PRESIDENCY UNIVERSITY

GAIN MORE KNOWLEDGE REACH GREATER HEIGHTS

PROGRAMME REGULATIONS & CURRICULUM

2024-28

PRESIDENCY SCHOOL OF COMPUTER SCIENCE & ENGINEERING

BACHELOR OF TECHNOLOGY (B.TECH.)

INFORMATION SCIENCE AND 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

Information Science and Engineering

B. Tech. [ISE]
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-24.5/SOCSE04/ISE/2024-28

Resolution No.05 of the 24th Meeting of the Academic Council held on 03rd August 2024, and ratified by the Board of Management in its 24th Meeting held on 05th August, 2024.

AUGUST-2024

Table of Contents

23.	Course Catalogue of all Courses Listed including the Courses Offered by other School / Department and Discipline / Program Electives	40

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 transforming Society.

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, 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. "PSOE" means the Presidency School of 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;

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)

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

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

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

6. Minimum and Maximum Duration

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

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

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

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

6.5 The enrolment of the student who fails to complete the mandatory requirements for the award of the concerned Degree (refer Section 19.**Error! Reference source not found.** of Academic Regulations) in the prescribed maximum duration (Clauses 18.1 and 18.2 of Academic Regulations), shall stand terminated and no Degree shall be awarded.

7 Programme Educational Objectives (PEO)

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

PEO 01: Demonstrate as a Computer Engineering Professional with innovative skills and moral and ethical values.

PEO 02: Engage in lifelong learning through research and professional development

PEO 03: Serve as a leader in the profession through consultancy, extension activities and/ or entrepreneurship

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

8.1 Programme Outcomes (PO)

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

PO1. Engineering Knowledge: Apply 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: Employability

Develop technical and managerial skills that enhance employability and prepare graduates for successful careers in the field of Information Science and Engineering.

PSO 02: Research

Gain a strong theoretical foundation in core courses, enabling the application of knowledge to solve real-world problems through research and innovation.

PSO 3: Ethics and Entrepreneurship

Demonstrate leadership, teamwork, and ethical responsibility while leveraging technology for entrepreneurial ventures and sustainable societal impact.

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.

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 (5^{th} and 6^{th} 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 1^{st} 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 3^{rd} Semester (commencement of the 2^{nd} Year) of the B.Tech. Program and culminating with the 8^{th} Semester (end of the 4^{th} 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 1^{st} year (1^{st} or 2^{nd} 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 3^{rd} Semester of the Program. i.e., the Program Structure and Curriculum from the 3^{rd} to 8^{th} 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 1^{st} 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. (Information Science and 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 (Information Science and 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 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 Clause 12.5 of academic regulations) shall be clearly defined in the Course Plan for every Course, and approved by the DAC.

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

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

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

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

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

12.5 Assessment Components and Weightage

S. No	Credit Structure [L-T-P-C]	Percentage/ Marks	CA		Mid-Term		End-term		Project	Total	Exam Conducted by
			Theory	Practical	Theory	Practical	Theory	Practical			
1	3-0-0-3	Percentage	25%	-	25%	-	50%	-	-	100 %	Mid-Term & End Term 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 Term by CoE * Except for full 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 Term 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 by IC at School level
		Marks	-	50	-	-	-	-	50	100	
6	0-0-2-1	Percentage	-	100%	-	-	-	-	-	100%	Only CA at School Level
		Marks	-	100	-	-	-	-	-	100	
7	3-0-2-4	Percentage	12.50%	12.50%	15%	10%	30%	20%	-	100%	Mid-Term & End Term by CoE
		Marks	25	25	30	20	60	40	-	200	
8	2-0-0-2	Percentage	25%	-	25%	-	50%	-	-	100%	Mid-Term & End Term by 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 **Error! Reference source not found.** of the Academic Regulations), the method of evaluation shall be based only on Continuous Assessments. The various components of Continuous Assessments, the distribution of weightage among such components, and the method of evaluation/assessment, shall be as decided and indicated in the Course Plan/PRC. The same shall be approved by the respective DAC.

12.6 Minimum Performance Criteria:

12.6.1 Theory only Course and Lab/Practice Embedded Theory Course

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

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

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

12.6.2 Lab/Practice only Course and Project Based Courses

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

12.6.3 A student who fails to meet the minimum performance criteria listed above in a Course shall be declared as "Fail" and given "F" Grade in the concerned Course. For theory Courses, the student shall have to re-appear in the "Make-Up Examinations" as scheduled by the University in any subsequent semester, or, re-appear in the End Term Examinations of the same Course when it is scheduled at the end of the following Semester or Summer Term, if offered. The marks obtained in the Continuous Assessments (other than the End Term Examination) shall be carried forward and be included in computing the final grade, if the student secures the minimum requirements (as per Sub-Clause 12.6.1 and 12.6.2) in the "Make-Up Examinations" of the concerned Course. Further, the student has an option to re-register for the Course and clear the same in the summer term/ subsequent semester if he/she wishes to do so, provided the Course is offered.

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

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

13.1 The transfer of credits shall be examined and recommended by the Equivalence Committee (as per 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 (as per Academic Regulations) and transfer equivalent credits to partially or fully complete the mandatory credit requirements of Discipline Elective Courses and/or the mandatory credit requirements of Open Elective Courses as prescribed in the concerned Curriculum Structure. However, it is the sole responsibility of the student to complete the mandatory credit requirements of the Discipline Elective Courses and the Open Elective Courses as prescribed by the Curriculum Structure of the concerned Program.

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

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

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

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

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

13.3.7 A student who has successfully completed the approved SWAYAM/NPTEL/ other approved MOOCs and wants to avail the provision of transfer of equivalent credits, must submit the original Certificate of Completion, or such similar authorized documents to the HOD concerned, with a written request for the transfer of the equivalent credits. On verification of the Certificates/Documents and approval by the HOD concerned, the Course(s) and equivalent Credits shall forwarded to the COE for processing of results of the concerned Academic Term.

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

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

13.3.9 The maximum permissible number of credits that a student may request for credit transfer from MOOCs shall not exceed 20% of the mandatory minimum credit requirements specified by the concerned Program Regulations and Curriculum 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.**Error! Reference source not found.**), 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. (Artificial Intelligence and Machine Learning) Program Structure (2024-2028) totalling 160 credits. Table 7 summarizes the type of baskets, number of courses under each basket and the associated credits that are mandatorily required for the completion of the Degree.

Table 3: B.Tech. (Information Science and 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

Table 3: B.Tech. (Information Science and Engineering) 2024-2028: Summary of Mandatory Courses and Minimum Credit Contribution from various Baskets		
Sl. No.	**Baskets**	**Credit Contribution**
5	Professional Elective Courses (PEC)	18
6	Open Elective Courses (OEC)	6
7	Project Work (PRW)	16
8	Mandatory Courses (MAC)	0
	Total Credits	**160 (Minimum)**

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. (Artificial Intelligence and Robotics) program of four years duration.

15. Minimum Total Credit Requirements of Award of Degree

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

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

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

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

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

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

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

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

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	1	0	2	2
4	Integral Transforms and Partial Differential Equations	3	0	0	3
5	Numerical Computing	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	Competitive Programming and Problem Solving	**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 Structures	**3**	**0**	**0**	3
2	Web Technologies	**2**	**0**	**0**	2
3	Data Communication and Computer Networks	**3**	**0**	**0**	3
4	Data Structures Lab	**0**	**0**	**4**	2
5	Web Technologies Lab	**0**	**0**	**2**	1
6	Data Communication and Computer Networks Lab	**0**	**0**	**2**	1
7	Analysis of Algorithms	3	1	0	4
8	Database Management Systems	**3**	**0**	**0**	3
9	Object Oriented Programming Using Java	**3**	**0**	**0**	3
10	Essentials of AI	**3**	**0**	**0**	3
11	Essentials of AI Lab	**0**	**0**	**4**	2
12	Database Management Systems Lab	**0**	**0**	**2**	1
13	Object Oriented Programming Using Java Lab	**0**	**0**	**4**	2
14	Analysis of Algorithms Lab	**0**	**0**	**2**	1
15	Theory of Computation	3	0	0	3
16	Software Design and Development	3	0	0	3
17	Cloud Computing	**2**	**0**	**0**	2
18	Machine Learning	**3**	**0**	**0**	3
19	Automation Design and Development	**3**	**0**	**0**	**3**
20	Automation Design and Development Lab	**0**	**0**	**4**	**2**
21	Information Retrieval	**3**	**0**	**0**	3
22	Machine Learning Lab	**0**	**0**	**4**	2
23	Cloud Computing Lab	**0**	**0**	**2**	1
24	Mobile Application Development	2	0	0	2
25	Software Testing and Quality Assurance	**3**	**0**	**0**	3
26	Operating Systems	3	0	0	3
27	Cryptography and Network Security	**3**	**0**	**0**	3
28	Software Testing and Quality Assurance Lab	**0**	**0**	**2**	1
29	Mobile Application Development Lab	0	0	4	2
30	Operating Systems Lab	0	0	2	1
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, 2021). 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) during the 5th / 6th / 7th Semester as applicable, subject to the following conditions:

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

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

18.3 Capstone Project

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

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

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

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

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

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

18.4 Research Project / Dissertation

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

18.4.1.1 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.						
Specialization Track 1: Emerging AI and Computational Intelligence						**12**
1	CAI3400	Image Processing and Analysis	2	0	2	3
2	CAI3401	Big Data Analytics for AI	2	0	2	3
3	CAI3402	Optimization Techniques for Machine Learning	2	0	2	3
4	CAI3403	Reinforcement Learning	2	0	2	3
5	CAI3404	AI in Cybersecurity	2	0	2	3
6	CAI3405	Explainable AI	2	0	2	3
7	CAI3406	Responsible AI	2	0	2	3
8	CAI3407	Agentic AI	2	0	2	3
9	CAI3408	Deep Neural Networks	2	0	2	3
10	CAI3409	Speech Recognition and Synthesis	2	0	2	3
11	CAI3410	AI Chatbots without Programming	2	0	2	3
12	CAI3411	Generative AI	2	0	2	3
13	CAI3412	Machine Learning for Finance	2	0	2	3
Specialization Track 2: AI driven Autonomous Systems						**12**
1	CAI3413	Industrial IoT	2	0	2	3
2	CAI3414	Smart Farming	2	0	2	3
3	CAI3415	AI for Autonomous Systems	2	0	2	3
4	CAI3416	Edge Computing	2	0	2	3
5	CAI3417	Cognitive Computing	2	0	2	3

6	CAI3418	Geospatial Data Analytics	2	0	2	3
7	CAI3419	AI for energy consumption optimization	2	0	2	3
Specialization Track 3: Healthcare Data Analytics						**12**
1	CAI3420	Bio Medical Informatics	2	0	2	3
2	CAI3421	Intelligent system for disease prediction and drug discovery	2	0	2	3
3	CAI3422	AI for Medical Imaging	2	0	2	3
4	CAI3423	Genomic Data Science	2	0	2	3
5	CAI3424	Clinical Data Science	2	0	2	3
6	CAI3425	AI in Epidemiology and Public Health Analytics	2	0	2	3
7	CAI3426	Time Series Analysis for Patient Monitoring	2	0	2	3
Specialization Track 4: Applied AI and Full Stack Development						**12**
1	CSE3425	Programming in C# and .NET	1	0	4	3
2	CSE3426	Front End Full Stack Development	2	0	2	3
3	CSE3427	Java Full Stack Development	2	0	2	3
4	CSE3428	.Net Full Stack Development	2	0	2	3
5	CAI3427	Language Models for Text Mining	2	0	2	3
6	CAI3428	Practical Deep Learning with TensorFlow	2	0	2	3
7	CAI3429	Deep Learning for Computer Vision	2	0	2	3
Specialization Track 4:Intelligent Systems and Automation						**12**
1	RAI3400	Advanced Automation Design and Development	2	0	2	3
2	RAI3401	Business Analysis with Automation Solutions	2	0	2	3
3	RAI3402	AI for IoT Applications	2	0	2	3
4	RAI3403	AI for Robotics	2	0	2	3
5	RAI3404	Robotic System Design	2	0	2	3
6	RAI3405	Robot Operating System	2	0	2	3
7	RAI3406	Robot Perception and Control	2	0	2	3
8	RAI3407	Autonomous Systems and Path Planning	2	0	2	3
9	RAI3408	Swarm Intelligence	2	0	2	3
10	RAI3409	Humanoid Robots	2	0	2	3
Specialization Track 5: Cybersecurity and Privacy						
1	CIT2400	Cyber-Physical Systems	3	0	0	3
2	CIT3403	Embedded Systems for IoT	3	0	0	3
3	CCS3412	Blockchain Security	3	0	0	3
4	CCS3414	Security in Internet of Things (IoT)	3	0	0	3
5	CIT3407	IoT Data Analytics and Machine Learning	3	0	0	3

6	CIT2504	AI and Deep Learning for IoT	3	0	0	3
Specialization Track 6: Networking and Cloud Computing						
1	CIT3406	Cloud Computing for IoT	3	0	0	3
2	CDV3402	Serverless Computing	3	0	0	3
3	CSE3418	Network Security and Firewall Management	2	0	2	3
4	CSE3420	Network Intrusion Detection and Prevention	3	0	0	3
5	CIT2501	Wireless Communication in IoT	3	0	0	3
6	CDV3406	Edge Computing & Hybrid Cloud	3	0	0	3
7	COM3404	Cloud Security and Governance	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 6											
Sl. No.	**Course Code**	**Course Name**	**L**	**T**	**P**	**C**	**Type of Skill/ Focus**	**Course Caters to**	**Prerequisites/ Core quisites**	**Anti requisites**	**Future Courses that need this as a Prerequisite**
Chemistry Basket											
1	CHE1003	Fundamentals of Sensors	3	0	0	3	S	ES	-	-	-
2	CHE1004	Smart materials for IOT	3	0	0	3	S	ES	-	-	-
3	CHE1005	Computational Chemistry	2	0	0	2	S	ES	-	-	-
4	CHE1006	Introduction to Nano technology	3	0	0	3	S	ES	-	-	-
5	CHE1007	Biodegradable electronics	2	0	0	2	S	ES	-	-	-
6	CHE1008	Energy and Sustainability	2	0	0	2	S	ES	-	-	-
7	CHE1009	3D printing with Polymers	2	0	0	2	S	ES	-	-	-
8	CHE1010	Bioinformatics and Healthcare IT	2	0	0	2	S	ES	-	-	-
9	CHE1011	Chemical and Petrochemical catalysts	3	0	0	3	S	ES	-	-	-
10	CHE1012	Introduction to Composite materials	2	0	0	2	S	ES	-	-	-
11	CHE1013	Chemistry for Engineers	3	0	0	3	S	ES	-	-	-
12	CHE1014	Surface and Coatings technology	3	0	0	3	S	ES	-	-	-
13	CHE1015	Waste to Fuels	2	0	0	2	S	ES	-	-	-
14	CHE1016	Forensic Science	3	0	0	3	S	ES	-	-	-
Civil Engineering Basket											
1	CIV1001	Disaster mitigation and management	3	0	0	3	S	-	-	-	-

2	CIV1002	Environment Science and Disaster Management	3	0	0	3	FC	-	-	-	-
3	CIV2001	Sustainability Concepts in Engineering	3	0	0	3	S	-	-	-	-
4	CIV2002	Occupational Health and Safety	3	0	0	3	S	-	-	-	-
5	CIV2003	Sustainable Materials and Green Buildings	3	0	0	3	EM	-	-	-	-
6	CIV2004	Integrated Project Management	3	0	0	3	EN	-	-	-	-
7	CIV2005	Environmental Impact Assessment	3	0	0	3	EN	-	-	-	-
8	CIV2006	Infrastructure Systems for Smart Cities	3	0	0	3	EN	-	-	-	-
9	CIV2044	Geospatial Applications for Engineers	2	0	2	3	EM	-	-	-	-
10	CIV2045	Environmental Meteorology	3	0	0	3	S	-	-	-	-
11	CIV3046	Project Problem Based Learning	3	0	0	3	S	-	-	-	-
12	CIV3059	Sustainability for Professional Practice	3	0	0	3	EN	-	-	-	-
Commerce Basket											
1	COM2001	Introduction to Human Resource Management	2	0	0	2	F	HP/GS	-	-	-
2	COM2002	Finance for Non Finance	2	0	0	2	S	-	-	-	-
3	COM2003	Contemporary Management	2	0	0	2	F	-	-	-	-
4	COM2004	Introduction to Banking	2	0	0	2	F	-	-	-	-
5	COM2005	Introduction to Insurance	2	0	0	2	F	-	-	-	-
6	COM2006	Fundamentals of Management	2	0	0	2	F	-	-	-	-
7	COM2007	Basics of Accounting	3	0	0	3	F	-	-	-	-
Computer Science Basket (not to be offered for PSCS students)											
1	CSE2002	Programming in Java	2	0	2	3	S/EM	-	-	-	-
2	CSE2003	Social Network Analytics	3	0	0	3	S	GS	-	-	-
3	CSE2004	Python Application Programming	2	0	2	3	S/ EM	-	-	-	-
4	CSE2005	Web design fundamentals	2	0	2	3	S/ EM/EN	-	-	-	-
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/G S	-	-

3	LAW2002	Introdcution to Law of Succession	2	0	0	0	2	F	HP/GS	-	-
4	LAW2003	Introduction to Company Law	2	0	0	0	2	F	HP	-	-
5	LAW2004	Introduction to Contracts	2	0	0	2	F	HP	-	-	-
6	LAW2005	Introduction to Copy Rights Law	2	0	0	2	F	HP	-	-	-
7	LAW2006	Introduction to Criminal Law	2	0	0	2	F	HP	-	-	-
8	LAW2007	Introduction to Insurance Law	2	0	0	2	F	HP	-	-	-
9	LAW2008	Introduction to Labour Law	2	0	0	2	F	HP	-	-	-
10	LAW2009	Introduction to Law of Marriages	2	0	0	2	F	HP/GS	-	-	-
11	LAW2010	Introduction to Patent Law	2	0	0	2	F	HP	-	-	-
12	LAW2011	Introduction to Personal Income Tax	2	0	0	2	F	HP	-	-	-
13	LAW2012	Introduction to Real Estate Law	2	0	0	2	F	HP	-	-	-
14	LAW2013	Introduction to Trademark Law	2	0	0	2	F	HP	-	-	-
15	LAW2014	Introduction to Competition Law	3	0	0	3	F	HP	-	-	-
16	LAW2015	Cyber Law	3	0	0	3	F	HP	-	-	-
17	LAW2016	Law on Sexual Harrassment	2	0	0	2	F	HP/GS	-	-	-
18	LAW2017	Media Laws and Ethics	2	0	0	2	F	HP/GS	-	-	-
Mathematics Basket											
1	MAT2008	Mathematical Reasoning	3	0	0	3	S	-	-	-	-
2	MAT2014	Advanced Business Mathematics	3	0	0	3	S	-	-	-	-
3	MAT2041	Functions of Complex Variables	3	0	0	3	S	-	-	-	-
4	MAT2042	Probability and Random Processes	3	0	0	3	S	-	-	-	-
5	MAT2043	Elements of Number Theory	3	0	0	3	S	-	-	-	-
6	MAT2044	Mathematical Modelling and Applications	3	0	0	3	S	-	-	-	-
Mechanical Basket											
1	MEC1001	Fundamentals of Automobile Engineering	3	0	0	3	F	-	-	-	-
2	MEC1002	Introduction to Matlab and Simulink	3	0	0	3	S/EM	-	-	-	-
3	MEC1003	Engineering Drawing	1	0	4	3	S	-	-	-	-
4	MEC2001	Renewable Energy Systems	3	0	0	3	F	ES	-	-	-
5	MEC2002	Operations Research & Management	3	0	0	3	F	-	-	-	-
6	MEC2003	Supply Chain Management	3	0	0	3	S/ EM/ EN	-	-	-	-
7	MEC2004	Six Sigma for Professionals	3	0	0	3	S/EM	-	-	MEC2008	-
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 for B.Tech. (Information Science and Engineering) with 12 weeks

21.1 NPTEL - Open Elective Courses for B.Tech. (Information Science and Engineering)

Table 5: SWAYAM-NPTEL Course Durations and Credit Equivalence		
Sl. No.	**Course Duration**	**Credit Equivalence for Transfer of Credits**
1	4 Weeks	1 Credit
2	8 Weeks	2 Credits
3	12 Weeks	3 Credits

Grading System for SWAYAM-NPTEL Courses		
Sl. No.	**Final Score on the SWAYAM-NPTEL Certificate**	**Grade Awarded**
1	90% and above	O
2	From 80% to 89%	A+
3	From 70% to 79%	A
4	From 60% to 69%	B+
5	From 50% to 59%	B
6	From 40% to 49%	C

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

Sl. No.	Course Code	Course Name	L	T	P	Credits	Contact Hours	Basket
Semester 1 - Physics Cycle						**19**	**26**	
1	MAT1001	Calculus and Linear Algebra	3	0	2	4	5	BSC
2	PHY1002	Optoelectronics and Device Physics	2	0	2	3	4	BSC

3	MEC1006	Engineering Graphics	2	0	0	2	2	ESC
4	ENG1002	Technical English	1	0	2	2	3	HS MC
5	PPS1001	Introduction to soft skills	0	0	2	1	2	HSM C
6	CSE1004	Problem Solving using C	1	0	4	3	5	ESC
7	ECE2007	Digital Design	2	0	2	3	4	ESC
8	DES1146	Introduction to Design Thinking	1	0	0	1	1	HSM C
Semester 2 - BES Cycle						**15**	**23**	
1	MAT1003	Applied Statistics	2	0	0	2	2	BSC
2	CHE1018	Environmental Science	1	0	2	0	3	MA C
3	CIV1008	Basic Engineering Sciences	2	0	0	2	2	ESC
4	CSE1006	Problem Solving using JAVA	1	0	4	3	5	ESC
5	ENG2001/FR LXXXX	Advanced English / Foreign Language courses	1	0	2	2	3	HS MC
6	PPS1012	Enhancing Personality Through Soft Skills	0	0	2	1	2	HS MC
7	EEE1007	Basics of Electrical and Electronics Engineering	3	0	2	4	5	ESC
8	LAW1007	Indian Constitution and Professional Ethics for Engineers	1	0	0	0	1	MA C
9	ECE2010	Innovative Projects Using Arduino	-	-	-	1	0	ESC
Semest er 3						**21**	**26**	
1	MAT2501	Integral Transforms and Partial Differential Equations	3	0	0	3	3	BSC
2	CSE1508	Data Structures	**3**	**0**	**0**	3	3	PCC
3	CSE1504	Web Technologies	**2**	**0**	**0**	2	2	PCC
4	CSE1506	Data Communication and Computer Networks	**3**	**0**	**0**	3	3	PCC
5	CSE1500	Computational Thinking using Python	2	0	2	3	4	ESC
6	MGTXXXX	Managerial Economics and Financial Analysis	**3**	**0**	**0**	3	3	HS MC
7	CSE1509	Data Structures Lab	**0**	**0**	**4**	2	4	PCC
8	CSE1505	Web Technologies Lab	**0**	**0**	**2**	1	2	PCC
9	CSE1507	Data Communication and Computer Networks Lab	**0**	**0**	**2**	1	2	PCC
Semest er 4						**26**	**32**	
1	MAT2602	Numerical Computations	3	0	0	3	3	BSC
2	MAT2605	Discrete Mathematics	4	0	0	4	4	BSC
3	CSE1512	Analysis of Algorithms	3	1	0	4	4	PCC
4	CSE1510	Database Management Systems	**3**	**0**	**0**	3	3	PCC
5	CSE1514	Object Oriented Programming Using Java	**3**	**0**	**0**	3	3	PCC
6	CSE1700	Essentials of AI	**3**	**0**	**0**	3	3	PCC
7	CSE1701	Essentials of AI Lab	**0**	**0**	**4**	2	4	PCC

8	CSE1511	Database Management Systems Lab	0	0	2	1	2	PCC
9	CSE1515	Object Oriented Programming Using Java Lab	0	0	4	2	4	PCC
10	CSE1513	Analysis of Algorithms Lab	0	0	2	1	2	PCC
Semester 5						27	30	
1	CSE2500	Theory of Computation	3	0	0	3	3	PCC
2	CSE2000	Software Design and Development	3	0	0	3	3	PCC
3	CSE2506	Cloud Computing	2	0	0	2	2	PCC
4	CAI2500	Machine Learning	3	0	0	3	3	PCC
5	RAI2000	Automation Design and Development	3	0	0	3	3	PCC
6	RAI2001	Automation Design and Development Lab	0	0	4	2	4	PCC
7	ISE2502	Information Retrieval	3	0	0	3	3	PCC
8	CSEXXXX	Professional Elective – I	3	0	0	3	3	PEC
9	CAI2501	Machine Learning Lab	0	0	4	2	4	PCC
10	CSE2507	Cloud Computing Lab	0	0	2	1	2	PCC
11	CSE7000	Internship	-	-	-	2	0	PRW
Semester 6						23	31	
1	CSE2508	Mobile Application Development	2	0	0	2	2	PCC
2	ISE2500	Software Testing and Quality Assurance	3	0	0	3	3	PCC
3	CSE2502	Operating Systems	3	0	0	3	3	PCC
4	CSE2503	Cryptography and Network Security	3	0	0	3	3	PCC
5	CSEXXXX	Professional Elective – II	3	0	0	3	3	PEC
6	XXXXXXX	Open Elective – I	3	0	0	3	3	OEC
7	ISE2501	Software Testing and Quality Assurance Lab	0	0	2	1	2	PCC
8	CSE2509	Mobile Application Development Lab	0	0	4	2	4	PCC
9	CSE2514	Operating Systems Lab	0	0	2	1	2	PCC
10	PPSXXXX	Industry Preparedness Program	2	0	0	0	2	MAC
11	CSE2510	Competitive Programming and Problem Solving	0	0	4	2	4	ESC
Semester 7						19	15	
1	CSEXXXX	Professional Elective – III	3	0	0	3	3	PEC
2	CSEXXXX	Professional Elective – IV	3	0	0	3	3	PEC
3	CSEXXXX	Professional Elective – V	3	0	0	3	3	PEC
4	CSEXXXX	Professional Elective – VI	3	0	0	3	3	PEC
5	XXXXXXX	Open Elective – II	3	0	0	3	3	OEC
6	CSE7100	Mini Project	-	-	-	4	0	PRW
Semester 8						10	0	

1	CSE7300	Capstone Project	-	-	-	10	0	PR W

22. Course Catalogue

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

The Course Catalogues for the Courses offered in each basket are attached below:

Course Code: MAT1001	Course Title: Calculus and Linear Algebra Type of Course: Basic Sciences Theory		L-T- P-C	3	0	2	4
Version No.	3.0						
Course Pre-requisites	Basic Concepts of Limits, Differentiation, Integration						
Anti-requisites	NIL						
Course Description	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.						
Course Objective	The objective of the course is Skill Development of student by using Problem Solving Techniques.						
Course Out Comes	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.						
Course Content:							
Module 1	Linear Algebra						10 Sessions

<table>
<tr><td colspan="5">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>Partial Derivatives</td><td></td><td></td><td>10 Sessions</td></tr>
<tr><td colspan="5">Review: Differential calculus with single variable.

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.</td></tr>
<tr><td>Module 3</td><td>Advanced Integral calculus</td><td></td><td></td><td>12 Sessions</td></tr>
<tr><td colspan="5">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.</td></tr>
<tr><td>Module 4</td><td>Ordinary Differential Equations</td><td>Assignment</td><td>Programming</td><td>12 Sessions</td></tr>
</table>

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 eax, sinax, cosax, eaxf(x), xnf(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: Engineering Sciences Theory	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:			

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.	7 Sessions

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.	8 Sessions
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.	8 Sessions
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.	7 Sessions
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: Engineering Science & Theory Only	L- T-P- C	2	0	0	2
Version No.	1.2					
Course Pre-requisites	NIL					
Anti-requisites	NIL					

<table>
<tr><td>Course Description</td><td colspan="4">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.</td></tr>
<tr><td>Course Objective</td><td colspan="4">The objective of the course is to familiarize the learners with the concepts of “Engineering Graphics” and attain SKILL DEVELOPMENT through Problem solving methodologies.</td></tr>
<tr><td>Course Outcomes</td><td colspan="4">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.</td></tr>
<tr><td colspan="5">Course Content:</td></tr>
<tr><td>Module 1</td><td>Introduction to Drawing</td><td>Assignment</td><td>Standard technical drawing</td><td>02 Sessions</td></tr>
<tr><td colspan="5">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]</td></tr>
<tr><td>Module 2</td><td>Orthographic projections of Points, Straight Lines and Plane Surfaces</td><td>Assignment</td><td>Projection methods Analysis</td><td>10 Sessions</td></tr>
<tr><td colspan="5">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]</td></tr>
<tr><td>Module 3</td><td>Orthographic Projections of Solids</td><td>Assignment</td><td>Multi-view drawing Analysis</td><td>10 Sessions</td></tr>
</table>

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: Humanities Science / Theory	L-T-P-C	1-0-2-2
Version No.	V. 3		
Course Pre-requisites	Intermediate Level English		

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	Presentation s	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: https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=JSTOR1_3307. 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 Last,Suzan, et. al. Technical Writing Essentials. University of Victoria, British Columbia, 2019 (E- Book) 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	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						

<table>
<tr><td>Module 2</td><td>EFFECTIVE COMMUNICATION</td><td>Individual Assessment</td><td>10 Hours</td></tr>
<tr><td colspan="4">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.</td></tr>
<tr><td>Module 3</td><td>HABIT FORMATION</td><td>Worksheets & Assignment</td><td>4 Hours</td></tr>
<tr><td colspan="4">Topics: Professional and personal ethics for success, Identity based habits, Domino effect, Habit Loop, Unlearning, standing up for what is right</td></tr>
<tr><td>Module 4</td><td>Goal setting & Time Management</td><td>Goal sheet</td><td>8 Hours</td></tr>
<tr><td colspan="4">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</td></tr>
<tr><td colspan="4">Targeted Application & Tools that can be used: LMS</td></tr>
<tr><td colspan="4">Project work/Assignment: Mention the Type of Project /Assignment proposed for this course</td></tr>
<tr><td colspan="4">Individual Assessment

LMS MCQ</td></tr>
<tr><td colspan="4">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.</td></tr>
</table>

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.

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

Course Code: 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						

<table>
<tr><td>Module 2</td><td>Design Thinking in Action</td><td>Visual journal, book of essays, context-specific assignment/project</td><td></td><td>Visual output generation, by visual journal and narrative development.</td><td>12 hours</td></tr>
<tr><td colspan="6">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.</td></tr>
<tr><td colspan="6">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</td></tr>
<tr><td colspan="6">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</td></tr>
<tr><td colspan="6">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</td></tr>
</table>

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	Probability			6 classes
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.

Course Code: CHE1018	Course Title: Environmental Science	L- T- P- C	1	0	2	0
		Contact	1	0	2	3

	Type of Course: School Core- Theory and Lab	hours				
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: 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=DO AB_1_06082022_18126

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DO AB_1_06082022_8761

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DO AJ_1_02082022_3333

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DO AB_1_06082022_3063

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DO AB_1_06082022_20719

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DO AB_1_06082022_16824

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DO AB_1_06082022_3954

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=DO AB_1_06082022_491

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=CU STOM_PACKAGE_16012023_WORLD_BUSINESS_COUNCIL_SUSTAINABLE_488

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=CU STOM_PACKAGE_16012023_WORLD_BUSINESS_COUNCIL_SUSTAINABLE_583

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=SP RINGER_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=TE XTBOOK_LIBRARY01_06082022_395&xIndex=4 https://www.ugc.gov.in/oldpdf/modelcurriculum/env.pdf

Course Code: CIV1008	Course Title: Basic Engineering Sciences Type of Course: Theory Only	L-T-P-C	2	0	0	2
Version No.	1.0					
Course Pre-requisites	NIL					
Anti-requisites	NIL					
Course Description	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.					
Course Objective	The objective of the course is skill development of student by using Participative Learning techniques.					
Course Outcomes	On successful completion of this course the students shall be able to: 1] 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.					

Course Content:				
Module 1	Introduction to various fields in Civil Engineering	Assignment	Case studies on different Civil Engineering Projects	6 Sessions
Topics: Introduction to Civil Engineering: Definition, scope and branches of Civil Engineering, Role of Civil Engineer, Overview of Infrastructure.				
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: Lab Integrated	L- T-P-C	1	0	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 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 in 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 Objects, 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 the 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 the assessment component mentioned in the 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								

<table>
<tr><td colspan="5">Creating the presentation
Giving the presentation</td></tr>
<tr><td>Module 3</td><td>Writing Reviews</td><td>Prezi</td><td>Review Writing</td><td>4 Classes</td></tr>
<tr><td colspan="5">Topics:
Review Writing
Short film reviews
Advanced English Grammar (Self Study)</td></tr>
<tr><td>Module 4</td><td>Starting your Career</td><td>Online Writing Lab</td><td>Writing Skills</td><td>4 Classes</td></tr>
<tr><td colspan="5">Topics:
Preparing a Resume
Writing Effective Application Letter
Creating a Professional Portfolio</td></tr>
<tr><td colspan="5">Course Content: Practical Sessions</td></tr>
<tr><td>Module 1</td><td colspan="3">Critical Reasoning and Writing</td><td>8 Classes</td></tr>
<tr><td colspan="5">Reading and Analyzing
Level 1 – Annotation
Level 2 - Assumptions
Writing Narrative Essays
Level 1 – Draft 1
Level 2 – Draft 2</td></tr>
<tr><td>Module 2</td><td colspan="3">Technical Presentation</td><td>10 Classes</td></tr>
<tr><td colspan="5">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</td></tr>
<tr><td>Module 3</td><td colspan="3">Writing Reviews</td><td>Classes</td></tr>
<tr><td colspan="5">Practice Worksheets
Level 1 – Eliminating the Passive Voice
Level 2 – Simple, compound and complex sentences</td></tr>
</table>

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		
Topics Relevant to "employability": Critical Reasoning, Presentation, Review Writing and Starting Career Topics Relevant to "Human Values and Professional Ethics": Critical reasoning		

Course Code: PPS1012	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-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 “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 laws 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. Summarizethe 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 Zin input impedance and Zout 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 Developmentthrough Experiential Learning techniques. This is attained through assessment component mentioned in course plan.

Course Code: CSE1006	Course Title: Problem Solving using JAVA Type of Course: Integrated	L- T-P- C	1	0	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.

Course Code: FRL1002	Course Title: Basic French Type of Course: Open Elective	L- T-P- C	2-0-0-2
Version No.	4.0		
Course Pre-requisites	Not Applicable		
Anti-requisites	Not Applicable		
Course Description	This Course is for beginners and gives an introduction of the French Language (basic grammar, conjugation, daily used vocabulary words, and basic conversations) and French culture. This Course is designed to build up all of the basic skills of French listening, reading, speaking, and writing introduced in the lessons. Besides, this Course offers an access to the French world, helping students to break cultural boundaries and raise cultural literacy.		
Course Objective	This course is designed to improve the learners Employability skills by using participative learning techniques to develop students’ language proficiency and cross-cultural competence by active and participatory teaching methods.		

