Analyzing the space and time complexity of brute force solutions and designing efficient solutions. CO3 : Evaluating the applicability of suitable algorithmic approaches to solve relevant CP problems. CO4: Creating efficient solutions of CP problems using the learnt algorithmic approaches.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Competitive Programming and Problem Solving** and attain **Skill Development** through **Experiential Learning** techniques.					

Module 1: Introduction to Competitive Programming

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

Module 2: Number Theory for Problem-Solving

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

Module 3: Optimizing Time & Space Using Sequential Storage

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

Module 4: Non-Linear Data Structures

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

Module 5: Problem Solving using Advanced Topics

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

List of Laboratory Tasks:

1. You are given the finishing times of 'N' runners in a marathon. Write a program to find the runner who finished in the third position. **Focus:** Basic data structures (arrays), sorting algorithms (e.g., insertion sort, selection sort), and basic input/output.
2. In the same marathon, you are given the finishing times of 'N' runners and their bib numbers. Write a program to efficiently find the top 10 runners and their corresponding bib numbers. **Focus:** Efficient sorting algorithms (e.g., merge sort, quick sort), data structures like priority queues, and optimizing for large datasets.
3. A library maintains a list of books with their unique IDs. Write a program to check if a given book ID is present in the library. Focus: Searching algorithms (linear search), basic data structures (arrays or lists).
4. The library wants to implement a system to quickly find books by their titles. Suggest an efficient data structure (e.g., a hash table or a trie) and explain how to implement it to achieve fast book lookups. Focus: Understanding the trade-offs between different data structures, choosing the most appropriate data structure for a specific problem, and implementing efficient search operations.
5. An online store sells products with different prices. Write a program to calculate the total cost of a given list of products. **Focus:** Basic arithmetic operations, working with arrays or lists to store product prices.
6. The online store offers discounts based on the total purchase amount. Design an algorithm to efficiently calculate the final cost of an order, considering different discount rules (e.g., percentage discounts, fixed amount discounts, tiered discounts). Focus: Algorithmic design, conditional statements, handling complex scenarios with multiple rules, and potentially using dynamic programming techniques for optimization.
7. You are given two integers, 'a' and 'm'. Calculate 'a' raised to the power 'm' modulo a large prime number 'p'. **Focus:** Basic modular arithmetic operations (modular exponentiation), understanding the modulo operator.
8. In a secure communication system, you need to efficiently compute the modular exponentiation for very large values of 'm'. Implement and analyze the efficiency of the binary exponentiation algorithm for this task. **Focus:** Efficient algorithms for modular exponentiation (binary exponentiation), time complexity analysis, and understanding the importance of efficient algorithms in cryptography.

9. You have a deck of 'N' cards. Calculate the total number of possible hands of size 'K' that can be drawn from the deck. **Focus:** Basic combinatorics (combinations), factorial calculations.
10. In a card game, you need to calculate the probability of drawing certain combinations of cards (e.g., a pair, a three-of-a-kind) from a shuffled deck. Design an efficient algorithm to calculate these probabilities. **Focus:** Advanced combinatorics (permutations and combinations with repetitions), probability calculations, and optimizing calculations to avoid overflows.
11. You are given a network of devices represented as a graph. Determine if there is a path between two given devices in the network. **Focus:** Graph traversal algorithms (depth-first search or breadth-first search).
12. In a secure network, you need to detect and isolate compromised devices. Design an algorithm that efficiently identifies devices that exhibit anomalous behavior (e.g., unusual traffic patterns) using XOR-based techniques for data comparison and pattern matching. **Focus:** Applying XOR operations for data comparison and pattern recognition, understanding the properties of XOR (e.g., commutative, associative), and designing algorithms for network anomaly detection.
13. You are given an array representing the speeds of cars on a highway. Find the minimum time required for all cars to pass a certain point. **Focus:** Basic array traversal, finding the minimum element in an array.
14. In a more realistic scenario, cars have different lengths. Implement a two-pointer approach to simulate the movement of cars and determine the minimum time for all cars to pass a given point. **Focus:** Two-pointer technique, simulating real-world scenarios with arrays, optimizing time complexity.
15. Given a string, find the number of occurrences of a specific substring within the string. **Focus:** Basic string manipulation, string matching (brute-force approach).
16. Implement the KMP (Knuth-Morris-Pratt) string matching algorithm to efficiently find all occurrences of a given pattern within a large text document. **Focus:** Advanced string matching algorithms, understanding the concept of the "next" array in KMP, optimizing for large input sizes.
17. An online auction platform receives bids for different items. Implement a data structure (e.g., a priority queue) to efficiently track the highest bid for each item. **Focus:** Priority queues, insertion and extraction operations on priority queues, basic implementation of a priority queue using an array or a suitable library.
18. The auction platform needs to handle a large number of bids concurrently. Design and implement a system that efficiently processes bids, updates the highest bid for each item, and handles potential race conditions. **Focus:** Concurrent data structures and algorithms, thread safety, handling race conditions, optimizing for high-throughput scenarios.
19. A social network can be represented as a graph where users are nodes, and connections between users are edges. Write an algorithm to find if two given users are connected in the network. **Focus:** Graph traversal algorithms (depth-first search or breadth-first search), basic graph representation (adjacency list or adjacency matrix).
20. In a large social network, efficiently finding the shortest path between two users is crucial. Implement Dijkstra's algorithm to find the shortest paths between users in the network, considering edge weights (e.g., representing the strength of connections). **Focus:** Shortest path algorithms (Dijkstra's algorithm), graph algorithms with weighted edges, optimizing for large graphs.
21. A file system can be modeled as a tree structure. Implement a function to traverse the file system and print the names of all files and directories. **Focus:** Tree traversal algorithms (depth-first search or breadth-first search), basic tree representation (using nodes and pointers).
22. Design and implement a file system that supports efficient operations like creating directories, deleting files, and finding files based on their names or paths. Consider using a combination of tree structures and hash tables for efficient indexing and searching. **Focus:** Designing and implementing file system structures, using multiple data structures together, optimizing for common file system operations.

23. An online shopping cart can be represented as a tree, where each node represents an item or a category of items. Write an algorithm to calculate the total price of all items in the shopping cart. **Focus:** Tree traversal, calculating sums within a tree structure.
24. Implement a system that allows customers to apply discounts and coupons to their shopping carts. Consider using a combination of trees and other data structures (e.g., hash tables) to efficiently apply discounts and calculate the final price. **Focus:** Applying discounts and promotions to tree-like structures, efficient implementation of discount rules, optimizing for complex pricing scenarios.
25. In a social network, users can form groups. Given a list of friendships, determine if all users in a specific group are connected (directly or indirectly) through friendships. **Focus:** Disjoint set union (DSU) data structure, basic connectivity checks.
26. Design an efficient algorithm to find the minimum number of new friendships needed to connect all users in the social network into a single, connected component. **Focus:** Applying DSU for finding connected components, greedy algorithms, optimization for minimizing connections.
27. A treasure hunt involves a series of clues leading to the final treasure. Given a list of possible paths and their associated costs, find the cheapest path to reach the treasure. **Focus:** Greedy algorithms (e.g., Dijkstra's algorithm for shortest paths), basic graph representation.
28. In a more complex treasure hunt, there are time constraints associated with each path. Design an algorithm to find the fastest path to the treasure while considering both path costs and time constraints. **Focus:** Combining greedy approaches with other techniques (e.g., priority queues), handling multiple constraints, optimizing for time-critical scenarios.
29. In a simplified chess game with only rooks, determine the minimum number of moves required for a rook to reach a specific target square on an empty board. **Focus:** Breadth-first search (BFS) on a graph (the chessboard), basic graph traversal.
30. In a more realistic chess game with multiple pieces and obstacles, implement a minimax algorithm with alpha-beta pruning to determine the best move for a player. **Focus:** Game tree search, minimax algorithm, optimization techniques like alpha-beta pruning, handling complex game states.

Targeted Application & Tools that can be used:

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

Text Books:

1 **Guide to Competitive Programming: Learning and Improving Algorithms Through Contests" (3rd Edition),** ***Antti Laaksonen, springer, 2024***

2 **"Data Structures and Algorithms in Java: A Project-Based Approach" –** ***Dan S. Myers, Cambridge University Press***

Reference Books:

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

Web Resources

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

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

Assessment Type

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

Catalogue prepared by	Dr. Robin Rohit Vincent
Recommended by the Board of Studies on	
Date of Approval by the Academic Council	

Course Code: CSE1508	**Course Title:** Data Structures **Type of Course: Theory**	**L-T- P- C**	3	0	0	3
Version No.	1.0					
Course Pre-requisites						
Anti-requisites	NIL					
Course Description	This course introduces the fundamental concepts of data structures and to emphasize the importance of choosing an appropriate data structure and technique for program development .This course has theory and lab component which emphasizes on understanding the implementation and applications of data structures using Java programming language .With a good knowledge in the fundamental concepts of data structures and practical experience in implementing them, the student can be an effective designer, developer for new software applications.					
Course Objective	The objective of the course is SKILL DEVELOPMENT of student by using EXPERIENTIAL LEARNING techniques					
Course Out Comes	**On successful completion of the course the students shall be able to:** **CO1 :**Describe the concept of basic data structure, stacks, queues, and arrays and their operations. **[Understand]** **CO2:** Utilize linked lists for real-time scenarios. **[Apply]** **CO3:** Apply an appropriate non-linear data structure for a given scenario. **[Apply]** **CO4:** Demonstrate different searching and sorting techniques. **[Apply]**					
Course Content:						
Module 1	**Introduction to Data Structure and Linear Data Structure – Stacks and Queues**	Assignment	Program activity	**9 Hours**		
Introduction –Introduction to Data Structures, Types and concept of Arrays **.** **Stack -**Concepts and representation, Stack operations, stack implementation using array and Applications of Stack**.** **Queues -**Representation of queue, Queue Operations, Queue implementation using array, Types of Queue and Applications of Queue**.**						
Module 2	**Linear Data Structure -Linked List**	Assignment	Program activity	**12 Hours**		
Topics: Linked List - Singly Linked List, Operation on linear list using singly linked storage structures, Circular List, Applications of Linked list. **Recursion** - Recursive Definition and Processes.						

Module 3	**Non-linear Data Structures - Trees**	Assignment	Program activity	**12 Hours**
Topics: Trees - Introduction to Trees, Binary tree :Terminology and Properties, Use of Doubly Linked List, Binary tree traversals :Pre-Order traversal, In-Order traversal, Post - Order traversal , .Heaps , Expression Tree ,Red Black Tree - AVL Trees ,Binary Serach Tree				

Module 4	**Non-linear Data Structures - Graphs and Hashing**	Assignment	Program activity	**6 Hours**
Topics: Graphs: Basic Concept of Graph Theory and its Properties, Representation of Graphs . ADT, Elementary graph operations, Minimum Cost spanning trees, Shortest path and Transitive closure. **Hashing**: Introduction, Static Hashing, Dynamic Hashing				

Module 5	**Searching & Sorting**	Assignment	Program activity	**6 Hours**
Topic: Sorting & Searching - Sequential and Binary Search, Sorting – Selection and Insertion sort, Quick sort, Merge Sort, Bubble sort.				

List of Laboratory Tasks:

Lab sheet -1

Level 1: Prompt the user, read input and print messages.Programs using class, methods and objects

Level 2: Programming Exercises on fundamental Data structure - Arrays based on Scenario.

Lab sheet -2

Level 1: Programming Exercises on Stack and its operations

Level 2: Programming Exercises on Stack and its operations with condition

Lab sheet -3

Level 1: Programming on Stack application infix to postfix Conversion

Level 2: -

Lab sheet -4

Level 1: Programming on Stack application – Evaluation of postfix

Lab sheet -5

Level 1: Programming Exercises on Queues and its operations with conditions

Level 2: -

Lab sheet -6

Level 1: Programming Exercises on Linked list and its operations.

Level 2: Programming Exercises on Linked list and its operations with various positions

Lab sheet -7

Level 1: Programming Exercises on Circular Linked list and its operations.

Level 2: Programming Exercises on Circular Linked list and its operations with various positions

Lab sheet -8

Level 1: Programming Exercises on factorial of a number

Level 2: Programming the tower of Hanoi using recursion

Lab sheet -9

Level 1: -

Level 2: Programming the tower of Hanoi using recursion

Lab sheet -10

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

Level 2: -

Lab sheet -11

Level 1: Program to Construct Binary Search Tree and Graph

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

Lab sheet -12

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

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

Lab sheet -13

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

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

Lab sheet -14 (Beyond syllabus activity)

Level 1: Program to Construct AVL Tree

Level 2:

Lab sheet -15 (Beyond syllabus activity)

Level 1: Program to Construct RED BLACK Tree

Targeted Application & Tools that can be used

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

Project work/Assignment:

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

Text Book

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

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

References

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

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

Web resources:

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

Topics relevant to development of "Skill Development":

Linked list and stacks **Topics relevant to development of "Environment and sustainability: Queues**	
Catalogue prepared by	Muthuraj
Recommended by the Board of Studies on	09th BOS held on 04/05/19
Date of Approval by the Academic Council	Academic Council Meeting No. 11, Dated 11/06/19

Course Code: CSE1509	**Course Title:** Data Structures Lab **Type of Course:Lab**	**L-T- P- C**	0	0	4	2
Version No.	1.0					
Course Pre-requisites						
Anti-requisites	NIL					
Course Description	This course introduces the fundamental concepts of data structures and to emphasize the importance of choosing an appropriate data structure and technique for program development .This course has theory and lab component which emphasizes on understanding the implementation and applications of data structures using Java programming language .With a good knowledge in the fundamental concepts of data structures and practical experience in implementing them, the student can be an effective designer, developer for new software applications.					
Course Objective	The objective of the course is SKILL DEVELOPMENT of student by using EXPERIENTIAL LEARNING techniques					
Course Out Comes	**On successful completion of the course the students shall be able to:** **CO1 :**Describe the concept of basic data structure, stacks, queues, and arrays and their operations. **[Understand]** **CO2:** Utilize linked lists for real-time scenarios. **[Apply]** **CO3:** Apply an appropriate non-linear data structure for a given scenario. **[Apply]** **CO4:** Demonstrate different searching and sorting techniques. **[Apply]**					
Course Content:						
Module 1	**Introduction to Data Structure and Linear Data Structure – Stacks and Queues**	Assignment	Program activity	**9 Hours**		

Introduction –Introduction to Data Structures, Types and concept of Arrays **.**

Stack -Concepts and representation, Stack operations, stack implementation using array and Applications of Stack**.**

Queues -Representation of queue, Queue Operations, Queue implementation using array, Types of Queue and Applications of Queue**.**

Module 2	**Linear Data Structure -Linked List**	Assignment	Program activity	**12 Hours**

Topics: Linked List - Singly Linked List, Operation on linear list using singly linked storage structures, Circular List, Applications of Linked list.

Recursion - Recursive Definition and Processes.

Module 3	**Non-linear Data Structures - Trees**	Assignment	Program activity	**12 Hours**

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

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

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

Hashing: Introduction, Static Hashing, Dynamic Hashing

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

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

List of Laboratory Tasks:

Lab sheet -1

Level 1: Prompt the user, read input and print messages.Programs using class, methods and objects

Level 2: Programming Exercises on fundamental Data structure - Arrays based on Scenario.

Lab sheet -2

Level 1: Programming Exercises on Stack and its operations

Level 2: Programming Exercises on Stack and its operations with condition

Lab sheet -3

Level 1: Programming on Stack application infix to postfix Conversion

Level 2: -

Lab sheet -4

Level 1: Programming on Stack application – Evaluation of postfix

Lab sheet -5

Level 1: Programming Exercises on Queues and its operations with conditions

Level 2: -

Lab sheet -6

Level 1: Programming Exercises on Linked list and its operations.

Level 2: Programming Exercises on Linked list and its operations with various positions

Lab sheet -7

Level 1: Programming Exercises on Circular Linked list and its operations.

Level 2: Programming Exercises on Circular Linked list and its operations with various positions

Lab sheet -8

Level 1: Programming Exercises on factorial of a number

Level 2: Programming the tower of Hanoi using recursion

Lab sheet -9

Level 1: -

Level 2: Programming the tower of Hanoi using recursion

Lab sheet -10

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

Level 2: -

Lab sheet -11

Level 1: Program to Construct Binary Search Tree and Graph

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

Lab sheet -12

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

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

Lab sheet -13

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

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

Lab sheet -14 (Beyond syllabus activity)

Level 1: Program to Construct AVL Tree

Level 2:

Lab sheet -15 (Beyond syllabus activity)

Level 1: Program to Construct RED BLACK Tree

Targeted Application & Tools that can be used

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

Project work/Assignment:

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

Text Book

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

T2 Seymour Lipschutz, Data Structures Schaum's Outlines, Revised 1st Ed, McGraw Hill, 2014.
References **R1** Data structures and program design in C by Robert Kruse, Tondo C L, Bruce Leung, Pearson education publishers, 2017. **R2** Programming and Data Structure by Jackulin C Salini etal., Ane books publishers, 2019. **Web resources:** 3. For theory :https://onlinecourses.nptel.ac.in/noc20_cs85/preview 4. https://puniversity.informaticsglobal.com/login
Topics relevant to development of "Skill Development": Linked list and stacks **Topics relevant to development of "Environment and sustainability: Queues**

Catalogue prepared by	Muthuraj
Recommended by the Board of Studies on	09th BOS held on 04/05/19
Date of Approval by the Academic Council	Academic Council Meeting No. 11, Dated 11/06/19

Course Code: CSE1510	**Course Title: Database Management Systems** **Type of Course: Theory**	**L-T-P-C**	3	0	0	3
Version No.						
Course Pre-requisites	Foundational understanding of data types, data structures, basic programming knowledge, familiarity with operating systems and file management. Basic knowledge of set theory, logic, and discrete mathematics to understand relational algebra and query formulation.					
Anti-requisites	NIL					

Course Description	This course introduces the foundational principles of database management systems, including data models, schemas, and architectures. This course provides a solid foundation on the relational model of data and the use of relational algebra. It develops skills in SQL for data definition, manipulation, and control, enabling students to construct and execute complex queries. The course also introduces the concept of object oriented and object relational databases and modern database technologies like **NoSQL.** The also course allows the students to gain insights into data storage structures and indexing strategies for optimizing query performance.
Course Objective	The objective of the course is to familiarize the learners with the concepts of Database Management Systems and attain **Employability** through **Problem Solving** Methodologies.
Course Out Comes	On successful completion of the course the students shall be able to: 1. Describe the fundamental elements of relational database management systems. [Understand] 2. Examine databases using SQL query processing and Optimization. [Apply] 3. Design simple database systems applying the normalization constraints and demonstrate the database transaction processing, recovery, and security. [Apply] 4. Interpret the concept of advanced databases and its applications. [Apply]

Course Content:

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

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

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

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

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

Topics:
Relational database design: Problems in schema design, redundancy and anomalies, Normal Forms based on Primary Keys-(1NF,2NF, 3NF), Boyce-Codd Normal Form, Multi valued Dependency (Fourth Normal Form), 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 (Apply)	Assignment	Case Study	12 Sessions

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

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

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

Text Books:

T1. Elmasri R and Navathe S B, "Fundamentals of Database System", Pearson Publication, 7th Edition, 2018.

T2. RamaKrishna & Gehrke, "Database Management Systems" 3rd Edition, 2018, McGraw-Hill Education.

T3. W. Lemahieu, S. vanden Broucke and B. Baesens, "Principles of Database Management: Practical Guide to Storing, Managing and Analyzing Big and Small Data", Cambridge University Press, 2018.

References

R1 Avi Silberschatz, Henry F. Korth, S. Sudarshan, "Database System Concepts", McGraw-Hill ,7th Edition, 2019.

R2 M. Kleppmann, "Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems", O'Reilly, 2017.

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

Catalogue preparedby	
Recommended by theBoard o Studies on	
Date of Approval by the Academic Council	

Course Code: CSE1511	**Course Title: Database Management Systems Laboratory** **Type of Course: Lab**	**L-T-P-C**	0	0	2	1
Version No.	1.0					
Course Pre-requisites	Foundational understanding of data types, basic programming knowledge, operating systems and file management.					
Anti-requisites	NIL					
Course Description	The Database Management Systems (DBMS) Laboratory is designed to provide students with hands-on experience in database design, implementation, and management using SQL and database management tools such as MySQL. The lab complements theoretical concepts learned in database courses by allowing students to practice database creation, querying, and optimization techniques. The DBMS Lab enables students to develop industry-relevant skills in database management, preparing them for careers in **software development, data engineering, and database administration**.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of Database Management Systems and attain **Employability** through **Problem Solving** Methodologies.					
Course Out Comes	On successful completion of the course the students shall be able to: 5. Demonstrate the database concepts, practice, and SQL queries. [Apply] 6. Design and implement database schemas while applying normalization techniques to optimize structure. [Apply]] 7. Develop and implement stored procedures, triggers, and views for automation and efficiency. [Apply] 8. To Design and build database applications for real world problems. [Apply]					

Course Content:

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

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

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

databases].

Experiment No. 4: [2 Session]

4. To study and implement different types of Set and Join Operations [2 Slots]

Level 1: Demonstrate different types of Set Operations (UNION, UNION ALL, INTERSECT, MINUS) and Join Operations (INNER JOINs, OUTER JOINS, CROSS JOIN, NATURAL JOIN) on two or more tables of Airline Database. Level 2: Use Set and Join operations to retrieve the data from two or more relations(tables) as per the given
scenario. [Airline Database]

Labsheet-3 [2 Practical Sessions]

Experiment No. 5: [2 sessions]

5. To study and implement Views, and Procedures in MySQL DB.

Level 1: Implement MySQL Views, and Procedures in ORACLE DB on Employee database.

Level 2: Analyze the requirement and construct views, and Procedures on Mini Project Domain. [Banking Database]

Labsheet-4 [2 Practical Sessions]

Experiment No. 6: [2 Sessions]

6. To study and implement Functions, and Triggers in MySQL DB.

Level 1: Implement Oracle Functions and Triggers in Oracle on Employee database.

Level 2: Analyze the requirement and construct Functions and Triggers. [Supply chain Database]

Labsheet-5 [2 Practical Sessions]

Experiment No. 7: [2 Sessions]

7. To study and implement Functions, and Triggers in MySQL DB.

Level 1: Implement Oracle Functions and Triggers in Oracle on Employee database.

Level 2: Determine the requirement and construct Functions and Triggers. [Supply chain Database]

Labsheet-6 [4 Practical Sessions]

Experiment No. 8: [2 Sessions]

8. To implement the concept of forms and reports.

Level 1: Implement the concept of forms and reports.

Level 2: Examine the schema relationship.

Experiment No. 9: [2 Sessions]

9. Create the database using the given schema. (Flight Management)

Level 1: Implement a relational database based on the provided schema for the Flight Management system, including the creation of tables, relationships, and constraints.

Level 2: Demonstrate schema relationships by defining primary and foreign keys to ensure data integrity within the Flight Management database.

Labsheet-7 [4 Practical Sessions]

Experiment No. 10: [2 Sessions]

10. Create the database using the given schema. (Company database)

Level 1: Implement the database schema by defining tables, relationships, and constraints according to the given Company Database schema.

Level 2: Demonstrate the schema's relationships and data integrity by creating and linking tables as per the specified requirements.

Experiment No. 11: [2 Sessions]

11. Create the database using the given schema. (Student Library)

Level 1: Implement forms and reports based on the provided Student Library database schema, ensuring effective data entry and reporting mechanisms.

Level 2: Demonstrate the schema relationships within the Student Library database, demonstrating how these relationships influence the creation and functionality of forms and reports.

Labsheet-8 [1 Sessions]

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

Targeted Application & Tools that can be used:

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

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

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

Text Books:

T1. Elmasri R and Navathe S B, "Fundamentals of Database System", Pearson Publication, 7th Edition, 2018.

T2. RamaKrishna & Gehrke, "Database Management Systems" 3rd Edition, 2018, McGraw-Hill Education.

T3. W. Lemahieu, S. vanden Broucke and B. Baesens, "Principles of Database Management: Practical Guide to Storing, Managing and Analyzing Big and Small Data", Cambridge University Press, 2018.

References

R1 Avi Silberschatz, Henry F. Korth, S. Sudarshan, "Database System Concepts", McGraw-Hill ,7th Edition, 2019.

R2 M. Kleppmann, "Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems", O'Reilly, 2017.

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

Catalogue preparedby	

Recommended by theBoard o Studies on	
Date of Approval by the Academic Council	

Course Code: CSE1512	**Course Title:** Analysis of Algorithms **Type of Course:** Theory		**L- T-P- C**	3	1	0	4
Version No.	1.0					
Course Pre-requisites						
Anti-requisites	Nil					
Course Description	This course introduces techniques for the design and analysis of efficient algorithms and methods of applications. This course discusses the classic approaches for algorithm design such as Divide and Conquer, Dynamic Programming, Greedy method. This course also describes other basic strategies searching solution space. The core concepts of analyzing algorithms and classifying them into various complexity classes is covered in the end.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Analysis of Algorithms** and attain **Skill Development** through **Problem Solving** Methodologies.					
Course Out Comes	On successful completion of the course the students shall be able to: 1. Compute efficiency of a given algorithm.[Apply] 2. Apply divide and conquer technique for searching and sorting Problems.[Apply] 3. Apply the Dynamic Programming technique for a given problem. [Apply] 4. Apply greedy technique for solving a Problem.[Apply] 5. Demonstrate Back tracking technique and limitations of Algorithms.[Apply]					
Course Content:						
Module 1	**Introduction**	Assignment	Simulation/Data Analysis	**10 Sessions**		
Introduction, Asymptotic Notations and its properties, Best case, worst case and average case- Sequential search, Sorting; Mathematical analysis for Recursive and Non-recursive algorithms: Substitution method and Master's Theorem.						
Module 2	**Divide-and-conquer**	Assignment	Simulation/Data Analysis	**08 Sessions**		
Introduction. Insertion Sort; Merge sort, Quick sort, Binary search.						
Module 3	**Dynamic programming**	Term paper/Assignment	Simulation/Data Analysis	**10 Sessions**		
Introduction with examples, Principles of Memoization, 0-1 Knapsack Problem, Bellman-Ford algorithm, Floyd-Warshall's Algorithms. Chain Matrix Multiplication.						
Module 4	**Greedy technique**	Term paper/Assignment	Simulation/Data Analysis	**09 Sessions**		
Introduction, Fractional Knapsack Problem, Minimal Spanning Tree: Prim's Algorithm and Kruskal's Algorithm, Single-source Shortest Path: Dijkstra's Algorithm						
Module 5	**Complexity Classes**	Term paper/Assignment	Simulation/Data Analysis	**08 Sessions**		
Complexity Classes- P,NP- NP Hard and NP Complete - Boolean Satisfiability Problem (SAT). Branch and Bound: Knapsack problem; Backtracking, - N-Queens problem.						
Text Book						

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

References

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

Web-Resources

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

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

Catalogue prepared by	Dr Murali Parameswaran
Recommended by the Board of Studies on	BOS NO: XXth BOS, held on N/NN/202N
Date of Approval by the Academic Council	Academic Council Meeting No. XX^{th,} Dated N/NN/202N

C o u r s e C	**C o u r s e T**					

Code: CSE XXXX	Title: Analysis of Algorithms Laboratory					

	r y T y p e o f C o u r s e : I n t e g r a t e d					
V e r s i o n N o	1					

.	
Course Pre-requisites	**CSE2001 - Data Structures and Algorithms.**
Anti-requisite	NIL

s	
Course Description	This course introduces techniques for the design and analysis of efficient algorithms and methods of applications. This course discusses the classic approaches for algorithm design such as Divide and Conquer, Dynamic Programming, Greedy method. This course also describes other basic strategies searching solution space. The core concepts of analyzing algorithms and classifying them into various complexity classes is covered in the end.
Course Objective	The objective of the course is to familiarize the learners with the concepts of **Analysis of Algorithms** and attain **Skill Development** through **Experiential Learning** Methodologies.

Course Outcomes	**On successful completion of the course the students shall be able to:** 1. Compute efficiency of a given algorithm. [Applying] 2. Apply divide and conquer technique for searching and sorting Problems.[Applying] 3. Apply the Dynamic Programming technique for a given problem. [Applying] 4. Apply greedy technique for solving a Problem.[Applying] 5. Demonstrate Back tracking technique and limitations of Algorithms.[Applying]	
Course Content		
Module 1	**Introduction**	**3 Sessio**

		n s
Measuring running time of an algorithm, Compare running time of algorithms, Implement sorting algorithms such as bubble sort, selection sort		
M o d u l e 2	**Divide-and-conquer**	**3 S e s s i o n s**
Compare searching algorithms: Linear Search, Binary Search; Compare Sorting algorithms: Insertion Sort, Merge Sort, QuickSort.		
M o d u l e 3	**Dynamic programming**	**3 S e s s i o n s**
Introduction and memorization: Factorial; Coin Change Problem ; Floyd-Warshall's Algorithm.		
M o d u l e 4	**Greedy technique**	**3 S e s s i o**

		n s
Fractional Knapsack Problem; Minimal Spanning Tree Algorithms-Prim's Algorithm, Kruskal's algorithm		
M o d u l e 5	**Complexity Classes**	**3 S e s s i o n s**
Branch and Bound: Knapsack problem; Backtracking, - N-Queens problem.		
	List of Laboratory Tasks: **1. Measuring running time of an algorithm** **Objective: To experimentally determine the running time of basic algorithms for input size n=10, 100, 1000, etc. by taking difference of starting time and ending time.** **2. Compare running time of algorithms** **Objective: To execute two algorithms to solve the same problem, and to comparatively evaluate the better algorithm for large values of N.** **3. Implement sorting algorithms such as bubble sort, selection sort** **Objective: To implement comparison based sorting strategies.** **4. Compare searching algorithms** **Objective: To implement two searching strategies and compare their performance.** **5. Compare Sorting algorithms**	

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

	15. Case Study **Objective: To demonstrate how various techniques can be used to solve the same problem with the help of Knapsack problem.**
	Targeted Application & Tools that can be used 1. 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
Catalogue prepared by	Dr. Murali Parameswaran
Recomm	BOS NO: XXth BOS, held on N/NN/202N

ended by the Board of Studies on	
Date of Appr	Academic Council Meeting No. XX^{th,} Dated N/NN/202N

o v a l b y t h e A c a d e m i c C o u n c i l	

Course Code: CSE1700	Course Title: Essentials of AI Type of Course: Theory	L- T-P- C	3	0	0	3

Version No.	2.0			
Course Pre-requisi Data **tes**	Basic knowledge of programming, mathematics, understanding of data handling			
Anti-requisites	**NIL**			
Course Description	This course is a comprehensive introductory course designed to equip learners with the fundamental Python programming skills necessary to work with artificial intelligence (AI) technologies. This course is aimed at individuals who are new to AI but have a basic understanding of programming concepts. It combines Python programming fundamentals with hands-on experience in implementing AI techniques such as machine learning, neural networks, and natural language processing.			
Course Objective	The objective of the course is to Understand Python Programming Fundamentals, Manipulate and Process Data with Python, Implement Machine Learning Algorithms and Build and Train Neural Networks for AI Applications.			
Course Outcomes	On successful completion of the course the students shall be able to: CO 1: Apply Python Programming to AI Projects CO 2: Build and Train Machine Learning Models CO 3: Develop Deep Learning Models with Neural Networks CO 4: Deploy AI Solutions and Understand Ethical Implications			
Course Content:				
Module 1	Introduction to Python Programming for AI	Assign ment	Implementation	**10 Sessions**
Topics: Python Basics: Variables, Data Types, Operators, and Control Flow Functions, Loops, and Conditionals statements, Data Structures: Lists, Tuples, Dictionaries, Sets ,Introduction to Libraries: NumPy and Pandas for data manipulation, Basic Input/Output and File Handling Introduction to Python for AI: Libraries and Frameworks Overview				
Module 2	Data Processing, Visualization	Assign ment	Implementation	10 Sessions
Topics:				

cleaning and preprocessing with Pandas,Handling missing data, outliers, and duplicates, Data transformation (Normalization, Encoding), Introduction to Matplotlib and Seaborn for Data Visualization, Exploratory Data Analysis (EDA), Visualizing datasets to understand patterns and relationships.