<table>
<tr><td>Course Outcomes</td><td colspan="4">On successful completion of the course the students shall be able to:
Identify the basics of French Grammar, vocabulary and Conjugation
Apply the basics strategies of listening, reading, speaking and writing skills
Use of French on everyday topics such as greetings, personal information, time and schedule
Practice conversations in French language with peer speakers in different situations</td></tr>
<tr><td>Course Content:</td><td colspan="4">Learning of Basic French skills</td></tr>
<tr><td>Module 1</td><td>Greetings and Introducing yourself</td><td></td><td>[Remember]</td><td>6 Periods</td></tr>
<tr><td colspan="5">Chapter 1. Greetings
Objectives: Greetings, introducing yourself, how to welcome someone,
Grammar: Construction of a sentence, the days of the weeks and the months

Chapter 2. Introducing yourself
Objectives: Introduce oneself / ask for someone’s personal information.
Grammar: Mas or Fem noun, adjectives, present tense of the 1st group
Usage of audio visual files</td></tr>
<tr><td>Module 2</td><td>Expressing likes/dislikes and introducing someone</td><td></td><td>[Apply]</td><td>6 Periods</td></tr>
<tr><td colspan="5">Chapter 3. Expressing likes and dislikes
Objectives: How to expressing what you like and dislike.
Grammar: Negative form, singular and plural.
Culture: The polite way to address people in French
Assignment

Chapter 4. Introducing someone
Objectives: How to describe someone,
Grammar: Vocabulary of the family, Demonstrative adjectives,
Present tense of verbs of the 2nd and 3rd group</td></tr>
<tr><td>Module 3</td><td>Inviting someone and asking questions</td><td></td><td>[Apply]</td><td>9 Periods</td></tr>
</table>

Chapter 5. Inviting someone Objectives: How to invite someone, accept or refuse the invitation, Read the time, Grammar: Future tense, Interrogation. Culture: The art of accepting and declining an invitation politely in French Internal Chapter 6. Asking for information, Objectives: How to ask for information, giving information				
Module 4	Making a reservation and giving directions		[Apply]	9 Periods
Chapter 7: Making a Reservation Objectives: How to make a reservation, future tense Chapter 8 : Giving directions Objectives: How to ask for directions, Imperative tense Group discussions				
Targeted Application & Tools that can be used				
Project work /Assignments Assignment (Essay writing / presentation) Internal Group work / Group discussions				
Text Book L'Atelier 1 - - Méthode de Français--- Niveau A1 (Didier – 2019) Festival 1- - Méthode de Français--- Niveau A1 (CLE International – 2005)				
References Learning materials designed by the instructor				
Topics relevant to development of 'Employability Skills' through participative learning techniques: Foreign language proficiency and cross-cultural competence by active and participatory teaching methods.				

<table>
<tr><td>Course Code:
ECE2010</td><td>Course Title: Innovative Projects using Arduino</td><td>L- T-P- C</td><td>-</td><td>-</td><td>-</td><td>1</td></tr>
<tr><td>Version No.</td><td colspan="6">1.0</td></tr>
<tr><td>Course Pre-requisites</td><td colspan="6">NIL</td></tr>
<tr><td>Anti-requisites</td><td colspan="6">NIL</td></tr>
<tr><td>Course Description</td><td colspan="6">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.</td></tr>
<tr><td>Course Objective</td><td colspan="6">The objective of the course is Employability Skills of student by using PARTICIPATIVE LEARNING techniques.</td></tr>
<tr><td>Course Outcomes</td><td colspan="6">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.</td></tr>
<tr><td>Course Content:</td><td colspan="6"></td></tr>
<tr><td>Module 1</td><td>Basic concepts of Arduino</td><td>Hands-on</td><td colspan="3">Interfacing Task and Analysis</td><td>4 Sessions</td></tr>
<tr><td colspan="7">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.</td></tr>
<tr><td>Module 2</td><td>Sensory Devices</td><td>Hands-on</td><td colspan="3">Interfacing Task and Analysis</td><td>4 Sessions</td></tr>
<tr><td colspan="7">Arduino Sensors: Humidity Sensor, Temperature Sensor, Water Detector / Sensor, PIR Sensor, Ultrasonic Sensor, Connecting Switches and actuators, sensor interface with Arduino.</td></tr>
</table>

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:

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

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.

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.

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: MAT2501	Course Title: Integral Transforms and Partial Differential Equations Type of Course:1] School Core	L-T- P- C	3	0	0	3
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 Laplace transform, Fourier transform and Z-transform in addition to expressing functions in terms of Fourier series. The course covers applications of Laplace transform to LCR circuits and solutions of different equations using Z-transform. The course also deals with the analytical methods for solving partial differential equations and the classical applications of partial differential equations.					
Course Objective	The objective of the course is to familiarize the learners with the concepts of “Transform Techniques, Partial Differential Equations” and attain Skill 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 Classes)
Definition and Laplace transform of elementary functions. Properties of Laplace transform, and Laplace transform of periodic function, unit-step function and Impulse function – related problems. Inverse Laplace transform of standard functions - problems, initial and final value theorem. Convolution theorem, solution of linear and simultaneous differential equations and LCR Circuit.			
Module 2	Fourier Series	Assignment	(8 Classes)
Fourier Series: Periodic functions, Dirichlet’s condition. Fourier series of periodic functions period 2π and arbitrary period. Half range Fourier series. Practical harmonic analysis.			
Module 3	Fourier Transforms and Z - Transforms		(13 Classes)
Fourier Transforms: Definitions, infinite Fourier transforms, Fourier sine and cosine transforms, inverse Fourier transforms, Problems. Difference equations and Z-transforms: Z-transforms – Basic definitions, Standard Z-transforms, Linearity property, Damping rule, Shifting rule, Initial value theorem, Final value			

theorem, Inverse Z-transforms. Difference equations – Basic definitions, Application of Z-transforms to solve difference equations.			
Module 4	Partial Differential Equations	Assignment	(12 Classes)

Formation of PDE, Solution of non-homogeneous PDE by direct integration, Solution of homogeneous PDE involving derivative with respect to one independent variable only (Both types with given set of conditions) Method 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 these by the method of separation of variables. D'Alembert's solution of wave equation. Two-dimensional Laplace's equation – various possible solutions. Solution of all these equations with specified boundary conditions (Boundary 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 statistical tools to tackle engineering and real-life problems.

Assignment:

Newton-Raphson Methods, Gauss-Seidel Method, LU Decomposition, Trapezoidal Rule, Simpson's rule, Runge-Kutta 4th Order.

Text Book

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

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

E-resources/ Web links:

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_140238

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_233298

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_204892

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_223548

https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_30102024_134719 https://presiuniv.knimbus.com/user#/viewDetail?searchResultType=ECATALOGUE_BASED&unique_id=EBSCO95_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 differential equation with reference to specific engineering problems. The course is of both conceptual and analytical type in nature through Problem solving. This is attained through the assessment component mentioned in course handout.

Course Code: CSE1508	Course Title : Data Structures Type of Course: Theory	L-T - P -C	3	0	0	3	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 ,Red Black Tree – AVL Trees ,Binary Serach Tree , .Heaps , Expression 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 :

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: CSE1504	Course Title: Web Technology Type of Course: Program core Theory Only			L- T-P-C	2-0-0-2
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	8 Sessions	

			XHTML, simple applications	
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	7 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	7 Sessions
Topics: PHP: Introduction to server-side Development with PHP, Arrays, $GET and $ POST, $_Files Array, Reading/Writing Files, PHP Classes and Objects, Working with Databases, SQL, Database APIs, Managing a MySQL Database. Accessing MySQL in PHP.				
Targeted Application & Tools that can be used: Xampp web server to be used to demonstrate PHP.				
Project work/Assignment:				
Assignments are given after completion of each module which the student need to submit within the stipulated deadline.				
Textbook(s): 1] Robert. W. Sebesta, "Programming the World Wide Web", Pearson Education, 8th Edition, 2015.				

2] CSS Notes for Professionals, ebook available at https://books.goalkicker.com/CSSBook/ (Retrieved on Jan. 20, 2022) 3] Deitel, Deitel, Goldberg,"Internet & World Wide Web How to Program", Fifth Edition, Pearson Education, 2021.
References 1] Randy Connolly, Ricardo Hoar, "Fundamentals of Web Development", Pearson Education India, 1st. Edition.2016. 2] Jeffrey C. Jackson, "Web Technologies: A Computer Science Perspective", Pearson Education, 1st Edition,2016.
Topics related to development of "FOUNDATION": Web, WWW, Web browsers, Web servers, Internet. CSS, PHP. Designing for healthcare. for Skill Development through Experiential Learning techniques. This is attained through assessment component mentioned in course handout. E-References pu.informatics.global, https://sm-nitk.vlabs.ac.in/

Course Code: CSE1506	Course Title: Data Communications and Computer Networks Type of Course: Theory & Integrated Laboratory	L-T-P- C	3	0	0	3
Version No.	1.0					
Course Pre-requisites	ECE2007 - 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 Outcomes	On successful completion of this course, the students shall be able to: CO1: Illustrate The Basic Concepts Of Data Communication And Computer Networks. (Apply) CO2: Analyze the functionalities of the Data Link Layer. (Analyse) CO3: Apply the Knowledge of IP Addressing and Routing Mechanisms in Computer Networks.(Apply) CO4: Demonstrate the working principles of the Transport layer and Application Layer. (Apply)			
Course Content:				
Module 1	Introduction and Physical layer-CO1	Assignments	Problem Solving	17 Sessions (L9 + P8)
Introduction to Computer Networks and Data communications, Network Components – Topologies, Transmission Media –Reference Models - TCP/IP Suite, OSI Model . Physical Layer -Analog and Digital Signals – Digital and Analog Signals – Transmission - Multiplexing and Spread Spectrum.				
Module 2	Data Link Layer –CO2	Assignments	Problem Solving	20 Sessions (L12 + P08)
Data Link Layer - Error Detection and Correction– Parity, CRC, Hamming Distance Flow Control and Error Control, Stop and Wait, Multiple Access Protocols, CSMA/CD,CSMA/CA				
Module 3	Network Layer –CO3	Assignments	Problem Solving	21 Sessions (L13 + P8)
Network Layer Services - Network Layer Services, Switching Techniques, IP Addressing methods- IPv4 IPV6 – Subnetting. Routing, - Distance Vector Routing, Link State Routing, RIP, OSPF, BGPV4.				
Module 4	Transport and Application Layer - CO4	Assignments	Problem Solving	17 Sessions (L11 + P6)
Transport Layers - Connection management – Flow control-Sliding Window, Go-Back N ARQ, Selective Repeat ARQ, UDP, TCP, congestion control, Congestion avoidance The Application Layer: Domain Name System (DNS), Domain Name Space, FTP, Electronic Mail (SMTP), HTTP.				
Targeted Application & Tools that can be used: Cisco Packet Tracer, Wireshark, and NS2 Simulator.				
Project work/Assignment: Choose and analyse a network from any organization/Assignment proposed for this course in CO1-CO4				
Topics related to 1. Problem Solving: Choose and appropriate devices and implement various network concepts. 2. Employability: Simulation of any network using Cisco Packet Tracer/NS2.				

Textbook(s): T1. Behrouz A. Forouzan, “Data Communications and Networking with TCP/IP Protocol Suite”, 6th Edition, Tata McGraw-Hill, 2022. T2. Andrew S Tanenbaum, Nick Feamster & David J Wetherall, “Computer Networks” Sixth Edition, Pearson Publication, 2022.
References R1. “Computer Networking: A Top-Down Approach”, Eighth Edition, James F. Kurose, Keith W. Ross, Pearson publication, 2021. R2. William Stallings, Data and Computer Communication, 8th Edition, Pearson Education, 2007. R3. Behrouz A. Forouzan, “Data Communications and Networking 5E”, 5th Edition, Tata McGraw-Hill, 2012 E-Resources: https://www.geeksforgeeks.org/what-is-spread-spectrum/ https://www.geeksforgeeks.org/difference-between-fdma-tdma-and-cdma/ https://archive.nptel.ac.in/courses/106/105/106105183/ http://www.nptelvideos.com/course.php?id=393 https://www.digimat.in/keyword/106.htmlhttps://puniversity.informaticsglobal.com/login

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 Kenneth H. Rosen, "Discrete Mathematics and its Applications", McGraw-Hill,s 8th Edition,2019. Harary – Graph Theory, Addison-Wesley Publishing Company.
References: Arthur Gill, "Applied Algebra for Computer Science", Prentice Hall. K.D. Joshi, "Discrete Mathematics", Wiley Eastern Ltd. 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.

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

CSE1500</td><td>Course Title: Computational Thinking Using Python

Type of Course: Integrated</td><td>L- T- P- C</td><td>2</td><td>0</td><td>2</td><td>3</td></tr>
<tr><td>Version No.</td><td colspan="6">1.0</td></tr>
<tr><td>Course Pre-requisites</td><td colspan="6">NIL</td></tr>
<tr><td>Anti-requisites</td><td colspan="6">NIL</td></tr>
<tr><td>Course Description</td><td colspan="6">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</td></tr>
<tr><td>Course Objective</td><td colspan="6">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</td></tr>
<tr><td>Course

Outcomes</td><td colspan="6">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.</td></tr>
<tr><td>Course Content:</td><td colspan="6"></td></tr>
</table>

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 "Computational Thinking for the Modern Problem Solver" – David D. Riley & Kenny A. Hunt “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: 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 ,Red Black Tree – AVL Trees ,Binary Serach Tree , .Heaps , Expression 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 : 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: 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:						

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):

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”: Web, WWW, Web browsers, Web servers, Internet. CSS, PHP. Designing the website for healthcare.
The objective of the course is to familiarize the learners with the concepts of Web Technology and attain Skill Development through Experiential Learning techniques.

Course Code: CSE1507		Course Title: Data Communication and Computer Networks Lab Type of Course: Lab	L-T-P-C	0	0	2	1
Course Pre-requisites		NIL					
Anti-requisites		NIL					
Course Description		This lab course is to get practical knowledge of working principles of various communication protocols. Analyse structure and formats of TCP/IP layer protocols using network tools such as Wireshark and network simulators. Implementing various network algorithms such as error control, error detection, routing, and security related algorithms.					
Course Objective		The objective of the course is to familiarize the learners with the concepts of Computer Networks and attain Skill Development through Participative Learning techniques					
Course Out Comes		On successful completion of the course the students shall be able to: To understand the working principle of various communication protocols. To understand the network simulator environment and visualize a network topology and observe its performance. To analyze the traffic flow and the contents of protocol frames. To analyze data flow in wired and wireless environment					

Course Content						
Module 1	Introduction to Computer Networks					7 Sessions
	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					8 Sessions
	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					7 Sesions
	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	Assignment	Problem Solving			08 Classes
	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 Laryer for Skill development through Participative Learning techniques. This is attained through the assessment component mentioned in the course handout.

Course Code: MAT2602	Course Title: Numerical Computations		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 C.F.Gerald and P.O.Wheatley", Applied Numerical Analysis", McGraw-Hill, 1981. Cheneg and Kincaid, "Introduction to Numerical Computing", Tata McGraw-Hill, 1998.
References: SRK Iyengar & RK Jain, Numerical Methods, New Age Internationals. Erwin Kreyzig, Advanced Engineering Mathematics, John Wiley and sons, Inc.10th Edition 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.

Course Code: CSE1512	Course Title: Analysis of Algorithms Type of Course: THEORY Only	L- T-P-C	3	0	0	3
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: CSE2502	Course Title: Operating Systems Type of Course: Program Core and Theory Only	L-T- P- C	3	0	0	3
Version No.	1.0					
Course Pre-requisites	CSE2009- Computer Organization, Problem solving using C					

	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 course introduces the concepts of operating system operations, operating system structure and its design and implementation. It covers the classical operating systems internal algorithms such as process scheduling, synchronization, deadlocks detection and recovery and memory management. The course also enhances the problem solving, systems programming ability and case studies.			
Course Object	The objective of the course is to familiarize the learners with the concepts of Operating Systems and attain Employability through Problem Solving Methodologies.			
Course Out Comes	On successful completion of the course the students shall be able to: Describe the fundamental concepts of operating Systems and case studies. [Knowledge] Demonstrate various CPU scheduling algorithms. .[Application] Apply various tools to handle synchronization problems.[Application] 4] Demonstrate deadlock detection and recovery methods [Application] 5] Illustrate various memory management techniques.[Application]			
Course Content:				
Module 1	Introduction to Operating System	Assignment	Programming	9 Hours
Topics: Introduction to OS , Operating-System Operations, Operating System Services, , System Calls and its types, Operating System Structure, System Program and its types, Linkers and Loaders, Overview of OS design and implementation, Open-source operating system				
Module 2	Process Management	Assignment/Case Study	Programming/Simulation	11 Hours
Topics: Process Concept, Operations on Processes, Inter Process Communication, Communication in client-server systems (sockets, RPC, Pipes), Introduction to threads - Multithreading Models, Thread Libraries, Threading Issues, Process Scheduling– Basic concepts, Scheduling Criteria, Scheduling Algorithms: FCFS, SJF, SRTF, RR and Priority.				
Module 3	Process Synchronization and Deadlocks	Assignment	Programming	11 Hours

Topics: The Critical-Section Problem- Peterson's Solution, Synchronization hardware, Semaphores, Classic Problems of Synchronization with Semaphore Solution- Producer-Consumer Problem, Reader-Writer problems, Dining Philosopher's Problem, . Introduction to Deadlocks, Necessary conditions for deadlock, Resource allocation Graph, Methods for handling deadlock: Deadlock Prevention and Implementation, Deadlock Avoidance and Implementation, Deadlock detection & Recovery from Deadlock.				
Module 4	Memory Management	Assignment	Programming/Simulation	10 Hours

Topics:

Introduction to Memory Management, Basic hardware-Base and Limit Registers, Memory Management Unit(MMU), Dynamic loading and linking, Swapping, Contiguous and Non-Contiguous Memory Allocation, Segmentation, Paging - Structure of the Page Table – Virtual Memory and Demand Paging – Page Faults and Page Replacement Algorithms, Copy-on-write, Allocation of Frames, Thrashing

Introduction to File system management: File System Interface (access methods, directory structures), File system implementation.

Targeted Application:

Application area is traffic management system, banking system, health care and many more systems where in there are resources and entities that use and manage the resources.

Software Tools:

Oracle Virtual Box/VMWare Virtualization software [Virtual Machine Managers]. Used to install and work on multiple guest Operating Systems on top of a host OS.

Intel Processor identification utility: This software is used to explain about multi-core processors. It helps to identify the specifications of your Intel processor, like no of cores, Chipset information,

technologies supported by the processor etc.

Project work/Assignment

Demonstrate process concepts in LINUX OS.

Simulation of CPU scheduling algorithms.

Develop program to demonstrate use of Semaphores in threads.

Develop program to demonstrate use of deadlock avoidance algorithms.

Develop program to demonstrate use of page replacement algorithms.

Simulation of memory allocation strategies [first fit, best fit and worst fit].

Text Book 1. Silberschatz A, Galvin P B and Gagne G , “Silberschatz's Operating System Concepts”, Paperback, Global Edition Wiley, 2019 2.
References Silberschatz A, Galvin P B and Gagne G, “Operating System Concepts”, 10th edition Wiley, 2018. William Stallings, “Operating Systems”,Ninth Edition, By Pearson Paperback ,1 March 2018. Sundaram RMD, Shriram K V, Abhishek S N, B Chella Prabha, “ Cracking the Operating System skills”, Dreamtech, paperback, 2020 Remzi H. Arpaci-Dusseau Andrea C. Arpaci-dusseau , “Operating Systems: Three Easy Pieces, Amazon digital Services”, September 2018. E-resources/Weblinks https://www.os-book.com/OS9/ https://pages.cs.wisc.edu/~remzi/OSTEP/ https://codex.cs.yale.edu/avi/os-book/OS10/index.html

Course Code: 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

Course Code: CSE1510	Course Title: Database Management Systems Type of Course: 1) Program Core & Theory only	L-T-P-C	3	0	0	3
Version No.	1.0					
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: Describe the fundamental elements of relational database management systems. [Understand] Examine databases using SQL query processing and Optimization. [Apply] Design simple database systems applying the normalization constraints and demonstrate the database transaction processing, recovery, and security. [Apply]					

	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, Native XML databases (NXD), Document-oriented databases, Statistical databases.

Targeted Application & Tools that can be used:

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

Also demonstration of ORACLE DB on object-relational database creation and JDBC connection.

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

<table>
<tr><td>Course Code:
CSE2503</td><td colspan="3">Course Title: Cryptography and Network Security</td><td>L- T-P-C</td><td>3</td><td>0</td><td>0</td><td>3</td></tr>
<tr><td>Version No.</td><td></td><td colspan="7">1</td></tr>
<tr><td>Course Pre-requisites</td><td></td><td colspan="7">“Data Communications and Computer Networks”.</td></tr>
<tr><td>Anti-requisites</td><td></td><td colspan="7">NIL</td></tr>
<tr><td>Course Description</td><td colspan="8">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.</td></tr>
<tr><td>Course Objective</td><td colspan="8">The objective of the course is SKILL DEVELOPMENT of student by using PARTICIPATIVE LEARNING techniques.</td></tr>
<tr><td>Course Outcomes</td><td colspan="8">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)</td></tr>
<tr><td>Course Content:</td><td colspan="8"></td></tr>
<tr><td>Module 1</td><td colspan="2">Introduction to Cryptography</td><td>Assignment</td><td colspan="3">Identify the Concepts</td><td colspan="2">08 Sessions</td></tr>
<tr><td colspan="9">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.</td></tr>
<tr><td>Module 2</td><td colspan="2">Private Key Cryptography and Number Theory</td><td>Assignment</td><td colspan="3">Analysis of requirement of complexity in cryptography</td><td colspan="2">13 Sessions</td></tr>
<tr><td colspan="9">Topics:</td></tr>
</table>

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: CSE2000	Course Title: Software Design and Development Type of Course: School Core [Theory Only]	L-T-P- C	3-0-0-3
Version No.	1.0		
Course Pre-requisites	NIL		
Anti-requisites	NIL		
Course Description	The objective of this course is to provide the fundamentals concepts of Software Engineering process and principles. The course covers software requirement engineering processes, system analysis, design, implementation and testing aspects of software system development. The course covers software quality, configuration management and maintenance.		
Course Objectives	The objective of the course is to familiarize the learners with the concepts of Software Engineering and attain Skill Development through Participative Learning techniques.		
Course Out Comes	On successful completion of this course the students shall be able to: 1] Describe the Software Engineering principles, ethics and process models(Knowledge) 2] Identify the requirements, analysis and appropriate design models for a given application(Comprehension) 3] Understand the Agile Principles(Knowledge)		

	4] Apply an appropriate planning, scheduling, evaluation and maintenance principles involved in software(Application)			
Module 1	Introduction to Software Engineering and Process Models (Knowledge level)	Quiz		10 Hours
Introduction: Need for Software Engineering, Professional Software Development, Software Engineering Ethics, Software Engineering Practice-Essence of Practice, General Principles Software Development Life Cycle Models: Waterfall Model – Classical Waterfall Model, Iterative Waterfall Model, Evolutionary model-Spiral, Prototype.				
Module 2	Software Requirements, Analysis and Design (Comprehension level)	Assignment	Development of SRS documents for a given scenario	12 Hours
Requirements Engineering: Eliciting requirements, Functional and non- Functional requirements, Software Requirements Specification (SRS), Requirement Analysis and validation. Requirements modelling- Introduction to Use Cases, Activity diagram and Swim lane diagram. CASE support in Software Life Cycle, Characteristics of CASE Tools, Architecture of a CASE Environment. Design: Design concepts, Architectural design, Component based design, User interface design.				
Module 3	Agile Principles & Devops (Knowledge level)	Quiz		10 Hours
Agile: Scrum Roles and activities, Sprint Agile software development methods - Scaling, User Stories, Agile estimation techniques, Product backlogs, Stake holder roles, Dynamic System Development Method. Devops: Introduction, definition, history, tools.				
Module 4	Software Testing and Maintenance (Application Level)	Assignment	Apply the testing concepts using Programing	13 Hours
Software Testing-verification and validation, Test Strategies - White Box Testing, Black box Testing. Automation Tools for Testing. Software Quality Assurance-Elements of software quality assurance, SQA Tasks, Goals and Metrics, Software configuration management- SCM process, SCM Tools (GitHub).				

Maintenance- Characteristics of Software Maintenance, Software Reverse Engineering, Software Maintenance Process Models.
Targeted Application & Tools that can be used: Selenium, GitHub, CASE Tools
Text Book 1] R1. Roger S. Pressman, “Software Engineering – A Practitioner’s Approach”, VII Edition, McGraw-Hill, 2017. 2] B2. Bob Hughes, Mike Cotterell, Rajib Mall, “Software Project Management”, VI Edition, McGraw-Hill, 2018.
References Rajib Mall, “Fundamentals of Software Engineering”, VI Edition, PHI learning private limited, 2015. Ian Sommerville, “Software Engineering”, IX Edition, Pearson Education Asia, 2011. Agile Software Development Principles, Patterns and Practices.1st Edition, Wiley, 2002
Topics Relevant to “Skill Development: Balck box Testing, White box Testing, Automated Testing for Skill development through Participative Learning Techniques. This is attained through assessment mentioned in the course handout

Course Code: CSE1511	Course Title: Database Management Systems Lab 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.

Course Code: CSE2500	Course Title: Theory of Computation Type of Course: Theory Only	L- T-P- C	3	0	0	3
Version No.	2.0					
Course Pre-requisites	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 of Computation 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: Describe various components of Automata. (Knowledge) Illustrate Finite Automata for the given Language. (Application) Distinguish between Regular grammar and Context free grammar. (Comprehension) Construct Push down Automata. (Application) Construct Turing machine for a Language. (Application)			
Course Content:				
Module 1	Introduction to automata theory	Assignment	Problems on Strings and Language operations	06 Sessions
Topics: Introduction to Automata Theory, Applications of Automata Theory, Alphabets, Strings, Languages & operations on languages, Representation of automata, Language recognizers, Finite State Machines (FSM): Deterministic FSM, Regular languages, Designing FSM, Nondeterministic FSMs				
Module 2	Finite Automata	Assignment	Problems on DFA, NFA’s	13 Sessions
Topics: Basic concepts of Finite automata, DFA- definitions of DFA, Deterministic Accepters Transition Graphs and Languages and DFA's, Regular Languages, NFA- Definition of a Nondeterministic Accepter, Languages and NFA’s Why Non-determinism? Equivalence of Deterministic and Nondeterministic Finite Accepters, Reduction of the Number of States in Finite Automata.				
Module 3	Regular Expressions & Context Free Grammar	Assignment	Problems on RE, CFG, PT, PL and Ambiguity	12 Sessions

Topics: Formal Definition of a Regular Expression, Languages Associated with Regular Expressions, Languages, Regular Languages (RL) and Non-regular Languages: Closure properties of RLs, to show some languages are not RLs, Closure Properties of Regular Context Free Grammars-Examples of Context-Free Languages, Leftmost and Rightmost Derivations, Derivation Trees, Relation Between Sentential Forms and Derivation Trees, Ambiguity in Grammars and Languages: Ambiguous Grammars, Removing Ambiguity, Chomsky Normal Form, Gribiche Normal Form.				
Module 4	Push down Automata	Assignment	Problems on pushdown Automaton	08 Sessions
Topics: Definition of a Pushdown Automaton, Language Accepted by a Pushdown Automaton, Acceptance by Final State, Acceptance by Empty Stack, From Empty Stack to Final State, From Final State to Empty Stack Equivalence of PDA's and CFG's: From Grammars to Pushdown Automata.				
Module 5	Turing Machine	Assignment	Problems on Turning Machine	07 Sessions
Topics: Definition of a Turing Machine, Turing Machines as Language Accepters, Example Languages to construct Turing machine, Turing Machines as Transducers, Halting Programming Techniques for Turing Machines				
Targeted Application & Tools that can be used: Targeted Application: Text Processing Compilers Text Editors Robotics Applications Artificial Intelligence Tools: JFLAP (Java Formal Language and Automata Package) Software simulation tool. It's interactive educational software written in Java to experiment topics in automata theory. Turing machine Online simulators.				
Text Book Peter Linz, "An introduction to Formal Languages and Automata", Jones and Bartlett Publications 6th Ed, 2018.				
References Aho, Ullman and Hopcroft, "Theory of Computation", Pearson India 3rd Edition 2008. 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: CSE1513	Course Title: Analysis of Algorithms Lab Type of Course: Integrated	L- T-P-C	0	0	2	1
Version No.	1					
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 Experiential Learning Methodologies.					
Course Out Comes	On successful completion of the course the students shall be able to: 1. Compute efficiency of a given algorithm. [Applying] 2. Apply divide and conquer technique for searching and sorting Problems.[Applying] 3. Apply the Dynamic Programming technique for a given problem. [Applying] 4. Apply greedy technique for solving a Problem.[Applying] 5. Demonstrate Back tracking technique and limitations of Algorithms.[Applying]					
Course Content						
Module 1	Introduction	3 Sessions				
Measuring running time of an algorithm, Compare running time of algorithms, Implement sorting algorithms such as bubble sort, selection sort						

Module 2	Divide-and-conquer	3 Sessions
Compare searching algorithms: Linear Search, Binary Search; Compare Sorting algorithms: Insertion Sort, Merge Sort, QuickSort.		
Module 3	Dynamic programming	3 Sessions
Introduction and memorization: Factorial; Coin Change Problem ; Floyd-Warshall's Algorithm.		
Module 4	Greedy technique	3 Sessions
Fractional Knapsack Problem; Minimal Spanning Tree Algorithms-Prim's Algorithm, Kruskal's algorithm		
Module 5	Complexity Classes	3 Sessions
Branch and Bound: Knapsack problem; Backtracking, - N-Queens problem.		
	List of Laboratory Tasks: 1. Measuring running time of an algorithm Objective: To experimentally determine the running time of basic algorithms for input size n=10, 100, 1000, etc. by taking difference of starting time and ending time. 2. Compare running time of algorithms Objective: To execute two algorithms to solve the same problem, and to comparatively evaluate the better algorithm for large values of N. 3. Implement sorting algorithms such as bubble sort, selection sort Objective: To implement comparison based sorting strategies. 4. Compare searching algorithms Objective: To implement two searching strategies and compare their performance. 5. Compare Sorting algorithms Objective: To implement searching strategies that follow top down design approach(Insertion sort, merge sort). 6. Quick Sort Objective: To demonstrate Quick sort and its variants, and their impact on running time. 7. Dynamic Programming Objective: To demonstrate Dynamic Programming approach with the help of Factorial algorithm. 8. Coin Change Problem	

	Objective: To implement an efficient algorithm for the Coin Change problem. 9. Floyd-Warshall’s Algorithm Objective: To demonstrate how dynamic programming is used with the help of Floyd-Warshall’s algorithm. 10. Fractional Knapsack Problem Objective: To demonstrate how greedy method can be used to solve the Fractional Knapsack Problem. 11. Minimal Spanning Tree Algorithm Objective: To implement greedy strategy to solve the Minimal Spanning Tree problem using Prim’s Algorithm. 12. Kruskal’s Minimal Spanning Tree Algorithm Objective: To implement greedy strategies to solve the Minimal Spanning Tree problem using Kruskal’s Algorithm. 13. Knapsack Problem Objective: To implement Knapsack problem using branch and bound technique. 14. N-Queen’s Problem Objective: To demonstrate backtracking method with the help of N-Queen’s problem. 15. Case Study Objective: To demonstrate how various techniques can be used to solve the same problem with the help of Knapsack problem.
	Targeted Application & Tools that can be used PyTorch/Jupyter Notebook – For Python programming
	Text Book T1 Anany Levitin, “Introduction to the Design and Analysis of Algorithms”, 3rd edition, Pearson Education, 2018. T2 Thomas H.Cormen, Charles E.Leiserson, Ronald L. Rivest and Clifford Stein, “Introduction to Algorithms”, 4th edition, MIT Press, 2022.