Module 3	Introduction to Machine Learning	Mini - Project	Implementation	10 Sessions

Topics:

What is Machine Learning? Types of ML algorithms Supervised Learning: Regression, Classification, Unsupervised Learning: Clustering, Key ML Algorithms: Linear Regression, Decision Trees, K-Means ,Introduction to Scikit-learn library

Model evaluation (Accuracy, Precision, Recall, Confusion Matrix)

Module 4	Neural Networks and Deep Learning	Quiz	Implementation	10 Sessions

Topics:

Introduction to Neural Networks and Deep Learning, Perceptron Model and Backpropagation

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

Targeted Application & Tools that can be used:

Applications:

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

Tools:

- **Pandas**: For data manipulation and cleaning (e.g., handling missing values, merging

datasets).

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

NLTK: The Natural Language Toolkit for various text processing tasks like tokenization, stemming, and part-of-speech tagging.

spaCy: A fast NLP library for advanced NLP tasks such as named entity recognition and dependency parsing.

Transformers (by Hugging Face): A powerful library for using pre-trained Transformer-based models like BERT, GPT, and others for advanced NLP tasks.

Text Book(s):

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

Reference(s):

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

Catalogue prepared by	
Recommended by the Board of Studies on	
Date of Approval by the Academic Council	

Course Code: CSE1701	**Course Title:** Essentials of AI Lab **Type of Course:** Lab		L			
Version No.	2.0					
Cou	Basic Java Programming Knowledge, Mathematics: Linear Algebra and Probability, Basic Data Structures and Algorithms, Familiarity with Libraries and Tools, Understanding of Basic Machine Learning Concepts.					

Prerequisites	
Anti-requisites	**NIL**
Course	This course introduces students to the essential concepts and techniques of Artificial Intelligence (AI) with a focus on practical implementation using Python. Students will explore core AI topics such as search algorithms, knowledge representation, machine learning, and neural networks, while gaining proficiency in using popular Python libraries like NumPy, pandas, scikit-learn, and TensorFlow. Through a series of lab exercises and projects,

Description	students will apply AI principles to solve real-world problems, develop intelligent applications, and understand how AI systems function at a foundational level.
Course Objective	The primary objectives of the course are to Gain Proficiency in AI Concepts and Python Implementation, Develop and Implement Machine Learning Models, Understand and Build Neural Networks, Apply AI to Real-World Problems
Course Outco	On successful completion of the course the students shall be able to: 1. Proficiency in Implementing AI Algorithms Using Python 2. Ability to Build and Evaluate Machine Learning Models 3. Hands-on Experience with Neural Networks and Deep Learning 4. Practical Application of AI to Solve Real-World Problems

mes				
Course Content:				
Module 1	InIntroduction to AI and Python for AI	Assignment	Implementation	**10 Sessions** I

Lab Assignment 1: Setting Up the Python Environment

- **Objective:** Get familiar with setting up a Python environment for AI projects.
- **Tasks:**

1. Install Python, Anaconda, and Jupyter Notebook.
2. Set up a virtual environment for AI development.
3. Install essential Python libraries: numpy, pandas, matplotlib, and scikit-learn.
4. Write and execute simple Python code to verify installation (e.g., print a "Hello AI" message).

Lab Assignment 2: Basic Python Programming for AI

- **Objective:** Understand and practice the basic Python syntax and data structures used in AI.
- **Tasks:**

1. Write Python code to work with basic data types (integer, float, string, boolean).
2. Implement and manipulate Python lists, tuples, sets, and dictionaries.
3. Create basic control flow structures: if-else, for loops, while loops.
4. Use functions and lambda functions to solve small AI-related problems, such as calculating factorial or Fibonacci numbers.

Lab Assignment 3: Data Exploration and Preprocessing

- **Objective:** Learn how to work with data for AI models.
- **Tasks:**

1. Load a dataset (e.g., Titanic or Iris dataset) using pandas.
2. Clean the dataset by handling missing values, removing duplicates, and converting data types if needed.
3. Explore the dataset by visualizing it using matplotlib and seaborn.
4. Perform basic data preprocessing tasks such as feature scaling, encoding categorical variables, and splitting data into training and testing sets.

Module	Data Processing, Visualization	A	Implem	10 Sess

e 2			e n t a t i o n	i o n s

Lab Assignment 1: Data Preprocessing with Pandas

Objective:

Learn the fundamentals of data preprocessing, including cleaning, handling missing values, and performing basic transformations using **Pandas**.

Tasks:

1. **Load and Inspect the Dataset:**
   - Load a dataset (e.g., **Iris**, **Titanic**, **Wine Quality** dataset) using pandas.read_csv() or pandas.read_excel().
   - Inspect the first few rows of the dataset using .head() and check basic information using .info().
2. **Handle Missing Values:**
   - Identify missing values in the dataset using .isnull() or .isna().
   - Handle missing data by imputing with mean, median, or mode using SimpleImputer from sklearn, or remove rows with missing data using .dropna().
3. **Data Transformation:**
   - Convert categorical variables to numerical values using one-hot encoding or label encoding.
   - Normalize/standardize numerical columns using StandardScaler or MinMaxScaler from sklearn.
4. **Subset and Filter Data:**
   - Create subsets based on certain conditions (e.g., select rows where a specific feature value is greater than a threshold).

- Filter outliers from numerical data using interquartile range (IQR).

Lab Assignment 2: Data Aggregation and Grouping with Pandas

Objective:

Master aggregation and grouping techniques using **Pandas** for summarizing data.

Tasks:

1. **Group Data by Category:**
   - Group data by one or more categorical features (e.g., "class" in the Iris dataset or "embarked" in Titanic dataset).
   - Use .groupby() to calculate aggregate statistics such as mean, median, sum, and count.
2. **Pivot Tables:**
   - Create a pivot table to summarize data (e.g., aggregate the average age of passengers in the Titanic dataset by class and gender).
   - Use .pivot_table() to perform multi-dimensional aggregation.
3. **Data Aggregation and Custom Functions:**
   - Apply custom aggregation functions to the grouped data (e.g., calculate custom metrics or perform complex transformations within each group).
4. **Sorting and Ranking Data:**
   - Sort the dataset by multiple columns (e.g., sorting by "age" or "fare").
   - Rank data based on specific metrics (e.g., assign ranks to passengers by fare in the Titanic dataset).

Lab Assignment 3: Data Visualization with Matplotlib and Seaborn

Objective:

Learn to visualize datasets using **Matplotlib** and **Seaborn** for better understanding and insights.

Tasks:

1. **Basic Plotting with Matplotlib:**

- Create simple plots like line plots, bar plots, and histograms using **Matplotlib**.
- Customize the plots by setting titles, labels, and legends.
- Create scatter plots to visualize relationships between two variables.

2. **Advanced Plotting with Seaborn:**
   - Use **Seaborn** to create advanced visualizations like pair plots, heatmaps, box plots, and violin plots.
   - Customize visualizations with color palettes, styling, and themes.
   - Create a correlation heatmap to visualize correlations between features in the dataset.
3. **Distribution Visualizations:**
   - Plot distributions of continuous variables using **Seaborn's** distplot() or kdeplot().
   - Create bar plots for categorical variables to understand their frequency distribution.
4. **Multi-Plot Grid Layouts:**
   - Use **Matplotlib's** subplots() function to create multiple plots in a grid layout for comparison (e.g., scatter plot and histogram in the same figure).

Lab Assignment 4: Visualizing Relationships and Feature Importance

Objective:

Understand how to visualize relationships between features and evaluate feature importance for predictive models.

Tasks:

1. **Scatter Plot Matrix:**
   - Use **Seaborn's** pairplot() to create a scatter plot matrix to visualize the relationships between multiple features.
   - Analyze the pairwise relationships between features and identify any patterns or correlations.
2. **Heatmap of Correlation Matrix:**
   - Use **Pandas** to calculate the correlation matrix of numeric features.

- Visualize the correlation matrix using **Seaborn's** heatmap() to understand feature correlations and multicollinearity.

3. **Feature Importance from Models:**
   - Train a decision tree or random forest model using **scikit-learn** on a dataset (e.g., **Iris** or **Titanic**).
   - Visualize feature importance using a bar chart to understand which features have the most impact on the model.
4. **Visualizing Predictions vs. Actual Values:**
   - For regression tasks, visualize the predicted values against the actual values using a scatter plot.
   - For classification tasks, visualize the classification results with a confusion matrix.

Lab Assignment 5: Time Series Data Visualization and Processing

Objective:

Learn how to process and visualize time series data, which is common in AI applications like forecasting and trend analysis.

Tasks:

1. **Load and Preprocess Time Series Data:**
   - Load a time series dataset (e.g., stock market data, weather data).
   - Parse dates properly and set the date column as the index using pd.to_datetime() and .set_index().
2. **Plot Time Series Data:**
   - Plot a time series line chart using **Matplotlib** to visualize trends over time.
   - Create rolling averages (e.g., 7-day, 30-day) to smooth out short-term fluctuations in the time series data.
3. **Seasonal Decomposition of Time Series:**
   - Use **statsmodels** to decompose a time series into seasonal, trend, and residual components.

- Visualize the decomposed components to understand seasonal variations.

4. **Forecasting with Simple Models:**

   - Use simple forecasting models (e.g., moving average, ARIMA) to predict future values.
   - Visualize the forecasted data along with actual historical data.

Module 3	Introduction to Machine Learning	A	Implementation	10 Sessions

Lab Assignment 3: Implementing Linear Regression

- **Tasks:**

1. Load a real-world dataset (e.g., **Boston Housing Price** dataset).
2. Train a **Linear Regression** model using LinearRegression() from scikit-learn.
3. Evaluate the model using **Mean Squared Error (MSE)** and **R-squared Score**.
4. Visualize the regression line using Matplotlib.

Lab Assignment 4: Logistic Regression for Classification

- **Tasks:**

1. Load the **Iris** or **Breast Cancer** dataset.

2. Preprocess the dataset (handle missing values, encode categorical variables, scale data).
3. Train a **Logistic Regression** model using LogisticRegression().
4. Evaluate performance using **Accuracy, Precision, Recall, F1-score**.
5. Plot the **Confusion Matrix** and **ROC Curve**.

Lab Assignment 5: Implementing K-Nearest Neighbors (KNN)

- **Tasks:**

1. Load the **Iris dataset** and split it into training and testing sets.
2. Train a **KNN classifier** using KNeighborsClassifier().
3. Experiment with different values of **K** and evaluate performance.
4. Visualize decision boundaries using a **scatter plot**.

Lab Assignment 6: Decision Trees and Random Forests

- **Tasks:**

1. Train a **Decision Tree classifier** on the Titanic dataset.
2. Visualize the tree structure using plot_tree().
3. Train a **Random Forest classifier** and compare performance with the decision tree.
4. Determine the **feature importance** using feature_importances_.

		Quiz	Implement	10 Sessions

			a t i o n	

Lab Assignment 7: Introduction to Perceptron and Activation Functions

Tasks:

1. Implement a **single-layer perceptron** using NumPy.
2. Train the perceptron to classify **AND, OR, XOR** gates.
3. Experiment with different **activation functions** (Sigmoid, ReLU, Tanh).
4. Visualize decision boundaries.

Lab Assignment 8: Building a Simple Neural Network with Keras

Tasks:

1. Load the **MNIST dataset** from keras.datasets.
2. Preprocess the data (normalize pixel values, reshape input).
3. Create a **fully connected neural network** using Sequential API.
4. Train and evaluate the model using **categorical cross-entropy loss** and **accuracy**.

Lab Assignment 9: Implementing CNN from Scratch

Tasks:

1. Load the **CIFAR-10 dataset**.
2. Build a CNN with **Conv2D, MaxPooling2D, Flatten, Dense, Dropout** layers.
3. Use **Adam optimizer** and **categorical cross-entropy loss**.
4. Train and visualize loss/accuracy curves.

Lab Assignment 10: Image Augmentation & Regularization

Tasks:

1. Apply **data augmentation** (rotation, zoom, flipping) using ImageDataGenerator.
2. Add **dropout and batch normalization** to prevent overfitting.
3. Compare model performance with and without augmentation.

Lab Assignment 11: Transfer Learning with Pre-trained Models

Tasks:

1. Use **VGG16 or ResNet50** pre-trained on ImageNet.
2. Replace the output layer to classify **new images**.
3. Freeze earlier layers and fine-tune deeper layers.
4. Evaluate the model on a custom **dataset (e.g., Cats vs. Dogs)**.

Lab Assignment 12: Implementing RNN for Text Classification

Tasks:

1. Load **IMDB movie reviews dataset** from keras.datasets.
2. Preprocess text (tokenization, padding sequences).
3. Build an **RNN** with **Embedding, SimpleRNN, Dense** layers.
4. Train and evaluate the model.

Lab Assignment 13: Building an LSTM for Time Series Prediction

Tasks:

1. Load a **time series dataset** (e.g., stock prices, temperature data).
2. Preprocess the data (normalize, reshape).

3. Build an **LSTM-based model**.
4. Predict future values and visualize trends.

Targeted Application & Tools that can be used:

Applications:

5. **Data Preprocessing**: Clean and manipulate data from various sources such as CSV, Excel, SQL databases, and APIs.

- **Exploratory Data Analysis (EDA)**: Gain insights into datasets by identifying trends, patterns, and outliers.
- **Predictive Modeling**: Build models for classification (e.g., spam detection) and regression (e.g., house price prediction).

6. **Clustering**: Group data into clusters for unsupervised learning tasks (e.g., customer segmentation).

- **Model Evaluation**: Assess model performance using appropriate metrics such as accuracy, precision, recall, and F1-score.

Tools:

- **Pandas**: For data manipulation and cleaning (e.g., handling missing values, merging datasets).
- **NumPy**: For numerical operations and working with arrays and matrices.
- **Matplotlib**: For creating static, animated, and interactive visualizations.
- **Seaborn**: For advanced data visualizations (e.g., heatmaps, pair plots).
- **Plotly**: For creating interactive visualizations, especially useful for large datasets.
- **Scikit-learn**: The go-to library for implementing machine learning algorithms (e.g., linear regression, decision trees, k-means clustering).
- **XGBoost**: For advanced gradient boosting models, particularly for large-scale machine learning tasks.
- **TensorFlow** (for deep learning in Module 4): A powerful open-source library for building machine learning and deep learning models.

- **Keras**: High-level neural network API, built on top of TensorFlow, to easily create deep learning models.

NLTK: The Natural Language Toolkit for various text processing tasks like tokenization, stemming, and part-of-speech tagging.

spaCy: A fast NLP library for advanced NLP tasks such as named entity recognition and dependency parsing.

Transformers (by Hugging Face): A powerful library for using pre-trained Transformer-based models like BERT, GPT, and others for advanced NLP tasks.

Text Book(s):

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

Reference(s):

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

Course Code: CSE1702	**Course Title:Machine Learning Techniques** **Type of Course: Program Core -Theory**
Version No.	1.0
Course Pre-requisites	

Anti-requisites	NIL	
Course Description	Machine Learning algorithms are the key to develop intelligent as Regression learning, Bayesian learning, Ensemble learning, Course lectures covers both the theoretical foundations as we intelligent systems for real life problems.	
Course Objectives	The objective of the course is EMPLOYBILITY of student by using	
Course Out Comes	On successful completion of this course the students shall be ab 18. **Apply** advanced supervised machine learning methods for predictive 19. **Produce** machine learning models with better predictive performanc 20. **Create** predictive models using Perceptron learning algorithms [Appl 21. **Employ** advanced unsupervised learning algorithms for clustering, co 22. **Implement** machine learning based intelligent models using Python	
Course Content:		
Module 1	**Supervised Learning**	**Assignment**
An overview of Machine Learning(ML); ML workflow; types of ML; Types of features, Feature Engineering – Softmax Regression with cross entropy as cost function; Bayesian Learning – Bayes Theorem, estimating c Vector Machines – soft margin and kernel tricks. Evaluation Methodologies – Testing Dataset, Train-Valid		
Module 2	**Ensemble Learning**	**Assignment**
Ensemble Learning – using subset of instances – Bagging, Pasting, using subset of features –random patche Stacking.		
Module 3	**Perceptron Learning**	**Assignment**
Perceptron Learning – from biological to artificial neurons, Perceptrons, Linear Threshold Units, logical com Perceptrons and the Backpropagation algorithm using Gradient Descent.		

Module 4	Unsupervised Learning	Assignment
Unsupervised Learning – simple k Means clustering- simple and mini-batch; updating centroids incrementa hierarchical clustering – bisecting k-means, clustering using Minimum Spanning Tree (MST). **Competitive L** Mixture Models (GMM) with EM algorithm ; Outlier Detection methods – Isolation Forest, Local Outlier Fa		
Targeted Application & Tools that can be used: 11. Google Colab 12. Python IDEs like PyCharm		
Project work/Assignment: Mention the Type of Project /Assignment proposed for this course 3. Students can be assigned a mini project to develop a machine learning application for real-life problems in various		
Textbook(s): 7. Aurélien Géron. *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow*, Oreilly, 3rd Edition, 2022.		
References: R1. Andreas C Muller, and Sarah Guido. *Introduction to Machine Learning with Python: A Guide for Data Sci* **Weblinks** **W1**. NPTEL Courses: https://nptel.ac.in/courses/106106139 (IIT M), https://nptel.ac.in/courses/106105152		
Catalogue prepared by		
Recommended by the Board of Studies on		
Date of Approval by the Academic Council		

Course Code: CSE1703	Course Title: Machine Learning Techniques Type of Course: Program Core -Laboratory	L-T-P-C	0	0	2	1

Version No.	1.0
Course Pre-requisites	
Anti-requisites	NIL
Course Description	Machine Learning algorithms are the key to develop intelligent systems such as Apple's Siri, Google's self-driving cars etc. This course introduces the concepts of the core machine learning techniques such as Regression learning, Bayesian learning, Ensemble learning, Perceptron learning, Unsupervised learning, Competitive learning, learning from Gaussian mixture models and learning to detect outliers. Course lectures covers both the theoretical foundations as well as the essential algorithms for the various learning methods. Lab sessions complement the lectures and enable the students in developing intelligent systems for real life problems.
Course Objectives	The objective of the course is EMPLOYBILITY of student by using EXPERIENTIAL LEARNING techniques.
Course Out Comes	On successful completion of this course the students shall be able to: 23. **Apply** advanced supervised machine learning methods for predictive modeling. [Apply] 24. **Produce** machine learning models with better predictive performance using meta learning algorithms [Apply] 25. **Create** predictive models using Perceptron learning algorithms [Apply] 26. **Employ** advanced unsupervised learning algorithms for clustering, competitive learning and outlier detection [Apply] 27. **Implement** machine learning based intelligent models using Python libraries. [Apply]

Course Content: **No. of Sessions: 15 (30 hours)**

Experiment No. 1: File Handling Using Python
Level 1: Read a CSV file using Python
Level 2: Read a text file using Python

Experiment No. 2: Methods for handling missing values
Level 1: Given a data set from UCI repository, implement the different ways of handling missing values in it using Scikit-learn library of Python
Level 2: Implement one of these methods using a custom defined function in Python.

Experiment No. 3: Data Visualization
Level 1: Perform Exploratory Data Analysis for a given data set by creating Scatter Plot, Pair Plot, Count Plot using Matplotlib and Seaborn
Level 2: Create Heat Maps, WordCloud

Experiment No. 4: Regression learning
Level 1: Given a data set from UCI repository, implement the simple linear regression algorithm and estimate the models parameters and the performance metrics. Plot the learning curves.
Level 2: Implement the polynomial regression algorithm. Compare the learning curves of Polynomial and Linear Regression.

Experiment No. 5: Logistic Regression

Level 1: Write custom code for generating the logistic/sigmoid plot for a given input
Level 2: Given a data set from UCI repository, implement the Logistic regression algorithm. Estimate the class probabilities for a given test data set. Plot and analyze the decision boundaries.

Experiment No. 6: Bayesian Learning
Level 1: Given a data set from UCI repository, implement a classification model using the Bayesian algorithm.
Level 2: Implement a Naïve Bayes classifier using 5-fold cross-validation

Experiment No. 7: Support Vector Machine (SVM)
Level 1: Given data sets from UCI repository, implement a linear SVM and a non-linear SVM based classification model.
Level 2: Construct kernels with 5-fold cross-validation for SVM.

Experiment No. 8 & 9: Ensemble Learning
Level 1: Implement Ensemble Learning algorithms such as Bagging, Pasting and Out-of Bag Evaluation
Level 2: Random Patches and Random Subspace Method, Adaboost and Gradient Boosting, Stacking.

Experiment No. 10: Perceptron Learning
Level 1: Implement the Perceptron Classifier
Level 2: An Image Classifier Using the Sequential API of Keras

Experiment No. 11 & 12: Unsupervised Learning
Level 1: K-means – simple and mini-batch. Finding the optimal number of clusters using Elbow method and Silhoutte Coefficient . Compare the inertia of both as k increases. Tuning the hyperparameter 'k' using GridSearchCV.
Level 2: Using clustering for Image segmentation and Preprocessing. Kmeans++

Experiment No. 13: Density Based Clustering
Level 1: Implement DBSCAN – clustering using the local density estimation. Perform hard and soft clustering for new instances.
Level 2: Outlier Detection using Isolation Forest and Local Outlier Factor

Experiment No. 14: Association Rule Mining
Level 1: Implement the Apriori Algorithm for Association Rule Mining
Level 2: Implement the Dynamic Itemset Counting Algorithm for Association Rule Mining.

Experiment No. 15: Collaborative Filtering
Level 1: Implement Collaborative Filtering using Item-Based Filtering
Level 2: Implement Collaborative Filtering using User-Based Filtering

Targeted Application & Tools that can be used:

13. Google Colab
14. Python IDEs like PyCharm

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

Students can be assigned a mini project to develop a machine learning application for real-life problems in various domains such as health care, business intelligence, environmental modeling, etc.

Textbook(s):

9. Aurélien Géron. *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow*, Oreilly, 3rd Edition, 2022.

References: R1. Andreas C Muller, and Sarah Guido. *Introduction to Machine Learning with Python: A Guide for Data Scientists*, O'Reilly, 1st Edition, 2016. **Weblinks** **W1**. NPTEL Courses: https://nptel.ac.in/courses/106106139 (IIT M), https://nptel.ac.in/courses/106105152 (IIT Kgp)	
Catalogue prepared by	
Recommended by the Board of Studies on	
Date of Approval by the Academic Council	

	Course Title: Software Design and Development **Type of Course: School Core [Theory Only]**	**L-** **T-** **P-** **C**	
	1.0		
	NIL		
	NIL		
	The objective of this course is to provide the fundamentals concepts of Software Engineering process and principles. The course covers software requirement engineering processes, system analysis, design, implementation and testing as software system development. The course covers software quality, configuration management and maintenance.		
	The objective of the course is to familiarize the learners with the concepts of Software Engineering and attain Skill Dev through Participative Learning techniques.		
	On successful completion of this course the students shall be able to: 1] Describe the Software Engineering principles, ethics and process models(Knowledge) 2] Identify the requirements, analysis and appropriate design models for a given application(Comprehension)		

	3] Understand the Agile Principles(Knowledge) 4] Apply an appropriate planning, scheduling, evaluation and maintenance principles involved in software(Application)			
	Introduction to Software Engineering and Process Models **(Knowledge level)**	Quiz		

n: Need for Software Engineering, Professional Software Development, Software Engineering Ethics, Software Engineering Practice-Essence of Pract
Software Development Life Cycle

terfall Model – Classical Waterfall Model, Iterative Waterfall Model, Evolutionary model-Spiral, Prototype.

	Software Requirements, Analysis and Design **(Comprehension level)**	Assignment	Development of SRS documents for a given scenario	

nts Engineering: Eliciting requirements, Functional and non- Functional requirements, Software Requirements Specification (SRS), Requirement Ana
Requirements modelling- Introduction to Use Cases, Activity diagram and Swim lane diagram. CASE support in Software Life Cycle, Characteristics o
re of a CASE Environment.

ign concepts, Architectural design, Component based design, User interface design.

	Agile Principles & Devops **(Knowledge level)**	Quiz		

n Roles and activities, Sprint Agile software development methods - Scaling, User Stories, Agile estimation techniques, Product backlogs, Stake holde
ystem Development Method.

roduction, definition, history, tools.

	Software Testing and Maintenance (Application Level)	Assignment	Apply the testing concepts using Programing	

sting-verification and validation, Test Strategies - White Box Testing, Black box Testing. Automation Tools for Testing.

uality Assurance-Elements of software quality assurance, SQA Tasks, Goals and Metrics, Software configuration management- SCM process, SCM To

e- Characteristics of Software Maintenance, Software Reverse Engineering, Software Maintenance Process Models.

pplication & Tools that can be used: Selenium, GitHub, CASE Tools

S. Pressman, "Software Engineering – A Practitioner's Approach", VII Edition, McGraw-Hill, 2017.

ughes, Mike Cotterell, Rajib Mall, "Software Project Management", VI Edition, McGraw-Hill, 2018.

ntals of Software Engineering", VI Edition, PHI learning private limited, 2015.

ftware Engineering", IX Edition, Pearson Education Asia, 2011.

pment Principles, Patterns and Practices.1st Edition, Wiley, 2002

vant to "Skill Development: Balck box Testing, White box Testing, Automated Testing for Skill development through Participative Learning Techniqu
rough assessment mentioned in the course handout

Course Code: CSE2504	**Course Title:** Scalable Application Development using Java **Type of Course:** Theory	**L- T-P- C**	3	**0**	0	**3**

Version No.	2.0			
Course Prerequisites	Basic Java Programming Knowledge, Java framework, understanding of databases and fundamentals of web development, basics of multithreading and concurrency.			
Anti-requisites	**NIL**			
Course Description	This course provides a comprehensive guide to designing, developing, and deploying scalable Java applications, covering high-performance architectures, distributed systems, microservices, cloud deployment, and DevOps integration.			
Course Objective	The objective of the course is to Understand Scalability Principles, developing high performance Java applications, design and implement scalable architecture, deploy and manage scalable data.			
Course Outcomes	On successful completion of the course the students shall be able to: CO 1: Design and Develop Scalable Java Applications CO 2: Design and Implement Scalable Architectures and Micro services CO 3: Implement Scalable Data Management Techniques CO 4: Deploy and Monitor Applications in Cloud Environments			
Course Content:				
Module 1	Foundations of Scalable Java Applications	Assignment	Implementation	**10 Sessions**
Topics: **Introduction to Scalability -** Defining scalability: vertical vs. horizontal scaling, Monolithic vs. Microservices Architecture, Performance bottlenecks and solutions. **Java Performance Optimization -** JVM internals and tuning, Garbage Collection (GC) strategies, Profiling and monitoring Java applications **Concurrency and Multithreading -** Java concurrency model and thread management, Executor framework, ForkJoinPool, Thread safety, locks, and synchronization.				
Module 2	Scalable Architectures and Microservices	Assignment	Implementation	10 Sessions
Topics: **Microservices and Distributed Systems -** Principles of microservices, Service-to-service, communication (REST, gRPC, Kafka), API Gateway, Service Discovery, and Load Balancing				

Design Patterns for Scalability - Singleton, Factory, and Builder patterns, CQRS (Command Query Responsibility Segregation), Circuit Breaker and Retry patterns (Resilience4j)

Event-Driven Architecture & Asynchronous Processing - Event Sourcing with Kafka, WebSockets and Reactive Programming, Handling failures in distributed systems.

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

Topics:

Database Scalability - SQL vs. NoSQL databases (MySQL, PostgreSQL, MongoDB, Cassandra), Sharding, Replication, and Partitioning strategies, Optimizing queries (Indexing, Query Caching)

Caching Strategies - In-memory caching with **Redis, Memcached,** Distributed caching techniques, Cache Invalidation and Eviction Policies

Data Streaming & Batch Processing - Apache Kafka for event-driven data pipelines, Apache Spark for large-scale data processing.