	References R1. J. Kleinberg and E. Tardos, “Algorithm Design”, Addison-Wesley, 2005. R2. Tim Roughgarden, “Algorithms Illuminated” (books 1 through 3), “Operating Systems Design and Implementation”, Soundlikeyourself Publishing, 2017-2019. R3. AV Aho, J Hopcroft, JD Ullman, “The Design and Analysis of Algorithms”, Addison-Wesley, 1974. R4. Donald E. Knuth, “The Art of Computer Programming”, Volumes 1and 3 Pearson. Web Based Resources and E-books: W1. NPTEL: https://onlinecourses.nptel.ac.in/noc19_cs47/preview W2. Coursera: Analysis of Algorithms by Princeton University W3. Algorithms Specialization in Coursera by Stanford University(Group of 4 courses). W4. Algorithms Coding Contest Links maintained by Prof Gerth Stølting Brodal of Aarhus University
	Topics relevant to “EMPLOYABILITY SKILLS”: The lab experiments and assessments enable the student to acquire Skill Development through Experiential Learning techniques

Course Code: CSE1700	Course Title: Essentials of AI Type of Course: Theory	L- T-P-C	3	0	0	3
Version No.	2.0					
Course Pre-requisiData 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	Assignment	Implementation	10 Sessions
Topics: Python Basics: Variables, Data Types, Operators, and Control Flow Functions, Loops, and Conditionals statements, Data Structures: Lists, Tuples, Dictionaries, Sets ,Introduction to Libraries: NumPy and Pandas for data manipulation, Basic Input/Output and File Handling Introduction to Python for AI: Libraries and Frameworks Overview				
Module 2	Data Processing, Visualization	Assignment	Implementation	10 Sessions
Topics: cleaning and preprocessing with Pandas,Handling missing data, outliers, and duplicates, Data transformation (Normalization, Encoding), Introduction to Matplotlib and Seaborn for Data Visualization, Exploratory Data Analysis (EDA), Visualizing datasets to understand patterns and relationships.				
Module 3	Introduction to Machine Learning	Mini - Project	Implementation	10 Sessions
Topics: What is Machine Learning? Types of ML algorithms Supervised Learning: Regression, Classification, Unsupervised Learning: Clustering, Key ML Algorithms: Linear Regression, Decision Trees, K-Means ,Introduction to Scikit-learn library Model evaluation (Accuracy, Precision, Recall, Confusion Matrix)				
Module 4	Neural Networks and Deep Learning	Quiz	Implementation	10 Sessions
Topics: Introduction to Neural Networks and Deep Learning, Perceptron Model and Backpropagation Deep Neural Networks and Activation Functions, Introduction to TensorFlow and Keras, Building and Training Neural Networks for Image and Text Classification, Overview of Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs)				

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

Course Code: CSE3146	Course Title: Scalable Application Development using Java Type of Course: 1] Program Core		L- T-P- C	3	0	0	3
Version No.		1.0					
Course Pre-requisites		[1] Problem Solving Using Java (CSE1001) [2] Database Management System (CSE3156)					
Anti-requisites		NIL					
Course Description		The purpose of this course is to provide students with an in-depth understanding of advanced concepts and techniques in Java development. The course is both conceptual and analytical and is understood with JDK 21 software & Eclipse IDE. This course involves essential core java concepts like multithreading, file handling, annotations, generics, lambda expressions etc. This course also develops critical thinking skills by augmenting the student's ability to develop web application for various modern management systems like banking management system, student information management system, , Library Management System etc. with the necessary API for communication with database.					
Course Objectives		The objective of the course is to familiarize the learners with the concepts of Advanced Java Programming and attain Employability Skills through Experiential Learning techniques.					
Course Outcomes		On successful completion of this course the students shall be able to: CO1. Apply Concurrent Programming using Java Multi-Threading. [Apply] CO2. Practice the access mechanism to read/write file systems using Java I/O Operations. [Apply]					

		CO3. Interpret Communication/Connection mechanisms of Java with DBMS. [Apply] CO4. Implement Generics, Annotations & Lambda expressions using Java Programs. [Apply] CO5. Develop & Test Web application using Servlet & JSP. [Apply]		
Course Content:				
Module 1	Multi-Threading	Assignment	Multi-Threading	9 Sessions
	Multi-Threading in Java: Understanding Threads , Needs of Multi-Threaded Programming ,Thread Life-Cycle, Thread Priorities , Synchronizing Threads, Inter Communication of Threads , Dead lock, Concurrency Framework .			
Module 2	Input / Output & File Handling	Assignment	File Operations	9 Sessions
	Java I/O Operations : Input/ Output Operation in Java(java.io Package), Streams and the new I/O Capabilities ,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.			
Module 3	Collection and Database programming using JDBC	Assignment	Collection & Connection to DB	9 Sessions
	Collection – Enum, Collection Framework: Collections of Objects, Collection Types, Sets, Sequence, Map, Understanding Hashing, Use of Array List & Vector, Comparable and Comparator Interfaces. Database Programming using JDBC- Introduction to JDBC, JDBC Drivers & Architecture, CRUD operation Using JDBC, Connecting to non-conventional Databases.			
Module 4	Modern Java Features	Assignment	Advanced Java Features	9 Sessions
	Annotation : Basics, Type and Repeating Annotation - Generics : Generic Class, Bounded Types using wild card arguments, Generic Methods, Generic Interfaces- Lambda Expressions : Block Lambda, Generic functional Interfaces, Passing Lambda expressions as arguments, Lambda Expressions & Exceptions, Variable Capture, Method & constructor references, Reflection			
Module 5	Distributed Programming with Servlet	Assignment	Distributed Programming	9 Sessions
	Web Application Basics: Introduction to Servlet & JSP, Servlet life cycle, Developing and Deploying Servlets, create and compile servlet source code, Web Server, servlet API, Handling HTTP Requests and Responses: Handling HTTP GET requests and POST request, Using Cookies, Session Tracking, Simple Servlet Program to fetch database records			

	Text Books Herbert Schildt, “Java 2: The Complete Reference”, Tata McGraw-Hill Education, 12th Edition,2021.
	References Y.Daniel Liang, “Introduction to Java programming Comprehensive Version”, Pearson Education, 10th Edition, 2018. Cay S Horstmann and Gary Cornell, “CORE JAVA volume II-Advanced Features, 9th Edition,2016. Core and Advanced Java Black Book, Dream Tech Press. e-Resources https://docs.spring.io/spring-framework/reference/core.html https://docs.oracle.com/javaee/7/api/javax/servlet/Servlet.html https://docs.oracle.com/javaee/5/tutorial/doc/bnajo.html https://docs.oracle.com/javase/tutorial/jdbc/basics/index.html

Course Code: CSE3146	Course Title: Scalable Application Development using Java Lab Type of Course: 1] Laboratory Program Core		L- T-P-C	0	0	4	2
Version No.		1.0					
Course Pre-requisites		[1] Problem Solving Using Java (CSE1001) [2] Database Management System (CSE3156)					
Anti-requisites		NIL					
Course Description		The purpose of this course is to provide students with an in-depth hands-on on implementing the advanced concepts and techniques in Java development. This course is implemented with JDK 21 & Eclipse IDE. This course involves implementation of essential core java concepts like multithreading, file handling, annotations, generics, lambda expressions etc. In this course the students also implements development of web application for various modern management systems like banking management system, student information management system, , Library Management System etc. with the necessary API for communication with database.					

Course Objectives		The objective of the course is to implement the the concepts of Advanced Java Programming and attain Employability Skills through Experiential Learning techniques.
Course Outcomes		On successful completion of this course the students shall be able to: CO1. Implement Concurrent Programming using Java Multi-Threading. [Apply] CO2. Develop the access mechanism to read/write file systems using Java I/O Operations. [Apply] CO3. Develop the Communication/Connection mechanisms of Java with DBMS. [Apply] CO4. Implement Generics, Annotations & Lambda expressions using Java Programs. [Apply] CO5. Develop, Test and Deploy Web application using Servlet, JSP and Web Servers. [Apply]
Course Content:		

Module 1	Multi-Threading	Assignment	Multi-Threading	6 Sessions
Multi-Threading in Java: Implementation of Multi-Threaded Programming , Constructing Program to demo Thread Life-Cycle and Thread Priorities , Develop program on Synchronizing Threads and Inter Communication of Threads , Implement Dead lock and Concurrency Framework .				
Module 2	Input / Output & File Handling	Assignment	File Operations	6 Sessions
Java I/O Operations : Develop program on Input/ Output Operation in Java(java.io Package), Constrcut program with Streams and the new I/O Capabilities , Implement File Object, File I/O Basics to Read and Write to Files, Buffer and Buffer Management, Read/Write Operations with File Channel, Implement Serialization & De-Serialization. Construct program using Observer and Observable Interfaces.				
Module 3	Collection and Database programming using JDBC	Assignment	Collection & Connection to DB	6 Sessions
Collection – Implementing Collections Collection Framework: Collections of Objects, Collection Types, Sets, Sequence, Map, Hashing, Developing program by using Array List & Vector, Comparable and Comparator Interfaces. Database Programming using JDBC- Implementation of Connecting to MySQL database using JDBC and fetch records. Developing program for performing CRUD operation Using JDBC.				
Module 4	Modern Java Features	Assignment	Advanced Java Features	6 Sessions

Annotation : Implementation of Annotation, Generics, Lambda Expressions & Exceptions and Reflection				
Module 5	Distributed Programming with Servlet	Assignment	Distributed Programming	6 Sessions

	Web Application Basics: Development of Simple Servlet Program to perform CRUD operations, Implementation of Session Tracking, Development of Basic Web Application using Servlet & JSP List of Laboratory Tasks: Labsheet -1 Level 1 – Demonstration of Thread Class and Runnable Interface. Level 2 – Implementation of Producer-Consumer Problem. Level 2 – Implementation of inter-thread communication. Labsheet -2 Level 1 – Develop java programs to utilize Java.io.* package. Level 2 – Practice java programs to perform fiile operations with a case study. Level 2 – Implement Serialize / De-serialize the objects Labsheet – 3 Level 1 – Create classes using Collections to perform add, remove, sort operations. Level 1 – Implement JDBC Connection to Database to perform basic CRUD Operation Level 2 – Implement Student Information Management (Standalone). [Group wise] Labsheet – 4 Level 2 – Create a custom annotation @MinValue that enforces a minimum value on integer fields in a class. Use reflection to validate the annotated fields at runtime. Level 2 – Implement a generic class Pair<T, U> that stores two values of different types. Add methods to swap values and print the pair. Level 2 – Implement a list of employees (name, salary). Use lambda expressions to sort by salary and name. Level 2 – Create a User class with a method greetUser(). Use reflection to dynamically invoke the method at runtime.

	Labsheet – 5 Level 1 – Web page creation using HTML, Dynamic web page using java.servlet and JDBC Level 2 – Implementation of Student Information Management (WEB based). [Group wise] Labsheet – 6 Develop web application as mini-project for any management system using Spring Framework
	Targeted Application & Tools that can be used: Java 8 / MYSQL 8 / Eclipse /IntelliJ (IDE)
	Project work/Assignment: Mention the Type of Project /Assignment proposed for this course
	Build a Standalone database application using Java Swing as Front End. Indicative areas include; TimeTable Management, Student Expense Tracker, Important Mail Fetcher, etc. Build a real time database application using J2EE as Front End. Indicative areas include; health care, education, industry, Library, Transport and supply chain, etc.
	Text Books Herbert Schildt, “Java 2: The Complete Reference”, Tata McGraw-Hill Education, 12th Edition,2021.
	References Y.Daniel Liang, “Introduction to Java programming Comprehensive Version”, Pearson Education, 10th Edition, 2018. Cay S Horstmann and Gary Cornell, “CORE JAVA volume II-Advanced Features, 9th Edition,2016. Core and Advanced Java Black Book, Dream Tech Press. e-Resources https://docs.spring.io/spring-framework/reference/core.html https://docs.oracle.com/javaee/7/api/javax/servlet/Servlet.html https://docs.oracle.com/javaee/5/tutorial/doc/bnajo.html https://docs.oracle.com/javase/tutorial/jdbc/basics/index.html

Course Code: CSE1701	Course Title: Essentials of AI LAB Type of Course: Lab	L- T-P-C	0	0	4	2
Version No.	2.0					
Course Prerequisites	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.					
Anti-requisites	NIL					
Course Description	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, 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 Outcomes	On successful completion of the course the students shall be able to: Proficiency in Implementing AI Algorithms Using Python Ability to Build and Evaluate Machine Learning Models Hands-on Experience with Neural Networks and Deep Learning Practical Application of AI to Solve Real-World Problems					
Course Content:						

Module 1	Introduction to AI and Python for AI	Assignment	Implementation	8 Sessions l

Lab Assignment 1: Setting Up the Python Environment

Objective: Get familiar with setting up a Python environment for AI projects.

Tasks:

Install Python, Anaconda, and Jupyter Notebook.

Set up a virtual environment for AI development.

Install essential Python libraries: numpy, pandas, matplotlib, and scikit-learn.

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:

Write Python code to work with basic data types (integer, float, string, boolean).

Implement and manipulate Python lists, tuples, sets, and dictionaries.

Create basic control flow structures: if-else, for loops, while loops.

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:

Load a dataset (e.g., Titanic or Iris dataset) using pandas.

Clean the dataset by handling missing values, removing duplicates, and converting data types if needed.

Explore the dataset by visualizing it using matplotlib and seaborn.

Perform basic data preprocessing tasks such as feature scaling, encoding categorical variables, and splitting data into training and testing sets.

Module 2	Data Processing, Visualization	Assignment	Implementation	8 Sessions

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:

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().

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().

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.

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:

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.

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.

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).

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:

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.

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.

Distribution Visualizations:

Plot distributions of continuous variables using Seaborn's distplot() or kdeplot().

Create bar plots for categorical variables to understand their frequency distribution.

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:

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.

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.

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.

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:

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(). 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. 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. 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	Assignments	Implementation	8 Sessions
Lab Assignment 3: Implementing Linear Regression Tasks: Load a real-world dataset (e.g., Boston Housing Price dataset). Train a Linear Regression model using LinearRegression() from scikit-learn. Evaluate the model using Mean Squared Error (MSE) and R-squared Score. Visualize the regression line using Matplotlib. Lab Assignment 4: Logistic Regression for Classification Tasks: Load the Iris or Breast Cancer dataset. Preprocess the dataset (handle missing values, encode categorical variables, scale data). Train a Logistic Regression model using LogisticRegression(). Evaluate performance using Accuracy, Precision, Recall, F1-score. Plot the Confusion Matrix and ROC Curve. Lab Assignment 5: Implementing K-Nearest Neighbors (KNN) Tasks: Load the Iris dataset and split it into training and testing sets.				

Train a KNN classifier using KNeighborsClassifier(). Experiment with different values of K and evaluate performance. Visualize decision boundaries using a scatter plot. Lab Assignment 6: Decision Trees and Random Forests Tasks: Train a Decision Tree classifier on the Titanic dataset. Visualize the tree structure using plot_tree(). Train a Random Forest classifier and compare performance with the decision tree. Determine the feature importance using feature_importances_.				
Module 4	Neural Networks and Deep Learning	Quiz	Implementation	6 Sessions
Lab Assignment 7: Introduction to Perceptron and Activation Functions Tasks: Implement a single-layer perceptron using NumPy. Train the perceptron to classify AND, OR, XOR gates. Experiment with different activation functions (Sigmoid, ReLU, Tanh). Visualize decision boundaries. Lab Assignment 8: Building a Simple Neural Network with Keras Tasks: Load the MNIST dataset from keras.datasets. Preprocess the data (normalize pixel values, reshape input). Create a fully connected neural network using Sequential API. Train and evaluate the model using categorical cross-entropy loss and accuracy. Lab Assignment 9: Implementing CNN from Scratch Tasks: Load the CIFAR-10 dataset. Build a CNN with Conv2D, MaxPooling2D, Flatten, Dense, Dropout layers. Use Adam optimizer and categorical cross-entropy loss. Train and visualize loss/accuracy curves.				

Lab Assignment 10: Image Augmentation & Regularization

Tasks:

Apply data augmentation (rotation, zoom, flipping) using ImageDataGenerator.

Add dropout and batch normalization to prevent overfitting.

Compare model performance with and without augmentation.

Lab Assignment 11: Transfer Learning with Pre-trained Models

Tasks:

Use VGG16 or ResNet50 pre-trained on ImageNet.

Replace the output layer to classify new images.

Freeze earlier layers and fine-tune deeper layers.

Evaluate the model on a custom dataset (e.g., Cats vs. Dogs).

Lab Assignment 12: Implementing RNN for Text Classification

Tasks:

Load IMDB movie reviews dataset from keras.datasets.

Preprocess text (tokenization, padding sequences).

Build an RNN with Embedding, SimpleRNN, Dense layers.

Train and evaluate the model.

Lab Assignment 13: Building an LSTM for Time Series Prediction

Tasks:

Load a time series dataset (e.g., stock prices, temperature data).

Preprocess the data (normalize, reshape).

Build an LSTM-based model.

Predict future values and visualize trends.

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

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: Identify the engineering problems related to local, regional, national or global needs. (Understand) Apply appropriate techniques or modern tools for solving the intended problem. (Apply) Design the experiments as per the standards and specifications. (Analyze) Interpret the events and results for meaningful conclusions. (Evaluate)					

Course Code: CSE2506	Course Title: Cloud Computing Type of Course: Theory	L- T-P- C	2	0	0	2
Version No.	1.0					

Course Pre-requisites	Data Communication and Computer Networks (CSE2011)				
Anti-requisites	Nil				
Course Description	Cloud Computing provides a hands-on comprehensive study of Cloud concepts and capabilities across the various Cloud service models including Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). It dives into all of the details that a student needs to know in order to plan for developing applications on the cloud and what to look for when using applications or services hosted on a cloud.				
Course Objectives	The objective of the course is to familiarize the learners with the concepts of CLOUD COMPUTING and is designed to improve the learners' SKILL DEVELOPMENT through PARTICIPATIVE LEARNING TECHNIQUES.				
Course Out Comes	On successful completion of the course the students shall be able to: Describe the fundamental components and layers of Cloud Computing Architecture. [Remember] Identify appropriate Virtualization techniques to virtualize infrastructures [Understand] Summarize various Cloud mechanisms to optimize the QoS parameters [Understand] Apply cloud platforms to develop various applications [Apply]				
Course Content:					
Module 1	Introduction to Cloud services	Assignment	Theory		L: 10
Evolution of cloud computing, Computing Platforms and Technologies, Cloud Computing Architecture, IaaS, PaaS, SaaS, Types of Clouds, Cloud Computing Environments. [Understanding]					
Module 2	Virtualization Techniques	Assignment	Theory		L: 10
Basics of Virtualization - Types of Virtualizations, Taxonomy of Virtualization Techniques, Implementation Levels of Virtualization. [Understanding]					
Module 3	Cloud QoS and Management	Assignment	Theory		L: 10
Cloud Infrastructure Mechanisms- Logical Network Perimeter, Virtual Server, Cloud Storage Device, Cloud Usage Monitor, Ready-Made Environment, SLAs, Specialized Cloud Mechanisms- Automated Scaling Listener, Load Balancer, SLA Monitor, Pay-Per-Use Monitor, Audit Monitor, Cloud Security Mechanisms. [Understanding]					
Module 4	Cloud Application development in Cloud	Assignment	Theory		L: 10

Programming Models for Cloud Computing – MapReduce, CGL Mapreduce, Cloud Haskell, Development environments for service development (Demonstration using AWS Cloud/Saturn Cloud); Dockers and Containers. [Apply]
Targeted Application & Tools that can be used : Applications: Cloud Platform, Use of cloud technology in different applications like healthcare, agriculture etc. Tools: Google App Engine AWS, Saturn Cloud etc.
Project work/Assignment: Mention the Type of Project /Assignment proposed for this course
Students can design and implement dynamic resource allocation for virtual machine using cloud computing environment. Design and Implementation of a Scalable Cloud-Based Data Storage System Development of a Multi-Cloud Management Platform
Text Book Rajkumar Buyya, Christian Vecchiola, and Thamarai Selvi, “Mastering Cloud Computing”, McGraw Hill Education, 2013 edition. John Rittinghouse and James Ransome, “Cloud Computing, Implementation, Management and Security”, CRC Press, 2010 edition.
References Thomas Erl, Zaigham Mahmood, and Ricardo Puttini, “Cloud Computing Concepts, Technology & Architecture”, PHI publisher 2013 edition. K. Chandrasekaran, “Essentials of CLOUD COMPUTING”, CRC Press, 2015 edition. David E.Y. Sarna, “Implementing and Developing Cloud Applications”, CRC Press, 2018 edition. Manvi, Sunilkumar, and Gopal K. Shyam. “Cloud Computing: Concepts and Technologies”. CRC Press, 2021. Web Based Resources and E-books: W1. IEEE Transactions on Cloud Computing- https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=6245519 W2. International Journal of Cloud Computing- https://www.inderscience.com/jhome.php?jcode=ijcc W3. CloudSim Resources https://javadoc.io/doc/org.cloudsimplus/cloudsim-plus/latest/org/cloudbus/cloudsim/resources/class-use/Resource.html

W4. Journal of Network and Computer Networking- https://www.journals.elsevier.com/journal-of-network-and-computer- applications
Topics relevant to "Skill Development": AWS, Azure, APIs, Aneka Cloud Platform, Virtualization, Cloud Platforms in Industry, EC2, Installation of VM Workstation, Cloud Infrastructure and Challenges for Skill Development through Participative Learning techniques. This is attained through assessment component mentioned in course handout.

Course Code: CSE2507	Course Title: Cloud Computing Lab	L- T-P- C	0	0	2	1
Version No.	1.0					
Course Pre-requisites	Data Communication and Computer Networks (CSE2011)					
Anti-requisites	Nil					
Course Description	Cloud Computing provides a hands-on comprehensive study of Cloud concepts and capabilities across the various Cloud service models including Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). It dives into all of the details that a student needs to know in order to plan for developing applications on the cloud and what to look for when using applications or services hosted on a cloud.					
Course Objectives	The objective of the course is to familiarize the learners with the concepts of CLOUD COMPUTING and is designed to improve the learners' SKILL DEVELOPMENT through PARTICIPATIVE LEARNING TECHNIQUES.					
Course						

Content:	
Targeted Application & Tools that can be used : Applications: Cloud Platform, Use of cloud technology in different applications like healthcare, agriculture etc. Tools: Google App Engine AWS, Saturn Cloud etc.	
Project work/Assignment: Mention the Type of Project /Assignment proposed for this course	
Students can design and implement dynamic resource allocation for virtual machine using cloud computing environment. Design and Implementation of a Scalable Cloud-Based Data Storage System Development of a Multi-Cloud Management Platform	
List of Laboratory Tasks: Experiments: Create a simple cloud software application and provide it as a service using any Cloud Service Provider to demonstrate Software as a Service (SaaS). Create a Virtual Machine with 1 vCPU, 2GB RAM and 15GB storage disk using a Type 2 Virtualization Software Create a Virtual Hard Disk and allocate the storage using VM ware Workstation Create a Snapshot and Cloning of a VM and Test it by loading the Previous Version/Cloned VM Demonstrate Infrastructure as a Service (IaaS) by Creating a Virtual Machine using a Public Cloud Service Provider (Azure/GCP/AWS), configure with minimum CPU, RAM, and Storage and Launch the VM image. Create a Simple Web Application using Java or Python and host it in any Public Cloud Service Provider (Azure/GCP/AWS) to demonstrate Platform as a Service (PaaS) Create a Storage service using any Public Cloud Service Provider (Azure/GCP/AWS) and check the public accessibility of the stored file to demonstrate Storage as a Service Create a SQL storage service and perform a basic query using any Public Cloud Service Provider (Azure/GCP/AWS) to demonstrate Database as a Service (DaaS) Perform the basic configuration setup for Installing Hadoop 2.x like Creating the HDUSER and SSH localhost Install Hadoop 2.x and configure the Name Node and Data Node. Launch the Hadoop 2.x and perform MapReduce Program for a Word Count problem	

Text Book Rajkumar Buyya, Christian Vecchiola, and Thamarai Selvi, “Mastering Cloud Computing”, McGraw Hill Education, 2013 edition. John Rittinghouse and James Ransome, “Cloud Computing, Implementation, Management and Security”, CRC Press, 2010 edition.
References Thomas Erl, Zaigham Mahmood, and Ricardo Puttini, “Cloud Computing Concepts, Technology & Architecture”, PHI publisher 2013 edition. K. Chandrasekaran, “Essentials of CLOUD COMPUTING”, CRC Press, 2015 edition. David E.Y. Sarna, “Implementing and Developing Cloud Applications”, CRC Press, 2018 edition. Manvi, Sunilkumar, and Gopal K. Shyam. “Cloud Computing: Concepts and Technologies”. CRC Press, 2021.   Web Based Resources and E-books: W1. IEEE Transactions on Cloud Computing- https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=6245519 W2. International Journal of Cloud Computing- https://www.inderscience.com/jhome.php?jcode=ijcc W3. CloudSim Resources https://javadoc.io/doc/org.cloudsimplus/cloudsim-plus/latest/org/cloudbus/cloudsim/resources/class-use/Resource.html W4. Journal of Network and Computer Networking- https://www.journals.elsevier.com/journal-of-network-and-computer- applications
Topics relevant to “Skill Development”: AWS, Azure, APIs, Aneka Cloud Platform, Virtualization, Cloud Platforms in Industry, EC2, Installation of VM Workstation, Cloud Infrastructure and Challenges for Skill Development through Participative Learning techniques. This is attained through assessment component mentioned in course handout.

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:

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.

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.

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).

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.

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.

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.

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.

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.

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.

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.

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).

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.

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.

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.

Given a string, find the number of occurrences of a specific substring within the string. Focus: Basic string manipulation, string matching (brute-force approach).

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.

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.

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.

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).

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.

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).

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.

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.

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.

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.

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.

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.

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.

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.

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:

C or C++ Compiler (g++): The standard compiler for CP. Familiarize students with compilation flags (e.g., -O2 for optimization).

IDE (Integrated Development Environment): Code:: Blocks, Visual Studio, CLion, or similar IDEs. These provide debugging capabilities, code completion, and other helpful features.

Online Judges (CodeChef, Codeforces, LeetCode, HackerRank): Essential for practicing and submitting solutions.

Debugger (gdb): Crucial for understanding code execution and finding bugs. Origin, excel and Mat lab soft wares for programming and data analysis.

Number Theory Libraries: Some libraries provide pre-built functions for number theory operations (though often it's better to implement them yourself for learning). Wolfram Alpha: A useful tool for verifying number theory calculations and exploring concepts. String Libraries: Familiarize students with the string manipulation functions available in C++. Graph Visualization Tools: Tools like Graphviz can be helpful for visualizing graphs and understanding graph algorithms. DP Debugging Techniques: Practice debugging DP solutions, as they can be complex. Visualizing the DP table can be helpful.
Text Books: Guide to Competitive Programming: Learning and Improving Algorithms Through Contests" (3rd Edition), Antti Laaksonen, springer, 2024 "Data Structures and Algorithms in Java: A Project-Based Approach" – Dan S. Myers, Cambridge University Press Reference Books: Data Structures and Algorithmic Thinking with Python/C++/Java", Narasimha Karumanchi, 5th Edition, Career Monk, 2017. Introduction to Algorithms, Thomas H. Cormen (Author), Charles E. Leiserson (Author), Ronald L. Rivest , fourth edition April 2022 Web Resources https://nptel.ac.in/courses/106106231
Project work/Assignment: Mention the Type of Project /Assignment proposed for this course
Assessment Type Midterm exam Assignment (review of digital/ e-resource from PU link given in references section - mandatory to submit screen shot accessing digital resource.) Quiz End Term Exam Self-Learning

Course Code: 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: Identify the engineering problems related to local, regional, national or global needs. (Understand) Apply appropriate techniques or modern tools for solving the intended problem. (Apply) Design the experiments as per the standards and specifications. (Analyze) Interpret the events and results for meaningful conclusions. (Evaluate) Appraise project findings and communicate effectively through scholarly publications. (Create)

Course Code: CSE 7300	Course Title: Capstone Project Type of Course:	L- T-P- C	0	0	0	10
Version No.	1.0					

Course Pre-requisites	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: Identify problems based on societal /research needs. (Understand) Apply Knowledge and skill to solve societal problems in a group. (Apply) Develop interpersonal skills to work as member of a group or leader. (Apply) Analyze the inferences from available results through theoretical / Experimental / Simulations. (Analyze) Analyze the impact of solutions in societal and environmental context for sustainable development. (Analyze) Improve in written and oral communication. (Create) Demonstrate capabilities of self-learning in a group, which leads to lifelong learning. (Understand)

<table>
<tr><td>Course Code: CAI3418</td><td colspan="3">Course Title: Geospatial Data Analytics
Type of Course: Integrated</td><td>L- T-P- C</td><td>2</td><td>0</td><td>2</td><td>3</td></tr>
<tr><td>Version No.</td><td colspan="8">1.0</td></tr>
<tr><td>Course Pre-requisites</td><td colspan="8">Essentials of AI</td></tr>
<tr><td>Anti-requisites</td><td colspan="8">NIL</td></tr>
<tr><td>Course Description</td><td colspan="8">This course introduces the fundamentals of geospatial data and spatial analysis using modern analytical tools and technologies. It focuses on acquiring, visualizing, processing, and analyzing geographic data to extract meaningful patterns. Applications include urban planning, environmental monitoring, remote sensing, and location-based services.</td></tr>
<tr><td>Course Objective</td><td colspan="8">To understand the fundamentals of geospatial data and coordinate systems
To perform spatial data processing and analysis using GIS tools
To visualize geospatial data using mapping libraries and dashboards
To apply machine learning and statistical models to spatial datasets</td></tr>
<tr><td>Course Outcomes</td><td colspan="8">Upon successful completion of this course, students will be able to:
Work with different types of geospatial data and formats
Perform spatial analysis and geoprocessing
Build visualizations using GIS platforms and Python libraries
Apply geospatial techniques for problem-solving in real-world scenarios</td></tr>
<tr><td colspan="9">Course Content:</td></tr>
<tr><td>Module 1</td><td>Introduction to Geospatial Data and GIS</td><td>Assignment</td><td colspan="4"></td><td colspan="2">18[8L+10P]
Sessions</td></tr>
<tr><td colspan="9">Topics:
Basics of geospatial data: raster vs vector, Coordinate reference systems (CRS) and map projections, GIS architecture and components, Geospatial data sources: satellite, drone, GPS, open data.</td></tr>
<tr><td>Module 2</td><td>Geospatial Data Handling and Visualization</td><td>Assignment</td><td colspan="4"></td><td colspan="2">14[7L+7P]
Sessions</td></tr>
<tr><td colspan="9">Topics:
Shapefiles, GeoJSON, and TIFF formats, Loading and manipulating geospatial data with GeoPandas, Map visualizations with Folium, Kepler.gl, and Plotly, Spatial joins, clipping, filtering, and reprojection</td></tr>
</table>

Module 3	Spatial Analysis and Remote Sensing	Assignment		14[6L+8P] Sessions
Topics: Overlay operations, buffering, and geoprocessing, Raster analysis using rasterio, Remote sensing concepts and multispectral image analysis, NDVI, LST, and change detection from satellite images.				
Module 4	Applications and Machine Learning in Geospatial Analytics	Assignment		14[6L+8P] Sessions
Topics: Spatial clustering and hotspot analysis, Geospatial ML: land cover classification, object detection, Time series and spatiotemporal data analysis, Case studies in disaster management, urban planning, and climate monitoring.				

List of Lab Tasks

Lab 1: Introduction to QGIS and Map Projections
Objective: Understand GIS interface and coordinate systems
Tasks: Load vector data and explore CRS
Activity: Visualize administrative boundaries and reproject layers

---

Lab 2: Handle Vector Data using GeoPandas
Objective: Load and manipulate shapefiles in Python
Tasks: Read, filter, and plot shapefile data
Activity: Analyze India’s state-level boundaries using GeoPandas

---

Lab 3: Perform Spatial Joins and Buffering
Objective: Learn spatial joins and geoprocessing
Tasks: Combine population and boundary datasets
Activity: Create buffer zones around city centers

Lab 4: Visualize Maps using Folium and Kepler.gl
Objective: Create interactive maps
Tasks: Generate heatmaps and choropleth maps
Activity: Visualize crime or COVID-19 data on an interactive map

---

Lab 5: Work with Raster Data using Rasterio
Objective: Read and manipulate satellite imagery
Tasks: Load and clip raster images
Activity: Display elevation or vegetation maps

Lab 6: Calculate NDVI from Satellite Images
Objective: Perform vegetation index analysis
Tasks: Use Red and NIR bands to compute NDVI
Activity: Visualize vegetation cover changes

Lab 7: Geocoding and Reverse Geocoding
Objective: Convert addresses to coordinates
Tasks: Use Geopy and Google Maps API
Activity: Geocode location list and map them

Lab 8: Perform Land Use Classification using K-Means
Objective: Apply clustering to satellite images
Tasks: Use unsupervised learning on raster data
Activity: Classify urban vs green areas

Lab 9: Detect Change over Time in Remote Sensing Data
Objective: Analyze temporal changes
Tasks: Compare satellite images from different years
Activity: Detect urban expansion or deforestation

Lab 10: Build Dashboard using Plotly and Dash
Objective: Create an interactive geospatial dashboard
Tasks: Combine charts and maps
Activity: Display flood zones with demographic data

Lab 11: Perform Hotspot and Cluster Analysis
Objective: Identify spatial patterns
Tasks: Use DBSCAN or Getis-Ord Gi* statistic
Activity: Detect crime or disease hotspots

Lab 12: Integrate GPS Data for Route Mapping
Objective: Process and visualize GPS tracks
Tasks: Load GPX/CSV files and plot paths
Activity: Analyze cycling or delivery routes

Lab 13: Apply Spatial Regression Models
Objective: Model spatial relationships
Tasks: Fit spatial autoregression (SAR) or GWR
Activity: Predict housing prices based on location

Lab 14: Use Google Earth Engine for Remote Sensing Analysis
Objective: Access cloud-based satellite processing
Tasks: Load and analyze Sentinel/Landsat data
Activity: Monitor water bodies or land surface temperature

Lab 15: Capstone – Geospatial Data Analytics Project
Objective: Apply geospatial techniques to a real dataset
Tasks: Perform end-to-end analysis
Activity: Present findings via dashboard or report

REFERENCE MATERIALS

TEXTBOOKS

Paul Longley et al., Geographic Information Systems and Science, Wiley, 4th Edition, 2015

Bolstad, Paul, GIS Fundamentals: A First Text on Geographic Information Systems, Eider Press, 6th Edition, 2019

REFERENCE BOOKS

Michael Dorman, Spatial Data Analysis in Python, Manning Publications, 2023

Andrew Cutts, Geospatial Analysis: A Comprehensive Guide, Winchelsea Press

Bonny P. McClain, Mastering Geospatial Analysis with Python, Packt Publishing, 2022

JOURNALS / MAGAZINES

International Journal of Geographical Information Science

Remote Sensing of Environment

Journal of Spatial Information Science

GIScience & Remote Sensing

SWAYAM / NPTEL / MOOCs

NPTEL – Introduction to GIS (IIT Roorkee)
https://nptel.ac.in/courses/105107120

Coursera – Geospatial and Environmental Analysis (UC Davis)
https://www.coursera.org/learn/environmental-analysis

edX – Geospatial Data Science and Applications (Tsinghua University)
https://www.edx.org/course/geospatial-data-science

Google Earth Engine Tutorials
https://developers.google.com/earth-engine/tutorials

Course Code: CAI3419	Course Title: AI for energy consumption optimization Type of Course: Integrated		L- T-P- C	2	0	2	3
Version No.	1.0						
Course Pre-requisites	Essesntails Of AI						
Anti-requisites	NIL						
Course Description	This course provides an in-depth study of how artificial intelligence can be leveraged to optimize energy consumption across various domains such as smart buildings, industrial systems, smart grids, and renewable energy. Students will learn how to use machine learning and deep learning models to analyze consumption data, predict future usage, and automate control for improved efficiency and sustainability.						
Course Objective	This course aims to impart practical knowledge in applying AI methods for optimizing energy usage, with a focus on reducing energy waste, maximizing efficiency, and integrating sustainable technologies through experiential learning.						
Course Outcomes	Course Outcomes On successful completion of this course, students will be able to: Understand the fundamentals of energy consumption systems and optimization methods. (Understand) Apply AI models to forecast and control energy usage. (Apply) Analyze and derive insights from energy data for efficient decision-making. (Analyze) Design intelligent energy optimization systems using modern AI tools. (Create)						
Course Content:							
Module 1	Introduction to Energy Systems and Optimization	Assignment				18[8L+10P] Sessions	
Topics: Basics of energy generation, transmission, and consumption,Key challenges in energy optimization, Overview of smart energy systems and IoT integration,Fundamentals of optimization: cost, usage, scheduling							

Module 2	Machine Learning for Energy Forecasting	Assignment		14[7L+7P] Sessions
Topics: Energy datasets and data preprocessing, Regression techniques for load and usage prediction, Time-series modeling with ARIMA and LSTM, Evaluation metrics and model selection				
Module 3	Deep Learning and Reinforcement Learning for Energy Control	Assignment		14[6L+8P] Sessions
Topics: Neural networks for pattern recognition in energy data, Deep learning for anomaly detection, Reinforcement learning for energy-efficient control systems, AI in smart HVAC and lighting systems				
Module 4	Applications and Emerging Trends	Assignment		14[6L+8P] Sessions
Topics: AI in smart homes and smart cities,Integration with renewable energy sources, Case studies: AI in smart grids, buildings, and industry, Future trends: Edge AI, Federated learning in energy networks				
Project work/Assignment:				
Develop a forecasting model for household energy optimization,Implement an RL-based agent to control energy consumption in a simulated environment, Design a smart energy dashboard with real-time analytics				
List of Lab Tasks: Lab 1: Load and explore energy consumption datasets (e.g., UCI, OpenEI) Lab 2: Clean and preprocess data for modeling Lab 3: Build a linear regression model for consumption forecasting Lab 4: Apply decision tree and random forest for energy classification Lab 5: Implement an ARIMA model for time-series analysis Lab 6: Develop a deep learning (LSTM) model for energy prediction				