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

Topics:

Cloud-Native Development - Containerization with **Docker,** Kubernetes for scaling and orchestration, Cloud deployment strategies (AWS, GCP, Azure),

CI/CD & DevOps for Scalable Applications - Building CI/CD pipelines (Jenkins, GitHub Actions), Canary & Blue-Green Deployments, Infrastructure as Code (Terraform, Kubernetes Helm)

Security & Reliability - Securing APIs with **OAuth2, JWT, API Gateway,** Load Balancing & Rate Limiting, Monitoring with **Prometheus, ELK Stack**

Targeted Application & Tools that can be used:

Applications:

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

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

Tools:

Programming & Frameworks

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

Database & Caching

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

Messaging & Event-Driven Architecture

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

Cloud & Deployment

5. **Docker** – Containerization for application portability.
6. **Kubernetes** – Scaling, orchestration, and auto-recovery.
7. **AWS / GCP / Azure** – Cloud deployment and auto-scaling.

CI/CD & DevOps

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

Text Book(s):

T1: "Designing Data-Intensive Applications" – *By Martin Kleppmann*

T2: "Java Concurrency in Practice" – *By Brian Goetz*

T3: "Spring Microservices in Action" – *By John Carnell*

T4. "Cloud Native Java" – *By Josh Long & Kenny Bastani*

Reference(s):

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

Catalogue prepared by	
Recommended by the Board of Studies on	
Date of Approval by the Academic Council	

Course Code: **CSE2505**	**Course Title:** Scalable Application Development using Java **Type of Course:** Lab	**L- T-P- C**	0	**0**	4	**2**
Version No.	2.0					
Course Prerequisites	Basic Java Programming Knowledge, Java framework, understanding of databases and fundamentals of web development, basics of multithreading and concurrency.					
Anti-requisites	**NIL**					
Course Description	This course provides a hands-on, practical approach to building scalable, high-performance applications using Java and related technologies. This course is designed to complement theoretical concepts by offering real-world lab exercises focused on the development of microservices architectures, cloud-native applications, and distributed systems. In this lab-intensive course, students will work on building and deploying scalable applications using Spring Boot, Spring Cloud, Docker, Kubernetes, and Apache					

	Kafka. Students will gain experience in implementing RESTful APIs, asynchronous messaging, data caching, and load balancing to ensure that applications can handle increased traffic and scale efficiently. The course will also cover essential techniques for optimizing performance, including JVM tuning, database optimization, and memory management.			
Course Objective	The primary objectives of the course are to Develop hands-on expertise in building scalable applications using Java and modern frameworks like Spring Boot, Spring Cloud, and Apache Kafka, Implement microservices architectures that enable applications to handle increasing loads efficiently through distributed systems and cloud-native practices, Gain practical experience in optimizing performance by leveraging tools for JVM tuning, database optimization, and memory management to improve application responsiveness and scalability., Work with containerization technologies such as Docker and Kubernetes to deploy Java applications in cloud environments with automated continuous integration/continuous deployment (CI/CD) pipelines, Master service discovery, load balancing, and caching mechanisms to ensure high availability, fault tolerance, and low-latency operations in production-grade applications and Apply event-driven architectures to build scalable and resilient systems using tools like Apache Kafka for real-time data processing and messaging.			
Course Outcomes	On successful completion of the course the students shall be able to: ● Implement Performance Optimization Techniques ● Design and Build Scalable Microservices ● Integrate Event-Driven Architectures and Caching. ● Deploy and Scale Applications in Cloud Environments.			
Course Content:				
Module 1	Foundations of Scalable Java Applications	Assignment	Implementation	**10 Sessions**

Lab Assignment 1: Setting Up Development Environment

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

Tasks:

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

Deliverables:

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

Lab Assignment 2: Performance Optimization with JVM

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

5. Implement a **simple Java application** that performs a memory-intensive task (e.g., sorting a large dataset).
6. Use **VisualVM** to monitor **JVM memory usage, CPU usage, and garbage collection**.
7. Optimize the application by adjusting **JVM flags** (e.g., **heap size**, **garbage collection strategy**).
8. Measure the impact of optimizations on **execution time and memory usage**.
   Deliverables:
9. Profiled and optimized **Java application** with performance comparison charts.
10. Detailed report on **JVM tuning** and optimization strategies.

Lab Assignment 3: Implementing Multi-threading

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

5. Implement a **multi-threaded Java application** that simulates multiple tasks (e.g., processing large files, image processing).
6. Use the **Executor framework** to manage thread pools.
7. Measure the **execution time** and compare the performance of **single-threaded** vs **multi-threaded** approaches.
   Deliverables:
8. Source code for **multi-threaded application** with explanations of thread management.
9. Execution time comparison chart.

Module 2	Scalable Architectures and Microservices	Assignment	Implementation	10 Sessions

Lab Assignment 4: Building a Simple Microservice with Spring Boot

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

5. Create a **Spring Boot microservice** that provides a RESTful API (e.g., a book or customer service).
6. Implement **basic CRUD operations** (Create, Read, Update, Delete).
7. Use **Spring Data JPA** to interact with an **SQL database** (e.g., MySQL).
8. Write unit tests using **JUnit** for API endpoints.
   Deliverables:
9. Working **Spring Boot microservice** with API documentation.

10. Source code with unit tests.

Lab Assignment 5: Implementing Service Discovery & Load Balancing

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

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

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

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

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

- Screenshots or logs showing messages being passed from producer to consumer.

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

Lab Assignment 7: Integrating SQL and NoSQL Databases

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

- Integrate a **MySQL** database into a **Spring Boot microservice** and implement CRUD operations.
- Set up a **MongoDB** instance (locally or via Docker) and create a second microservice using **Spring Data MongoDB**.
- Compare the performance and scalability aspects of both databases.
  Deliverables:
- Source code for Spring Boot microservices using **MySQL** and **MongoDB**.
- Database performance comparison with benchmarks.

Lab Assignment 8: Implementing Redis Caching

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

5. Integrate **Redis** with your Spring Boot application.
6. Cache frequently accessed data (e.g., product information, user profiles) in Redis.
7. Implement **cache expiration** and **cache invalidation** strategies.
8. Measure the performance improvement by comparing **cache hits vs. misses**.
   Deliverables:
9. Redis-integrated Spring Boot application with caching logic.
10. Performance comparison between cached and non-cached operations.

Lab Assignment 9: Data Streaming with Kafka

Objective: Implement a **data streaming pipeline** with Kafka for real-time data processing.
Tasks:

- Build a **data pipeline** where **Kafka producers** stream events (e.g., logs, transactions) to **Kafka brokers**.
- Use **Kafka consumers** to process these events in real-time (e.g., updating a database or triggering a workflow).
- Implement **Kafka Streams** to process data within the Kafka ecosystem.
  Deliverables:
- Kafka producer-consumer setup with real-time data flow.

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

Lab Assignment 10: Dockerizing a Spring Boot Application

Objective: Containerize a Spring Boot application using **Docker** for scalability.
Tasks:

- Write a **Dockerfile** for your Spring Boot application.
- Build a **Docker image** and run the application in a Docker container.
- Test the application running in the container and compare performance to local deployment.
  Deliverables:
- Dockerized **Spring Boot application** with a functional test.
- Docker image and run command documentation.

Lab Assignment 11: Deploying with Kubernetes

Objective: Deploy a Spring Boot microservice to a **Kubernetes cluster** for horizontal scaling.
Tasks:

- Deploy the **Dockerized Spring Boot application** to a **Kubernetes cluster** (local Minikube or cloud-based).

- Set up **Kubernetes Pods, Services, and Deployments** to scale the microservice.
- Implement **auto-scaling** based on CPU or memory usage.

Deliverables:

- Kubernetes **deployment YAML files** for Spring Boot application.
- Running Kubernetes cluster with auto-scaling behavior.

Lab Assignment 12: Setting Up CI/CD for Scalable Applications

Objective: Automate the deployment process using **CI/CD pipelines**.

Tasks:

- Set up a **Jenkins** or **GitHub Actions** CI/CD pipeline to automate the build, test, and deployment process.
- Implement **continuous deployment** (CD) for deploying a **Dockerized microservice** to a Kubernetes cluster.
- Test the pipeline by pushing changes to a GitHub repository and triggering the deployment process automatically.

Deliverables:

- **CI/CD pipeline configuration** (Jenkins or GitHub Actions).
- Deployment automation logs and screenshots.

Targeted Application & Tools that can be used:

Applications:

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

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

Tools:

Programming & Frameworks

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

Database & Caching

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

Messaging & Event-Driven Architecture

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

Cloud & Deployment

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

CI/CD & DevOps

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

Text Book(s):

T1: "Spring in Action" by Craig Walls T2: "Java Performance: The Definitive Guide" by Scott Oaks T3: "Designing Data-Intensive Applications" by Martin Kleppmann T4. "Spring Microservices in Action" by John Carnell
Reference(s): 8. **"Designing Data-Intensive Applications"** – *Martin Kleppmann* 9. **"Java Performance: The Definitive Guide"** – *Scott Oaks* 10. **"Spring Microservices in Action"** – *John Carnell* 11. **"Kubernetes Up & Running"** – *Kelsey Hightower, Brendan Burns, Joe Beda*

Catalogue prepared by	
Recommended by the Board of Studies on	
Date of Approval by the Academic Council	

Course Code: MAT2011	Course Title: Numerical Computation Type of Course:1] School Core	L-T- P- C	3	0	0	3
Version No.	1.0					
Course Pre-requisites	Calculus, Linear Algebra, Differential Equations					
Anti-requisites	**NIL**					
Course Description	The course explores mathematical techniques used to approximate solutions to complex problems that are difficult to solve analytically, often utilizing computers to perform calculations, including methods for root finding, interpolation, numerical differentiation and integration, solving systems of linear equations, and approximating solutions to differential equations, with applications across various scientific and engineering fields. It focuses on understanding the theoretical basis behind these methods, their implementation in programming languages, and analyzing their accuracy and stability.					
Course Objective	The objective of the course is to equip students with understanding and ability to apply various numerical techniques to approximate solutions to complex mathematical problems that are difficult or impossible to solve analytically, particularly focusing on areas like solving systems of equations, finding roots of functions, interpolation, numerical differentiation, and integration, often utilizing computational tools to implement these methods.					
Course Out Comes	On successful completion of the course the students shall be able to: CO1 - Calculate errors induced in the values by truncation of a series expansion. CO2 - Demonstrate the applications of numerical methods to find the roots of polynomial equations and eigen values of real symmetric matrices. CO3 - Apply the knowledge of numerical methods in modelling of various physical and engineering phenomena. CO4 - Apply various numerical methods for solving linear Ordinary & Partial differential equations arising in engineering field.					
Course Content:						
Module 1	**Solution of Linear Systems of Equation**					**(12 Classes)**
Numerical Computation: Motivation and Objectives, Number Representation, Machine Precision, Round-of Error, Truncation Error, Random Number Generation. Solution of algebraic and transcendental equations: Various types of errors - Bisection method, Regula-Falsi method, Newton-Raphson method, Graffe's method - Bairstow's method - Newton's method for solving f(x,y) = 0 and g(x,y) = 0, secant method, Fixed point iteration method, Solution of linear system of equations, Gauss elimination method, Pivoting, Gauss Jordan method, Iterative methods of Gauss Jacobi and Gauss Seidel, Sufficient conditions for convergence - LU decomposition method, Eigenvalues of a matrix by Power method and Jacobi's method for symmetric matrices.						
Module 2	**Interpolation and Approximation**	**Assignment**				**(8 Classes)**
Interpolation with equal intervals, Newton's forward and backward difference formulae, Interpolation with unequal intervals, Lagrange's interpolation, Newton's divided difference interpolation, Cubic Splines, Difference operators and relations.						
Module 3	**Numerical Differentiation and Integration**					**(10 Classes)**
Numerical differentiation, Approximation of derivatives using interpolation polynomials, Numerical integration using Trapezoidal rule, Simpson's one-third rule, Simpson's three-eighth rule, Weddle's rule, Romberg's Method, Two point and three point Gaussian quadrature formulae, Evaluation of double integrals by Trapezoidal rule and Simpson's one-third rule						
Module 4	**Initial & Boundary Value Problems for Ordinary & Partial Differential Equations**	**Assignment**				**(15 Classes)**
Single step methods — Taylor's series method, Modified Euler's method, Fourth order Runge-Kutta method for solving first order equations, Multi step methods, Milne's and Adams, Bash forth predictor corrector methods for solving first order equations. Finite difference methods for solving second order, two-point linear boundary value problems, Finite difference techniques for the solution of two-dimensional Laplace's and Poisson's equations on rectangular domain, One-dimensional heat flow equation by explicit and implicit (Crank Nicholson) methods, One-dimensional wave equation by explicit method.						
Targeted Application & Tools that can be used: The contents of this course has direct applications in most of the core engineering courses for problem formulations, Problem Solution and system Design.						

Tools Used: Python.
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. C.F.Gerald and P.O.Wheatley", Applied Numerical Analysis", McGraw-Hill, 1981. 2. Cheneg and Kincaid, "Introduction to Numerical Computing", Tata McGraw-Hill, 1998.
References: 1. SRK Iyengar & RK Jain, Numerical Methods, New Age Internationals. 2. Erwin Kreyzig, Advanced Engineering Mathematics, John Wiley and sons, Inc.10th Edition 3. B. S. Grewal (2017), Higher Engineering Mathematics by, 44th Edition, Khanna Publishers. **E-resources/ Web links:** https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_135224 https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_141727 https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_217628 http://.ac.in/courses.php?disciplineID=111 http://www.class-central.com/subject/math(MOOCs) http://academicearth.org/ https://www.math.hkust.edu.hk/~maqian/ma006_0607F.html https://www.scu.edu.au/study-at-scu/units/math1005/2022/
Topics relevant to SKILL DEVELOPMENT: The course focuses on the concepts of calculus and differential equation with reference to specific engineering problems. The course is of both conceptual and analytical type in nature through Problem solving**.** This is attained through the assessment component mentioned in course handout.

Catalogue prepared by		**Dr. Mohan S**
Recommended by the Board of Studies on		**13th BOS held on 04/01/2025**
Date of Approval by the Academic Council		**24th ACM held in 3rd August 2024**

Course Code: PPS 1001	**Course Title: Introduction to Soft Skills** **Type of Course: Practical Only Course**	L- P- C	0	2	1
Version No.	1.0				
Course Pre-requisites	Students are expected to understand Basic English.				

	Students should have desire and enthusiasm to involve, participate and learn.		
Anti-requisites	NIL		
Course Description	This course is designed to enable students understand soft skills concepts and improve confidence, communication and professional skills to give the students a competitive advantage and increase chances of success in the professional world. 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 "Soft Skills" and attain SKILL DEVELOPMENT through PARTICIPATIVE LEARNING techniques.**		
Course Out Comes	**On successful completion of this course the students shall be able to:** CO1: **Recognize** significance of soft skills CO2: **Illustrate** effective communication while introducing oneself and others CO3: **List** techniques of forming healthy habits CO4: **Apply** SMART technique to achieve goals and increase productivity		
Course Content:			
Module 1	INTRODUCTION TO SOFT SKILLS	**Classroom activity**	**04 Hours**
Topics: Setting Expectations, Ice Breaker, Significance of soft skills, Formal grooming, punctuality			
Module 2	EFFECTIVE COMMUNICATION	**Individual Assessment**	**10 Hours**
Topics: Different styles of communication, Difference between hearing and listening, Effective communication for success, Email etiquette, Self-introduction framework, Video introduction, email- writing, Resume Building- Digital, Video, Traditional.			
Module 3	HABIT FORMATION	**Worksheets & Assignment**	**4 Hours**
Topics: Professional and personal ethics for success, Identity based habits, Domino effect, Habit Loop, Unlearning, standing up for what is right			
Module 4	Goal setting & Time Management	**Goal sheet**	**8 Hours**
A session where students will be introduced to Time management, setting SMART Goals, Introduction to OKR Techniques, Time Management Matrix, steps to managing time through outbound group activity, making a schedule, Daily Plan and calendars (To Do List), Monitoring/charting daily activity			
Targeted Application & Tools that can be used: LMS			

Project work/Assignment: Mention the Type of Project /Assignment proposed for this course
1) Individual Assessment 2) LMS MCQ
The topics related to Skill Development: Communication and professional grooming, Goal setting and presentation for skill development through participative learning techniques. This is attained through assessment component mentioned in course handout.