Lab 7: Perform anomaly detection on smart meter data Lab 8: Develop a clustering model to group usage patterns Lab 9: Design a dashboard for visualizing real-time and historical energy data Lab 10: Build a rule-based control system for smart appliances Lab 11: Implement Q-learning for controlling a simulated thermostat Lab 12: Reinforcement learning for lighting system optimization Lab 13: Forecast solar energy generation using weather and usage data Lab 14: Combine IoT data with AI models for intelligent decision-making Lab 15: Final capstone: Smart energy optimization prototype using AI
REFERENCE MATERIALS TEXTBOOKS Abhishek Kumar, Machine Learning and Data Science in the Energy Sector, Wiley, 2022. Peter Palensky et al., Energy Informatics: Fundamentals and Applications, Springer, 2021. REFERENCES Soteris Kalogirou, Artificial Intelligence in Energy and Renewable Energy Systems, Nova Science Publishers, 2013. Klaus-Dieter Thoben et al., AI Methods for Smart Energy Systems and Industry 4.0, Springer, 2020. Subramanian Vadari, Smart Grid Redefined: Transformation of the Electric Utility, CRC Press, 2020. JOURNALS / MAGAZINES IEEE Transactions on Smart Grid Energy and AI (Elsevier) Renewable & Sustainable Energy Reviews SWAYAM/NPTEL/MOOCs NPTEL – Smart Grid Technology by IIT Kharagpur Coursera – AI for Energy edX – Data Science and Machine Learning for Energy Systems

Course Code: CAI3420	Course Title : Bio Medical Informatics Type of Course :Integrated			L – P – –TC	2	0	2	3
Version No.	1.0							
Course Pre-requisites	Basic knowledge of Machine Learning / Data Science							
Anti-requisites	NIL							
Course Description	This course introduces students to Bio Medical Informatics. The focus is on understanding domain-specific data and AI methods applicable to healthcare. The course includes both theoretical and practical components that prepare students for real-world applications in the health sector.							
Course Objective	The objective of the course is SKILL DEVELOPMENT of students using EXPERIENTIAL LEARNING techniques.							
Course Out Comes	On successful completion of the course the students shall be able to: On successful completion of the course the students shall be able to: CO1: Describe the scope and importance of Bio Medical Informatics. [Understand] CO2: Identify and pre-process health-related data for analysis. [Apply] CO3: Build intelligent models to support medical diagnostics or analytics. [Apply] CO4: Evaluate AI models for performance and reliability in healthcare. [Analyze] CO5: Design ethical, efficient AI-based systems for healthcare applications. [Create]							
Course Content:								
Module 1	Introduction to Bio Medical Informatics	Assignment	Program activity	22 Hours				
This module introduces the fundamental concepts of Bio Medical Informatics, outlining its scope and significance in modern healthcare. Students will learn about the different types of biomedical data including structured, semi-structured, and unstructured formats. The role of information systems such as Electronic Health Records (EHRs) and medical standards like ICD, SNOMED CT, and HL7 will be discussed. The application of data science and AI in areas such as diagnostics, treatment planning, and clinical workflows will be explored through real-world use cases. This module sets the foundation for understanding how informatics bridges technology and medicine.								
Module 2	Data collection, preprocessing techniques and AI model design	Assignment	Program activity	22 Hours s				

In this module, the focus shifts to data acquisition and preprocessing techniques essential for biomedical data analysis. Students will explore various data collection methods used in healthcare, including clinical trials, IoT-based sensors, and surveys. Key topics include data cleaning, integration of heterogeneous sources, handling missing values, and feature engineering for biomedical signals and images. Privacy concerns and de-identification practices in patient data will also be addressed. Practical exposure to preprocessing tools and techniques will prepare students for effective data handling in real-world scenarios.				
Module 3	Model training, validation, interpretation, and performance metric	Assignment	Program activity	18 Hours
This module delves into the application of machine learning algorithms within the biomedical context. Students will implement supervised learning methods such as logistic regression, decision trees, and support vector machines, along with unsupervised techniques like clustering for patient stratification. Emphasis will be placed on performance evaluation using metrics tailored to healthcare (e.g., sensitivity, specificity, AUC). Techniques for improving model robustness, including cross-validation and model interpretability tools like SHAP and LIME, will be discussed. Real-life datasets will be used to build disease prediction models, offering insights into practical implementation challenges.				
Module 4	Deployment, ethics in AI for healthcare, and case studies	Assignment	Program activity	13 Hours
The final module focuses on deploying AI models in clinical environments and understanding the ethical, legal, and social implications of Bio Medical Informatics. Students will learn about lightweight deployment tools such as Flask and Streamlit for building user interfaces and dashboards. Case studies on clinical decision support systems (CDSS) will illustrate the integration of AI into healthcare systems. The module also explores recent trends such as personalized medicine, genomic data analysis, and the role of AI in epidemiology. Ethical dilemmas, bias mitigation, and fairness in AI applications are also critically examined to ensure responsible use of technology in healthcare.				
List of Lab Tasks： Lab Sheet 1: Introduction to healthcare datasets using Python; Data visualization using matplotlib and seaborn Lab Sheet 2: Data cleaning techniques; Feature selection methods for health data Lab Sheet 3: Logistic regression for disease prediction; Evaluate with confusion matrix Lab Sheet 4: Decision trees and random forests; Cross-validation comparisons Lab Sheet 5: Clustering with K-means; Hierarchical clustering				

Lab Sheet 6: Time series forecasting with ARIMA; Anomaly detection Lab Sheet 7: Genomic sequence preprocessing; Feature extraction Lab Sheet 8: Medical image preprocessing; CNN classification (e.g., chest X-rays) Lab Sheet 9: NLP on clinical notes; Named Entity Recognition Lab Sheet 10: SHAP/LIME interpretability; Deployment with Flask/Streamlit Lab Sheet 11: Dashboard design; Integration with cloud or mobile apps Lab Sheet 12: Bias detection and mitigation Lab Sheet 13: COVID-19 case study; Real-time data visualizatio
Targeted Application & Tools that can be used Python, scikit-learn, pandas, matplotlib, seaborn, Jupyter, TensorFlow/PyTorch, Streamlit
Project work／Assignment：
Assignment：Assignments include module-wise exercises and real-world project implementation.
Text Book T1: Adam Bohr & Kaveh Memarzadeh – Artificial Intelligence in Healthcare, Academic Press, 2020 T2: Kevin Franks – Machine Learning for Healthcare, Apress, 2022
References R1: Recent journal articles from IEEE, Nature, and PubMed R2: Online materials from NPTEL, CourseraWeb resources ：
Topics relevant to development of “Skill Development ：” Health informatics, AI modeling, data analytics Topics relevant to development of “Environment and sustainability: Public health data analysis, epidemiology

<table>
<tr><td>Course Code:
CAI3421</td><td colspan="3">Course Title : Intelligent System for Disease Prediction and Drug Discovery
Type of Course :Integrated</td><td>L - P - -TC</td><td>2</td><td>0</td><td>2</td><td>3</td></tr>
<tr><td>Version No.</td><td colspan="8">1.0</td></tr>
<tr><td>Course Pre-requisites</td><td colspan="8">Basic knowledge of Machine Learning / Data Science</td></tr>
<tr><td>Anti-requisites</td><td colspan="8">NIL</td></tr>
<tr><td>Course Description</td><td colspan="8">This course provides in-depth understanding and hands-on exposure to advanced techniques in Intelligent System for Disease Prediction and Drug Discovery. It aims to enhance technical skills for solving complex problems in healthcare using AI.</td></tr>
<tr><td>Course Objective</td><td colspan="8">The objective of the course is SKILL DEVELOPMENT of students using EXPERIENTIAL LEARNING techniques.</td></tr>
<tr><td>Course Out Comes</td><td colspan="8">On successful completion of the course the students shall be able to:
On successful completion of the course the students shall be able to:
CO1: Describe the scope and role of intelligent systems in disease prediction and drug discovery. [Understand]
CO2: Identify and prepare relevant clinical and molecular data for AI model development. [Apply]
CO3: Build machine learning and deep learning models for disease prediction and drug target identification. [Apply]
CO4: Evaluate the effectiveness of AI models using appropriate metrics in healthcare settings. [Analyze]
CO5: Design and propose intelligent, ethical systems for real-world clinical and pharmaceutical applications. [Create]</td></tr>
<tr><td>Course Content:</td><td colspan="8"></td></tr>
<tr><td>Module 1</td><td>Fundamentals of Intelligent Systems in Healthcare</td><td>Assignment</td><td colspan="5">Program activity</td><td>22 Hours</td></tr>
<tr><td colspan="9">This module introduces students to the fundamentals of intelligent systems and their significance in disease prediction and drug discovery. It covers knowledge-based systems, expert systems, and decision support systems with examples from clinical settings. Students learn how AI mimics human reasoning for healthcare applications and the use of knowledge engineering in rule-based diagnosis.</td></tr>
<tr><td>Module 2</td><td>Data-Driven Approaches in Disease Prediction</td><td>Assignment</td><td colspan="5">Program activity</td><td>22 Hours s</td></tr>
</table>

The focus here is on data-driven intelligent systems using machine learning and deep learning. It covers data acquisition from clinical databases, molecular and drug databases, and preprocessing techniques suitable for biological data. Techniques for classifying disease patterns and identifying drug targets using AI are discussed.				
Module 3	AI in Computational Drug Discovery	Assignment	Program activity	18 Hours
This module delves into computational drug discovery, covering ligand-based and structure-based approaches. Students explore QSAR modeling, molecular docking, and neural networks for compound activity prediction. Ethical concerns and challenges in in-silico trials are also introduced.				
Module 4	Integration and Applications in Clinical Workflows	Assignment	Program activity	13 Hours
The final module addresses the integration of intelligent systems in clinical workflows and pharmaceutical research. Topics include AI-based platforms for precision medicine, clinical validation, regulatory compliance, and case studies on successful AI-driven drug discovery pipelines.				

List of Lab Tasks：

Lab Sheet 1

Level 1: Introduction to healthcare and molecular datasets

Level 2: Data loading and exploration using Python (pandas, NumPy)

Lab Sheet 2

Level 1: Data cleaning and preprocessing (missing values, normalization)

Level 2: Feature selection for clinical and drug datasets

Lab Sheet 3

Level 1: Implementation of classification models (Logistic Regression, Decision Trees)

Level 2: Model evaluation using confusion matrix and ROC-AUC

Lab Sheet 4

Level 1: Application of deep learning models for disease prediction

Level 2: Hyperparameter tuning and performance comparison

Lab Sheet 5

Level 1: Introduction to molecular representations (SMILES, fingerprints)

Level 2: Compound similarity calculation and clustering

Lab Sheet 6

Level 1: QSAR modeling using regression techniques

Level 2: Interpretation of chemical descriptors and activity prediction

Lab Sheet 7

Level 1: Structure-based drug discovery: basics of molecular docking Level 2: Running docking simulations using open-source tools (e.g., AutoDock) Lab Sheet 8 Level 1: Development of a basic rule-based expert system for disease diagnosis Level 2: Knowledge base and inference engine simulation Lab Sheet 9 Level 1: Data visualization with seaborn and matplotlib Level 2: Creating dashboards to visualize model predictions Lab Sheet 10 Level 1: Ethical case study analysis in AI-driven healthcare Level 2: Design of fairness-aware AI models for drug discovery Lab Sheet 11–15 Capstone mini-project development based on real-world data Includes problem definition, data handling, model building, evaluation, and report writing
Targeted Application & Tools that can be used Python, scikit-learn, pandas, matplotlib, seaborn, Jupyter, TensorFlow/PyTorch, Streamlit
Project work／Assignment：
Assignment：Assignments include module-wise exercises and real-world project implementation.
Text Book T1: Adam Bohr & Kaveh Memarzadeh – Artificial Intelligence in Healthcare, Academic Press, 2020 T2: Kevin Franks – Machine Learning for Healthcare, Apress, 2022
References R1: Krittanawong, C., Johnson, K.W., Rosenson, R.S., et al. Deep learning for cardiovascular medicine: A practical primer. European Heart Journal, 2020. R2: Ekins, S., Puhl, A.C., Zorn, K.M., et al. Exploiting machine learning for end-to-end drug discovery and development. Nature Materials, 2019. R3: Topol, E. Deep Medicine: How Artificial Intelligence Can Make Healthcare Human Again. Basic Books, 2019. Web resources ： R4: NPTEL Course – AI for Drug Discovery and Healthcare, https://nptel.ac.in

R5: Coursera Specialization – AI in Healthcare Specialization, Stanford University, https://coursera.org
Topics relevant to development of “Skill Development :” Development of classification and regression models Topics relevant to development of “Environment and sustainability: Reduction of animal testing through AI-based drug screening and in-silico trials

Course Code: CAI3422	Course Title : AI for Medical Imaging Type of Course :Integrated	L – P– –T C	2	0	2	3
Version No.	1.0					
Course Pre-requisites	Basic knowledge of Machine Learning / Data Science					
Anti-requisites	NIL					
Course Description	This course focuses on applying artificial intelligence and deep learning to medical image analysis. Students will explore medical imaging modalities, preprocessing techniques, computer vision models, and deployment strategies. Emphasis is placed on real-world applications such as disease diagnosis, image segmentation, and anomaly detection.					
Course Objective	The objective of the course is SKILL DEVELOPMENT of students using EXPERIENTIAL LEARNING techniques..					
Course Out Comes	On successful completion of the course the students shall be able to: CO1: Describe different medical imaging modalities and AI applications. [Understand] CO2: Preprocess and annotate medical images for AI pipelines. [Apply] CO3: Build and train computer vision models for disease detection. [Apply] CO4: Evaluate performance of AI models using imaging metrics. [Analyze] CO5: Design and deploy ethical and explainable AI solutions in medical imaging. [Create]					
Course Content:						

Module 1	Medical Imaging Modalities and Preprocessing	Assignment	Program activity	22 Hours
This module covers the foundations of medical imaging modalities such as X-rays, CT, MRI, and ultrasound. Students are introduced to DICOM formats, imaging physics, and image acquisition principles. The importance of image preprocessing for AI applications is emphasized.				
Module 2	Deep Learning for Image Analysis	Assignment	Program activity	22 Hours s
Focus is placed on computer vision and deep learning techniques in imaging. Key concepts include convolutional neural networks (CNNs), image segmentation, classification, and detection. Hands-on exercises include building image classification models for disease identification.				
Module 3	Advanced Imaging Techniques and Evaluation	Assignment	Program activity	18 Hours
Advanced topics such as 3D imaging, multi-modal fusion, and transfer learning are covered. Students also explore annotation tools, dataset creation, and model performance metrics like Dice coefficient and IoU.				
Module 4	Deployment and Real-world Applications	Assignment	Program activity	13 Hours
This module discusses AI deployment in radiology workflows, regulatory and ethical considerations, and real-world implementation. Use cases like tumor detection, fracture analysis, and pneumonia prediction are analyzed for impact and accuracy.				
List of Lab Tasks： Lab Sheet 1 Level 1: Loading and visualizing medical images (DICOM, PNG, JPEG) using Python libraries (e.g., pydicom, OpenCV) Level 2: Image enhancement techniques such as histogram equalization, denoising, and contrast adjustment Lab Sheet 2 Level 1: Image annotation using tools like LabelImg or CVAT Level 2: ROI (Region of Interest) extraction and mask creation for segmentation tasks Lab Sheet 3 Level 1: Building a basic Convolutional Neural Network (CNN) for classifying binary medical images Level 2: Fine-tuning pretrained models (e.g., VGG16, ResNet) on a labeled medical image dataset Lab Sheet 4 Level 1: Semantic segmentation using U-Net architecture				

Level 2: Evaluation using IoU and Dice coefficient Lab Sheet 5 Level 1: Multi-class classification with medical datasets (e.g., chest X-ray with normal, pneumonia, COVID-19 labels) Level 2: Performance evaluation using confusion matrix, sensitivity, specificity, and ROC-AUC Lab Sheet 6 Level 1: Heatmap generation for explainability using Grad-CAM Level 2: Visual interpretation of model decisions in medical diagnosis Lab Sheet 7 Level 1: 3D image visualization using volumetric data (CT/MRI) with SimpleITK Level 2: Slice-wise analysis and conversion between formats (e.g., NIfTI to PNG) Lab Sheet 8 Level 1: Building a simple web interface using Streamlit for AI-based image diagnosis Level 2: Deployment and testing of the model in the interface Lab Sheet 9 Level 1: Comparative study of model performance with and without data augmentation Level 2: Real-world use case: early detection of breast cancer using mammography Lab Sheet 10 Capstone Project: End-to-end implementation of an AI model for a medical imaging problem (e.g., lung opacity classification, brain tumor segmentation, or fracture detection
Targeted Application & Tools that can be used Programming Language: Python Deep Learning Frameworks: TensorFlow, Keras, PyTorch Computer Vision Libraries: OpenCV, PIL (Python Imaging Library) Medical Image Processing: pydicom, nibabel, SimpleITK Annotation Tools: LabelImg, CVAT, VGG Image Annotator (VIA) Visualization: Matplotlib, Seaborn, Grad-CAM for explainability

Model Deployment: Streamlit, Flask Datasets: NIH Chest X-ray, COVID-19 Radiography Dataset, BraTS for brain tumor segmentation, LIDC-IDRI
Project work／Assignment：
Assignment：Assignments include module-wise exercises and real-world project implementation.
Text Book T1: Adam Bohr & Kaveh Memarzadeh, Artificial Intelligence in Healthcare, Academic Press, 2020. T2: S. Kevin Zhou, Hayit Greenspan, Dinggang Shen, Deep Learning for Medical Image Analysis, Academic Press, 2017.
References R1: M. A. Haidekker, Medical Imaging Technology, Springer, 2013. R2: Geert Litjens et al., A survey on deep learning in medical image analysis, Medical Image Analysis, Elsevier, 2017. R3: Online resources including Coursera’s AI for Medical Diagnosis, Stanford’s CS231n: Convolutional Neural Networks for Visual Recognition, and NPTEL’s Medical Image Computing Web resources ： https://www.coursera.org/learn/ai-for-medical-diagnosis – AI for Medical Diagnosis by DeepLearning.AI https://cs231n.stanford.edu/ – CS231n: Convolutional Neural Networks for Visual Recognition, Stanford University https://nptel.ac.in/courses/106/106/106106213/ – Medical Image Computing, NPTEL https://www.kaggle.com/datasets – Public datasets for medical image classification and segmentation https://grand-challenge.org/ – AI challenges and annotated datasets for medical imaging research
Topics relevant to development of “Skill Development ：” Image preprocessing and augmentation techniques Design and training of deep learning models for medical image classification and segmentation Evaluation of AI models using healthcare-specific metrics (e.g., sensitivity, specificity, IoU, Dice score) Interpretation of model predictions using explainability tools (e.g., Grad-CAM)

Building and deploying real-time diagnostic tools using Python, Streamlit, and cloud platforms Collaborative problem-solving through project-based learning with real medical datasets Topics relevant to development of "Environment and sustainability: Use of AI to reduce redundant imaging procedures, minimizing patient exposure to radiation and resource use Energy-efficient model architectures and deployment practices to lower computational footprint in healthcare AI Early detection and screening with AI to reduce the need for invasive follow-up procedures and hospital admissions Cost-effective diagnostic solutions that support equitable access to healthcare in under-resourced or rural areas Adoption of paperless workflows and digital tools to support green healthcare initiatives

Course Code: CAI3423	Course Title : Genomic Data Science Type of Course :Integrated	L – P– -T C	2	0	2	3
Version No.	1.0					
Course Pre-requisites	Basic knowledge of Machine Learning / Data Science					
Anti-requisites	NIL					
Course Description	This course provides an in-depth understanding of genomic data and the computational approaches used to analyze it. Students will explore genome structures, sequencing technologies, data preprocessing techniques, and the application of machine learning and statistical tools for interpreting genomic data. The course aims to equip students with the skills needed to extract meaningful insights from genomic datasets for biomedical and healthcare applications.					
Course Objective	The objective of the course is SKILL DEVELOPMENT of students using EXPERIENTIAL LEARNING techniques.					
Course Out Comes	On successful completion of the course the students shall be able to: CO1: Describe fundamental genomic concepts and technologies. [Understand] CO2: Preprocess and manage large-scale genomic datasets. [Apply] CO3: Apply bioinformatics and ML techniques to genomic data. [Apply]					

	CO4: Analyze and interpret patterns in gene expression and sequence data. [Analyze] CO5: Design solutions for personalized healthcare using genomic insights. [Create]			
Course Content：				
Module 1	Introduction to Genomics and Data Sources	Assignment	Program activity	22 Hours
An introduction to the field of genomics and its role in personalized medicine forms the basis of this module. Topics include genome structure, sequencing technologies, types of genomic data, and public genomic databases such as ENSEMBL, NCBI, and 1000 Genomes.				
Module 2	Genomic Data Preprocessing and Feature Engineering	Assignment	Program activity	22 Hours s
Students learn to handle large-scale genomic data, including sequence alignment (BLAST, BWA), variant calling (VCF), and gene expression analysis (microarray, RNA-seq). Focus is also given to quality control, normalization, and handling missing values.				
Module 3	Machine Learning in Genomic Analysis	Assignment	Program activity	18 Hours
Machine learning approaches are applied to tasks such as disease susceptibility prediction, biomarker discovery, and patient stratification. Techniques include classification, clustering, and feature selection using scikit-learn and bioinformatics tools like BioPython and Bioconductor.				
Module 4	Applications in Personalized Medicine	Assignment	Program activity	13 Hours
This module explores case studies in cancer genomics, pharmacogenomics, and population genomics. Ethical considerations, data privacy (e.g., HIPAA, GDPR), and the integration of genomic data into clinical workflows are discussed.				
List of Lab Tasks： Lab Sheet 1 Level 1: Retrieve DNA and protein sequences from NCBI and Ensembl databases Level 2: Visualize and annotate genomic regions using the UCSC Genome Browser Lab Sheet 2 Level 1: Perform sequence alignment using BLAST (Basic Local Alignment Search Tool)				

Level 2: Use BWA (Burrows-Wheeler Aligner) for short-read alignment; convert to SAM/BAM formats Lab Sheet 3 Level 1: Variant calling from aligned reads using bcftools or GATK Level 2: Annotate variants using tools like VEP (Variant Effect Predictor) or SnpEff Lab Sheet 4 Level 1: Preprocess microarray gene expression data (normalization, filtering) Level 2: Perform RNA-seq pipeline: FASTQ to aligned reads and read counts Lab Sheet 5 Level 1: Feature engineering: convert genomic data into machine learning-ready formats Level 2: Apply dimensionality reduction (PCA, t-SNE) on gene expression datasets Lab Sheet 6 Level 1: Train a simple classifier (e.g., SVM, Decision Tree) to predict disease from gene expression Level 2: Evaluate model performance using accuracy, precision, recall, and AUC Lab Sheet 7 Level 1: Cluster genomic samples using k-means and hierarchical clustering Level 2: Visualize clusters using heatmaps and dendrograms Lab Sheet 8–10 Capstone Project: Use genomic datasets to identify candidate biomarkers or predict disease risk; includes full workflow from data preprocessing to model evaluation and report presentation
Targeted Application & Tools that can be used Data Repositories: NCBI, Ensembl, 1000 Genomes Project, UCSC Genome Browser Sequence Analysis: BLAST, BWA, SAMtools, FASTQC Variant Calling & Annotation: bcftools, GATK, VEP, SnpEff Gene Expression Analysis: DESeq2, edgeR (via R/Bioconductor), limma Machine Learning Libraries: scikit-learn, TensorFlow, XGBoost Bioinformatics Libraries: BioPython, Bioconductor (R), pyVCF Visualization: matplotlib, seaborn, pheatmap, genome browsers Web Platforms: Galaxy, EMBL-EBI Tools Environments: Jupyter Notebook, RStudio, Google Colab
Project work／Assignment：

Assignment：Assignments include module-wise exercises and real-world project implementation.
Text Book T1: Michael C. Schatz et al., Genomic Data Science, Cold Spring Harbor Lab Press, 2022 T2: Jason H. Moore & Scott M. Williams, Bioinformatics for Geneticists, Wiley, 2020
References R1: R. Durbin et al., Biological Sequence Analysis, Cambridge University Press, 1998 R2: Online resources from Coursera (e.g., Genomic Data Science Specialization by Johns Hopkins), NPTEL, and EMBL-EBI Web resources ： https://www.ncbi.nlm.nih.gov/ – NCBI Genomics Portal https://www.ensembl.org/ – Ensembl Genome Browser https://galaxyproject.org/ – Web-based bioinformatics analysis platform https://bioconductor.org/ – Open software for genomic data analysis
Topics relevant to development of “Skill Development ：” Genome browsing and data retrieval Variant analysis and functional annotation Machine learning applications in genomics Development of predictive models using omics data Project-based learning with publicly available datasets Topics relevant to development of “Environment and sustainability: Use of genomics to develop sustainable agriculture and precision nutrition Minimizing clinical trial waste through AI-based patient stratification Genetic screening for early disease detection to reduce long-term healthcare burden Efficient use of cloud computing for large-scale genomic data analysis

Course Code： CAI3424	Course Title ： Clinical Data Science Type of Course ：Integrated	L – P – –TC	2	0	2	3
Version No.	1.0					

<table>
<tr><td>Course Pre-requisites</td><td colspan="4">Basic knowledge of Machine Learning / Data Science</td></tr>
<tr><td>Anti-requisites</td><td colspan="4">NIL</td></tr>
<tr><td>Course Description</td><td colspan="4">This course introduces students to clinical data and the computational methods used for its analysis in healthcare. The curriculum covers electronic health records (EHR), clinical coding systems, data preprocessing, predictive modeling, and visualization. Students gain skills in transforming raw clinical data into meaningful insights for improving patient outcomes and supporting evidence-based decision-making.</td></tr>
<tr><td>Course Objective</td><td colspan="4">The objective of the course is SKILL DEVELOPMENT of students using EXPERIENTIAL LEARNING techniques.</td></tr>
<tr><td>Course Out Comes</td><td colspan="4">On successful completion of the course the students shall be able to:

CO1: Describe clinical data sources, structures, and standards. [Understand]

CO2: Preprocess and integrate structured and unstructured clinical data. [Apply]

CO3: Build analytical models for clinical decision support. [Apply]

CO4: Evaluate model accuracy and interpret outputs using relevant metrics. [Analyze]

CO5: Design data-driven solutions for real-world healthcare scenarios. [Create]</td></tr>
<tr><td>Course Content:</td><td colspan="4"></td></tr>
<tr><td>Module 1</td><td>Introduction to Clinical Data and Coding Systems</td><td>Assignment</td><td>Program activity</td><td>22 Hours</td></tr>
<tr><td colspan="5">This module introduces electronic health records (EHRs), their components, and commonly used clinical coding systems like ICD, CPT, and LOINC. Students understand how structured and unstructured clinical data are generated and stored in hospitals and research environments.</td></tr>
<tr><td>Module 2</td><td>Clinical Data Processing and Integration</td><td>Assignment</td><td>Program activity</td><td>22 Hours s</td></tr>
<tr><td colspan="5">Focuses on data cleaning, missing value handling, temporal sequencing, and merging data from disparate hospital systems. Students explore preprocessing tools such as pandas and SQL, with real-world clinical datasets for practice.</td></tr>
<tr><td>Module 3</td><td>Predictive Analytics and Risk Modeling</td><td>Assignment</td><td>Program activity</td><td>18 Hours</td></tr>
</table>

Students learn to develop and evaluate machine learning models for predicting hospital readmissions, disease progression, and patient risk scores. Survival analysis and calibration techniques are introduced.				
Module 4	Visualization and Deployment in Clinical Workflows	Assignment	Program activity	13 Hours
Covers best practices for visualizing clinical KPIs, creating dashboards for hospital use, and regulatory requirements like HIPAA and HL7 compliance. Students build real-time reporting tools for clinical insights.				
List of Lab Tasks： Lab Sheet 1 Level 1: Explore the structure of synthetic electronic health records (EHRs) Level 2: Extract and visualize demographic information using SQL or pandas Lab Sheet 2 Level 1: Preprocess structured clinical data: handle missing values, outliers Level 2: Transform unstructured clinical notes using basic NLP (e.g., tokenization, stemming) Lab Sheet 3 Level 1: Join and merge multi-source clinical datasets (e.g., labs + diagnosis + medication) Level 2: Time-based filtering and patient cohort generation using temporal queries Lab Sheet 4 Level 1: Build a logistic regression model to predict hospital readmission Level 2: Evaluate model performance using confusion matrix and ROC curve Lab Sheet 5 Level 1: Perform survival analysis using Kaplan-Meier estimator Level 2: Cox proportional hazards model to analyze patient survival risk Lab Sheet 6 Level 1: Create a visual dashboard of patient metrics using tools like matplotlib or seaborn Level 2: Build a real-time monitoring dashboard with Streamlit Lab Sheet 7–10 Capstone Project: End-to-end clinical analytics project: data preprocessing, modeling, evaluation, and reporting using real-world or simulated data				
Targeted Application & Tools that can be used				

Data Repositories & Formats: MIMIC-III, eICU, FHIR, CSV, HL7 Programming & Data Handling: Python, SQL, pandas, NumPy Data Visualization: matplotlib, seaborn, Plotly, Streamlit Machine Learning Frameworks: scikit-learn, XGBoost Survival Analysis: Lifelines (Python), R survival package NLP for Clinical Text: spaCy, NLTK, SciSpacy Dashboards & Reporting: Power BI, Streamlit, Tableau (optional) Environment: Jupyter Notebook, Google Colab, Anaconda
Project work／Assignment：
Assignment：Assignments include module-wise exercises and real-world project implementation.
Text Book T1: Mark L. Braunstein, Practitioner's Guide to Health Informatics, Springer, 2015. T2: Pradeep Menon, Applied Clinical Informatics: A Practical Guide for Healthcare Professionals, CRC Press, 2021.
References R1: J. D. Dalianis, Clinical Text Mining: Secondary Use of Electronic Patient Records, Springer, 2018. R2: Steinar Carlsen et al., Health Informatics: An Interprofessional Approach, Elsevier, 2020. R3: Online resources including MIMIC-III tutorials (MIT-LCP), NPTEL Health Analytics courses, and Coursera’s Data Science in Stratified Healthcare and Precision Medicine. Web resources ： https://physionet.org/about/mimic/ – MIMIC-III: Medical Information Mart for Intensive Care https://www.hl7.org/ – Health Level Seven International (HL7 standards) https://nptel.ac.in/courses/106/106/106106213 – NPTEL: Health Informatics and Analytics https://www.coursera.org/learn/clinical-data-science – Coursera: Clinical Data Science Specialization https://streamlit.io/ – Streamlit: Rapid development of clinical data dashboards
Topics relevant to development of “Skill Development ：” Handling and preprocessing large-scale clinical datasets (structured and unstructured) Using SQL and Python for real-world healthcare data analysis

Developing predictive models for risk scoring and clinical decision support Conducting survival analysis and time-to-event modeling Visualizing healthcare data using professional dashboard tools Building real-time, interactive applications for clinical reporting and monitoring Applying NLP techniques for extracting information from clinical notes Topics relevant to development of "Environment and sustainability: Promoting data-driven, paperless clinical workflows for sustainable healthcare management Reducing unnecessary diagnostic procedures through predictive analytics Enhancing resource optimization in hospitals via data-informed decision-making Supporting public health sustainability through early risk detection and preventive care models Minimizing environmental burden by deploying digital dashboards and remote monitoring systems

Course Code: CAI3425	Course Title : AI in Epidemiology and Public Health Analytics Type of Course :Integrated	L – P – –TC	2	0	2	3
Version No.	1.0					
Course Pre-requisites	Basic knowledge of Machine Learning / Data Science					
Anti-requisites	NIL					
Course Description	This course explores how artificial intelligence can be applied in epidemiology and public health to understand, predict, and manage health outcomes at a population level. Students will gain skills in analyzing health indicators, working with public datasets, and developing AI-based models for forecasting outbreaks and informing health policy decisions					
Course Objective	The objective of the course is SKILL DEVELOPMENT of students using EXPERIENTIAL LEARNING techniques.					
Course Out Comes	On successful completion of the course the students shall be able to: CO1: Explain the principles of epidemiology and population health data. [Understand]					

	CO2: Preprocess and analyze large-scale public health datasets. [Apply] CO3: Develop AI models for outbreak prediction and risk estimation. [Apply] CO4: Evaluate model outcomes for public health decision-making. [Analyze] CO5: Design data-driven tools and dashboards for health policy and surveillance. [Create]			
Course Content：				
Module 1	Foundations of Epidemiology and Health Indicators	Assignment	Program activity	22 Hours
Introduces the core concepts of epidemiology including incidence, prevalence, mortality, morbidity, and health surveillance systems. Students learn about the structure and utility of public health datasets such as NHANES, DHS, and WHO databases.				
Module 2	Public Health Data Analytics	Assignment	Program activity	22 Hours s
Focuses on collecting, cleaning, and analyzing epidemiological data using statistical and AI tools. Students apply descriptive analytics and GIS-based visualizations to identify trends and disparities in population health.				
Module 3	AI Models for Population Health Forecasting	Assignment	Program activity	18 Hours
Students build predictive models for disease outbreak forecasting, disease burden estimation, and vaccination coverage prediction using machine learning algorithms like decision trees, SIR models, and ensemble methods.				
Module 4	Visualization and Ethical Implications in Public Health AI	Assignment	Program activity	13 Hours
Students develop dashboards and visual reports to support public health decision-making. The module also explores privacy, fairness, and ethical issues in population-level AI applications.				
List of Lab Tasks： Lab Sheet 1 Level 1: Load and explore public health datasets (e.g., NHANES, DHS)				

Level 2: Perform descriptive statistical analysis (mean, median, incidence rates) Lab Sheet 2 Level 1: Clean and preprocess public health data using pandas Level 2: Perform demographic segmentation and cohort analysis Lab Sheet 3 Level 1: Visualize disease distribution geographically using plotly or geopandas Level 2: Create choropleth maps and interactive visualizations Lab Sheet 4 Level 1: Train a decision tree or logistic regression model to classify health risk groups Level 2: Evaluate classification performance using ROC and precision-recall curves Lab Sheet 5 Level 1: Apply time series forecasting (e.g., ARIMA) to model disease trends Level 2: Compare model forecasts with actual data using RMSE Lab Sheet 6 Level 1: Implement a simple SIR model to simulate disease spread Level 2: Calibrate parameters to fit real-world outbreak data Lab Sheet 7 Level 1: Build a public health dashboard using Streamlit Level 2: Integrate multiple charts and summary statistics into an interactive interface Lab Sheet 8–10 Capstone Project: Design an AI-based early warning system or policy dashboard using historical epidemic or immunization data
Targeted Application & Tools that can be used Public Health Datasets: NHANES, DHS, WHO Global Health Observatory, India NFHS Programming & Analysis: Python, pandas, NumPy, SciPy, statsmodels Visualization: matplotlib, seaborn, plotly, geopandas, folium AI/ML Frameworks: scikit-learn, XGBoost, Prophet, TensorFlow (optional) Epidemiological Modeling: SIR/SEIR models using custom Python functions Geospatial Mapping: QGIS (optional), plotly choropleths, mapbox Dashboards & Reporting: Streamlit, Tableau (optional), Power BI Environments: Jupyter Notebook, Google Colab, Anaconda
Project work / Assignment:

Assignment：Assignments include module-wise exercises and real-world project implementation.
Text Book T1: Michael J. Paul & Mark Dredze, Social Monitoring for Public Health, Morgan & Claypool, 2017. T2: R. Bonita, R. Beaglehole & T. Kjellström, Basic Epidemiology, 2nd Edition, WHO Press, 2006.
References R1: David L. Streiner & Geoffrey R. Norman, Health Measurement Scales: A Practical Guide to Their Development and Use, Oxford University Press, 2015. R2: Online resources including CDC WONDER database, WHO Health Data Platform, and training modules from Johns Hopkins' Coursera series on Public Health Data Science R3: Tutorials and documentation for SIR/SEIR modeling, GIS-based health data visualization, and epidemiological surveillance dashboards Web resources ： https://www.who.int/data – WHO Global Health Observatory (GHO) https://www.cdc.gov/datastatistics – CDC Data & Statistics https://www.coursera.org/specializations/public-health-data-science – Coursera: Public Health Data Science Specialization https://ourworldindata.org/coronavirus – COVID-19 data and public health analysis tools https://www.healthdata.org/ – Institute for Health Metrics and Evaluation (IHME)
Topics relevant to development of “Skill Development ：” Epidemiological data cleaning, exploration, and statistical summarization Application of machine learning for disease prediction and outbreak forecasting Use of geospatial tools for mapping and hotspot analysis Development of interactive public health dashboards using real datasets Deployment of AI models for real-time public health monitoring Communication of data-driven insights for policy formulation and community awareness Topics relevant to development of “Environment and sustainability: Early detection and mitigation of disease outbreaks to reduce public health burden AI-based models to track the impact of climate change on health outcomes Data-driven planning for sustainable healthcare infrastructure and resource allocation

Monitoring of environmental hazards (e.g., air/water pollution) and their epidemiological impact Supporting sustainable development goals (SDGs) related to health and well-being (e.g., SDG 3, SDG 6, SDG 13)

Course Code: CAI3426	Course Title : Time Series Analysis for Patient Monitoring Type of Course :Integrated	L – P– –T C	2	0	2	3
Version No.	1.0					
Course Pre-requisites	Basic knowledge of Machine Learning / Data Science					
Anti-requisites	NIL					
Course Description	This course focuses on time series data generated from continuous patient monitoring systems such as ICU sensors, wearable devices, and medical records. Students will learn to preprocess, analyze, and forecast temporal data using classical and machine learning-based time series models to detect anomalies and predict patient conditions.					
Course Objective	The objective of the course is SKILL DEVELOPMENT of students using EXPERIENTIAL LEARNING techniques.					
Course Out Comes	On successful completion of the course the students shall be able to: CO1: Explain the nature and components of time series data in healthcare. [Understand] CO2: Preprocess and extract meaningful features from patient monitoring data. [Apply] CO3: Build forecasting and anomaly detection models using time series techniques. [Apply] CO4: Evaluate the performance of time series models for clinical applications. [Analyze] CO5: Design intelligent patient monitoring solutions using real-time data streams. [Create]					
Course Content:						

Module 1	Fundamentals of Healthcare Time Series Data	Assignment	Program activity	22 Hours
Introduces sources and structures of time series data in healthcare, such as vital signs, ECG, glucose levels, and ICU telemetry. Covers time series components (trend, seasonality, noise) and visualization techniques.				
Module 2	Time Series Preprocessing and Feature Engineering	Assignment	Program activity	22 Hours s
Focuses on handling missing data, irregular sampling, outliers, normalization, resampling, and windowing. Techniques for extracting rolling statistics and domain-specific temporal features are covered.				
Module 3	Forecasting and Anomaly Detection Models	Assignment	Program activity	18 Hours
Students implement statistical models like ARIMA, SARIMA, and exponential smoothing, as well as ML-based approaches such as LSTM, GRU, and hybrid models. Emphasis is placed on model selection, tuning, and interpretability.				
Module 4	Deployment and Real-time Monitoring Applications	Assignment	Program activity	13 Hours
Covers real-time patient monitoring frameworks, streaming data pipelines, alert systems, and dashboard integration using lightweight deployment tools (e.g., Streamlit, MQTT). Includes ethical and regulatory issues in monitoring.				