Catalogue prepared by	L&D Department Faculty members
Recommended by the Board of Studies on	
Date of Approval by the Academic Council	

Course Code: MAT2501	**Course Title: Integral Transforms and Partial Differential Equations** **Type of Course:1] School Core**	**L-T- P- C**	**3**	**0**	**0**
Version No.	1.0				
Course Pre-requisites	Calculus and Differential Equations				
Anti-requisites	**NIL**				
Course Description	This course aims to introduce various transform techniques such as Lapla transform, Fourier transform and Z-transform in addition to expressing functio in terms of Fourier series. The course covers applications of Laplace transform LCR circuits and solutions of different equations using Z-transform. The cou also deals with the analytical methods for solving partial differential equations a the classical applications of partial differential equations.				
Course Objective	The objective of the course is to **familiarize the learners with the concepts "Transform Techniques, Partial Differential Equations**" and attain **Sk Development through Problem Solving Techniques**.				
Course Out Comes	On successful completion of the course the students shall be able to: CO1 - Express functions in terms of uniformly convergent Fourier series. CO2 - Apply Laplace transform technique to solve differential equations. CO3 - Employ Z-transform techniques to solve difference equations. CO4 - Solve a variety of partial differential equations analytically.				

Course Content:			
Module 1	**Laplace Transforms**		**(12 Class**
Definition and Laplace transform of elementary functions. Properties of Laplace transform, and Laplace transfo of periodic function, unit-step function and Impulse function – related problems. Inverse Laplace transform standard functions - problems, initial and final value theorem. Convolution theorem, solution of linear a simultaneous differential equations and LCR Circuit.			
Module 2	**Fourier Series**	**Assignment**	**(8 Class**
Fourier Series: Periodic functions, Dirichlet's condition. Fourier series of periodic functions period 2π a arbitrary period. Half range Fourier series. Practical harmonic analysis.			
Module 3	**Fourier Transforms and Z - Transforms**		**(13 Class**
Fourier Transforms: Definitions, infinite Fourier transforms, Fourier sine and cosine transforms, inverse Four transforms, Problems. **Difference equations and Z-transforms:** Z-transforms – Basic definitions, Standard Z-transforms, Linear property, Damping rule, Shifting rule, Initial value theorem, Final value theorem, Inverse Z-transforms. Differen equations – Basic definitions, Application of Z-transforms to solve difference equations.			
Module 4	**Partial Differential Equations**	**Assignment**	**(12 Class**
Formation of PDE, Solution of non-homogeneous PDE by direct integration, Solution of homogeneous PI involving derivative with respect to one independent variable only (Both types with given set of conditions) Meth of separation of variables. (First and second order equations) Solution of Lagrange's linear PDE. of the type P p Q q = R. Applications of PDE: Derivation of one-dimensional wave and heat equations. Various possible solutions of the by the method of separation of variables. D'Alembert's solution of wave equation. Two-dimensional Laplac equation – various possible solutions. Solution of all these equations with specified boundary conditions (Bounda value problems).			
Targeted Application & Tools that can be used: The objective of the course is to familiarize students with a variety of numerical techniques and the theoretical concepts of probability and statistics to equip them with the necessary numerical approaches and basic statistica tools to tackle engineering and real-life problems.			
Assignment:			
Newton-Raphson Methods, Gauss-Seidel Method, LU Decomposition, Trapezoidal Rule, Simpson's rule, Rung Kutta 4th Order.			
Text Book 1. Erwin Kreyzig, Advanced Engineering Mathematics, John Wiley and sons, Inc.10th Edition 2. B. S. Grewal (2017), Higher Engineering Mathematics by, 44th Edition, Khanna Publishers.			
References: 1. Victor Henner, Tatyana Belozerova, Mickhail Khenner, Ordinary and Partial Differential Equations, CRC Press, Edition, 2013. 2. Walter Ledermann, Multiple integrals, Springer, 1st edition **E-resources/ Web links:** https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EB CO95_30102024_140238 https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EB CO95_30102024_233298 https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EB CO95_30102024_204892 https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EB CO95_30102024_246791 https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EB CO95_30102024_223548 https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EB CO95_30102024_134719			

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EB
CO95_30102024_32614
https://www.math.hkust.edu.hk/~maqian/ma006_0607F.html
https://www.scu.edu.au/study-at-scu/units/math1005/2022/

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

Catalogue prepared by		**Dr. Husna**
Recommended by the Board of Studies on		**13th BOS held on 04/01/2025**
Date of Approval by the Academic Council		**24th ACM held in 3rd August 2024**

Course Code: MAT2502	Course Title: Numerical Methods and Complex Variables Type of Course:1] School Core	L-T- P- C	3	0	0	3
Version No.	2.0					
Course Pre-requisites	Calculus & Differential Equations					
Anti-requisites	**NIL**					
Course Description	Numerical methods contain solutions of system of linear equations, roots of non-linear equations, interpolation, numerical differentiation and integration. It plays an important role in solving various engineering sciences problems. Complex Variable is functions involving complex numbers as variables, exploring concepts like limits, continuity, differentiation, integration, and series within the complex plane, with a focus on key topics like Cauchy-Riemann equations, complex exponentials, contour integration, residues, and applications to solving real-world problems in physics and engineering.					
Course Objective	Numerical methods is to provide approximate, yet accurate solutions to complex mathematical problems that are often difficult or impossible to solve analytically, by using computational techniques to generate solutions through iterative processes, especially when dealing with real-world scenarios involving large datasets or intricate equations. Complex variable is to study the techniques of complex variables and functions together with their derivatives, Contour integration and transformations. To study complex power series, classification of singularities, calculus of residues and its applications in the evaluation of integrals, and other concepts and properties.					
Course Out Comes	On successful completion of the course the students shall be able to: CO1 - Demonstrate the applications of numerical methods to find the roots of polynomial equations and eigen values of real symmetric matrices. CO2 - Interpret the fitted parameters and apply curve fitting techniques to real-world data analysis problems. CO3 - Apply various numerical methods for solving linear Ordinary & Partial differential equations arising in engineering field. CO4 - Apply the Cauchy-Riemann equations to identify analytic functions.					
Course Content:						

Module 1	**Solution of Linear Systems of Equation**		***(10 Classes)***
Solution of algebraic and transcendental equations: Various types of errors - Bisection method, Regula-Falsi method, Newton-Raphson method, Graffe's method - Bairstow's method - Newton's method for solving $f(x,y) = 0$ and $g(x,y) = 0$, secant method, Fixed point iteration method, Solution of linear system of equations, Gauss elimination method, Pivoting, Gauss Jordan method, Iterative methods of Gauss Jacobi and Gauss Seidel, Sufficient conditions for convergence - LU decomposition method, Eigenvalues of a matrix by Power method and Jacobi's method for symmetric matrices.			
Module 2	**Interpolation and Curve Fitting**	**Assignment**	***(10 Classes)***
Newton's forward and backward interpolation, Divided difference method, Lagrange's method. Method of least squares to fit equations of the form $y = ax + b$, $y = ax^2 + bx + c$, $y = ae^{bx}$, $y = ab^x$ and $y = ax^b$.			
Module 3	**Numerical Differentiation and Integration**		***(10 Classes)***
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. Euler's method - Taylor's method - Runge-Kutta method of fourth order - Numerical solution of Laplace equation - One-dimensional heat flow equation and wave equation by finite difference methods.			
Module 4	**Complex Variables**	**Assignment**	***(15 Classes)***
Introduction, Cauchy-Riemann equations, analytic functions, harmonic functions, finding harmonic conjugate; Conformal mappings. Complex Integration: Contour integrals, Cauchy-Goursat theorem (without proof), Cauchy Integral formula (without proof), Liouville's theorem and Maximum-Modulus theorem (without proof); Taylor's series, zeros of analytic functions, singularities, Laurent's series; Residues, Cauchy Residue theorem (without proof).			
Targeted Application & Tools that can be used:			

Numerical methods are widely applied in various fields like engineering, physics, finance, and biology, primarily used to solve complex problems where analytical solutions are difficult or impossible to find, allowing for the approximation of solutions through computational algorithms.
Complex variable methods are applied to elliptical problems in fluid mechanics, and linear elasticity. The techniques presented for solving parabolic problems are the Laplace transform and separation of variables, illustrated for problems of heat flow and soil mechanics.

Assignment:

1. Calculate its absolute and relative errors for different input values using a numerical method like the Taylor series approximation.
2. Given sin 45° = 0.7071, sin 50° = 0.7660, sin 55° = 0.8192, sin 60° = 0.8660 find sin 57° $and\ sin$ 52° using an appropriate interpolation formula.
3. Find the equation of the polynomial which passes through the points (4,-43), (7, 83), (9, 327), (12, 1053) using Newton's divided difference interpolation formula.

Text Book

1. Brown & Churchill, Complex Variables and Applications, McGraw Hill Higher Education; 9th edition.
2. B. S. Grewal (2017), Higher Engineering Mathematics by, 44th Edition, Khanna Publishers.

References:

1. Erwin Kreyzig, Advanced Engineering Mathematics, John Wiley and sons, Inc.10th Edition.
2. M.K. Jain, S.R.K. Iyengar and R.K. Jain, Numerical Methods for Scientific and Engineering Computations, 6th Edition, New age Publishing House, 2015.
3. Carlos A. Berenstein & Roger Gay, Complex Variables - An Introduction, Springer-Verlag New York Inc.

E-resources/ Web links:

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_166145
https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_141727
https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_135224
https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_246791
https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_190270
https://www.math.hkust.edu.hk/~maqian/ma006_0607F.html
https://www.scu.edu.au/study-at-scu/units/math1005/2022/

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

Catalogue prepared by		**Dr. Chandni Kumar & Dr. Heena Firdose**
Recommended by the Board of Studies on		**13th BOS held on 04/01/2025**
Date of Approval by the Academic Council		**24th ACM held in 3rd August 2024**

Course Code: MAT2602	Course Title: Numerical Computations Type of Course:1] School Core	L-T- P- C	3	0	0	3
Version No.	1.0					
Course Pre-requisites	Calculus, Linear Algebra, Differential Equations					
Anti-requisites	**NIL**					
Course Description	The course explores mathematical techniques used to approximate solutions to complex problems that are difficult to solve analytically, often utilizing computers to perform calculations, including methods for root finding, interpolation, numerical differentiation and integration, solving systems of linear equations, and approximating solutions to differential equations, with applications across various scientific and engineering fields. It focuses on understanding the theoretical basis behind these methods, their implementation in programming languages, and analyzing their accuracy and stability.					
Course Objective	The objective of the course is to equip students with understanding and ability to apply various numerical techniques to approximate solutions to complex mathematical problems that are difficult or impossible to solve analytically, particularly focusing on areas like solving systems of equations, finding roots of functions, interpolation, numerical differentiation, and integration, often utilizing computational tools to implement these methods.					
Course Out Comes	On successful completion of the course the students shall be able to: CO1 - Calculate errors induced in the values by truncation of a series expansion. CO2 - Demonstrate the applications of numerical methods to find the roots of polynomial equations and eigen values of real symmetric matrices. CO3 - Apply the knowledge of numerical methods in modelling of various physical and engineering phenomena. CO4 - Apply various numerical methods for solving linear Ordinary & Partial differential equations arising in engineering field.					
Course Content:						
Module 1	**Solution of Linear Systems of Equation**					**(12 Classes)**
Numerical Computation: Motivation and Objectives, Number Representation, Machine Precision, Round-of Error, Truncation Error, Random Number Generation. Solution of algebraic and transcendental equations: Various types of errors - Bisection method, Regula-Falsi method, Newton-Raphson method, Graffe's method - Bairstow's method - Newton's method for solving f(x,y) = 0 and g(x,y) = 0, secant method, Fixed point iteration method, Solution of linear system of equations, Gauss elimination method, Pivoting, Gauss Jordan method, Iterative methods of Gauss Jacobi and Gauss Seidel, Sufficient conditions for convergence - LU decomposition method, Eigenvalues of a matrix by Power method and Jacobi's method for symmetric matrices.						
Module 2	**Interpolation and Approximation**	**Assignment**				**(8 Classes)**
Interpolation with equal intervals, Newton's forward and backward difference formulae, Interpolation with unequal intervals, Lagrange's interpolation, Newton's divided difference interpolation, Cubic Splines, Difference operators and relations.						
Module 3	**Numerical Differentiation and Integration**					**(10 Classes)**
Numerical differentiation, Approximation of derivatives using interpolation polynomials, Numerical integration using Trapezoidal rule, Simpson's one-third rule, Simpson's three-eighth rule, Weddle's rule, Romberg's Method, Two point and three point Gaussian quadrature formulae, Evaluation of double integrals by Trapezoidal rule and Simpson's one-third rule						
Module 4	**Initial & Boundary Value Problems for Ordinary & Partial Differential Equations**	**Assignment**				**(15 Classes)**
Single step methods — Taylor's series method, Modified Euler's method, Fourth order Runge-Kutta method for solving first order equations, Multi step methods, Milne's and Adams, Bash forth predictor corrector methods for solving first order equations. Finite difference methods for solving second order, two-point linear boundary value problems, Finite difference techniques for the solution of two-dimensional Laplace's and Poisson's equations on rectangular domain, One-dimensional heat flow equation by explicit and implicit (Crank Nicholson) methods, One-dimensional wave equation by explicit method.						
Targeted Application & Tools that can be used: The contents of this course has direct applications in most of the core engineering courses for problem formulations, Problem Solution and system Design.						

Tools Used: Python.

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

3. C.F.Gerald and P.O.Wheatley", Applied Numerical Analysis", McGraw-Hill, 1981.
4. Cheneg and Kincaid, "Introduction to Numerical Computing", Tata McGraw-Hill, 1998.

References:

4. SRK Iyengar & RK Jain, Numerical Methods, New Age Internationals.
5. Erwin Kreyzig, Advanced Engineering Mathematics, John Wiley and sons, Inc.10th Edition
6. B. S. Grewal (2017), Higher Engineering Mathematics by, 44th Edition, Khanna Publishers.