List of Lab Tasks：

Lab Sheet 1

Level 1: Load and visualize patient time series data (e.g., heart rate, ECG)

Level 2: Decompose time series into trend, seasonality, and residuals

Lab Sheet 2

Level 1: Handle missing data using interpolation and imputation techniques

Level 2: Resample irregularly spaced data and smooth noisy signals

Lab Sheet 3

Level 1: Extract rolling statistics (mean, std) and domain-specific features (e.g., HRV)

Level 2: Apply time windowing techniques for model input preparation

Lab Sheet 4

Level 1: Build an ARIMA model for forecasting a physiological signal

Level 2: Evaluate model performance using MAE, RMSE, and residual plots

Lab Sheet 5 Level 1: Train an LSTM model for predicting vital signs Level 2: Tune hyperparameters and visualize learning curves Lab Sheet 6 Level 1: Detect anomalies in ICU data using Z-score and Isolation Forest Level 2: Compare detection rates across different methods Lab Sheet 7 Level 1: Create a real-time streaming simulation using stored sensor data Level 2: Trigger alerts when predefined clinical thresholds are breached Lab Sheet 8–10 Capstone Project: Develop an end-to-end patient monitoring pipeline (data ingestion, processing, forecasting, alerting, and visualization)
Targeted Application & Tools that can be used Programming Languages & Libraries: Python, pandas, NumPy, matplotlib, seaborn, statsmodels Time Series & Forecasting: ARIMA, SARIMA, Holt-Winters, Prophet, scikit-learn, pmdarima Deep Learning Models: TensorFlow, Keras, LSTM, GRU, Autoencoders Anomaly Detection: Isolation Forest, Z-score, One-Class SVM Data Visualization & Dashboards: Plotly, Streamlit, Dash Healthcare Datasets & Simulators: MIMIC-III Waveform Database, PhysioNet, openICPSR vital sign data Streaming & Real-time Tools (optional): MQTT, Apache Kafka (for advanced setups), Flask for alerting interfaces Development Environment: Jupyter Notebook, Google Colab, VS Code
Project work / Assignment：
Assignment：Assignments include module-wise exercises and real-world project implementation.
Text Book T1: Aileen Nielsen, Practical Time Series Analysis: Prediction with Statistics and Machine Learning, O'Reilly Media, 2019

T2: Paolo Emiliozzi, Time Series Forecasting in Python, Leanpub, 2021
References R1: Rob J. Hyndman & George Athanasopoulos, Forecasting: Principles and Practice, OTexts, 3rd edition (freely available online) R2: Online resources and datasets from https://physionet.org/ – MIMIC, eICU, and waveform databases R3: Tutorials from Coursera’s Time Series Forecasting, NPTEL’s Healthcare Analytics, and GitHub repositories for clinical time series modeling using LSTM and GRU Web resources : https://physionet.org/ – PhysioNet: Free access to physiological time series datasets (e.g., MIMIC, ECG, ICU signals) https://otexts.com/fpp3/ – Online book: Forecasting: Principles and Practice https://www.coursera.org/learn/time-series – Coursera: Time Series Forecasting Specialization https://github.com/awslabs/gluon-ts – GluonTS: Probabilistic time series modeling with deep learning https://tensorflow.org/tutorials/structured_data/time_series – TensorFlow Time Series tutorials
Topics relevant to development of “Skill Development :” Preprocessing and analyzing time series data from patient monitoring systems Building forecasting models using ARIMA, LSTM, and hybrid techniques Detecting anomalies in vital signs and physiological signals Designing end-to-end data pipelines for real-time health monitoring Developing interactive dashboards and alerting systems for clinical decision support Hands-on experience with publicly available ICU and wearable sensor datasets Topics relevant to development of “Environment and sustainability: Reducing unnecessary hospital visits through continuous remote monitoring of patients Promoting sustainable healthcare by enabling early detection and preventive interventions Minimizing resource usage (e.g., tests, ICU beds) through AI-based forecasting and anomaly alerts Supporting environmentally friendly healthcare models by integrating low-power wearable devices Enhancing public health outcomes with minimal ecological impact via digital monitoring platforms

Course Code: CAI3427	Course Title: Language Models for Text Mining Type of Course: Integrated	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: Process text data to derive information from text. [Apply] Apply insights from textual information to real-world business. [Apply] Develop solutions for a particular NLP problem using different machine learning and deep learning techniques. [Apply] 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

<table>
<tr><td colspan="5">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.</td></tr>
<tr><td>Module 4</td><td>Natural Language Processing with Keras</td><td>Adversarial Quiz Tests</td><td>Module Tests</td><td>No. of
Sessions: 06</td></tr>
<tr><td colspan="5">Word Embeddings vs. One-Hot Encoding. Contextual Bag of Words (CBOW). Skipgram. Deep Learning for Document Classification.</td></tr>
<tr><td colspan="5">List of Lab 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.</td></tr>
</table>

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: Google Colab Python IDEs like PyCharm
Project work/Assignment: Mention the Type of Project /Assignment proposed for this course 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): 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).

Aditya Joshi, Pushpak Bhattacharyya. “Natural Language Processing”, Wiley Publication, 2023 (1st Edition).
References: R1. Chris Manning and Hinrich Schutze, “Foundations of Statistical Natural Language Processing”, 1st Edition, MIT Press. 1999. R2. Pawan Goyal. “Natural Language Processing”. 1st Edition, 2016. Weblinks W1. E-Book link or R2: https://drive.google.com/file/d/10nbwAJd-dv6htOOZVBgAvLd1WscI0RqC/view W2. Web Resource for T1: https://web.stanford.edu/~jurafsky/slp3/ - VERY VERY IMPORTANT!!! W3. NPTEL Courses: https://nptel.ac.in/courses/106106211 CMI), https://nptel.ac.in/courses/106105158 (IIT Kgp), https://nptel.ac.in/courses/106101007 (IITB), https://nptel.ac.in/courses/106105572 (IIT Kgp - NEW)

Course Code: CAI3428	Course Title: Practical Deep Learning with TensorFlow Type of Course: Integrated	L- T-P- C	2	0	2	3
Version No.	1.0					
Course Pre-requisites	CSE 3001-Artificial Intelligence and Machine Learning					
Anti-requisites	NIL					
Course Description	This course introduces students to the concepts of deep neural networks and state of the art approaches to develop deep learning models. In this course students will be given an exposure to the details of neural networks as well as deep learning architectures and to develop end-to-end models for such tasks. It will help to design and develop an application-specific deep learning models and also provide the practical knowledge handling and analyzing end user realistic applications.					
Course Objective	This course is designed to improve the learners EMPLOYABILITY SKILLS by using EXPERIENTIAL LEARNING techniques.					
Course Outcomes	On successful completion of this course the students shall be able to: Implement backpropagation and gradient descent techniques to train neural networks effectively. (Apply)					

	Build and train deep learning models using Python libraries such as TensorFlow and Keras for real-world applications. (Apply) 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:				
Assignment 1 on (Module 1 and Module 2) Assignment 2 on (Module 3)				
List of Lab 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

François Chollet, "Deep Learning with Python", 2nd Edition, Manning Publications, 2022

Ian Goodfellow, Yoshua Bengio, Aaron Courville, "Deep Learning", MIT Press, 2017.

REFERENCES

Amlan Chakrabarti Amit Kumar Das, Saptarsi Goswami, Pabitra Mitra , "Deep Learning", Pearson Publication, 2021.

David Foster, "Generative Deep Learning" O'Reilly Publishers, 2020.

John D Kellehar, "Deep Learning", MIT Press, 2020.

JOURNALS/MAGAZINES

IEEE Transactions on Neural Networks and Learning Systems

https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=5962385

IEEE Transactions on Pattern Analysis and Machine Intelligence

https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=34http://ijaerd.com/papers/special_papers/IT032.pdf International Journal of Intelligent Systems https://onlinelibrary.wiley.com/journal/1098111x SWAYAM/NPTEL/MOOCs: Swayam Nptel – Deep Learning – IIT Ropar https://onlinecourses.nptel.ac.in/noc21_cs35/preview Coursera – Neural Networks and Deep Learning Andrew Ng Coursera - Neural Networks for Machine Learning by Geoffrey Hinton in Coursera

Course Code: UG COURSE: CAI3429	Course Title: Deep Learning for Computer Vision Type of Course: Integrated	L-T-P-C	2	0	2	3
Version No.	1.0					
Course Pre-requisites	MAT1003 Applied Statistics, Knowledge of Python, Machine Learning, and Digital image processing					
Anti-requisites	NIL					
Course Description	This course covers the fundamentals and advanced concepts of deep learning for computer vision applications. Students will explore convolutional neural networks (CNNs), object detection, image segmentation, and generative models. Hands-on lab experiments will reinforce theoretical concepts using frameworks like TensorFlow and PyTorch.					
Course Out Comes	On successful completion of the course the students shall be able to: Understand the Fundamentals of Deep Learning for Vision Explain the core concepts of neural networks and deep learning architectures for image processing. Implement and optimize convolutional neural networks (CNNs) for classification tasks. Apply Object Detection and Image Segmentation Techniques Implement and analyze state-of-the-art object detection algorithms such as YOLO, Faster R-CNN, and SSD. Develop and evaluate image segmentation models like U-Net and Mask R-CNN.					

	Explore Advanced Deep Learning Techniques for Vision Utilize Vision Transformers (ViTs) and attention mechanisms for image classification. Generate and manipulate images using Generative Adversarial Networks (GANs). Deploy and Optimize Deep Learning Models for Real-World Applications			
Course Content:				
Module 1	Fundamentals of Deep Learning for Vision	Assignment	Practical	No. of Classes:8
Introduction to Deep Learning & Neural Networks, Convolutional Neural Networks (CNNs) Architecture Backpropagation & Optimization in CNNs, Transfer Learning & Pretrained Models.				
Module 2	Object Detection & Image Segmentation	Assignment	Practical	No. of Classes:14
Introduction to Object Detection (R-CNN, SSD, YOLO), Region Proposal Networks (Faster R-CNN) Semantic & Instance Segmentation (U-Net, Mask R-CNN), Real-time Object Detection Applications				
Module 3	Advanced Topics in Vision	Assignment	Practical	No. of Classes:8
Attention Mechanisms & Vision Transformers (ViTs), Generative Adversarial Networks (GANs) for Image Generation, Self-supervised Learning for Vision, Multi-modal Learning (CLIP, DALL·E)				
Module 4	Applications & Deployment	Assignment	Practical	No. of Classes:8
Edge AI & Mobile Deployment (TensorFlow Lite, ONNX), Adversarial Attacks & Robustness in Vision Models, Explainability & Interpretability of Vision Models, Case Studies & Industry Applications				
Lab Experiments are to be conducted on the following topics:- Lab Sheet 1: Keras Sequential API model Read in the data and explore				

Define a Sequential API model

Define the hyperparameters and optimizer

Train the model and visualize the history

Testing

Keras Functional API model:

Define a Functional API model

Train the model and visualize the history

Lab Sheet 2:

Softmax regression with Keras

Read in the data and prepare

Define a Sequential API model

Define the hyperparameters and optimizer

Train the model and visualize the history

Testing

Lab Sheet 3:

Convolutional Neural Network with Keras (grayscale images)

Read in the data:

Visualize the data:

Prepare the data:

Define a CNN model:

Define the hyperparameters and optimizer:

Train the model and visualize the history:

Testing:

Lab Sheet 4:

Convolutional Neural Network with Keras (color images):

Read in the data:

Visualize the data:

Prepare the data:

Define a CNN model:

Define the hyperparameters and optimizer:

Train the model and visualize the history:

Testing:

Lab Sheet 5:

Time series and prediction:

Read in the data and explore:

Apply the exponential smoothing method and predict

Recurrent neural network (RNN):

Pre-processing:

Do the necessary definitions: (Hyper parameters, Model,

Train the model:

Predict the future:

Lab Sheet 6:

Document classification with LSTM network:

Read in the data:

Explore the data:

Data preprocessing:

Define the model:

Define the optimizer and compile:

Train the model and visualize the history:

Testing:

Lab Sheet 7:

Document classification with LSTM network (Binary):

Read in the data:

Explore the data:

Data preprocessing:

Define the model:

Define the optimizer and compile:

Train the model and visualize the history:

Testing:

Lab Sheet 8:

Document classification with LSTM + CNN network (Binary):

Read in the data:

Explore the data:

Data preprocessing:

Define the model:

Define the optimizer and compile:

Train the model and visualize the history:

Testing:

Lab Sheet 9:

Softmax regression to recognize the handswritten digits:

Download the MNIST data:

Take a look at the dataset:

Do the necessary definitions:

Training and Testing:

Multi-layer neural network to recognize the handswritten digits:

Download the MNIST data:

Take a look at the dataset:

Do the necessary definitions:

Training and Testing:

Lab Sheet 10:

Object Detection using YOLOv5

Lab Sheet 11:

Image Segmentation using U-Net

Custom Object Detection using Faster R-CNN

Lab Sheet 12:

Implementing Vision Transformers for Image Classification

Generating Images using GANs (DCGAN, StyleGAN)

(Group Project)

Object Detection and Recognition:

Haar cascade object detection (e.g., face detection or object detection using pre-trained classifiers).

Feature-based object detection using techniques like Speeded-Up Robust Features (SURF) or Scale-Invariant Feature Transform (SIFT).

Deep learning-based object detection using Convolutional Neural Networks (CNNs) or You Only Look Once (YOLO) algorithm.

Optical Character Recognition (OCR):

Preprocessing of text images (e.g., binarization, noise removal, or skew correction).

Text localization using techniques like connected component analysis or Stroke Width Transform (SWT). Character recognition using machine learning algorithms like Support Vector Machines (SVM) or Convolutional Neural Networks (CNNs). Gesture Recognition: Hand segmentation using techniques like background subtraction or skin color detection. Feature extraction from hand regions (e.g., finger counting, hand shape descriptors). Classification of gestures using machine learning algorithms (e.g., k-Nearest Neighbors or Support Vector Machines).
Tools/Software Required : OpenCV 4 Python 3.7 MATLAB
Text Books "Deep Learning for Computer Vision Image Classification, Object Detection and Face Recognition in Python" Jason Brownlee (2019) "Deep Learning for Computer Vision with python" Adrian Rosebrock (2017)
References Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press. A foundational book covering deep learning principles, including CNNs, optimization, and generative models. Raschka, S., & Mirjalili, V. (2022). Machine Learning with PyTorch and Scikit-Learn. Packt Publishing. Covers practical deep learning techniques using PyTorch, including CNNs and transfer learning. Geron, A. (2022). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow (3rd Edition). O'Reilly Media. Provides hands-on implementations of deep learning for computer vision using TensorFlow and Keras. Zhang, A., Lipton, Z. C., Li, M., & Smola, A. J. (2021). Dive into Deep Learning. Available online (https://d2l.ai). Open-access book covering CNNs, object detection, and advanced vision techniques with PyTorch and TensorFlow. Chollet, F. (2021). Deep Learning with Python (2nd Edition). Manning Publications. Explains deep learning fundamentals and applications with Keras, including image classification and segmentation.

Ballé, J., Laparra, V., & Simoncelli, E. P. (2017). Deep Learning for Computer Vision: A Brief Introduction. A concise introduction to CNNs, object detection, and generative models.

Course Code: CAI2507	Course Title: Deep Reinforcement Learning Type of Course: Theory	L- T- P- C	2	0	0	2
Version No.	1.0					
Course Pre-requisites	CSE1700 Essentials of AI					
Anti-requisites	NIL					
Course Description	This course provides an in-depth understanding of Deep Reinforcement Learning (DRL), an area at the intersection of machine learning and artificial intelligence that enables agents to make decisions by interacting with dynamic environments. Starting from the foundational concepts of reinforcement learning such as agents, rewards, policies, and value functions, the course progresses to advanced topics including Markov Decision Processes (MDPs), Q-learning, and policy optimization. Students will explore how deep learning techniques are integrated with reinforcement learning to solve complex problems in areas like robotics, game playing, autonomous driving, and recommendation systems.					
Course Outcomes	On successful completion of the course the students shall be able to: 1. Describe dynamic programming concepts to find an optimal policy in a gaming environment [Understanding] 2. Identify on-policy and off-policy Monte Carlo methods for finding an optimal policy in a reinforcement learning environment. [Understanding] 3. Apply Temporal Difference learning techniques to the Frozen Lake RL environment [Apply] 4. Distinguish various exploration-exploitation strategies of the Multi-Armed Bandit (MAB) problem [Analyse]					

	CO1: Explain the fundamental concepts of reinforcement learning, including agents, environments, rewards, and policies. [Level: Understand] CO2: Develop deep reinforcement learning models using deep Q-networks (DQNs) and policy gradient methods for complex environments. [Level: Understand] CO3: Implement model-free reinforcement learning algorithms such as Q-learning and SARSA to solve decision-making tasks. [Level: Apply] CO2: Analyze the mathematical foundations of Markov Decision Processes (MDPs) and solve simple RL problems using dynamic programming methods. [Level: Analyze]			
Course Content:				
Module 1	Introduction to Reinforcement Learning	Assignment		8 Classes
Topics: Topics : Elements of RL, Agent, environment Interface, Goals and rewards, RL platforms, Applications of RL, Maths essentials of RL, Policy and its types, episodic and continuous tasks, return and discount factor, fundamental functions of RL – value and Q functions, model-based and model-free learning, types of RL environments				
Module 2	Deep Q Networks and Policy Gradient methods	Assignment		8 Classes
Topics: Introduction to Deep Q-Learning, Q-Network Architecture, Experience Replay, Target Networks, Training the DQN, Applications of DQN, Limitations of Value-Based Methods, Introduction to Policy Gradients				
Module 3	Temporal Difference(TD) Learning	Assignment		7 Classes
Topics: Temporal difference learning: TD Prediction, TD Control : On-policy TD control – SARSA, computing the optimal policy using SARSA, Off-policy TD control – Q learning, computing optimal policy using Q learning, Examples, Difference between SARSA and Q-learning, Comparison of DP, MC and TD methods				
Module-4	Markov Decision Process(MDP)	Assignment		7 Classes
Topics:				

Solving MDP using Bellman Equation, Algorithms for optimal policy using Dynamic Programming -Value iteration and policy iteration, Example : Frozen Lake problem, Limitations and Scope
Project work/Assignment:
Assignment 1 on (Module 1 and Module 2) Assignment 2 on (Module 3 and Module 4)
REFERENCE MATERIALS: TEXTBOOKS Richard S. Sutton and Andrew G. Barto, “Reinforcement Learning: An Introduction”, MIT press, Second Edition, 2018. SudharshanRavichandiran, “Deep Reinforcement Learning with Python”, Packt Publishers, Second Edition, 2020 REFERENCES Maxim Lapan , “Deep Reinforcement Learning Hands-On”, Packt Publishing,2023 LaurraGraesser and Wan Loon Keng, “Foundations of Deep Reinforcement Learning”, Pearson, 2022 Marco Wiering, Martijn van Otterlo,”Reinforcement Learning: State-of-the-Art”, Springer, https://www.udemy.com/course/artificial-intelligence-reinforcement-learning-in-python/ JOURNALS/MAGAZINES IEEE Transactions on Deep Reinforcement Learning: A Survey https://ieeexplore.ieee.org/document/9904958 IEEE Transactions on A Survey on Offline Reinforcement Learning: Taxonomy, Review, and Open Problems https://ieeexplore.ieee.org/document/9904958 IEEE Spectrum https://www.technologyreview.com/ SWAYAM/NPTEL/MOOCs: SwayamNptel – Reinforcement Learning By Prof. BalaramanRavindran|IITMadrashttps://onlinecourses.nptel.ac.in/noc20_cs74/preview

Coursera- Reinforcement Learning Specialization https://www.coursera.org/specializations/reinforcement-learning 3.Coursera - Unsupervised Learning, Recommenders, Reinforcement Learning https://www.coursera.org/learn/unsupervised-learning-recommenders-reinforcement-learning

Course Code: CAI2508	Course Title: Deep Reinforcement Learning Lab Type of Course: Lab	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 lab course offers hands-on experience in implementing and experimenting with Deep Reinforcement Learning (DRL) algorithms in simulated environments. Students will explore core concepts such as Q-learning, Deep Q-Networks (DQNs), Policy Gradient methods, and Actor-Critic algorithms through practical coding exercises. The course emphasizes building and training DRL models using libraries like TensorFlow, PyTorch, and OpenAI Gym. Learners will design agents that can learn optimal behaviors through interaction with dynamic environments such as games, navigation tasks, and robotic simulations. By the end of the course, students will be equipped to design, implement, and evaluate deep reinforcement learning solutions for complex decision-making 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: 1. Apply dynamic programming concepts to find an optimal policy in a gaming environment [Application] 2. Implement on-policy and off-policy Monte Carlo methods for finding an optimal policy in a reinforcement learning environment. [Application]					

	3. Apply Temporal Difference learning techniques to the Frozen Lake RL environment [Application] 4. Apply various exploration-exploitation strategies of the Multi-Armed Bandit (MAB) problem[Application] Here are well-structured Course Outcomes (COs) for the Deep Reinforcement Learning Lab, aligned with practical and implementation-based objectives: --- Course Title: Deep Reinforcement Learning Lab Course Outcomes (COs) By the end of this course, students will be able to: CO1: Implement basic reinforcement learning algorithms such as Q-learning and SARSA in simulated environments. [Level: Apply] CO2: Design and train Deep Q-Networks (DQNs) to solve complex decision-making tasks using neural networks. [Level: Create] CO3: Apply policy gradient methods and understand their use in continuous and high-dimensional action spaces. [Level: Apply] CO4: Analyze the performance of different DRL algorithms based on metrics such as learning efficiency, stability, and convergence. [Level: Analyze]
Course Content:	
List of Lab Tasks: 1 .Software Setup :installalling Anaconda, OpenAI Gym and Universe. Basic simulations of some gaming environments in Gym 2. Working with Gym environments to create agents with random policy 2.1 Create the Frozen Lake GYM environment and explore the states, action, transition probability, reward functions and generating episodes. 2.2 Create an agent for the Cart-Pole environment using a random policy and record the game 3. Finding the optimal policy for the agent using Dynamic Programming 3.1 Compute the optimal policy for the Frozen Lake Environment using value iteration method 3.2 Compute the optimal policy for the Frozen Lake Environment using policy iteration method	

4. Build and train a basic DQN agent using PyTorch or TensorFlow 5. Add experience replay and target network updates to enhance learning 6. Implementing Temporal Difference prediction for the Frozen lake environment for a random policy 7. Computing the optimal policy using on-policy TD control – SARSA 8. Computing the optimal policy using off-policy TD control – Q-learning 9. Multi-Armed Bandit problem 9.1 Creating a MAB in Gym 9.2 Compute the best arm using various exploration strategies such as epsilon-greedy and softmax exploration method. 10. Implement variations of DQN to improve performance and stability
Targeted Application & Tools that can be used : Execution of the RL algorithms will be done using the environments provided by OpenAI's Gym and Gymnasium of Farama Foundation in "Colab", available at https://colab.research.google.com/ or Jupyter Notebook. Lab tasks will be implemented using the necessary libraries available in Python
Project work/Assignment: Mention the Type of Project /Assignment proposed for this course
Students can be given group assignments to develop different gaming environments and implement the RL algorithms
Text Book Richard S. Sutton and Andrew G. Barto, "Reinforcement Learning: An Introduction", MIT press, Second Edition, 2018. Sudharshan Ravichandiran, "Deep Reinforcement Learning with Python", Packt Publishers, Second Edition, 2020
References LaurraGraesser and Wan Loon Keng, "Foundations of Deep Reinforcement Learning", Pearson, 2022 https://www.udemy.com/course/artificial-intelligence-reinforcement-learning-in-python/

Course Code:	Course Title: Image Processing and Analysis	L- T-P- C				

CAI3400	Type of Course: Integrated		2	0	2	3
Version No.	1.0					
Course Pre-requisites	Essentials of AI					
Anti-requisites	NIL					
Course Description	This course provides a solid foundation in the fundamentals and applications of image processing and analysis. Students will learn techniques for enhancing, segmenting, and interpreting digital images using various algorithms. It also covers essential image transformation and computer vision concepts relevant to real-world problems.					
Course Objective	To equip learners with the theoretical knowledge and practical skills in digital image processing techniques, enabling them to develop analytical solutions for computer vision problems.					
Course Outcomes	On successful completion of this course the students shall be able to: 1. Understand the basic concepts and techniques of image processing. (Understand) 2. Apply spatial and frequency domain operations to enhance images. (Apply) 3. Analyze images using segmentation, filtering, and transformation techniques. (Analyze) 4. Implement real-time image processing tasks using Python and OpenCV. (Apply)					
Course Content:						

Module 1	Introduction to Image Processing	Assignment 1	18[8L+10P] Sessions
Topics: Fundamentals of digital images, pixel operations, image acquisition, sampling and quantization, basic image transformations. Understanding image histograms and intensity transformations.			
Module 2	Image Enhancement and Restoration	Assignment 1	14[7L+7P] Sessions
Topics: Spatial domain techniques, histogram equalization, smoothing and sharpening filters. Noise models, restoration techniques, and Wiener filtering in the frequency domain.			

Module 3	Image Segmentation and Morphological Processing	Assignment 2	14[6L+8P] Sessions
Topics: Edge detection, thresholding, region growing and splitting. Morphological operations for shape analysis and binary image processing.			
Module 4	Image Analysis and Applications		14[6L+8P] Sessions
Topics: Feature extraction, texture analysis, object recognition. Applications in biomedical imaging, satellite imagery, and computer vision systems			
Project work/Assignment:			
Assignment 1 on (Module 1 and Module 2) Assignment 2 on (Module 3)			
List of Lab Tasks: 1. Perform basic image operations like read, write, and display using OpenCV. 2. Implement image enhancement using histogram equalization. 3. Apply various smoothing and sharpening filters to an image. 4. Convert color images to grayscale and binary formats. 5. Perform geometric transformations: rotation, scaling, and translation. 6. Add Gaussian, salt-and-pepper noise and apply noise removal filters. 7. Implement edge detection using Sobel, Prewitt, and Canny methods. 8. Segment an image using thresholding and region-based techniques. 9. Perform morphological operations: erosion, dilation, opening, and closing. 10. Extract features using contour detection and bounding boxes. 11. Analyze texture patterns using GLCM (Gray Level Co-occurrence Matrix). 12. Object detection using color segmentation and contour properties. 13. Image classification using histogram features and a simple classifier. 14. Real-time video processing using webcam and OpenCV. 15. Mini-project: Build an end-to-end image processing pipeline for a selected application.			

REFERENCE MATERIALS: TEXTBOOKS 1. Rafael C. Gonzalez and Richard E. Woods, “Digital Image Processing”, 4th Edition, Pearson Education, 2018. 2. Anil K. Jain, “Fundamentals of Digital Image Processing”, Prentice Hall, 1989. REFERENCES 1. Bernd Jähne, “Digital Image Processing”, Springer, 2005. 2. Milan Sonka, Vaclav Hlavac, Roger Boyle, “Image Processing, Analysis, and Machine Vision”, Cengage Learning, 2014. 3. Mark Nixon and Alberto Aguado, “Feature Extraction and Image Processing for Computer Vision”, Academic Press, 2019. JOURNALS/MAGAZINES 1. IECE Journal of Image Analysis and Processing (JIAP): 2. Medical Image Analysis: 3. IPOL Journal (Image Processing On Line): SWAYAM/NPTEL/MOOCs: 1. Digital Image Processing 2. Computer Vision and Image Processing – Fundamentals and Applications 3 Medical Image Analysis

Course Code: CAI3402	Course Title: Optimization Techniques for Machine Learning Type of Course: Integrated	L- T- P- C	2	0	2	3
Version No.	1.0					
Course Pre-requisites	Essentials of AI					
Anti-requisites	NIL					
Course Description	This course introduces a range of machine learning models and optimization tools that are used to apply these models in practice. Course will introduce what lies behind the optimization tools often used as a black box as well as an understanding of the trade-offs of numerical accuracy and theoretical and empirical complexity. For the students with some optimization background this course will introduce a variety of applications arising in machine learning and statistics as well as novel optimization methods targeting these applications.					

Course Objective	The objective of the course is to familiarize the learners with the concepts of Optimization Techniques for Machine Learning and attain Skill Development through Participative Learning techniques.			
Course Outcomes	On successful completion of this course the students shall be able to: Describe fundamentals of Optimization Techniques [Remember]. Explain Optimization Techniques for Machine learning. [Understand]. Discuss Convex optimization models [Understand]. Apply Methods for convex optimization [Apply].			
Course Content:				
Module 1:	Optimization Basics	Quiz	Knowledge based Quiz	16[8L+8 P]Sessions
Topics: Introduction, The Basics of Optimization: Bivariate and Multivariate, Convex Objective Functions, Properties of Optimization in Machine Learning: Least-Square Classification, Support Vector Machines, Logistic Regression, Optimization Models for Binary Targets, Optimization Models for the Multiclass Setting, Coordinate Descent.				
Module 2:	Optimization Solutions	Quiz	Comprehension based Quiz	15[8L+7 P]Sessions
Topics: Introduction, Challenges in Gradient-Based Optimization: Momentum-Based Learning, RMSProp, Newton Method, Newton Methods in Machine Learning: Computationally Efficient Variations of Newton Method, The Subgradient Method, Proximal Gradient Method, Non-differentiable Optimization Functions: Designing Surrogate Lose Functions.				
Module 3	Constrained Optimization	Assignment	Batch-wise Assignments	14[7L+7 P]Sessions
Topics: Introduction, Primal Gradient Descent Methods: Primal Gradient Descent, Lagrangian Relaxation and Duality: Fundamentals of SVM Dual, Optimization Algorithms for the SVM Dual				
Module 4:	Optimization in Computational Graphs	Assignment and Presentation	Batch-wise Assignment and Presentations	15[7L+8 P]Sessions
Topics: Introduction, basics, Optimization in Directed Acyclic Graphs: Optimizations in Directed Acyclic Graphs, Broad Framework, Application: Node-to-Node derivations using Brute Force				
Targeted Application & Tools that can be used: Use of Matlab tool				
Project work/Assignment: Survey on Methods for convex optimization Survey on Machine learning models related to optimization				
Introduction to Optimization Problems using Python/Matlab.				

Implement Bivariate and Multivariate Optimization. Solve Least-Square Classification Problem. Implement Support Vector Machine (SVM) Optimization. Logistic Regression Model Optimization. Coordinate Descent Algorithm Implementation. Gradient Descent and Stochastic Gradient Descent Techniques. Implement Momentum-based Gradient Descent. RMSProp Optimization Method Application. Newton Method Implementation for Machine Learning. Subgradient Method for Non-differentiable Functions. Proximal Gradient Method Implementation. Solve Constrained Optimization Problems with Lagrangian Methods. Optimization in Directed Acyclic Graphs. Survey and Comparative Analysis of Optimization Algorithms.
Text Book T1. Charu C. Aggarwal, " Linear Algebra and Optimization for Machine Learning", Springer, 2020. T2. Sra Suvrit, Nowozin Sebastian, and Wright Stephen J, "Optimization for Machine Learning", The MIT Press,2012.
References R1.Guanghui Lan, "First-order and Stochastic Optimization Methods for Machine Learning", Springer Cham, 2020. Web References W1. https://sm-nitk.vlabs.ac.in/ W2. https://nptel.ac.in/courses/
Topics relevant to SKILL DEVELOPMENT: Concepts of Convex optimization models and Methods for convex optimization for Skill Development through Problem Solving methodologies. This is attained through assessment component mentioned in course handout.