E-resources/ Web links:

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_135224
https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_141727
https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_217628
http://.ac.in/courses.php?disciplineID=111
http://www.class-central.com/subject/math(MOOCs)
http://academicearth.org/
https://www.math.hkust.edu.hk/~maqian/ma006_0607F.html
https://www.scu.edu.au/study-at-scu/units/math1005/2022/

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

Catalogue prepared by		**Dr. Mohan S**
Recommended by the Board of Studies on		**13th BOS held on 04/01/2025**
Date of Approval by the Academic Council		**24th ACM held in 3rd August 2024**

Course Code: MAT2605	Course Title: Discrete Mathematics Type of Course:1] School Core	L-T- P- C	4	0	0	4
Version No.	1.0					
Course Pre-requisites	Linear Algebra					
Anti-requisites	**NIL**					
Course Description	The course explores the study of mathematical structures that are fundamentally discrete (not continuous), focusing on concepts like set theory, logic, graph theory, combinatorics, and number theory, with applications primarily in computer science fields like algorithms, software development, and cryptography; it covers topics such as propositional logic, proof techniques, relations, functions, counting principles, and basic graph algorithms, providing a foundation for analyzing discrete problems and structures within computer science.					
Course Objective	The main objective of the course is that students should learn a particular set of mathematical facts and how to apply them. It teaches students how to think logically and mathematically through five important themes: mathematical reasoning, combinatorial analysis, discrete structures, algorithmic thinking, and applications and modeling. A successful discrete mathematics course should carefully blend and balance all five themes.					
Course Outcomes	On successful completion of the course the students shall be able to: CO1 - Explain logical sentences through predicates, quantifiers and logical connectives. CO2 - Deploy the counting techniques to tackle combinatorial problems CO3 - Comprehend the basic principles of set theory and different types of relations. CO4 - Apply different types of structures of trees for developing programming skills					
Course Content:						
Module 1	**Fundamentals of Logic**					**(10 Classes)**
Basic Connectives and Truth Tables, Propositional Logic, Applications of Propositional Logic, Propositional Equivalences, Predicates and Quantifiers, Nested Quantifiers, Rules of Inference, Introduction to Proofs, Proof Methods and Strategy.						
Module 2	**Principle of Counting**	**Assignment**				**(15 Classes)**
The Well Ordering Principle – Mathematical Induction The Basics of Counting, Permutations and Combinations, Binomial Coefficients and Identities, Generalized Permutations and Combinations, Generating Permutations and Combinations Advanced Principle Counting: The Principle of Inclusion and Exclusion, Generalizations of the Principle, Derangements – Nothing is in its Right Place, Rook Polynomials.						
Module 3	**Relations and Functions**					**(10 Classes)**
Cartesian Products and Relations, Functions, One-to-One, Onto Functions. The Pigeon-hole Principle, Function Composition and Inverse Functions. Relations, Properties of Relations, Computer Recognition – Zero-One Matrices and Directed Graphs, Partial Orders, Lattice, Hasse Diagrams, Equivalence Relations and Partitions.						
Module 4	**Recurrence Relations and Generating Functions**					**(10 Classes)**
Homogeneous and inhomogeneous recurrences and their solutions - solving recurrences using generating functions - Repertoire method - Perturbation method - Convolutions - simple manipulations and tricks.						
Module 5	**Graph Theory & Algorithms on Networks**	**Assignment**				**(15 Classes)**
Definitions and basic results - Representation of a graph by a matrix and adjacency list - Trees - Cycles - Properties - Paths and connectedness - Sub graphs - Graph Isomorphism - Operations on graphs - Vertex and edge cuts - Vertex and edge connectivity, Euler and Hamilton Paths, Shortest-Paths. Tree - Definitions, Properties, and Examples, Routed Trees, Binary search tree, Decision tree, spanning tree: BFS, DFS. Algorithms on Networks - Shortest path algorithm- Dijikstra's algorithm, Minimal spanning tree- Kruskal algorithm and Prim's algorithm.						
Targeted Application & Tools that can be used:						

Discrete mathematics provides the mathematical foundations for many computer science courses including data structures, algorithms, database theory, automata theory, formal languages, compiler theory, computer security, and operating systems.

Assignment:

Assignment 1: Logic Equivalences and Predicate calculus.
Assignment 2: Equivalence Relations and Lattices
Assignment 3: Recurrence Relations

Text Book

1. Kenneth H. Rosen, "Discrete Mathematics and its Applications", McGraw-Hill,s 8th Edition,2019.
2. Harary – Graph Theory, Addison-Wesley Publishing Company.

References:

1. Arthur Gill, "Applied Algebra for Computer Science", Prentice Hall.
2. K.D. Joshi, "Discrete Mathematics", Wiley Eastern Ltd.
3. Ralph. P. Grimaldi., "Discrete and Combinatorial Mathematics: An Applied Introduction", 4th Edition, Pearson Education Asia.

E-resources/ Web links:

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_54588
https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_375
https://www.math.hkust.edu.hk/~maqian/ma006_0607F.html
https://www.scu.edu.au/study-at-scu/units/math1005/2022/

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

Catalogue prepared by		**Dr. V. Ramalatha**
Recommended by the Board of Studies on		**13th BOS held on 04/01/2025**
Date of Approval by the Academic Council		**24th ACM held in 3rd August 2024**

Course Code: **CAI2500**	**Course Title: Machine Learning** **Type of Course: 1] Program Core 2] Theory**	**L-T-P-C**	3	0	0	3
Version No.	1.0					
Course Pre-requisites	Calculus and Differential Equations					
Anti-requisites	**NIL**					

Course Description	Machine Learning algorithms are the key to develop intelligent systems such as Apple's Siri, Google's self-driving cars etc. This course introduces the concepts of the core machine learning techniques such as Regression learning, Bayesian learning, Ensemble learning, Perceptron learning, Unsupervised learning, Competitive learning, learning from Gaussian mixture models and learning to detect outliers. Course lectures cover both the theoretical foundations as well as the essential algorithms for the various learning methods. Lectures enable the students to develop intelligent systems for real life problems.			
Course Objectives	This course is designed to improve the learners 'EMPLOYABILITY SKILLS' by using EXPERIENTIAL LEARNING techniques.			
Course Out Comes	On successful completion of the course the students shall be able to: CO1: Apply advanced supervised machine learning methods for predictive modeling. [Apply] CO2: Employ advanced unsupervised learning algorithms for clustering, competitive learning and outlier detection [Apply] CO3: Build machine learning models with better predictive performance using Ensemble learning algorithms [Apply] CO4: Build predictive models using Perceptron learning algorithms [Apply]			
Course Content:				
Module 1	Supervised Learning	Participative Learning	Brainstorming session/Quiz	**No. of Sessions L – 10**
Topics: An overview of Machine Learning(ML); ML workflow; types of ML; Types of features, Feature Engineering -Data Imputation Methods; Regression – introduction; simple linear regression, loss functions; Polynomial Regression; Logistic Regression; Softmax Regression with cross entropy as cost function**; Bayesian Learning –** Bayes Theorem, estimating conditional probabilities for categorical and continuous features, Naïve Bayes for supervised learning; Bayesian Belief networks; Support Vector Machines – soft margin and kernel tricks.				
Module 2	Unsupervised Learning	Participative Learning	Brainstorming session/Quiz	**No. of Sessions L-8**
Topics: **Unsupervised Learning –** k Means clustering- simple and mini-batch**;** updating centroids incrementally; finding the optimal number of clusters using Elbow method ; Silhouette coefficient, drawbacks of kMeans, kMeans++ **;** Divisive hierarchical clustering – bisecting k-means, clustering using Minimum Spanning Tree (MST), Density Based Spatial Clustering – DBSCAN; Outlier Detection methods – Isolation Forest, Local Outlier Factor(LOF)				
Module 3	Ensemble Learning	Participative Learning	Brainstorming session/Quiz	**No. of Sessions L-6**
Topics: **Ensemble Learning –** using subset of instances – Bagging, Pasting, using subset of features –random patches and random subspaces method; Voting Classifier, Random Forest; Boosting – AdaBoost, Gradient Boosting, Stacking.				

Module 4	Perceptron Learning	Participative Learning	Brainstorming session/Quiz	**No. of Sessions L-6**
Topics: **Perceptron Learning –** from biological to artificial neurons, Perceptron, Linear Threshold Units, logical computations with Perceptron, common activation functions – sigmoid, tanh, relu and SoftMax, common loss functions, multi-layer Perceptron and the Backpropagation algorithm using Gradient Descent.				
Targeted Application & Tools that can be used: Students can be assigned a mini project to develop a machine learning application for real-life problems in various domains such as health care, business intelligence, environmental modeling, etc.				
Project work/Assignment: 1. Certification course in Machine Learning through NPTEL 2. Mini Project on (Module 1 to Module 4)				
Textbooks 1. Aurélien Géron, "Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow", Oreilly, Second Edition, 2019. 2. Andreas C Muller, Sarah Guido, "Introduction to Machine Learning with Python: A Guide for Data Scientists", Oreilly, First Edition, 2018				
References 1. Giuseppe Bonaccorso, "Machine Learning Algorithms: A reference guide to popular algorithms from data science and machine learning", Packt Publishing, 2017				

Course Code: **CAI2501**	**Course Title: Machine Learning Lab** **Type of Course: 1] Program Core** **2] Laboratory**	0	0	4	2
Version No.	1.0				
Course Pre-requisites	Computational Thinking using Python Lab				
Anti-requisites	**NIL**				
Course Description	Machine Learning algorithms are the key to develop intelligent systems such as Apple's Siri, Google's self-driving cars etc. This course introduces the concepts of the core machine learning techniques such as Regression learning, Bayesian learning, Ensemble learning, Perceptron learning, Unsupervised learning, Competitive learning, learning from Gaussian mixture models and learning to detect outliers. Course lectures cover both the theoretical foundations as well as the essential algorithms for the				

	various learning methods. Lab sessions complement the lectures and enable the students to develop intelligent systems for real life problems.
Course Objectives	This course is designed to improve the learners 'EMPLOYABILITY SKILLS' by using EXPERIENTIAL LEARNING techniques.
Course Out Comes	On successful completion of the course the students shall be able to: CO1: Apply advanced supervised machine learning methods for predictive modeling. [Apply] CO2: Employ advanced unsupervised learning algorithms for clustering, competitive learning and outlier detection [Apply] CO3: Build machine learning models with better predictive performance using Ensemble learning algorithms [Apply] CO4: Build predictive models using Perceptron learning algorithms [Apply]

List of Laboratory Tasks:

Experiment N0 1: Methods for handling missing values
Level 1: Given a data set from UCI repository, implement the different ways of handling missing values in it using Scikit-learn library of Python
Level 2: Implement one of these methods using a custom defined function in Python.

Experiment No. 2: Data Visualization
Level 1 : Perform Exploratory Data Analysis for a given data set by creating Scatter Plot, Pair Plot, Count Plot using Matplotlib and Seaborn
Level 2: Create Heat Maps, Word Cloud

Experiment No. 3: Regression learning
Level 1: Given a data set from UCI repository, implement the simple linear regression algorithm and estimate the model's parameters and the performance metrics. Plot the learning curves.
Level 2: Implement the polynomial regression algorithm. Compare the learning curves of Polynomial and Linear Regression.

Experiment No.4: Logistic regression
Level 1: Write custom code for generating the logistic/sigmoid plot for a given input
Level 2 : Given a data set from UCI repository, implement the Logistic regression algorithm. Estimate the class probabilities for a given test data set. Plot and analyze the decision boundaries.

Experiment No.5: Bayesian Learning
Level 1: Given a data set from UCI repository, implement a classification model using the Bayesian algorithm

Experiment No.6: Support Vector Machine (SVM)
Level 1: Given data sets from UCI repository, implement a linear SVM based classification model.
Level 2: Given data sets from UCI repository, implement a non-linear SVM based classification model.

Experiment No. 7: Unsupervised Learning
Level 1: K–means – simple and mini-batch. Finding the optimal number of clusters using Elbow method and Silhouette Coefficient. Compare the inertia of both as k increases. Tuning the hyperparameter 'k' using GridSearchCV.
Level 2: Using clustering for Image segmentation and Preprocessing. Kmeans++

Experiment No. 8: Density Based Clustering
Level 1: Implement DBSCAN – clustering using the local density estimation.

Level 2: Perform hard and soft clustering for new instances.

Experiment No. 9: Ensemble Learning using Subset of Instances
Level 1: Implement Ensemble Learning algorithms such as Bagging, Pasting and Out-of Bag Evaluation
Level 2: Random Patches and Random Subspace Method

Experiment No. 10: Ensemble Learning using Subset of Features
Level 1: Apply ensemble learning techniques such as AdaBoost and Gradient Boosting
Level 2: Apply ensemble learning techniques such as Stacking

Experiment No. 11: Perceptron Learning
Level 1: Implement the Perceptron Classifier
Level 2: – An Image Classifier Using the Sequential API of Keras

Experiment No. 12: Outlier Detection
Level 1 Outlier Detection using Isolation Forest
Level 2: Outlier Detection using Local Outlier Factor

Targeted Application & Tools that can be used:

1. Execution of the ML algorithms will be done using the Google's cloud service namely "Colab", available at https://colab.research.google.com/ or Jupyter Notebook.
2. The data sets will be from the bench marking repositories such as UCI machine learning repository available at: https://archive.ics.uci.edu/ml/index.php
3. Laboratory tasks will be implemented using the libraries available in Python such as Scikit learn, matplotlib, seaborn, perceptron and the deep learning framework namely Keras.

Project work/Assignment:

Students can be assigned a mini project to develop a machine learning application for real-life problems in various domains such as health care, business intelligence, environmental modeling, etc.

Textbook s 2. Aurélien Géron, "Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow", Oreilly, Second Edition, 2019. 3. Andreas C Muller, Sarah Guido, "Introduction to Machine Learning with Python :A Guide for Data Scientists", Oreilly, First Edition, 2018 3. Giuseppe Bonaccorso, "Machine Learning Algorithms: A reference guide to popular algorithms from data science and machine learning", Packt Publishing, 2017.
References 1. Giuseppe Bonaccorso, "Machine Learning Algorithms: A reference guide to popular algorithms from data science and machine learning", Packt Publishing, 2017

Course Code: CAI2504	Course Title: Natural Language Processing Type of Course: Program Core -Theory	L-T-P-C	3	0	0	3
Version No.	1.0					
Course Pre-requisites	CSE1700 Essentials of AI					
Anti-requisites	NIL					
Course Description	This course introduces the basics of Natural Language Processing methods with specific emphasis on modern applications. The course will teach students different concepts of natural language processing, such as word representations, text representations, part-of-speech tagging, word sense disambiguation, parsing, etc. Topics: Word representations, Part-of-Speech tagging, chunking, parsing, text classification, sentiment analysis, named entity recognition, and machine translation.					
Course Objectives	The objective of the course is EMPLOYBILITY of student by using EXPERIENTIAL LEARNING techniques.					
Course Out Comes	On successful completion of this course the students shall be able to: 28. Define different problems related to natural language processing. [Understand] 29. Discuss using NLP techniques for different applications. [Apply] 30. Propose solutions for a particular NLP problem using different machine learning and deep learning techniques. [Apply] 31. Learn to use different NLP tools and packages. [Apply]					
Course Content:						
Module 1	Introduction to Natural Language Processing	Assignment	Case Study on Text Classification			No. of sessions:08

Definition of Natural Language Processing; Overview of various NLP tasks; Sentence and word boundary detection; Introduction to word representation, PoS tagging, Chunking and Parsing, and text classification; Applications of NLP (Sentiment Analysis, Named Entity Recognition, Machine Translation).				
Module 2	Word and Text Representation	Hands-on coding	Implementing and Comparing Word Embeddings	No. of sessions:08
Introduction to Word Embeddings; Creation of word embeddings using Skipgram; Using word embeddings like GloVe / fastText; Cross-lingual word embeddings (e.g., MUSE); Pre-trained monolingual and multilingual language models; Text representations using BoW, feature-based, kernel, and embedding-based representations;				
Module 3	Part-of-Speech Tagging, Chunking and Parsing	Hands-on coding	Implementing PoS Tagging and Parsing	No. of sessions:08
Sequence Labeling and Hidden Markov Model; Viterbi Algorithm; Part-of-Speech Tagging; Using NLTK and Spacy for PoS Tagging; Building a PoS Tagger; Chunking and Constituency Parsing; Using Parser from NLTK; Introduction to Transformer Models (Basic concept of BERT and its applications in NLP).				
Module 4	NLP Applications and Ethical AI	Assignment	NLP Applications and Ethical AI	No. of Sessions: 06
Lexical Resource Creation – Creation and evaluation. Agreement metrics; Sentiment Analysis – Definitions, Challenges (Sarcasm, Thwarting, etc.); Named-Entity Recognition – Definition, Relationship between NER and PoS tagging; Machine Translation – Definition, Challenges, Approaches and Paradigms, Evaluation Techniques. Ethical NLP & Bias in AI.				