Course Code: CAI3403	Course Title: Reinforcement Learning Type of Course: Integrated	L-T-P-C	2	0	2	3
Version No.	1.0					

Course Pre-requisites	Artificial Intelligence and Machine Learning
Anti-requisites	NIL
Course Description	For both engineers and researchers in the field of Computer science, it is common to develop models of real-life situations and develop solutions based on those models. It is of utmost importance to come up with innovative solutions for scenarios that are highly stochastic. The objective of this course, is to introduce different reinforcement learning techniques which is a promising paradigm for stochastic decision making in the forthcoming era. Starting from the basics of stochastic processes, this course introduces several RL techniques that are as per the industry standard. With a good knowledge in RL, the students will be able to develop efficient solutions for complex and challenging real-life problems that are highly stochastic in nature.
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: 1. Apply dynamic programming concepts to find an optimal policy in a gaming environment [Application] 2. Implement on-policy and off-policy Monte Carlo methods for finding an optimal policy in a reinforcement learning environment. [Application] 3. Apply Temporal Difference learning techniques to the Frozen Lake RL environment [Application] 4. Apply various exploration-exploitation strategies of the Multi-Armed Bandit (MAB) problem[Application]
Course Content:	

Module 1	Introduction to Reinforcement Learning	Assignment	Programming using the OpenAI Gym environment	No. of Classes L – 5 P – 6
Topics : Elements of RL, Agent, environment Interface, Goals and rewards, RL platforms, Applications of RL, Markov decision process (MDP), RL environment as a MDP, Maths essentials of RL, Policy and its types, episodic and continuous tasks, return and discount factor, fundamental functions of RL – value and Q functions, model-based and model-free learning, types of RL environments, Solving MDP using Bellman Equation, Algorithms for optimal policy using Dynamic Programming -Value iteration and policy iteration, Example : Frozen Lake problem, Limitations and Scope				

Module 2	Monte-Carlo(MC) methods	Assignment	Programming using the OpenAI Gym environment	No. of Classes L-5 P-6
Topics: Monte Carlo methods, prediction and control tasks, Monte Carlo prediction : algorithm, types of MC prediction, examples , incremental mean updates, Monte Carlo Control : algorithm, on-policy MC control, MC with epsilon-greedy policy, off-policy MC control. Limitations of MC method.				
Module 3	Temporal Difference(TD) Learning	Assignment/Quiz	Programming using the OpenAI Gym environment	No. of Classes L-7 P -6
Topics: Temporal difference learning: TD Prediction, TD Control : On-policy TD control – SARSA, computing the optimal policy using SARSA, Off-policy TD control – Q learning, computing optimal policy using Q learning, Examples, Difference between SARSA and Q-learning, Comparison of DP, MC and TD methods.				
Module 4	Multi-Armed Bandit (MAB) problem	Assignment	Programming using the OpenAI Gym environment	No. of Classes L-6 P -4
Topics: Understanding the MAB problem, Various exploration strategies – epsilon-greedy, softmax exploration, upper confidence bound and Thompson sampling, Applications of MAB - finding the best advertisement banner for a web site, Contextual bandits, introduction to Deep Reinforcement Learning(DRL) Algorithm – Deep Q Network (DQN)				

List of Lab Tasks:

1 .Software Setup :installalling Anaconda, OpenAI Gym and Universe.

Basic simulations of some gaming environments in Gym

2. Working with Gym environments to create agents with random policy

2.1 Create the Frozen Lake GYM environment and explore the states, action, transition probability, reward functions and generating episodes.

2.2 Create an agent for the Cart-Pole environment using a random policy and record the game

3. Finding the optimal policy for the agent using Dynamic Programming

3.1 Compute the optimal policy for the Frozen Lake Environment using value iteration method

3.2 Compute the optimal policy for the Frozen Lake Environment using policy iteration method

4. Implementing Monte Carlo prediction method using blackjack game

4.1 Every-visit MC prediction

4.2 First-visit MC prediction

5. Implementing on-policy MC control method using the epsilon-greedy policy for the blackjack game 6. Implementing Temporal Difference prediction for the Frozen lake environment for a random policy 7. Computing the optimal policy using on-policy TD control – SARSA 8. Computing the optimal policy using off-policy TD control – Q-learning 9. Multi-Armed Bandit problem 9.1 Creating a MAB in Gym 9.2 Compute the best arm using various exploration strategies such as epsilon-greedy and softmax exploration method. 10. Application of MAB – Finding the best advertisement banner for a web site using MAB
Targeted Application & Tools that can be used : Execution of the RL algorithms will be done using the environments provided by OpenAI's Gym and Gymnasium of Farama Foundation in "Colab", available at https://colab.research.google.com/ or Jupyter Notebook. Lab tasks will be implemented using the necessary libraries available in Python
Project work/Assignment: Mention the Type of Project /Assignment proposed for this course
Students can be given group assignments to develop different gaming environments and implement the RL algorithms
Text Book Richard S. Sutton and Andrew G. Barto, "Reinforcement Learning: An Introduction", MIT press, Second Edition, 2018. SudharshanRavichandiran, "Deep Reinforcement Learning with Python", Packt Publishers, Second Edition, 2020
References LaurraGraesser and Wan Loon Keng, "Foundations of Deep Reinforcement Learning", Pearson, 2022 https://www.udemy.com/course/artificial-intelligence-reinforcement-learning-in-python/

Course Code: CAI3404	Course Title: AI in Cyber Security Type of Course: Integrated	L- T-P-C				

				2	0	2	3
Version No.	1.0						
Course Pre-requisites	CSE 3001-Artificial Intelligence and Machine Learning, CSE 2004 – Computer Networks, Basic Programming Skills in Python.						
Anti-requisites	NIL						
Course Description	This course introduces students to the fundamentals of applying artificial intelligence techniques to cyber security. It covers threat detection, anomaly detection, malware classification, and the application of machine learning/deep learning methods in securing data and systems. It includes hands-on lab exercises to help students build AI models for real-world cyber security challenges.						
Course Objective	This course is designed to improve learners' employability skills by applying experiential learning techniques in cybersecurity and artificial intelligence.						
Course Outcomes	On successful completion of this course the students shall be able to: Apply AI techniques to detect and mitigate cyber threats. Analyze network and system data to uncover anomalies. Build intelligent systems to classify and predict malicious activity. Use tools and frameworks to develop cyber security solutions using machine learning.						

Course Content:

Module 1	Introduction to AI in Cyber Security	Assignment		18[8L+10P] Sessions

Topics:

Fundamentals of Cyber Security: CIA Triad, Threats & Vulnerabilities, Role of Artificial Intelligence in Cyber Security, Cyber Attack Lifecycle and Defense Mechanisms, Overview of Machine Learning and Deep Learning Techniques, Introduction to Data Sources: Network logs, system logs, NetFlow, and packet capture data, Overview of Threat Intelligence Platforms (TIPs),

AI-based Cyber Security Use Cases in Industry, Challenges in Deploying AI for Cyber Defense (adversarial attacks, data imbalance, etc.).

Module 2	Anomaly and Intrusion Detection Systems	Assignment		14[7L+7P] Sessions

Topics:

Types of Intrusion Detection Systems (IDS): Signature-based vs. Anomaly-based, Dataset Exploration: KDDCup, NSL-KDD, CICIDS2017, Data Preprocessing: Feature Engineering, Label Encoding, Normalization, Supervised Learning for Intrusion Detection: SVM, Random

Forest, Decision Trees, Unsupervised Learning: K-Means Clustering, Isolation Forests, Autoencoders,

Model Evaluation: Confusion Matrix, ROC Curve, Precision/Recall, Real-time Detection Systems with Streaming Data (e.g., using Kafka or PySpark), Case Study: Building and Deploying a ML-based IDS.

Module 3	Malware Detection and Classification	Assignment		14[6L+8P] Sessions

Topics:

Introduction to Malware Types: Virus, Worm, Trojan, Ransomware, Spyware, Static and Dynamic Malware Analysis Techniques, Feature Extraction: Opcode, API Call Sequences, Binary Analysis,

Deep Learning Techniques: CNN for image-based malware classification, RNN/LSTM for sequence learning, Model Training with Malimg and Microsoft Malware Dataset, Model Optimization Techniques (Dropout, Early Stopping, Hyperparameter Tuning), Use of Embeddings for Malware Behavior Analysis, Adversarial Examples and Evasion Techniques in Malware Detection.

Module 4	AI for Cyber Threat Intelligence and Response			

Threat Intelligence Fundamentals and Sources (OSINT, commercial feeds), Natural Language Processing for Cyber Threat Intelligence (CTI) extraction, Entity Recognition and Classification from Threat Reports, URL and Email Phishing Detection using ML/NLP, Behavioral Biometrics: Keystroke Dynamics, Mouse Movement Analysis, Deep Learning for Security Information and Event Management (SIEM), AI in Incident Response and Automation (SOAR platforms),

Case Study: Detecting phishing websites using NLP and ensemble models.

Project work/Assignment:

Assignment 1: Threat Detection using Supervised Learning

Assignment 2: Malware Classification using Deep Learning

Mini Project (Team-based): AI-Driven Cyber Threat Intelligence Dashboard

Lab 1: Explore Python libraries for cyber security (Scikit-learn, TensorFlow, Keras, Pandas).

Lab 2: Data preprocessing and feature extraction from KDD Cup dataset.

Lab 3: Develop a basic binary classifier to detect malicious network traffic.

Lab 4: Implement an SVM model for intrusion detection.

Lab 5: Build a deep neural network to classify attacks using NSL-KDD dataset.

Lab 6: Train an autoencoder for anomaly detection in log files.

Lab 7: Use Random Forest for malware classification.

Lab 8: Text mining of phishing emails using NLP.

Lab 9: Create a spam classifier using Naïve Bayes.

Lab 10: Train an LSTM model for real-time anomaly detection.

Lab 11: Visualize threat patterns using t-SNE and PCA.

Lab 12: Use a GAN to generate synthetic attack data.

Lab 13: Build a model for phishing URL detection.

Lab 14: Implement behavioral biometrics using keystroke dynamics.

Lab 15: Develop a dashboard integrating AI-driven threat alerts.

REFERENCE MATERIALS:

TEXTBOOKS

Mark Stamp, Introduction to Machine Learning with Applications in Information Security, CRC Press, 2020.

Clarence Chio, David Freeman, Machine Learning and Security: Protecting Systems with Data and Algorithms, O'Reilly, 2018.

REFERENCES

Xiaofeng Chen, Cyber Security: AI and Big Data Perspective, Springer, 2021.

Sumeet Dua, Xian Du, Data Mining and Machine Learning in Cybersecurity, CRC Press, 2011.

Richard E. Smith, Elementary Information Security, Jones & Bartlett Learning, 2021.

JOURNALS/MAGAZINES

IEEE Transactions on Information Forensics and Security
https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=8858
Covers the theory and practice of information forensics, cyber defense, and security.

ACM Transactions on Privacy and Security (TOPS)
https://dl.acm.org/journal/tops
Peer-reviewed research on cyber security systems, privacy-preserving AI, and secure protocols.

Computers & Security (Elsevier)
https://www.sciencedirect.com/journal/computers-and-security
Practical and academic articles on cyber threats, security analytics, and AI applications in security.

Journal of Cybersecurity (Oxford Academic)
https://academic.oup.com/cybersecurity
Multidisciplinary research on digital threats and AI-driven defensive mechanisms.

IEEE Security & Privacy Magazine
https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=8013
Blends academic depth with industry relevance; covers AI, forensics, and emerging cyber threats.

Cybersecurity Magazine https://cybersecurity-magazine.com Regularly features expert opinion, trends, and technologies including AI in cyber security. SWAYAM/NPTEL/MOOCs: NPTEL – Introduction to Machine Learning (IIT Kharagpur) https://onlinecourses.nptel.ac.in/noc22-cs58 AI for Cybersecurity Specialization – IBM https://www.coursera.org/specializations/ai-cybersecurity Covers: AI and ML techniques to detect malware, phishing, and threats in network traffic. Hands-on labs included.

Course Code: CAI3405	Course Title: Explainable AI Type of Course: Integrated	L- T-P- C	2	0	2	3
Version No.	1.0					
Course Pre-requisites	Essentials of AI					
Anti-requisites	NIL					
Course Description	This course introduces the concepts, techniques, and challenges of Explainable Artificial Intelligence (XAI). Students will learn to build interpretable models and apply explainability tools to demystify complex AI decisions, making AI systems transparent, trustworthy, and accountable. Emphasis will be on practical implementation and evaluation using real-world datasets.					
Course Objective	1 Understand the need for and principles of explainable AI 2 Explore techniques for explaining model predictions 3 Gain hands-on experience with state-of-the-art XAI tools and libraries 4 Build models that are interpretable and meet regulatory or ethical standards					

Course Outcomes	On successful completion of this course the students shall be able to: Explain the importance and scope of explainability in AI Compare interpretable models with black-box models Apply XAI techniques (e.g., LIME, SHAP) to real-world datasets Develop systems with enhanced transparency and traceability Evaluate explainability metrics and their impact on model trustworthiness			
Course Content:				
Module 1	Introduction to Explainable AI	Understand		13[7L+4P] Sessions
Topics: What is Explainability? Why it matters, Challenges in interpreting ML/DL models AI Ethics and Responsible AI				
Module 2	Interpretable Models vs. Black-box Models	Apply		14[7L+7P] Sessions
Topics: Decision Trees, Linear Models, Rule-based Models, Black-box models: Neural Networks, Ensemble methods, Trade-offs between accuracy and explainability				
Module 3	Post-Hoc Explanation Techniques	Assignment		14[6L+8P] Sessions
Topics: Local vs. Global explanations, LIME (Local Interpretable Model-Agnostic Explanations), SHAP (SHapley Additive exPlanations), Partial Dependence Plots, Feature Importance				
Module 4	Visual and Textual Explanations	Assignment	14[6L+8P] Sessions	
Saliency maps for CNNs, Attention mechanisms in NLP, Counterfactual and contrastive explanations				
Project work/Assignment:				

Assignment 1 on (Module 1 and Module 2) Assignment 2 on (Module 3)
List of Lab Tasks: Lab 1 – Compare interpretable vs. black-box models Lab 2 – Implement LIME for image/text classification Lab 3 – Apply SHAP to a random forest classifier Lab 4 – Visualize CNN saliency maps for image predictions Lab 5 – Use What-If Tool (TensorBoard) for exploring model fairness Lab 6 – Build a decision support tool using explainable outputs Lab 7 – Case Study: Explainability in credit scoring models Lab 8 – Final Project: Explainable AI dashboard for real-world data
REFERENCE MATERIALS: TEXTBOOKS Christoph Molnar – Interpretable Machine Learning, 2022 Edition (Free online) Sameer Singh et al. – Explainable AI: A Guide for Practitioners Gunning & Aha – DARPA’s XAI Program Publications REFERENCES IEEE XAI publications Research papers from NeurIPS, ICML, and ACL on XAI XAI Fairness & Bias Toolkits by Google, IBM, and Microsoft JOURNALS/MAGAZINES IEEE Transactions on Artificial Intelligence Journal of Artificial Intelligence Research (JAIR) ACM Transactions on Intelligent Systems and Technology (TIST) Artificial Intelligence Journal (Elsevier) SWAYAM/NPTEL/MOOCs: NPTEL: Responsible AI by IIT Madras Coursera: Explainable AI with Google Cloud FastAI: Modules on Model Interpretation

IBM AI Explainability 360 Toolkit

Course Code: CAI3406	Course Title: Responsible AI Type of Course: Integrated	L- T-P- C	2	0	2	3

Version No.		1.0
Course Pre-requisites		Essentials of AI
Anti-requisites		NIL
Course Description		Responsible AI emphasizes transparency and explainability, ensuring that AI-driven decisions are understandable and justifiable. It also prioritizes security, reliability, and sustainability, aiming to create AI systems that are safe, efficient, and environmentally conscious. Ultimately, Responsible AI seeks to align technology with human values, promoting trust and ensuring that AI enhances rather than harms society.
Course Objective		The objective of Responsible AI is to develop and deploy artificial intelligence in a way that is ethical, fair, transparent, and aligned with human values.
Course Out Comes		On successful completion of this course the students shall be able to: To state aspects of responsible AI such as fairness, accountability, bias, privacy etc.[Remember] To assess the fairness and ethics of AI models.[Understand] To enforce fairness in models and remove bias in data.[Understand] To preserve the privacy of individuals while learning from them and apply it to various domains.[Apply]
Course Content:		

Module 1	Introduction to Responsive AI (Remember)	Assignment			11 Sessions
	Topics: Artificial Intelligence Fundamentals, definition of responsible AI, Importance of responsible AI, core principles of responsible AI, Regulations and Policies, challenges, Responsible AI in practice.				
Module 2	Fairness and Bias (Understand)	Assignment			11 Sessions
	Topics:				

	Sources of Biases, Exploratory data analysis, limitation of a dataset, Preprocessing, in processing and postprocessing to remove bias, Group fairness and Individual fairness, Counterfactual fairness					
Module 3		Interpretability and explainability, Ethics and Accountability (Understand)	Assignment			12 sessions
	Topics: Interpretability through simplification and visualization, Intrinsic interpretable methods Post Hoc interpretability, Explainability through causality, Model agnostic Interpretation, Auditing AI models, fairness assessment, Principles for ethical practices					
Module 4		Privacy preservation (Apply)	Assignment			11 sessions
	Topics: Attack models, Privacy-preserving Learning, Differential privacy, Federated learning, Case Study- Recommendation systems, Medical diagnosis, Hiring/ Education, Computer Vision, Natural Language Processing					
	Lab Experiments: Lab 1 – Real-time sentiment analysis from live social media feed Lab 2 – Build an AI-powered chatbot using Dialogflow or Rasa Lab 3 – Create a user-adaptive recommendation engine Lab 4 – Implement online learning for a dynamic classification problem Lab 5 – Emotion recognition from facial expressions using webcam input Lab 6 – Deploy a low-latency AI model using TensorFlow Lite Lab 7 – Build a real-time fraud detection prototype using streaming data Lab 8 – Mini Project: End-to-end responsive AI application					
	Targeted Application & Tools that can be used: ChatGPT, DeepSeek					
	Project work/Assignment: Mention the Type of Project /Assignment proposed for this course					
	Case Study in different domains					
	Text Book Virginia Dignum, “Responsible Artificial Intelligence: How to Develop and Use AI in a Responsible Way” Springer Nature, 04-Nov-2019;ISBN-10 : 3030303705, ISBN-13 : 978-3030303709					

	Christoph Molnar "Interpretable Machine Learning".Lulu, 1st edition, March 24, 2019; eBook. ISBN-10 : 0244768528, ISBN-13 : 978-0244768522 [available online]
	References R1. Voeneky S, Kellmeyer P, Mueller O, Burgard W, eds. The Cambridge Handbook of Responsible Artificial Intelligence. In: The Cambridge Handbook of Responsible Artificial Intelligence: Interdisciplinary Perspectives. Cambridge Law Handbooks. Cambridge University Press; 2022:i-ii. Web links W1. Responsible AI for generative models: Designing for responsibility W2. Responsible AI W3. Microsoft Responsible AI - Fairness
	Topics relevant to development of "Employability": Responsible AI ethics, Fairness and Bias, ethics and accountability

Course Code: CAI3407	Course Title: Agentic AI Type of Course: Integrated	L- T-P- C	2	0	2	3
Version No.	1.0					
Course Pre-requisites	Essentials of AI					
Anti-requisites	NIL					
Course Description	This course explores Agentic AI—AI systems that act autonomously with the ability to perceive, reason, and make decisions in complex environments. Students will learn about intelligent agents, multi-agent systems, autonomous planning, and goal-driven behavior, with a focus on both theory and real-world applications.					
Course Objective	1 Understand the architecture and behavior of autonomous intelligent agents 2 Design agents capable of interacting with environments and other agents 3 Explore decision-making, planning, and coordination in agent-based systems 4 Implement and evaluate simple agentic AI systems					

Course Outcomes	On successful completion of this course the students shall be able to: Describe agent architectures and types of intelligent agents Apply decision-making strategies for autonomous goal-directed agents Develop agents capable of environmental interaction and adaptation Build multi-agent systems with basic coordination and communication Analyze the ethical and societal impact of autonomous agentic systems			
Course Content:				
Module 1	Introduction to Intelligent Agents	Understand		13[7L+4P] Sessions
Topics: What is an agent? Reactive vs. deliberative agents, Perception, reasoning, and action loop, Environment types and agent performance				
Module 2	Agent Architectures and Planning and Decision Making	Apply		14[7L+7P] Sessions
Topics: Simple reflex agents, Goal-based and utility-based agents, Layered architectures (e.g., subsumption, BDI models), Goal formulation Search and planning algorithms (A*, STRIPS, etc.), Markov Decision Processes (MDPs), utility theory				
Module 3	Multi-Agent Systems	Assignment		14[6L+8P] Sessions
Topics: Communication and coordination among agents, Distributed problem-solving, Game theory basics, negotiation and cooperation				
Module 4	Agentic AI in Practice	Assignment	14[6L+8P] Sessions	
Agents in robotics, simulations, and digital environments, Autonomous vehicles, virtual assistants, and game agents,				

Ethics, safety, and alignment in agentic AI
Project work/Assignment:
Assignment 1 on (Module 1 and Module 2) Assignment 2 on (Module 3)
List of Lab Tasks: Lab 1 – Implement a simple reflex agent in Python Lab 2 – Design a goal-based agent with planning capability Lab 3 – Simulate MDPs for agent decision-making Lab 4 – Build a multi-agent gridworld simulation Lab 5 – Agent communication using JSON messages Lab 6 – Agent negotiation using basic game-theory Lab 7 – Develop a smart assistant with agentic behavior Lab 8 – Mini Project: Autonomous agent in a dynamic environment (e.g., a search-and-rescue sim)
REFERENCE MATERIALS: TEXTBOOKS Stuart Russell & Peter Norvig, Artificial Intelligence: A Modern Approach, 4th Ed. Michael Wooldridge, An Introduction to MultiAgent Systems, 2nd Ed. Gerhard Weiss, Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence REFERENCES AI research papers from IJCAI, AAAI, AAMAS OpenAI research on agentic models and autonomous systems Case studies on autonomous robotics and virtual agent behavior JOURNALS/MAGAZINES Autonomous Agents and Multi-Agent Systems (Springer) Journal of Artificial Intelligence Research (JAIR) Artificial Intelligence Journal (Elsevier) IEEE Transactions on Cognitive and Developmental Systems SWAYAM/NPTEL/MOOCs: NPTEL: Artificial Intelligence – Search Methods for Problem Solving

Coursera: Autonomous Agents – University of Alberta edX: Multi-Agent Systems and Distributed AI OpenAI Blog: Research articles on emerging agentic models (e.g., AutoGPT)

Course Code: CAI3408	Course Title: Deep Neural Networks Type of Course: Integrated		L- T-P- C	2	0	2	3
Version No.	1.0						
Course Pre-requisites							
Anti-requisites	NIL						
Course Description	This course introduces students to the concepts of deep neural networks and state of the art approaches to develop deep learning models. In this course students will be given an exposure to the details of neural networks as well as deep learning architectures and to develop end-to-end models for such tasks. It will help to design and develop an application-specific deep learning models and also provide the practical knowledge handling and analyzing end user realistic applications. Topics include Fundamental concepts of deep neural networks, Convolutional Neural Networks, Recurrent Network structures, Deep Unsupervised Learning, Generative Adversarial Networks and applications in various problem domains.						
Course Objective	This course is designed to improve the learners EMPLOYABILITY SKILLS by using EXPERIENTIAL LEARNING techniques.						
Course Outcomes	On successful completion of this course the students shall be able to: CO1: Apply Convolution Neural Network for image processing. CO2: Understand the basics of associative memory and unsupervised learning networks. CO3: Apply CNN and its variants for suitable applications. CO4: Analyze the key computations underlying deep learning and use them to build and train deep neural networks for various tasks. CO5: Apply autoencoders and generative models for suitable applications.						
Course Content:							
Module 1	INTRODUCTION	Assignment					13[7L+6P] Sessions

<table>
<tr><td colspan="5">Topics:

Neural Networks-Application Scope of Neural Networks-Artificial Neural Network: An Introduction Evolution of Neural Networks-Basic Models of Artificial Neural Network-Important Terminologies of ANNs-Supervised Learning Network.</td></tr>
<tr><td>Module 2</td><td>ASSOCIATIVE MEMORY AND UNSUPERVISED LEARNING NETWORKS</td><td>Assignment</td><td></td><td>18[8L+10P]

Sessions</td></tr>
<tr><td colspan="5">Topics:

Training Algorithms for Pattern Association-Autoassociative Memory Network-Heteroassociative Memory Network-Bidirectional Associative Memory (BAM)-Hopfield Networks-Iterative Autoassociative Memory Networks-Temporal Associative Memory Network-Fixed Weight Competitive Nets-Kohonen Self-Organizing Feature Maps-Learning Vector Quantization-Counter propagation Networks-Adaptive Resonance Theory Network.</td></tr>
<tr><td>Module 3</td><td>THIRD-GENERATION NEURAL NETWORKS</td><td>Assignment</td><td></td><td>16[8L+8P]

Sessions</td></tr>
<tr><td colspan="5">Topics:

Spiking Neural Networks-Convolutional Neural Networks-Deep Learning Neural Networks-Extreme Learning Machine Model-Convolutional Neural Networks: The Convolution Operation – Motivation – Pooling – Variants of the basic Convolution Function – Structured Outputs – Data Types – Efficient Convolution Algorithms – Neuroscientific Basis – Applications: Computer Vision, Image Generation, Image Compression.</td></tr>
<tr><td>Module-4</td><td>DEEP FEEDFORWARD NETWORKS</td><td>Assignment</td><td></td><td>13[7L+6P]

Sessions</td></tr>
<tr><td colspan="5">Topics:

History of Deep Learning- A Probabilistic Theory of Deep Learning- Gradient Learning – Chain Rule and Backpropagation - Regularization: Dataset Augmentation – Noise Robustness -Early Stopping, Bagging and Dropout - batch normalization- VC Dimension and Neural Nets.</td></tr>
<tr><td>Module-5</td><td>RECURRENT NEURAL NETWORKS</td><td>Assignment</td><td></td><td>13[7L+6P]

Sessions</td></tr>
<tr><td colspan="5">Recurrent Neural Networks: Introduction – Recursive Neural Networks – Bidirectional RNNs – Deep Recurrent Networks – Applications: Image Generation, Image Compression, Natural Language Processing. Complete Auto encoder, Regularized Autoencoder, Stochastic Encoders and Decoders, Contractive Encoders.</td></tr>
</table>

Project work/Assignment:
Assignment 1 on (Module 1 and Module 2) Assignment 2 on (Module 3 and Module 4)
List of Lab Tasks: Lab 1: Working with Deep Learning Frameworks Objective: Explore various Deep Learning Frameworks Tasks: Identify deep learning frameworks (Keras, Tensorflow, Matplotlib, etc) Activity: Practice with various methods available in DL Frameworks to develop a Model. Lab 2: Build a Basic Artificial Neural Network Objective: Create a ANN with DL frameworks. Task: Identify suitable ANN Layers using Keras and Tensorflow. Activity: Design a basic Artificial Neural Networks using Keras with TensorFlow (pima-indians-diabetes) Lab 3 and Lab 4: Build a MultiLayer Perceptron Objective: Create a MLP for classification task. Task: Identify suitable model for house price prediction. Activity: Design a MLP for implementing classification and fine-tuning using House price.csv Lab 5: Build a Convolutional Neural Network Objective: Create a CNN model. Task: Build CNN architecture for Dog-Cat classification problem. Activity: Implement a Convolution Neural Network (CNN) for dog/cat classification problem using keras Lab 6 and Lab 7: Build a Time-Series Model Objective: Create a RNN and LSTM Model Task: Build RNN/LSTM Model for predicting time series data. Activity Train a sentiment analysis model on IMDB dataset, use RNN layers with LSTM/GRU notes

Lab 8: Build a Gated Recurrent Unit architecture.

Objective: Create a Time Series Model.

Task: Build GRU Architecture for predicting time series data.

Activity: Implement a GRU architecture for language translations.

Lab 9 and Lab 10: Build a Transfer Learning Model.

Objective: Create a Seq2Seq Model

Task: Create Hugging-face API using Transfer learning model.

Activity: Implement Transfer Learning models for classification problems Exploring Hugging-face API

Lab 11: Build an Auto-Encoder model

Objective: Create an Unsupervised Deep Learning Model.

Task: Create AutoEncoder network Output Translations.

Activity: implement an Encoder-Decoder Recurrent neural network model for Neural Machine Translation.

Lab 12: Build Generative Adversarial Networks.

Objective: Create an Unsupervised Deep Learning Model.

Task: Design GAN Architecture for Image generations.

Activity: Design a Age Prediction model by Applying Generative Adversarial

REFERENCE MATERIALS:

TEXT BOOKS:

1. Ian Goodfellow, Yoshua Bengio, Aaron Courville, “Deep Learning”, MIT Press, 2016.

2. Francois Chollet, “Deep Learning with Python”, Second Edition, Manning Publications, 2021. 106

REFERENCES:

1. Aurélien Géron, “Hands-On Machine Learning with Scikit-Learn and TensorFlow”, Oreilly, 2018.

2. Josh Patterson, Adam Gibson, “Deep Learning: A Practitioner’s Approach”, O’Reilly Media, 2017.

3. Charu C. Aggarwal, “Neural Networks and Deep Learning: A Textbook”, Springer International Publishing, 1st Edition, 2018.

4. Learn Keras for Deep Neural Networks, Jojo Moolayil, Apress,2018

5. Deep Learning Projects Using TensorFlow 2, Vinita Silaparasetty, Apress, 2020

6. Deep Learning with Python, FRANÇOIS CHOLLET, MANNING SHELTER ISLAND,2017. 7. S Rajasekaran, G A Vijayalakshmi Pai, “Neural Networks, FuzzyLogic and Genetic Algorithm, Synthesis and Applications”, PHI Learning, 2017. 8. Pro Deep Learning with TensorFlow, Santanu Pattanayak, Apress,2017 9. James A Freeman, David M S Kapura, “Neural Networks Algorithms, Applications, and Programming Techniques”, Addison Wesley, 2003. JOURNALS/MAGAZINES IEEE Transactions on Neural Networks and Learning Systems https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=5962385 IEEE Transactions on Pattern Analysis and Machine Intelligence https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=34http://ijaerd.com/papers/special_papers/IT032.pdf International Journal of Intelligent Systems https://onlinelibrary.wiley.com/journal/1098111x SWAYAM/NPTEL/MOOCs: Swayam Nptel – Deep Learning – IIT Ropar https://onlinecourses.nptel.ac.in/noc21_cs35/preview Coursera – Neural Networks and Deep Learning Andrew Ng Coursera - Neural Networks for Machine Learning by Geoffrey Hinton in Coursera

Course Code: CAI3409	Course Title: Speech Recognition and Synthesis Type of Course: Integrated	L- T-P- C	2	0	2	3
Version No.	1.0					
Course Pre-requisites	Essentials of AI					
Anti-requisites	NIL					

<table>
<tr><td>Course Description</td><td colspan="4">This course introduces fundamental principles and practical aspects of speech signal processing. It covers techniques in speech analysis, recognition, and synthesis, enabling students to build interactive voice-based AI systems. Emphasis is placed on acoustic modeling, feature extraction, and the use of machine learning and deep learning models for speech applications.</td></tr>
<tr><td>Course Objective</td><td colspan="4">To understand the basic concepts and characteristics of speech signals.
To explore feature extraction and pattern matching techniques used in ASR.
To study the principles of speech synthesis and TTS systems.
To provide hands-on experience with speech processing tools and APIs.</td></tr>
<tr><td>Course Outcomes</td><td colspan="4">On successful completion of this course, students will be able to:
Analyze speech signals and extract key features.
Apply pattern recognition and machine learning methods for ASR.
Design speech synthesis systems using classical and deep learning models.
Develop real-time applications using speech APIs and open-source tools.</td></tr>
<tr><td colspan="5">Course Content:</td></tr>
<tr><td>Module 1</td><td>Speech Signal Fundamentals</td><td>Assignment</td><td></td><td>18[8L+10P]
Sessions</td></tr>
<tr><td colspan="5">Topics:
Human speech production and auditory perception, Speech signal representation: time and frequency domains, Preprocessing: sampling, quantization, windowing, pre-emphasis, Speech signal features: pitch, formants, energy, ZCR, Spectrogram and Short-Time Fourier Transform (STFT).</td></tr>
<tr><td>Module 2</td><td>Feature Extraction & Modeling Techniques</td><td>Assignment</td><td></td><td>14[7L+7P]
Sessions</td></tr>
<tr><td colspan="5">Topics:
MFCC, PLP, LPC, Delta features, DTW (Dynamic Time Warping), GMM-HMM based acoustic modelling, Viterbi decoding and alignment, Basics of phonetics and phoneme modelling</td></tr>
<tr><td>Module 3</td><td>Automatic Speech Recognition (ASR)</td><td>Assignment</td><td></td><td>14[6L+8P]
Sessions</td></tr>
<tr><td colspan="5">Topics:
Architecture of ASR systems, Language modeling: N-grams, smoothing, Deep learning for speech: DNN, CNN, LSTM, End-to-end models: CTC, Attention, Transformers, Tools: Kaldi, CMU Sphinx, DeepSpeech.</td></tr>
</table>

Module 4	Speech Synthesis and TTS	Assignment		14[6L+8P] Sessions

Topics:

Concatenative, Parametric, and Neural synthesis, TTS pipeline: text normalization, phoneme mapping, HMM-based synthesis, Tacotron and WaveNet architectures, APIs: Google TTS, Amazon Polly, Festival, eSpeak.

List of Lab Tasks:

Lab 1: Introduction to Speech Signal Processing

Objective: Understand the nature of speech signals.

Task: Record and visualize a speech waveform using a microphone and Python.

Activity: Capture audio input, analyze waveform, and visualize with matplotlib.

Lab 2: Time Domain Analysis of Speech Signals

Objective: Analyze speech features in the time domain.

Task: Compute Zero Crossing Rate (ZCR) and short-time energy of speech samples.

Activity: Implement ZCR and energy plots for different speech segments using Python.

Lab 3: Frequency Domain Analysis Using Spectrogram

Objective: Understand frequency components of speech.

Task: Apply Short-Time Fourier Transform (STFT) to speech signals.

Activity: Generate and interpret spectrograms using Librosa and Matplotlib.

Lab 4: Extraction of MFCC Features

Objective: Extract key speech features using MFCC.

Task: Extract and visualize MFCC features from recorded speech.

Activity: Use Librosa or python_speech_features to extract MFCCs.

Lab 5: Linear Predictive Coding (LPC)

Objective: Implement LPC-based feature extraction.

Task: Apply LPC to model the vocal tract.

Activity: Compute LPC coefficients and analyze spectral envelope.

Lab 6: Implement Dynamic Time Warping (DTW)

Objective: Understand time alignment of speech.

Task: Compare two speech signals using DTW.

Activity: Use fastdtw to align utterances of the same word.