Targeted Application & Tools that can be used:

15. Execution of the NLP task will be done using the Google's cloud service namely "Colab", available at https://colab.research.google.com/, Anaconda Navigator.
16. Laboratory tasks will be implemented using the libraries available in Python such as NLTK, Gensim, Spacy and Huggingface Transformers.

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

4. 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):

8. Daniel Jurafsky, James H. Martin. "Speech and Language Processing: An Introduction to Natural Language Processing", Computational Linguistics and Speech, Pearson Publication, 2024 (3rd Edition Draft).
9. 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: CA255	Course Title: Natural Language Processing Type of Course: Program Core -Laboratory	L-T-P-C	0	0	2	1
Version No.	1.0					
Course Pre-requisites	CSE1701 Essentials of AI Lab					
Anti-requisites	NIL					
Course Description	This course introduces the basics of Natural Language Processing methods with specific emphasis on modern applications. The course will teach students different concepts of natural language processing, such as word representations, text representations, part-of-speech tagging, word sense disambiguation, parsing, etc. Topics: Word representations, Part-of-Speech tagging, chunking, parsing, text classification, sentiment analysis, named entity recognition, and machine translation.					
Course Objectives	The objective of the course is EMPLOYBILITY of student by using EXPERIENTIAL LEARNING techniques.					
Course Out Comes	On successful completion of this course the students shall be able to: 32. Define different problems related to natural language processing. [Understand] 33. Discuss using NLP techniques for different applications. [Apply] 34. Propose solutions for a particular NLP problem using different machine learning and deep learning techniques. [Apply] 35. Learn to use different NLP tools and packages. [Apply]					

Course Content:	No. of Sessions: 15 (30 hours)
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:

17. Execution of the NLP task will be done using the Google's cloud service namely "Colab", available at https://colab.research.google.com/, Anaconda Navigator.
18. Laboratory tasks will be implemented using the libraries available in Python such as NLTK, Gensim, Spacy and Huggingface Transformers.

Project work/Assignment: Mention the Type of Project /Assignment proposed for this course To enhance their understanding and gain practical exposure to NLP concepts, students are encouraged to complete a certification related to Natural Language Processing (NLP). 📌 Natural Language Processing - NPTEL 📌 Deep Learning for NLP - NPTEL 📌 Applied Natural Language Processing - NPTEL
Textbook(s): 10. Daniel Jurafsky, James H. Martin. "Speech and Language Processing: An Introduction to Natural Language Processing", Computational Linguistics and Speech, Pearson Publication, 2024 (3rd Edition Draft). 11. 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)

Catalogue prepared by	Dr. Sandeep Albert Mathias Ms. Devi.S
Recommended by the Board of Studies on	BOS NO: SOCSE 2nd BOS held on 17/03/25
Date of Approval by the Academic Council	Academic Council Meeting No 21, Dated 17/03/25

Course Code: CSE 2503	**Course Title: Cryptography and Network Security** **Type of Course: Program Core & Theory only**	**L- T-P-C**	3	0	0	3
Version No.	1					

Course Pre-requisites		"Data Communications 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. **Topics**: 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	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: **CO1:** Identifies the basic concept of Cryptography **(Knowledge)** **CO2:** Express the different types of Cryptographic Algorithms. **(Comprehension)** **CO3:** Recognize the Public key Cryptographic Techniques for various applications. **(Comprehension)** **CO4:** Apply the network security concepts during their implementation of network security application developments. **(Application)**			
Course Content:				
Module 1	**Introduction to Cryptography**	Assignment	Identify the Concepts	08 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 Ciphers : Caesar, Mono alphabetic, Polyalphabetic, Play-fair and Hill Cipher, Introduction to Block Cipher and Stream Cipher, Festal Structure.				
Module 2	**Private Key Cryptography and Number Theory**	Assignment	Analysis of requirement of complexity in cryptography	13 Sessions
Topics: Symmetric Encryption Algorithms : Data Encryption Standard, Introduction to Galois Field, Advanced Encryption Standard, Modular Arithmetic, Prime numbers, Fermat's little theorem, brief about primality testing and factorization, Discrete Logarithmic Problem, Euclidean and Extended Euclidean Algorithm, Euler Totient Function, Chinese Remainder Theorem				

Module 3	**Public Key Cryptography and its Applications**	Assignment	Recognize the importance of various security concepts to achieve sufficient solutions	10 Sessions
	Topics: Overview of Public Key Cryptography, RSA, Diffie - Helman Key exchange, Man in the middle attack, Cryptographic Hash functions, Secure Hash Algorithm, Message Authentication Codes – HMAC, Digital Signature, Discussion on real time practices of Cryptography.			
Module 4	**Network Security**	Assignment	Implement the advanced network security algorithms in recent applications.	07 Sessions
Topics: Network Security fundamentals, Network Security applications: Authentication: Kerberos, PKI, Network Security applications: e-mail security: PGP, MIME, Network Security applications: IP Security: IP Sec architecture, Network Security applications: Web Security.				
Targeted Application & Tools that can be used: Students get the knowledge about cryptography techniques followed, the algorithms used for encryption and decryptions & the techniques for authentication and confidentiality of messages.				
Assignment:				
Assignment 1: Solve the problems of basic encryption techniques. **Assignment 2:** Solve and analyze the problems on symmetric and asymmetric encryption.				
Textbooks: 1.William Stallings, "Cryptography and Network Security - Principles and Practices", Prentice Hall, 8th Edition, 2019. 2. Wade Trappe and Lawrence C Washington, “Introduction to Cryptography with Coding Theory”, Pearson, 2020. **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. AtulKahate, "Cryptography and Network Security", Tata McGraw-Hill, 2nd Edition, 2019. 4. BruceSchneier, “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/opac-detail.pl?biblionumber=10133&query_desc=kw%2Cwrdl%3A%20Cryptography%20and%20Network%20Security				

4.http://182.72.188.195/cgi-bin/koha/opac-detail.pl?biblionumber=5875&query_desc=kw%2Cwrdl%3A%20Cryptography%20and%20Network%20Security.
Topics relevant to "Skill Development": Symmetric and Asymmetric Encryption Algorithms and its problems.

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

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

Course Content:						

Module 1	**Pillars of Computational Thinking**	Comprehension		**9 Sessions**
What is computational thinking? Why is it important? Pillars of computational thinking: decomposition; pattern recognition; data representation and abstraction; algorithms Applying computational thinking to case studies				
Module 2	**Algorithm Design & Problem-Solving Strategies**	Application		**9 Sessions**
Introduction to Algorithms, Introduction to Problem Solving techniques: Brute Force, Divide and conquer, Common algorithms: find-max, linear search, binary search and other simple Algorithms				
Module 3	**Applied Computational Thinking using Python**	Application		**12 Sessions**
Introduction to Python, Data representation: variables, lists, Conditionals, Loops and Iteration Basic Example programs to illustrate the programming constructs				
Targeted Application & Tools that can be used: **Google Colab, Python**				
Text Book 1. "Computational Thinking for the Modern Problem Solver" – David D. Riley & Kenny A. Hunt 2. "Mastering Python 3 Programming: Ultimate Guide to Learn Python Coding Fundamentals and Real-World Applications" Subburaj Ramaswamy, BPB publications				
References 1. • **Sweigart, Al.** *Automate the Boring Stuff with Python: Practical Programming for Total Beginners.* No Starch Press, 2015. https://automatetheboringstuff.com • **Severance, Charles.** *Python for Everybody: Exploring Data Using Python 3.* CreateSpace Independent Publishing, 2016. https://www.py4e.com				

- **Wing, Jeannette M.**

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

- **Downey, Allen B.**

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

.

E-Resources

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

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

Course Code: CSE2514	**Course Title: Operating Systems Lab** **Type of Course: Lab Only**	**L-T- P- C**	0	0	2	1
Version No.	1.0					
Course Pre-requisites	CSE2009- Computer Organization Students should have basic knowledge on computers, computer software & hardware, and Computer Organization. Prior programming experience in C is recommended.					
Anti-requisites	**NIL**					
Course Description	This laboratory course provides hands-on experience with the core concepts of operating systems through practical assignments, simulations, and case studies. It covers foundational aspects such as system calls, process and thread management, inter-process communication, synchronization, deadlocks, memory management, and file systems. Students will implement and simulate real-time OS components and scheduling algorithms, fostering deeper understanding of OS architecture and design. The lab also introduces modern OS tools, programming interfaces, and the basics of open-source OS environments.					
Course Object	The objective of the course is to familiarize the learners with the concepts of Operating Systems and attain **Employability** through **Problem Solving** Methodologies.					

Course Out Comes	On successful completion of the course the students shall be able to: 1] Demonstrate system-level programming using system calls and OS structures. [**Apply**] 2] Simulate process scheduling and multithreading techniques. [**Apply**] 3] Apply various tools to handle synchronization problems using semaphores and shared memory. [**Apply**] 4] Demonstrate memory management and file system concepts using simulation or scripting. **[Apply]**
Course Content:	

Targeted Application:

Application area is traffic management system, banking system, health care and many more systems where in there are resources and entities that use and manage the resources.

Software Tools:

Oracle Virtual Box/VMWare Virtualization software [Virtual Machine Managers]. Used to install and work on multiple guest Operating Systems on top of a host OS.

Intel Processor identification utility: This software is used to explain about multi-core processors. It helps to identify the specifications of your Intel processor, like no of cores, Chipset information, technologies supported by the processor etc.

List of Laboratory Tasks:

Lab sheet -1

L1: Write a program to demonstrate the use of fork() and exec() system calls in process creation.

L2: A system has limited memory and high-priority real-time processes. Design a scheduling algorithm that ensures responsiveness while preventing starvation.

Lab sheet -2

L1: Implement First-Come-First-Serve (FCFS) process scheduling using C or Python.

L2: You are designing a server that handles thousands of client connections. Compare multithreading and multiprocessing for this task and implement a basic server model.

Lab sheet -3

L1: Implement Round Robin Scheduling with a fixed time quantum.

L2: In a banking system, concurrent access to accounts leads to data corruption. Design a synchronization solution to avoid race conditions.

Lab sheet -4

L1: Write a program to create threads using Pthreads or Python's threading module.

L2: You're tasked with building a file access tracker in an OS. Implement a system to log file access patterns and identify frequent accesses.

Lab sheet -5

L1: Demonstrate inter-process communication (IPC) using pipes.

L2: A simulation tool needs to emulate process suspension and resumption. Design and implement such a mechanism using signals or condition variables.

Lab sheet -6

L1: Simulate the Producer-Consumer problem using semaphores.

L2: You're developing a system where sensor devices (producers) generate temperature readings, and data processors (consumers) store and process these readings. To prevent race conditions and ensure buffer safety, implement a synchronization mechanism using semaphores.

Lab sheet -7

L1: Implement Dining Philosophers Problem using threads and synchronization.

L2: In a multi-threaded cafeteria simulation, five philosophers sit around a circular table, each alternating between thinking and eating. To eat, a philosopher must hold two forks (represented by shared resources). Your task is to avoid deadlock and ensure no philosopher starves using thread synchronization techniques.

Lab sheet -8

L1: Write a program to simulate First Fit, Best Fit, and Worst Fit memory allocation strategies.

L2: A system with limited memory blocks needs to allocate memory to processes arriving with various size requests. Your task is to implement three classic memory allocation strategies—First Fit, Best Fit, and Worst Fit—to allocate memory to each process efficiently. Simulate and compare how memory gets allocated in each strateg

Lab sheet -9

L1: Demonstrate paging using a simple page table simulation.

L2: A program has a logical address space divided into pages. The system's memory is divided into equal-sized frames. When a program executes, its pages are loaded into available frames in main memory. Simulate the address translation process using a page table and demonstrate how a logical address is converted to a physical address.

Lab sheet -10

L1: Write a program to simulate page replacement algorithms like FIFO and LRU.

L2: In a virtual memory system, a process accesses pages in a specific order. The memory can only hold a limited number of pages (frames). When a page is needed and the memory is full, a page replacement algorithm is used to decide which page to evict. Simulate and compare FIFO and LRU algorithms for a given page reference string.

Lab sheet -11

L1: Simulate file directory structure (single level/two level).

L2: A university campus computer lab has limited memory space available for each student login session. When students open files or run programs, memory pages are loaded into available memory frames. Due to the limited number of frames, some pages must be replaced when new ones are needed. The lab system uses page replacement algorithms to decide which pages to evict when memory is full..

Lab sheet -12

L1: Write a shell script to demonstrate file handling commands in Linux.

L2: Design a command-line mini shell that can run background and foreground processes and handle basic built-in commands like cd, pwd, exit.

Project work/Assignment

Demonstrate process concepts in LINUX OS.

Simulation of CPU scheduling algorithms.

Develop program to demonstrate use of Semaphores in threads.

Develop program to demonstrate use of deadlock avoidance algorithms.

Develop program to demonstrate use of page replacement algorithms.

Simulation of memory allocation strategies [first fit, best fit and worst fit].

Text Book

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

References

Silberschatz A, Galvin P B and Gagne G, "Operating System Concepts", 10th edition Wiley, 2018.

William Stallings, "Operating Systems",Ninth Edition, By Pearson Paperback ,1 March 2018.

Sundaram RMD, Shriram K V, Abhishek S N, B Chella Prabha, " Cracking the Operating System skills", Dreamtech, paperback, 2020

Remzi H. Arpaci-Dusseau Andrea C. Arpaci-dusseau , "Operating Systems: Three Easy Pieces, Amazon digital Services", September 2018.

E-resources/Weblinks

https://www.os-book.com/OS9/

https://pages.cs.wisc.edu/~remzi/OSTEP/

https://codex.cs.yale.edu/avi/os-book/OS10/index.html

Ittagalpura, Rajanukunte, Yelahanka, Bengaluru 560 119