Lab 7: Acoustic Modeling using GMM-HMM

Objective: Implement basic acoustic modeling.

Task: Train a GMM-HMM model for a simple digit recognition task.

Activity: Use hmmlearn or HTK for acoustic modeling.

Lab 8: Viterbi Algorithm for Speech Alignment

Objective: Apply sequence decoding in speech.

Task: Use Viterbi decoding to find the best state path in an HMM.

Activity: Implement Viterbi algorithm in Python and visualize state paths.

Lab 9: Build a Simple ASR System using CMU Sphinx

Objective: Use open-source ASR toolkit.

Task: Set up CMU Sphinx and perform isolated word recognition.

Activity: Create language and acoustic models and test recognition.

Lab 10: Real-time Speech Recognition using DeepSpeech

Objective: Apply deep learning in ASR.

Task: Use Mozilla DeepSpeech to transcribe audio files.

Activity: Install DeepSpeech, load pre-trained model, and test transcription.

Lab 11: Evaluate ASR using Google Speech-to-Text API

Objective: Explore cloud-based speech recognition.

Task: Use Google Cloud API to transcribe different accents.

Activity: Send audio to API and analyze word error rate.

Lab 12: Introduction to Text-to-Speech (TTS) using Festival

Objective: Create basic speech synthesis output.

Task: Convert written text to speech using Festival.

Activity: Install Festival, input text, and generate synthesized speech.

Lab 13: Implement a Neural TTS system using Tacotron 2

Objective: Use deep learning for speech synthesis.

Task: Synthesize speech from text using Tacotron 2.

Activity: Use pre-trained Tacotron 2 model and generate speech samples.

Lab 14: Speech Synthesis using Google TTS and Amazon Polly

Objective: Use cloud-based TTS systems.

Task: Convert sample sentences into speech using APIs.

Activity: Use Python SDKs to access TTS services and compare outputs.

Lab 15: Mini Project – Build a Voice Assistant

Objective: Integrate ASR and TTS.

Task: Build a simple voice-controlled assistant.

Activity: Combine speech recognition, intent detection, and speech synthesis using tools like SpeechRecognition + gTTS + Pyttsx3.

REFERENCE MATERIALS

TEXTBOOKS

Lawrence Rabiner and Ronald Schafer, Theory and Applications of Digital Speech Processing, Pearson, 2011.

Daniel Jurafsky and James H. Martin, Speech and Language Processing, Pearson, 3rd Edition (Draft).
https://web.stanford.edu/~jurafsky/slp3/

REFERENCE BOOKS

Ben Gold and Nelson Morgan, Speech and Audio Signal Processing, Wiley, 2nd Edition, 2011.

Thomas Dutoit, An Introduction to Text-to-Speech Synthesis, Springer, 1997.

Xuedong Huang, Alex Acero, Hsiao-Wuen Hon, Spoken Language Processing, Prentice Hall, 2001.

Tokuda et al., Speech Synthesis: Algorithms for Text-to-Speech and Speech Modification, Springer, 2021.

JOURNALS / MAGAZINES

IEEE Transactions on Audio, Speech and Language Processing
https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=10376

Computer Speech & Language (Elsevier) https://www.sciencedirect.com/journal/computer-speech-and-language Speech Communication (Elsevier) https://www.journals.elsevier.com/speech-communication SWAYAM / NPTEL / MOOCs NPTEL – Spoken Language Processing (IIT Madras) https://nptel.ac.in/courses/106106168 Coursera – Audio Signal Processing for Music Applications (UPF) https://www.coursera.org/learn/audio-signal-processing edX – Speech Processing (KTH Royal Institute of Technology) https://www.edx.org/course/speech-processing Fast.ai – Practical Deep Learning (TTS + ASR projects) https://course.fast.ai/

Course Code: CAI3410	Course Title: AI Chatbots without Programming Type of Course: Integrated	L- T-P- C	2	0	2	3
Version No.	1.0					
Course Pre-requisites	Essesntails Of AI					
Anti-requisites	NIL					
Course Description	This course enables students to design and develop intelligent chatbots using no-code/low-code platforms. It covers the fundamentals of natural language processing, conversational AI, and chatbot design strategies. Students will gain hands-on experience building functional chatbots for domains such as customer service, education, and e-commerce using tools like Dialogflow, Chatfuel, Microsoft Power Virtual Agents, and other no-code frameworks.					
Course Objective	This course aims to democratize chatbot development by empowering students to create intelligent conversational agents without programming knowledge, using experiential learning on visual platforms.					

Course Outcomes	Course Outcomes On successful completion of this course, students will be able to: Understand the architecture and functioning of AI-powered chatbots. (Understand) Design conversational flows using drag-and-drop tools. (Apply) Analyze user intent and train NLP models using no-code tools. (Analyze) Create and deploy domain-specific chatbots for real-world use cases. (Create)			
Course Content:				
Module 1	Introduction to Conversational AI and Chatbots	Assignment		18[8L+10P] Sessions
Topics: History and evolution of chatbots,Types of chatbots: rule-based vs. AI-based,Components of chatbot architecture,Use cases across industries				
Module 2	Designing Conversational Experiences	Assignment		14[7L+7P] Sessions
Topics: Conversation design principles, User personas and intent mapping, Flowcharting and decision trees, Multilingual and accessibility considerations				
Module 3	Building AI Chatbots on No-Code Platforms	Assignment		14[6L+8P] Sessions
Topics: Overview of no-code tools: Dialogflow, Chatfuel, Landbot, Microsoft PVA,NLP basics: intents, entities, training phrases, Contexts and session handling, Integrations with messaging platforms (WhatsApp, Telegram, Web)				
Module 4	Testing, Deployment, and Analytics	Assignment		14[6L+8P] Sessions
Topics: Bot testing and improvement strategies, Connecting to APIs and databases, Deployment to websites and social media, Analyzing user behavior and feedback.				

Project work/Assignment:
Design a customer support chatbot using Chatfuel or Dialogflow Develop a feedback collection chatbot for educational use Group project: Cross-platform chatbot for a chosen domain
List of Lab Tasks: Lab 1: Introduction to chatbot interfaces and no-code tools Lab 2: Create a simple rule-based chatbot using Chatfuel Lab 3: Design a user flow using decision trees in Landbot Lab 4: Build an FAQ chatbot using Dialogflow intents and responses Lab 5: Implement intent recognition and entity extraction in Dialogflow Lab 6: Add context-based conversations in a Dialogflow chatbot Lab 7: Build a WhatsApp-integrated chatbot using Twilio Lab 8: Design a chatbot using Microsoft Power Virtual Agents Lab 9: Create multi-lingual responses and fallback messages Lab 10: Integrate a chatbot with Google Sheets to log user responses Lab 11: Embed a chatbot on a website using iframe or script Lab 12: Analyze user interaction logs for performance metrics Lab 13: Customize chatbot appearance and branding elements Lab 14: Develop a feedback chatbot with sentiment-based responses Lab 15: Final project: Design and deploy a fully functional domain-specific chatbot
REFERENCE MATERIALS TEXTBOOKS Jason D. Brown, Designing Bots: Creating Conversational Experiences, O'Reilly Media, 2017. Akshay Kulkarni and Adarsha Shivananda, Building Chatbots with Google Dialogflow, Apress, 2019. REFERENCES Rashid Khan, Build Better Chatbots: A Complete Guide to Getting Started with Chatbots, Apress, 2017. Michael McTear, Conversational AI: Dialogue Systems, Conversational Agents, and Chatbots, Springer, 2020.

Navin Sabharwal et al., Designing Human-Centric AI Experiences, Apress, 2021. JOURNALS / MAGAZINES ACM Transactions on Interactive Intelligent Systems AI Magazine (Association for the Advancement of Artificial Intelligence) Journal of Web Engineering SWAYAM/NPTEL/MOOCs Coursera – Building AI Powered Chatbots Without Programming (IBM) edX – Introduction to Chatbots (Microsoft) Udemy – Chatbot Development Without Coding

Course Code: CAI2506	Course Title: Generative AI Type of Course: Integrated		L-T-P-C	2	0	2	3
Version No.		1.0					
Course Pre-requisites		CSE1700 Essentials of AI					
Anti-requisites		NIL					
Course Description		This course builds the foundational insight of understanding generative AI models and to explore various architectures, algorithms and practices of Gen AI skills to accelerate strategic decision making with data and deliver cutting-edge products faster with GenAI-augmented software development and leverage Gen AI tools to optimize workflows.					
Course Objective		The objective of the course is to familiarize the learners to explore the competence in benchmarking and comprehend the potential generative AI models and techniques to revolutionize industries and create prominent Gen AI tools to attain Employability Skills through Experiential Learning techniques.					
Course Out Comes		On successful completion of the course the students shall be able to: CO 1: Infer the concepts of generative AI models and prompt engineering in tailoring customized outputs [Understand]. CO 2: Demonstrate attention mechanism and transformers architecture with practical Applications. [Apply]. CO 3: Practice advanced generative AI techniques using Langchain Python framework [Apply].					

		CO 4: Solve real-time applications using multi-modal generative AI models [Apply].			
Course Content:					
Module 1	Introduction to Generative AI	Participative Learning		Brainstorming session/Quiz	No. of classes L-6 P-8
	Topics: Introduction to Generative models: Historical perspective and evolution, Applications, Types of Generative models for different data modalities, Large Language Models (LLMs) – Introduction, evolution, Generative pre-trained transformers (GPT) and its variants, Google DeepMind's, PaLM2, LLaMa and its series of models by Meta AI, Claud and its variants by Anthropic, Prompt Engineering-basic prompting.				
Module 2	Text-based Generative models	Participative Learning		Fish bowl, Think-pair & share	No. of classes L-8 P-6
	Topics: Text-based Generative models: State-of-the Art models, RNN, LSTM, Transformer Architecture, Transformer based Generative models: BERT, GPT, Training and Fine tuning LLMs for Generative task, Open AI's Pre-trained transformers for Text Generation: ChatGPTs, Limitations of LLMs: Lack of context and Hallucination risks, Techniques to mitigate these limitations: chaining and retrieval augmentation, Workflow of an LLM application.				
Module 3	Introduction to Lang Chain	Experiential Learning		Implementation of Gen AI models using Langchain Framework	No. of classes L-8 P-8
	Topics: Introduction to Lang chain: Types, Components, Information retrieval using agents and tools in Lang chain, Retrieval Augmented Language Models (RaLM): Understanding Retrieval and vectors: Embeddings, Vector storage, Vector indexing, Vector Libraries, Vector Databases, Chatbot using memory and conversation buffer.				
Module 4	Generative models for other Data modalities	Project-based Learning		Multi-Modal Gen AI models for Realtime Applications	No. of classes L-8 P-8
	Topics: Generative Adversarial Networks (GAN): GAN Architecture, GAN variants, Neural Style transfer with GAN, Training GANs and common challenges, GAN applications in image and text generation, Variational Auto Encoders (VAEs) and its variants, Image generation models: Dall-E, MidJourney and stable diffusion: Architecture and components of stable diffusion, Text-to-image Generation, Parameter tuning, Image-to-image generation, Training custom models, In-Painting: Exchanging				

	classes, Multi-modal generative models using Whisper for Audio: Speech-to-Text generation.
	Project work/Assignment:
	Certification course in Generative AI through Google Mini Project on (Module 3 and Module 4)
	Targeted Application & Tools that can be used Open AI Generative AI models: GPT 3.5 Turbo, GPT 4.0 vision model, Dall-E 3.0, Lang Chain Framework in Python, Python IDE, Stable Diffusion, Gemini, Hugging Face,
	List of Lab Tasks:
	Experiment No.1: Setting up Python IDE(Spyder) and OpenAI API key. Introduction to OpenAI playground and prompting Level 1: Document the installation and the process for generating models in OpenAI Level 2: Solve various GenAI models of OpenAI from Playground using prompts
	Experiment No.2: Text classification, summarization, sentiment analysis, chatbot application, code explanation with generating single and multiple response(S). Level 1: Practice the text generation model of OpenAI and Spyder IDE to implement various applications.
	Experiment No.3: Embeddings – for words, similarity between words, text embeddings, plagiarism check of documents Level 1: Use generating embeddings for words, text and documents Level 2: Apply the embeddings API to develop applications for plagiarism check
	Experiment No.4: Image generation using Dall E. Using GPT-Vision model for text to image generation and image-to-text. Level 1: Apply GPT-vision model for text-to-image generation and image-to-image
	Experiment No.5: Transformer based text and email classification Level 1: Develop transformer-based AI models for classifying text/email
	Experiment No.6: BERT for masked token generation Level 1: Develop BERT based model for generating masked tokens
	Experiment No.7: Creating applications using different types of LangChains – Simple Sequential, Sequential and map reduce Level 1: List the various types of chains in Langchain Level 2: Practice different types of chains using Spyder IDE and OpenAI

	Experiment No.8: Information retrieval using agents and tools in Langchain. Level 1: Use agents and tools with Langchain for information retrieval
	Experiment No.9: Custom Document loading and retrieval in LangChain using ChromaDB Level 1: Understand ChromeDb Level 2: Apply chromed with Langchain to generate information retrieval model from custom document
	Experiment No.10: Create a GPT like Chatbot using the memory component and RALM in LangChain Level 1: Show GPT like chatbot using memory component and retrieval augmented language model
	Experiment No.11: Using action agents, human as a tool and plan and execute agents for information retrieval. Level 1: Understand action agents and plan and execute agents Level 2: Use agents and tools for information retrieval
	Experiment No.12: Implement GAN for neural style transfer Level 1: Demonstrate a style transfer algorithm using generative models and experiment with the transformation of images by applying different artistic styles, assessing both the technical aspects and the aesthetic outcomes
	Experiment No.13: Text to Image generation using Dall-e/stable diffusion using prompts Level 1: List various image generation models Level 2: Use an image generation model to generate image from prompts
	Experiment No.14: Image to Image generation using stable diffusion Level 1: Apply stable diffusion to generate image from an image using prompts
	Experiment No.15: Speech to text and multi-modal generative models using Whisper for Audio Level 1: Identify the generative model for text, image and audio data Level 2: Use Langchain to create models for generating different data modalities. Ex: Audio-to-text
	TEXT BOOKS: T1: Generative AI with LangChain, Ist Edition by Ben Auffarth, Packt. Inc. ISBN: 978-1-83508-346-8, Decemeber 2023. T2: Generative Deep Learning, 2nd Edition by David Foster, O'Reilly Media, Inc. ISBN: 9781098134181, May 2023. T3: Prompt Engineering for Generative AI, by James Phoenix, Mike Taylor, O'Reilly Media, Inc., ISBN:9781098153373, July 2024.

	REFERENCE BOOKS: R1. Bandi, A., Adapa, P. V. S. R., & Kuchi, Y. E. V. P. K. (2023). The power of Generative AI: a review of requirements, models, Input–Output formats, evaluation metrics, and challenges. Future Internet, 15(8), 260. https://doi.org/10.3390/fi15080260 R2. Barachini, F., & Stary, C. (2022). From digital twins to digital selves and beyond. In Springer eBooks. https://doi.org/10.1007/978-3-030-96412-2 R3. Hadi, M. U., Tashi, Q. A., Qureshi, R., Shah, A., Muneer, A., Irfan, M., Zafar, A., Shaikh, M. B., Akhtar, N., Wu, J., & Mirjalili, R4. S. (2023). Large Language Models: A Comprehensive Survey of its Applications, Challenges, Limitations, and Future Prospects. https://doi.org/10.36227/techrxiv.23589741.v4 R4. Hai-Jew, S. (n.d.). Generative AI in Teaching and Learning. IGI Global. R5. Salvaris, M., Dean, D., & Tok, W. H. (2018). Generative adversarial networks. In Apress eBooks (pp. 187–208). https://doi.org/10.1007/978-1-4842-3679-6_8
	MOOC's/Swayam Courses/Online Courses: h https://onlinecourses.swayam2.ac.in/imb24_mg116/preview Certification Course by Google : 1. https://www.cloudskillsboost.google Introduction to Generative AI (Beginner) Gemini for Google Cloud (Intermediate) Generative AI for Developers (Advanced) 2. https://www.credly.com/badges/90e3eae0-87f3-44e3-af82-658e837aad3d/public_url 3. https://www.coursera.org/learn/generative-ai-with-llms 4. https://www.coursera.org/specializations/prompt-engineering ONLINE RESOURCES: W1. https://openai.com W2: https://python.langchain.com/v0.2/docs/introduction/ W3: https://www.udemy.com/course/master-ai-image-generation-using-stable-diffusion/?kw=Image+generation+using&src=sac&couponCode=LETSLEARNNOWPP W4: https://huggingface.co/google-t5/t5-base W5: https://dominguezdaniel.medium.com/exploring-image-generative-ai-models-9359705b15d3

	W6: https://cloud.google.com/use-cases/retrieval-augmented-generation?hl=en# W7: https://ig.ft.com/generative-ai/ W8: https://medium.com/@samia.khalid/bert-explained-a-complete-guide-with-theory-and-tutorial-3ac9ebc8fa7c
	Topics relevant to "EMPLOYABILITY SKILLS": Topics of all four modules will help in developing Employability Skills through Experiential Learning techniques. This is attained through assessment component mentioned in course handout.

Course Code: CAI3412	Course Title: Machine Learning for Finance Type of Course: Integrated	L- T-P- C	2	0	2	3
Version No.	1.0					
Course Pre-requisites	Essentials of AI					
Anti-requisites	NIL					
Course Description	This course explores the intersection of machine learning and finance. Students will learn to model financial data, predict market trends, manage risk, and develop algorithmic trading strategies using ML techniques. It emphasizes hands-on experience with financial datasets and real-world case studies.					
Course Objective	To enable students to apply machine learning methods to financial data analysis, portfolio optimization, and risk modeling. Students will gain practical knowledge to build ML-based financial systems using Python and related libraries.					
Course Outcomes	On successful completion of this course the students shall be able to: 1. Understand machine learning applications in the financial domain. (Understand) 2. Apply supervised and unsupervised learning to model financial data. (Apply) 3. Analyze risk and optimize portfolios using ML models. (Analyze) 4. Design and implement algorithmic trading strategies. (Create)					
Course Content:						
Module 1	Introduction to Financial Data and ML	Assignment 1			18[8L+10P] Sessions	

Topics: Overview of financial markets, financial data types, returns and risk, ML in finance, financial time-series analysis, introduction to supervised learning models used in finance.			
Module 2	Financial Forecasting and Risk Modeling	Assignment 1	14[7L+7P] Sessions
Topics: Time series forecasting (ARIMA, LSTM), volatility modeling, Value at Risk (VaR), credit risk modeling, regression and classification models for default prediction.			
Module 3	Portfolio Optimization and Strategy Design	Assignment 2	14[6L+8P] Sessions
Topics: Portfolio theory, efficient frontier, ML for asset allocation, reinforcement learning in portfolio management, backtesting trading strategies.			
Module 4	Algorithmic Trading and Fraud Detection		14[6L+8P] Sessions
Topics: Overview of algorithmic trading, strategy development using ML, anomaly detection for fraud, real-time data analysis, regulatory and ethical considerations.			
Project work/Assignment:			
Assignment 1 on (Module 1 and Module 2) Assignment 2 on (Module 3)			
List of Lab Tasks: 1. Load and analyze historical stock prices using pandas and yfinance. 2. Visualize financial time-series data and calculate returns. 3. Implement linear regression for stock price forecasting. 4. Build logistic regression model to predict credit default. 5. Use Random Forests to classify high-risk vs low-risk customers. 6. Implement ARIMA model for time-series forecasting. 7. Use LSTM model to predict stock prices with Keras/TensorFlow. 8. Apply K-Means clustering on customer transaction data.			

9. Calculate and visualize Value at Risk (VaR) for a portfolio. 10. Optimize asset allocation using ML-based mean-variance optimization. 11. Backtest a trading strategy using historical stock data. 12. Implement reinforcement learning for portfolio management. 13. Detect financial fraud using anomaly detection techniques. 14. Build a real-time financial dashboard using Streamlit. 15. Mini-project: Develop and evaluate an ML model for a selected finance use case.
REFERENCE MATERIALS: TEXTBOOKS 1. Yves Hilpisch, “Python for Finance: Mastering Data-Driven Finance”, O’Reilly Media, 2nd Edition, 2018. 2. Marcos Lopez de Prado, “Advances in Financial Machine Learning”, Wiley, 2018. REFERENCES 1. Tucker Balch, “Machine Learning for Trading”, Udacity course materials. 2. David Aronson, “Evidence-Based Technical Analysis”, Wiley, 2006. 3. E. Tsang, “Foundations of Computational Finance with Machine Learning”, Springer, 2021 JOURNALS/MAGAZINES 1. Journal of Financial Data Science 2. Quantitative Finance 3. Journal of Computational Finance SWAYAM/NPTEL/MOOCs: 1. Machine Learning for Engineering and Science Applications 2. Artificial Intelligence: Search Methods for Problem Solving 3 Deep Learning

Course Code: CAI3413	Course Title: Industrial IoT Type of Course: Integrated	L- T-P- C	2	0	2	3
Version No.	1.0					

Course Pre-requisites	Fundamentals of IoT, Computer Networks, Python Programming			
Anti-requisites	NIL			
Course Description	This course introduces the foundational concepts and applications of the Industrial Internet of Things (IIoT), emphasizing industrial automation, smart manufacturing, and predictive analytics. Students will gain hands-on experience with sensor integration, cloud platforms, and industrial communication protocols necessary for developing and managing IIoT systems.			
Course Objective	To provide practical skills and theoretical knowledge for designing, implementing, and managing Industrial IoT applications in real-world industrial and manufacturing environments.			
Course Outcomes	On successful completion of the course, students will be able to: Understand the architecture and ecosystem of Industrial IoT systems. Integrate sensors and edge devices for real-time industrial data monitoring. Utilize cloud computing and analytics for IIoT-based applications. Apply communication protocols and security mechanisms relevant to industrial environments.			
Course Content:				
Module 1	Introduction to Industrial IoT	Assignment		18[8L+10P] Sessions
Topics: Fundamentals of IoT and IIoT: Definitions, history, evolution, IIoT vs. Consumer IoT, IIoT Architecture Layers: Perception, Network, Middleware, Application Industrial applications: Smart factories, Energy grids, Process automation Introduction to Industry 4.0 and its pillars (Cyber-Physical Systems, Big Data, Cloud Computing), Business benefits and ROI of IIoT deployment Challenges in IIoT: Scalability, legacy systems, interoperability, security, Standards and regulatory compliance in IIoT (ISA-95, ISO/IEC 30141).				
Module 2	Devices and Communication Protocols	Assignment		14[7L+7P] Sessions
Topics: Types of Sensors and Actuators: Environmental, Proximity, Motion, Flow, Vibration, Microcontrollers and Development Boards: Arduino, Raspberry Pi, ESP32, IIoT Gateways and Edge Devices: Intel NUC, Jetson Nano, Communication Technologies: Wired: Modbus RTU, CAN, Ethernet/IP,Wireless: Zigbee, LoRaWAN, Wi-Fi, NB-IoT, Bluetooth LE, Protocols for IIoT:MQTT: Publish-Subscribe Model,OPC-UA: Industrial interoperability and data exchange, CoAP, HTTP/HTTPS,Integration with SCADA and PLC Systems,Device				

provisioning and configuration management, Network Topologies and Addressing in IIoT environments.				
Module 3	IIoT Data Processing and Analytics	Assignment		14[6L+8P] Sessions
Topics: Data Acquisition: Sampling rate, ADC/DAC, Signal conditioning, Local Storage: SQLite, Edge databases, Stream Processing: Apache Kafka, Apache Flink, Data Preprocessing: Noise reduction, outlier detection, filtering techniques, Introduction to Edge Computing and Fog Computing, Real-time vs. Batch Analytics:Predictive Analytics in Manufacturing:Predictive maintenance models,Failure prediction using historical data,Machine Learning for Sensor Data: Regression, Clustering, Classification,Anomaly Detection in Industrial Systems,Data Encryption and Access Control at Edge and Cloud,Security Threats: DDoS, firmware tampering, man-in-the-middle attacks, Best practices for data governance and GDPR compliance				
Module 4	IIoT Cloud Integration and Applications			
Overview of Cloud Computing Models (IaaS, PaaS, SaaS) in IIot, IIoT Cloud Platforms: AWS IoT Core: Device Shadow, Rule Engine, Azure IoT Hub and IoT Central, Google Cloud IoT Core, Device-to-Cloud Communication Models, Storage Solutions: Amazon S3, Azure Blob Storage, InfluxDB, Visualization Tools: Grafana, Power BI, Google Data Studio, Case Studies:Digital Twin of a robotic arm, Smart HVAC system for energy optimization,Real-time supply chain visibility, Building Custom Dashboards and Alerts,Integration with ERP and MES Systems,Final Capstone Project:Design, simulate, and evaluate an end-to-end IIoT system, Include edge device, communication, cloud, analytics, and visualization.				
Project work/Assignment:				
Assignment 1: Sensor Data Acquisition and Visualization Assignment 2: Edge-Based Analytics for Predictive Maintenance Mini Project (Team-based): End-to-End IIoT System for a Smart Industry Scenario				
Setup Raspberry Pi/Arduino for IIoT applications. Interface DHT11, vibration, and ultrasonic sensors. Send sensor data via MQTT to a cloud broker. Create Node-RED dashboards for IIoT data. Build OPC-UA server and client communication. Log and visualize sensor data using Python. Apply edge analytics using Raspberry Pi and filtering techniques. Stream real-time data using Kafka. Integrate with AWS IoT Core for data monitoring. Forecast sensor values using LSTM (predictive maintenance).				

Detect anomalies in sensor data using Scikit-learn. Connect multiple IIoT devices into a secure network. Simulate a digital twin for a production line. Develop a simple IIoT-based security alert system. Capstone Project: Deploy an end-to-end IIoT prototype for a smart manufacturing scenario.
REFERENCE MATERIALS: TEXTBOOKS Alasdair Gilchrist, Industry 4.0: The Industrial Internet of Things, Apress, 2016. Sabina Jeschke, Christian Brecher, Houbing Song, Danda B. Rawat, Industrial Internet of Things, Springer, 2017. REFERENCES Arshdeep Bahga, Vijay Madisetti, Internet of Things: A Hands-On Approach, Universities Press, 2015. Hakima Chaouchi, The Internet of Things: Connecting Objects, Wiley, 2010. Rajkumar Buyya, Amir Vahid Dastjerdi, Internet of Things: Principles and Paradigms, Morgan Kaufmann, 2016. JOURNALS/MAGAZINES IEEE Internet of Things Journal https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=6488907 Publishes articles on the design, development, and deployment of IoT applications in industrial settings. ACM Transactions on Internet of Things (TIOT) https://dl.acm.org/journal/tiot Covers architectures, algorithms, and applications related to IoT including edge computing and real-time analytics. Sensors (MDPI - Special Issue on Industrial IoT) https://www.mdpi.com/journal/sensors Features studies on sensor development, wireless communication, and industrial sensor networks. Journal of Industrial Information Integration (Elsevier) https://www.sciencedirect.com/journal/journal-of-industrial-information-integration Focuses on integrating data from industrial systems for smart manufacturing and digital twins. IEEE Spectrum – IoT and Industry 4.0 https://spectrum.ieee.org/ Publishes accessible articles on cutting-edge tech including IIoT, smart robotics, and industrial AI. Industrial IoT World – Insights & Reports https://www.iiot-world.com/ Offers case studies, whitepapers, and expert commentary on real-world IIoT deployments.

Automation World https://www.automationworld.com/ Covers automation systems, IIoT trends, cloud integration, and edge devices. Industry 4.0 Magazine (Industry40.today) https://industry40.today/ Industry-focused magazine offering updates on cyber-physical systems, smart factories, and digital transformation. Regularly features expert opinion, trends, and technologies including AI in cyber security. SWAYAM/NPTEL/MOOCs: NPTEL – Industrial Internet of Things (IIT Kharagpur) Instructor: Prof. Sudip Misra https://onlinecourses.nptel.ac.in/noc23_cs69/preview NPTEL – Introduction to Industry 4.0 and Industrial Internet of Things (IIT Roorkee) Instructor: Prof. Sudeb Dasgupta https://onlinecourses.nptel.ac.in/noc21_me88/preview

Course Code: CAI3414	Course Title: Smart Farming Type of Course: Integrated	L- T-P- C	2	0	2	3
Version No.	1.0					
Course Pre-requisites	Fundamentals of IoT, Python Programming, Basics of Agriculture Science (desirable)					
Anti-requisites	NIL					
Course Description	This course introduces the interdisciplinary domain of Smart Farming, integrating IoT, AI, cloud computing, and sensor networks to enable precision agriculture. Students will learn how to design, develop, and deploy data-driven agricultural systems for real-time monitoring, crop management, and decision-making.					
Course Objective	To provide practical and theoretical insights into the application of smart technologies in agriculture, with emphasis on sustainable practices, yield optimization, and automation.					

Course Outcomes	On successful completion of this course, students will be able to: Understand the fundamentals of smart agriculture and its ecosystem. Apply IoT and sensors for environmental and crop monitoring. Analyze agricultural data for predictive insights using AI/ML. Design and deploy smart farming solutions using cloud and mobile platforms.			
Course Content:				
Module 1	Introduction to Smart Farming and Precision Agriculture	Assignment		18[8L+10P] Sessions
Topics: Evolution from traditional to precision agriculture, Components and architecture of Smart Farming systems, Applications: crop monitoring, irrigation control, pest detection, Overview of remote sensing and satellite-based agriculture, Soil health and weather data integration.				
Module 2	IoT and Sensor Systems in Agriculture	Assignment		14[7L+7P] Sessions
Topics: Sensor types: soil moisture, temperature, humidity, pH, NDVI, Microcontrollers: Arduino, ESP32, Raspberry Pi,Wireless communication protocols: LoRa, Zigbee, Wi-Fi, GSM, Energy harvesting and power management for field devices, Edge computing in agriculture.				
Module 3	Data Analytics and AI in Agriculture	Assignment		14[6L+8P] Sessions
Topics: Introduction to agri-data lifecycle, Data preprocessing and anomaly detection, AI/ML techniques: regression, classification, clustering for yield prediction and disease detection,Time series forecasting for rainfall, irrigation, and climate impact, Decision support systems and dashboards.				
Module 4	Cloud, Mobile, and Drone Integration			
Cloud platforms for agriculture: ThingSpeak, AWS IoT, Blynk, Azure IoT, Mobile app integration for farmer advisory systems, Role of drones in smart agriculture: aerial imaging, spraying, crop health assessment, Case studies: smart greenhouses, hydroponics, aquaponics, Capstone: End-to-end smart farming system design.				
Project work/Assignment:				
Assignment 1: Real-Time Crop and Soil Monitoring System Assignment 2: Crop Yield Prediction using Machine Learning Mini Project (Team-based): Smart Farm Automation System				

List of Lab Tasks :

- Setup Arduino/ESP32 for collecting soil and climate data
- Interface with soil moisture, DHT11, and pH sensors
- Transmit data wirelessly using LoRa or Wi-Fi
- Real-time dashboard for field data (using Blynk/ThingSpeak)
- Predict crop yield using linear regression
- Train an image classifier for leaf disease detection
- Setup automated irrigation control system
- Use GPS for geotagging sensor data
- Drone-based simulation for crop monitoring
- Preprocess and visualize multivariate agri-data using Python
- Connect field devices to cloud platform (AWS IoT or Firebase)
- Alert system for low soil moisture via SMS/email
- Forecast rainfall using time series techniques
- Build a mobile-based decision support system (low-code platform)
- Capstone: Build and present a complete Smart Farm prototype

REFERENCE MATERIALS:

TEXTBOOKS

- Rajesh Singh, Anita Gehlot, Bhupendra Singh, Internet of Things and Wireless Sensor Networks in Smart Agriculture, CRC Press, 2021.
- Subhas Chandra Mukhopadhyay, Internet of Things in Smart Agriculture, Springer, 2020.

REFERENCES

Himanshu Patel, Smart Farming Technologies for Sustainable Agricultural Development, IGI Global, 2020.

G. R. Kanagachidambaresan, Internet of Things for Sustainable Community Development, Springer, 2021.

IEEE Papers and Reports on Smart Agriculture, Remote Sensing, and Precision Farming

JOURNALS/MAGAZINES

- EEE Access – Special Section on Smart Agriculture
- Computers and Electronics in Agriculture (Elsevier)
- Agricultural Systems Journal
- Smart Farming Magazine
- IoT for Agri-Tech (IoT World Today)

SWAYAM/NPTEL/MOOCs: NPTEL – Introduction to Smart Agriculture (IIT Kanpur) https://onlinecourses.nptel.ac.in/noc22_ge15/preview   NPTEL – Applications of IoT in Agriculture (IIT Kharagpur) https://onlinecourses.nptel.ac.in/noc23_cs91/preview   Coursera – Smart Agriculture with IoT https://www.coursera.org/learn/smart-agriculture-iot

Course Code: CAI3415	Course Title: AI for Autonomous Systems Type of Course: Integrated	L- T-P- C	2	0	2	3
Version No.	1.0					
Course Pre-requisites	Artificial Intelligence, Machine Learning, Basic Robotics, Python Programming					
Anti-requisites	NIL					
Course Description	This course introduces the principles and technologies involved in developing intelligent autonomous systems using Artificial Intelligence. It focuses on perception, decision-making, control systems, and learning-based approaches to build real-time, autonomous agents such as self-driving vehicles, drones, and mobile robots.					
Course Objective	To equip students with knowledge and practical skills in using AI techniques to develop safe, adaptive, and intelligent autonomous systems that can sense, plan, and act in dynamic environments.					
Course Outcomes	On completion of this course, students will be able to: Understand the architectural components of autonomous systems. Apply computer vision and sensor fusion for environmental perception. Develop path planning and control strategies for autonomous navigation. Implement learning-based models for real-time decision making.					
Course Content:						

Module 1	Introduction to Autonomous Systems	Assignment		18[8L+10P] Sessions

<table>
<tr><td colspan="5">Definition and types of autonomous systems, Key components: perception, planning, control, learning, Applications: Self-driving cars, drones, mobile robots, AGVs, Architecture of an autonomous system: sensors, actuators, processors, ROS (Robot Operating System) overview, Ethical and safety concerns in autonomy.</td></tr>
<tr><td>Module 2</td><td>Perception and Sensor Fusion</td><td>Assignment</td><td></td><td>14[7L+7P]

Sessions</td></tr>
<tr><td colspan="5">Topics:

Sensor types: LiDAR, Radar, Ultrasonic, IMU, GPS, Cameras, Environment mapping: Occupancy grid, SLAM, Visual perception: Object detection (YOLO, SSD), semantic segmentation, Sensor fusion techniques: Kalman filter, EKF, particle filter,Case study: Lane detection using OpenCV</td></tr>
<tr><td>Module 3</td><td>Planning and Navigation</td><td>Assignment</td><td></td><td>14[6L+8P]

Sessions</td></tr>
<tr><td colspan="5">Topics:

Path planning algorithms: Dijkstra, A*, RRT, D*, Obstacle avoidance: Dynamic Window Approach, Potential Fields, Localization: Monte Carlo Localization, Graph-based SLAM,Trajectory generation and smoothing,Behavior trees and state machines for task execution,Indoor vs. outdoor navigation.</td></tr>
<tr><td>Module 4</td><td>Learning-based Control and Decision Making</td><td></td><td></td><td></td></tr>
<tr><td colspan="5">Reinforcement learning: Q-learning, Deep Q-Networks, Policy-based learning and imitation learning, Decision making under uncertainty (Markov Decision Processes)Learning from simulation (Gazebo, CARLA, Webots), Edge AI deployment for low-latency inference, Case studies: Autonomous driving stack, warehouse robot planner.</td></tr>
<tr><td colspan="5">Project work/Assignment:</td></tr>
<tr><td colspan="5">Assignment 1: Object Detection and Lane Tracking System

Assignment 2: Path Planning Simulator

Mini Project (Team-based):   Autonomous System Prototype</td></tr>
<tr><td colspan="5">List of Lab Tasks (15 Labs):

Simulate a differential drive robot in ROS

Interface camera and distance sensors with Raspberry Pi

Perform object detection using pre-trained YOLOv5

Apply color-based segmentation for road/lane detection

Fuse GPS and IMU data using a Kalman filter

Build a simple SLAM model using GMapping or Cartographer

Implement obstacle avoidance using LiDAR data</td></tr>
</table>

Visualize path planning using A* on a grid map

Simulate autonomous vehicle in CARLA/Gazebo

Implement PID control for motor speed regulation

Train a reinforcement learning agent to navigate a maze

Deploy a TensorFlow Lite model for edge object recognition

Evaluate decision trees for robotic task selection

Use OpenCV to detect traffic signs and lights

Capstone: Build and test a mini autonomous robot prototype

REFERENCE MATERIALS:

TEXTBOOKS

Raj Madhavan, Autonomous Systems: Issues and Challenges, Springer, 2021

Peter Corke, Robotics, Vision and Control: Fundamental Algorithms in MATLAB, Springer, 2017

REFERENCES

Sebastian Thrun et al., Probabilistic Robotics, MIT Press, 2005

Benjamin Kuipers, Principles of Intelligent Autonomous Systems, MIT Press, 2016

Deepak Khemani, A First Course in Artificial Intelligence, McGraw-Hill, 2013

JOURNALS/MAGAZINES

IEEE Transactions on Robotics

Journal of Field Robotics (Wiley)

Autonomous Robots (Springer)

IEEE Transactions on Intelligent Vehicles

SWAYAM/NPTEL/MOOCs:

NPTEL – Robotics: Perception, Planning and Control (IIT Kanpur)
https://onlinecourses.nptel.ac.in/noc21_ee68/preview

NPTEL – Artificial Intelligence: Search Methods for Problem Solving (IIT Madras)
https://onlinecourses.nptel.ac.in/noc21_cs53/preview

Self-Driving Cars Specialization (University of Toronto)
https://www.coursera.org/specializations/self-driving-cars

Course Code: CAI3416	Course Title: Edge Computing Type of Course: Integrated		L- T-P- C	2	0	2	3
Version No.	1.0						
Course Pre-requisites	Computer Networks, Cloud Computing, Python Programming						
Anti-requisites	NIL						
Course Description	This course introduces the fundamentals of edge computing, an emerging paradigm that enables data processing and analytics at the network's edge. It covers system architecture, communication protocols, edge AI, and real-time applications with a hands-on approach using edge devices.						
Course Objective	To equip students with knowledge and skills in designing and deploying edge computing systems that offer low-latency, energy-efficient, and scalable solutions for real-time and distributed applications.						
Course Outcomes	After successful completion, students will be able to: Explain the principles and architecture of edge computing systems. Develop and deploy applications on edge devices like Raspberry Pi, Jetson Nano, etc. Integrate edge computing with IoT, cloud, and AI systems. Analyze and optimize performance of edge-based applications.						
Course Content:							
Module 1	Introduction to Edge Computing	Assignment				18[8L+10P] Sessions	
Topics: Evolution from cloud to fog to edge, Edge computing architecture: core, edge, and device layers, Use cases: smart cities, industrial automation, healthcare, autonomous systems, Edge vs. Cloud: latency, bandwidth, privacy, scalability, Edge device overview: Raspberry Pi, Nvidia Jetson, Coral Dev Board							
Module 2	Edge Devices and Communication	Assignment				14[7L+7P] Sessions	
Topics: Edge device setup and configuration, Operating systems: Raspbian, Ubuntu Core, Yocto, Edge and fog networking protocols: MQTT, CoAP, 6LoWPAN, Containerization with Docker at the edge, Device orchestration and provisioning, Energy and resource management in constrained devices.							

Module 3	Edge AI and Analytics	Assignment		14[6L+8P] Sessions

Topics:

Edge AI: Concepts and advantages, Lightweight ML frameworks: TensorFlow Lite, ONNX, OpenVINO, Model quantization and compression for deployment, Real-time data processing and event-based analytics, Case studies: object detection, audio classification, sensor fusion, Ethical considerations and security at the edge

Module 4	Edge-Cloud Integration and Applications			

Edge-to-cloud communication pipelines, AWS Greengrass, Azure IoT Edge, Google Edge TPU, Microservices and serverless architecture on the edge, Industrial edge computing with OPC-UA and Modbus, Building dashboards for edge data insights, Capstone: End-to-end edge system for real-world application.

Project work/Assignment:

Assignment 1: IoT Data Stream Processing at the Edge

Assignment 2: Edge AI Inference Deployment

Mini Project (Team-based): Real-World Edge Solution

List of Lab Tasks :

Setting up Raspberry Pi or Jetson Nano for edge deployment

Installing and configuring Docker on an edge device

Connecting sensors (camera, DHT11, ultrasonic) to edge devices

Building and running MQTT-based data pipeline

Collecting and visualizing data using Node-RED or Grafana

Edge inference using TensorFlow Lite object detection model

Deploying ONNX model for sensor-based classification

Streaming video analytics at the edge

Building a containerized AI service on the edge

Edge-to-cloud integration using AWS Greengrass or Azure IoT Edge

Monitoring resource usage on constrained devices

Detecting anomalies in time-series data at the edge

Designing a simple mobile dashboard to view edge insights

Security setup: HTTPS, authentication, and encryption

Capstone: Build a complete edge-to-cloud application prototype.

REFERENCE MATERIALS:

TEXTBOOKS

Perry Lea, Edge Computing: From Hype to Reality, Packt Publishing, 2020

Mahmoud Tawfik, Edge Computing and Internet of Things for Smart Agriculture, Springer, 2021

REFERENCES

Satyanarayanan M., The Emergence of Edge Computing, IEEE Computer, 2017

Flavio Bonomi et al., Fog Computing and Its Role in the Internet of Things, MCC Workshop, ACM 2012

OpenFog Consortium Architecture White Paper, 2017

JOURNALS/MAGAZINES

IEEE Internet of Things Journal

ACM Transactions on Internet Technology (TOIT)

IEEE Edge Computing Magazine

Elsevier Future Generation Computer Systems – Special Issues on Edge Computing

SWAYAM/NPTEL/MOOCs:

NPTEL – Cloud Computing and Distributed Systems (IIT Kharagpur)
https://onlinecourses.nptel.ac.in/noc22_cs62/preview

NPTEL – Internet of Things (IIT Kharagpur)
https://onlinecourses.nptel.ac.in/noc21_cs60/preview

Coursera – Edge AI and Computer Vision (Intel + OpenVINO)
https://www.coursera.org/learn/introduction-to-edge-ai

Course Code: CAI3417	Course Title: Cognitive Computing Type of Course: Integrated	L- T-P- C	2	0	2	3
Version No.	1.0					
Course Pre-requisites	Essentials of AI					
Anti-requisites	NIL					

Course Description	This course explores the interdisciplinary domain of Cognitive Computing, inspired by the human brain’s capability to reason, learn, and interact naturally. Students will study the fundamentals of cognitive systems, natural language processing, machine learning, and AI-driven decision-making, with a focus on building intelligent applications using IBM Watson, transformers, and cognitive APIs.			
Course Objective	To understand the principles of cognitive systems and their architecture To explore the components of natural language understanding and reasoning To apply AI and ML models in cognitive tasks such as Q&A, dialogue, and speech To implement and evaluate real-world cognitive applications			
Course Outcomes	Upon successful completion, students will be able to: Explain the structure and functioning of cognitive computing systems Apply NLP and ML techniques to build cognitive applications Use cognitive computing APIs for speech, vision, and language understanding Develop intelligent solutions using IBM Watson and other platforms			
Course Content:				
Module 1	Introduction to Cognitive Computing	Assignment		18[8L+10P] Sessions
Topics: Definition and evolution of cognitive computing, Cognitive architecture: perception, reasoning, learning, decision-making, Differences between traditional AI and cognitive systems, Applications in healthcare, finance, education, and law.				
Module 2	Machine Learning in Cognitive Systems	Assignment		14[7L+7P] Sessions
Topics: Supervised, unsupervised, and reinforcement learning basics, Neural networks and deep learning, Introduction to cognitive frameworks (IBM Watson, Microsoft Azure AI, Google AI), Cognitive APIs for image, speech, and text processing				
Module 3	Natural Language Processing and Understanding	Assignment		14[6L+8P] Sessions
Topics:				

Language modeling, tokenization, stemming, lemmatization, Sentiment analysis, named entity recognition (NER), and summarization, Conversational AI: chatbots and voice assistants, Transformer models: BERT, GPT, and question answering systems.				
Module 4	Building Cognitive Applications	Assignment		14[6L+8P] Sessions
Topics: Knowledge representation and reasoning, Integrating cognitive APIs (speech-to-text, text-to-speech, visual recognition), Ethical considerations and trust in cognitive systems, Cognitive services deployment and case studies.				

List of Lab Tasks

Lab 1: Introduction to Cognitive Computing Platforms
Objective: Explore IBM Watson and Azure Cognitive Services
Task: Register and configure accounts for cloud-based cognitive tools
Activity: Build a simple "Hello Cognitive World" app

Lab 2: Sentiment Analysis using IBM Watson NLP API
Objective: Analyze emotional tone of user-generated content
Task: Use Watson’s NLP API to detect sentiment
Activity: Visualize sentiment trends in tweets or reviews

Lab 3: Named Entity Recognition using spaCy
Objective: Extract key entities from text
Task: Identify names, places, and organizations
Activity: Highlight named entities using Python and spaCy

Lab 4: Speech-to-Text using Google Cloud API
Objective: Convert speech input into text
Task: Use Google Cloud Speech API
Activity: Transcribe audio files with speaker diarization

Lab 5: Text-to-Speech using IBM Watson TTS
Objective: Generate human-like speech from text
Task: Convert user input to spoken voice
Activity: Play synthesized speech in a web app

Lab 6: Build a Visual Recognition App
Objective: Identify objects in images
Task: Use IBM Watson Visual Recognition API
Activity: Upload image and detect labels or tags

Lab 7: Design a Conversational Chatbot with Dialogflow
Objective: Implement intent-based conversation flow

Task: Create intents, entities, and responses
Activity: Deploy chatbot on a web interface

Lab 8: Question Answering with Transformers
Objective: Use BERT to answer questions based on context
Task: Load pre-trained model and context documents
Activity: Ask and receive accurate answers using Hugging Face

---

Lab 9: Create a Knowledge Graph using Neo4j
Objective: Represent relationships among entities
Task: Build and query knowledge graphs
Activity: Visualize connections in graph format

Lab 10: Sentiment Classification with LSTM
Objective: Classify text as positive/negative
Task: Train LSTM model for binary sentiment
Activity: Evaluate with accuracy and confusion matrix

Lab 11: Image Captioning with CNN-RNN Architecture
Objective: Generate captions for images
Task: Integrate image features and text generation
Activity: Display image and generate natural description

Lab 12: Facial Expression Recognition using OpenCV
Objective: Detect and classify facial emotions
Task: Use emotion classification models
Activity: Real-time expression detection from webcam

Lab 13: Design a Voice Assistant using Python
Objective: Enable basic voice interaction
Task: Use speech recognition and TTS
Activity: Query weather, date, and time via voice

Lab 14: Ethical Use of Cognitive Computing Tools
Objective: Understand ethical AI deployment
Task: Analyze bias and fairness in models
Activity: Present case studies on responsible AI

Lab 15: Capstone Project – Build a Multi-Modal Cognitive App
Objective: Integrate speech, vision, and language
Task: Develop a chatbot with voice and visual recognition
Activity: Demonstrate and document full application pipeline

REFERENCE MATERIALS

TEXTBOOKS

Judith Hurwitz, Marcia Kaufman, Cognitive Computing and Big Data Analytics, Wiley, 2015

Rajiv Mathur, Cognitive Computing: Theory and Applications, CRC Press, 2022

REFERENCE BOOKS

Rob High, The Era of Cognitive Systems: An Inside Look at IBM Watson and How it Works, IBM Redbooks

Adnan Masood, Cognitive Computing Recipes: AI and Machine Learning Applications Using IBM Watson, Apress, 2019

Sebastian Raschka, Natural Language Processing with Transformers, O'Reilly, 2021

JOURNALS / MAGAZINES

IEEE Intelligent Systems

ACM Transactions on Interactive Intelligent Systems (TIIS)

Cognitive Computation (Springer)

Journal of Artificial Intelligence Research (JAIR)

SWAYAM / NPTEL / MOOCs

NPTEL – Deep Learning for Computer Vision (IIT Hyderabad)
https://nptel.ac.in/courses/106106231

Coursera – Introduction to IBM Watson (IBM)
https://www.coursera.org/learn/ai-watson

edX – IBM Applied AI: Cognitive Services
https://www.edx.org/professional-certificate/ibm-applied-artificial-intelligence

Course Code: CAI2500	Course Title: Machine Learning Type of Course: 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 – 11
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-11
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-11
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-12
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: Certification course in Machine Learning through NPTEL Mini Project on (Module 1 to Module 4)				
Textbooks Aurélien Géron, "Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow", Oreilly, Second Edition, 2019. Andreas C Muller, Sarah Guido, "Introduction to Machine Learning with Python: A Guide for Data Scientists", Oreilly, First Edition, 2018				
References 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: Lab	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 Lab 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

Experiment No 13: Implementing k-Nearest Neighbors (k-NN) for Handwritten Digit Recognition

Level 1: Implement k-NN from scratch and test on a small subset of the MNIST dataset.

Level 2: Compare custom k-NN with scikit-learn's k-NN on the full MNIST dataset and evaluate accuracy, precision, and recall.

Experiment No 14: Anomaly Detection using One-Class SVM on Credit Card Fraud Dataset

Level 1: Use a sample dataset to apply One-Class SVM and identify outliers.

Level 2: Apply One-Class SVM on real-world credit card fraud data, analyze the ROC curve and optimize kernel parameters.

Experiment No 15: Hyperparameter Tuning and Model Selection using Grid Search and Cross-Validation

Level 1: Use GridSearchCV to tune parameters of a simple SVM model on the Iris dataset.

Level 2: Perform model comparison (SVM, Random Forest, k-NN) with nested cross-validation and report the best performing model.

Targeted Application & Tools that can be used:

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.

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

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

Course Code: CSE2502	Course Title: Deep Learning Type of Course: 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 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 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 application-specific deep learning models and provide practical knowledge handling and analyzing end user realistic applications. Topics include Fundamental concepts of					

	deep neural networks, Convolutional Neural Networks, Recurrent Network structures, Deep Unsupervised Learning, Generative Adversarial Networks and applications in various problem domains.			
Course Outcomes	On successful completion of this course the students shall be able to: CO1: Learn the Fundamental Principles of Deep Learning. (Remember). CO2: Identify the Deep Learning Algorithms for Various Types of Learning Tasks in various domains (Apply). CO3: Build Supervised and Unsupervised Deep Learning techniques to implement effective models for prediction or classification tasks. (Apply). CO4: Make use of appropriate validation metrics to evaluate the performance of Implemented Deep Neural Network. (Apply)			
Course Content:				
Module 1	Introduction to Deep Learning and Neural Networks	Assignment		12 Classes
Topics: Fundamentals of Deep Learning, Perceptron, Multilayer Perceptron, Optimizing Perceptions using Activation Functions, Loss Functions, Gradient Descent.Feedforward Neural Network, Training Neural Network with Back-propagation, Hyper parameters, Regularization, Dropouts, Batch Normalization, Practical Issues in Neural Network Training -The Problem of Overfitting, The Vanishing and Exploding Gradient Problems				
Module 2	Common Deep Learning Architectures:	Assignment		11 Classes
Topics: Convolutional Neural Network, Transfer learning Techniques, Variants of CNN: ResNet, AlexNet Sequence Modelling: Recurrent Neural Network and its variants - Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU)				
Module 3	Deep Generative Models	Assignment		11 Classes
Topics: Generative Adversarial Networks, Kohonen Networks, Autoencoders, Boltzman Machine, Restricted Boltzmann Machine, Deep Belief Network				

Module-4	Advanced Deep Learning Architectures	Assignment		11 Classes
Topics: Hopfield Network, Probabilistic Neural Network, Deep Reinforcement Learning - The Basic Framework of Reinforcement Learning Deep Learning applications: Image segmentation, Object detection, Attention model for computer vision tasks, Speech Recognition, Video Analytics				
Project work/Assignment:				
Assignment 1 on (Module 1 and Module 2) Assignment 2 on (Module 3 and Module 4)				
REFERENCE MATERIALS: TEXTBOOKS François Chollet, "Deep Learning with Python", 2nd Edition, Manning Publications, 2022 Ian Goodfellow, Yoshua Bengio, Aaron Courville, "Deep Learning", MIT Press, 2017. REFERENCES Amlan Chakrabarti Amit Kumar Das, Saptarsi Goswami, Pabitra Mitra, "Deep Learning", Pearson Publication, 2021. David Foster, "Generative Deep Learning" O'Reilly Publishers, 2020. John D Kellehar, "Deep Learning", MIT Press, 2020. JOURNALS/MAGAZINES IEEE Transactions on Neural Networks and Learning Systems https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=5962385 IEEE Transactions on Pattern Analysis and Machine Intelligence https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=34http://ijaerd.com/papers/special_papers/IT032.pdf International Journal of Intelligent Systems https://onlinelibrary.wiley.com/journal/1098111x SWAYAM/NPTEL/MOOCs: Swayam Nptel – Deep Learning – IIT Ropar https://onlinecourses.nptel.ac.in/noc21_cs35/preview Coursera – Neural Networks and Deep Learning Andrew Ng				

Coursera - Neural Networks for Machine Learning by Geoffrey Hinton in Coursera

Course Code: CAI2503	Course Title: Deep Learning Type of Course: LAB	L- T- P- C	0	0	4	2
Version No.	1.0					
Course Pre-requisites	CSE 1701 Essentials of AI Lab					
Anti-requisites	NIL					
Course Description	This course introduces students to the concepts of deep neural networks and state of the art approaches to develop deep learning models. In this course students will be given 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 application-specific deep learning models and provide practical knowledge handling and analyzing end user realistic applications. Topics include Fundamental concepts of deep neural networks, Convolutional Neural Networks, Recurrent Network structures, Deep Unsupervised Learning, Generative Adversarial Networks and applications in various problem domains.					
Course Outcomes	On successful completion of this course the students shall be able to: CO1: Learn the Fundamental Principles of Deep Learning. (Remember). CO2: Identify the Deep Learning Algorithms for Various Types of Learning Tasks in various domains (Apply). CO3: Build Supervised and Unsupervised Deep Learning techniques to implement effective models for prediction or classification tasks. (Apply). CO4: Make use of appropriate validation metrics to evaluate the performance of Implemented Deep Neural Network. (Apply)					
Course Content:						
List of Lab Tasks: Experiment No. 1: Working with Deep Learning Framework						

Level 1: Explore various Deep Learning Frameworks and identify deep learning frameworks (Keras, Tensorflow, Matplotlib, etc) with various methods available in DL Frameworks to develop a Model.

Experiment No. 2: Build a Basic Artificial Neural Network

Level 1: Create an ANN with DL frameworks and identify suitable ANN Layers using Keras and Tensorflow for pima-indians-diabetes.

Level 2: Create an ANN with DL frameworks and identify suitable ANN Layers using Keras and Tensorflow for any image dataset.

Experiment No. 3: Build a Multi-Layer Perceptron

Level 1: Create a MLP for classification task by identify suitable model for house price prediction.

Level 2: Design a MLP for implementing classification and fine-tuning for speech recognition

Experiment No. 4: Build a Convolutional Neural Network

Level 1: Build CNN architecture for Dog-Cat classification problem.

Level 2: Build Convolution Neural Network (CNN) for fine tuning hyperparameter for improving the performance of model.

Experiment No. 5: Build ResNet Model

Level 1: Build ResNet Model for Medical Imaging Datasets ChestX-ray14

Level 2: Build ResNet Model for Video datasets

Experiment No. 6: Build AlexNet Model

Level 1: Build ResNet Model for CIFAR10 Datasets.

Level 2: Build ResNet Model for Video datasets

Experiment No. 7: Build a Time-Series Model

Level 1: Build RNN/LSTM Model for predicting time series data for sentiment analysis model on IMDB dataset.

Experiment No. 8: Build a Time-Series Model

Level 1: Build RNN/GRU Model for predicting time series data for sentiment analysis model on IMDB dataset.

Experiment No. 9: Build GANs for CIFAR10

Level 1: Develop a GAN to Generate CIFAR10 Small Color Photographs

Experiment No. 10: Build a Transfer Learning Model.

Level 1: Create a Seq2Seq Model. Create Hugging-face API using Transfer learning model.

Experiment No. 11: Build an Auto-Encoder model

Level 1: Implement an Encoder-Decoder Recurrent neural network model for Neural Machine Translation.

Experiment No. 12: Build Generative Adversarial Networks.

Level 1: Design GAN Architecture for Image generations.

Level 2: Design a Age Prediction model by Applying Generative Adversarial

Experiment No. 13: Build an Attention Mechanism Model

Level 1: Implement a basic Attention mechanism for text classification using TensorFlow/Keras.

Level 2: Extend the Attention model for Machine Translation tasks (e.g., English to French translation).

Experiment No. 14: Build a BERT Model for Text Classification

Level 1: Fine-tune a pre-trained BERT model for binary text classification (e.g., movie reviews sentiment analysis).

Level 2: Fine-tune BERT for multi-label classification on a real-world dataset (e.g., news article categorization).

Experiment No. 15: Build a Deep Reinforcement Learning Model

Level 1: Implement a basic Deep Q-Network (DQN) for a simple environment like CartPole using TensorFlow.

Level 2: Implement a DQN-based agent for a complex game environment like Atari Breakout.

REFERENCE MATERIALS:

TEXTBOOKS

François Chollet, “Deep Learning with Python”, 2nd Edition, Manning Publications, 2022

Ian Goodfellow, Yoshua Bengio, Aaron Courville, “Deep Learning”, MIT Press, 2017.

REFERENCES

Amlan Chakrabarti Amit Kumar Das, Saptarsi Goswami, Pabitra Mitra , “Deep Learning”, Pearson Publication, 2021.

David Foster, “Generative Deep Learning” O’Reilly Publishers, 2020.

John D Kellehar, “Deep Learning”, MIT Press, 2020.

JOURNALS/MAGAZINES

IEEE Transactions on Neural Networks and Learning Systems

https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=5962385

IEEE Transactions on Pattern Analysis and Machine Intelligence

https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=34http://ijaerd.com/papers/special_papers/IT032.pdf

International Journal of Intelligent Systems https://onlinelibrary.wiley.com/journal/1098111x

SWAYAM/NPTEL/MOOCs:

Swayam Nptel – Deep Learning – IIT Ropar
https://onlinecourses.nptel.ac.in/noc21_cs35/preview

Coursera – Neural Networks and Deep Learning Andrew Ng

Coursera - Neural Networks for Machine Learning by Geoffrey Hinton in Coursera

Course Code: CAI2504	Course Title: Natural Language Processing Type of Course: 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: Define different problems related to natural language processing. [Understand] Discuss using NLP techniques for different applications. [Apply] Propose solutions for a particular NLP problem using different machine learning and deep learning techniques. [Apply] 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:12
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:11
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:11
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: 11

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:

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.

Lab 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

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):

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).

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: CAI2505	Course Title: Natural Language Processing Lab Type of Course: Lab	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: Define different problems related to natural language processing. [Understand] Discuss using NLP techniques for different applications. [Apply] Propose solutions for a particular NLP problem using different machine learning and deep learning techniques. [Apply] 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: 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. Lab 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): 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). 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: CAI2506	Course Title: Numerical Optimization in AI Type of Course: Discipline Elective - Theory	L- T-P- C	3	0	0	3
Version No.	1.0					
Course Pre-requisites	Essentials of AI					
Anti-requisites	NIL					
Course Description	This course explores numerical optimization techniques essential to Artificial Intelligence and Machine Learning. Students will understand optimization foundations, techniques for both unconstrained and constrained problems, and learn how optimization drives performance improvements in AI systems. Emphasis is on algorithmic understanding and real-world applications in training AI models.					
Course Objective	To introduce various types of optimization problems and solution strategies. To enable students to apply optimization techniques in AI models. To develop skills in both deterministic and stochastic optimization methods. To implement optimization algorithms in practical machine learning tasks.					
Course Outcomes	On successful completion of the course, students will be able to: Formulate and analyze various optimization problems in AI. Implement unconstrained and constrained optimization algorithms. Apply gradient-based and stochastic techniques to train AI models. Evaluate the effectiveness of optimization strategies in real-world AI tasks.					
Course Content:						

Module 1	Foundations of Optimization	Assignment		[12] Sessions
Topics: Optimization in AI and ML: Role and importance, Classification: Linear, Nonlinear, Convex, Non-convex problems, Mathematical tools: Vectors, matrices, gradients, Hessians Objective functions and optimality conditions.				
Module 2	Unconstrained Optimization Methods	Assignment		[11] Sessions
Topics: Gradient Descent, Momentum, Nesterov's Accelerated Gradient, Newton's Method, Conjugate Gradient Method, Quasi-Newton Methods (BFGS, L-BFGS), Line Search and Trust Region Strategies, Convergence behavior and numerical stability.				
Module 3	Constrained Optimization Techniques	Assignment		[11] Sessions
Topics: Equality and Inequality constraints, Lagrange multipliers, KKT conditions, Penalty and Barrier Methods, Sequential Quadratic Programming (SQP), Applications in AI models such as SVM and regularization.				
Module 4	Stochastic and Heuristic Optimization in AI			[11] Sessions
Topics: Stochastic Gradient Descent (SGD) and Adaptive Variants: Adam, Adagrad, RMSprop, Evolutionary Algorithms: Genetic Algorithms, Differential Evolution, Swarm Intelligence: PSO, Ant Colony Optimization, Applications in Deep Learning and Reinforcement Learning.				
REFERENCE MATERIALS: TEXTBOOKS Jorge Nocedal and Stephen J. Wright, Numerical Optimization, Springer, 2nd Edition, 2006. Edwin K. P. Chong and Stanislaw H. Zak, An Introduction to Optimization, Wiley, 4th Edition, 2013. REFERENCES Dimitri P. Bertsekas, Nonlinear Programming, Athena Scientific, 3rd Edition, 2016. Stephen Boyd and Lieven Vandenberghe, Convex Optimization, Cambridge University Press, 2004.				

Andreas Antoniou and Wu-Sheng Lu, Practical Optimization: Algorithms and Engineering Applications, Springer, 2007.

Yaochu Jin, Multi-Objective Machine Learning, Springer, 2006.

Russell & Norvig, Artificial Intelligence: A Modern Approach, Pearson – Latest Edition (selected chapters).

John D Kellehar, “Deep Learning”, MIT Press, 2020.

JOURNALS/MAGAZINES

IEEE Transactions on Evolutionary Computation
https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=4235

IEEE Transactions on Neural Networks and Learning Systems
https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=5962385

Journal of Optimization Theory and Applications (Springer)
https://www.springer.com/journal/10957

Mathematical Programming (Springer)
https://www.springer.com/journal/10107

International Journal of Approximate Reasoning (Elsevier)
https://www.sciencedirect.com/journal/international-journal-of-approximate-reasoning

SWAYAM/NPTEL/MOOCs:

NPTEL – Optimization Techniques (IIT Kharagpur)
Instructor: Prof. P.K. Biswas
https://nptel.ac.in/courses/108105088

NPTEL – Convex Optimization (IIT Madras)
Instructor: Prof. S. Sundaram
https://nptel.ac.in/courses/110106138

Coursera – Numerical Methods for Engineers (Georgia Tech)
https://www.coursera.org/learn/numerical-methods-engineers

Coursera – Discrete Optimization (University of Melbourne)
https://www.coursera.org/learn/discrete-optimization

edX – Convex Optimization (Stanford University – Stephen Boyd)
https://www.edx.org/course/convex-optimization

Course Code: ISE2500	Course Title: Software Testing and Quality Assurance Type of Course: Theory	L- T-P- C	3	0	0	3
Version No.	1.0					

<table>
<tr><td>Course Pre-requisites</td><td colspan="4">Software Design and Development</td></tr>
<tr><td>Anti-requisites</td><td colspan="4">NIL</td></tr>
<tr><td>Course Description</td><td colspan="4">This course provides a comprehensive understanding of software testing strategies and quality assurance practices essential for delivering reliable and high-quality software products. Students will learn various testing techniques, software quality metrics, standards, and tools that support automation and continuous integration in the software development lifecycle.</td></tr>
<tr><td>Course Objective</td><td colspan="4">To understand the principles and practices of software testing.
To introduce various levels and types of testing strategies.
To equip students with knowledge of quality assurance methodologies and standards.
To provide hands-on knowledge of software testing tools and frameworks.</td></tr>
<tr><td>Course Outcomes</td><td colspan="4">On successful completion of the course, students will be able to:
1. Design and implement effective software test strategies.
2. Analyze and evaluate testing coverage and effectiveness.
3. Apply QA practices, including reviews, audits, and process assessments.
4. Use automated testing tools for enhancing software reliability.</td></tr>
<tr><td colspan="5">Course Content:</td></tr>
<tr><td>Module 1</td><td>Introduction to Software Testing</td><td>Assignment</td><td></td><td>[8] Sessions</td></tr>
<tr><td colspan="5">Topics:
Software Development Life Cycle (SDLC) and Testing Life Cycle (STLC), Testing objectives and principles, Test levels: Unit, Integration, System, Acceptance, Test types: Functional, Non-functional, Regression, Smoke, and Sanity testing, Software Defect life cycle.</td></tr>
<tr><td>Module 2</td><td>Test Design and Execution Techniques</td><td>Assignment</td><td></td><td>[7] Sessions</td></tr>
<tr><td colspan="5">Topics:
Black-box and White-box testing, Equivalence class partitioning, Boundary value analysis,
Decision table, State transition, and Use case-based testing, Code coverage, Path testing, Mutation testing, Test case design and test data preparation.</td></tr>
<tr><td>Module 3</td><td>Software Quality Assurance (SQA)</td><td>Assignment</td><td></td><td>[7] Sessions</td></tr>
</table>

Topics:

SQA fundamentals and importance, Reviews, Walkthroughs, and Audits, Quality metrics and defect prevention, Process models: CMMI, ISO 9001, Six Sigma, Software configuration management and version control.

Module 4	**Test Automation and Tools**			**[8] Sessions**

Topics:

Automation Testing: Introduction and benefits, Selenium, JUnit, TestNG frameworks, Continuous Integration tools: Jenkins, GitLab CI/CD, Performance Testing using JMeter, Case studies and real-world QA practices.

REFERENCE MATERIALS:

TEXTBOOKS

1. Ron Patton, Software Testing, 2nd Edition, Pearson Education, 2005.
2. Paul Ammann and Jeff Offutt, Introduction to Software Testing, Cambridge University Press, 2nd Edition, 2016.

REFERENCES

1. Cem Kaner, Jack Falk, Hung Q. Nguyen, Testing Computer Software, Wiley, 2nd Edition, 1999.
2. Glenford J. Myers, The Art of Software Testing, Wiley, 3rd Edition, 2011.
3. Naresh Chauhan, Software Testing: Principles and Practices, Oxford University Press, 2010.
4. Pressman & Maxim, Software Engineering: A Practitioner’s Approach, McGraw-Hill, 8th Edition, 2014 – Selected Chapters.

JOURNALS / MAGAZINES

1. **IEEE Transactions on Software Engineering**
   https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=32
2. **Software Quality Journal (Springer)**
   https://www.springer.com/journal/11219
3. **ACM Transactions on Software Engineering and Methodology (TOSEM)**
   https://dl.acm.org/journal/tosem

SWAYAM / NPTEL / MOOCs

1. **NPTEL – Software Testing (IIT Kharagpur)**
   https://nptel.ac.in/courses/106105150

2. **Coursera – Software Testing and Automation (University of Minnesota)**
   https://www.coursera.org/learn/software-testing-automation
3. **Udemy – Automated Software Testing with Python**
   https://www.udemy.com/course/automated-software-testing-with-python/
4. **edX – Software Testing Fundamentals (TUM)**
   https://www.edx.org/course/software-testing

Course Code: ISE2501	Course Title: Software Testing and Quality Assurance Lab Type of Course: Lab	L- T-P- C	0	0	2	1
Version No.	1.0					
Course Pre-requisites	Software Design and Development					
Anti-requisites	NIL					
Course Description	This laboratory course offers hands-on practice in software testing techniques and tools to ensure software quality and reliability. Students will implement unit, integration, system, and regression tests, and gain experience with manual and automated testing tools. Emphasis is placed on testing strategies, test case development, defect tracking, and quality assurance practices.					
Course Objective	To provide practical knowledge of software testing life cycle and strategies To enable test design and execution using black-box and white-box techniques To introduce automated testing tools for functional and performance testing To develop defect tracking, reporting, and quality assurance processes					
Course Outcomes	Upon successful completion, students will be able to: 5. Design effective test cases for different levels of testing 6. Execute manual and automated test procedures 7. Analyze software quality through metrics and reports 8. Use software testing tools to automate and validate results					
List of Laboratory Tasks						

Lab 1: Introduction to Software Testing Tools
Objective: Understand basic tools and environment for testing
Task: Explore tools like JUnit, Selenium, Bugzilla
Activity: Set up testing environment for future labs

Lab 2: Requirement Analysis and Test Case Design
Objective: Design test cases from software requirements
Task: Derive test scenarios and test cases from a sample SRS
Activity: Document test cases in standard format

Lab 3: Black-Box Testing Techniques
Objective: Apply functional testing methods
Task: Perform equivalence partitioning and boundary value analysis
Activity: Design and execute test cases on login module

Lab 4: White-Box Testing Techniques
Objective: Explore code-based testing methods
Task: Perform statement and branch coverage testing
Activity: Generate test data for decision structures in Java/Python

---

Lab 5: Unit Testing with JUnit
Objective: Write and execute unit tests
Task: Create unit test cases using JUnit framework
Activity: Test Java functions and assert expected outputs

Lab 6: Integration Testing
Objective: Test interaction between modules
Task: Use stubs and drivers to simulate integration
Activity: Execute integration test cases on a student registration system

---

Lab 7: System and Acceptance Testing
Objective: Perform end-to-end testing
Task: Prepare test cases for system-level testing
Activity: Validate expected output on sample system software

Lab 8: Regression Testing
Objective: Validate unchanged functionality after changes

Task: Rerun previously passed test cases after updates
Activity: Identify regression defects and document reports

Lab 9: Automation with Selenium WebDriver
Objective: Automate web application testing
Task: Create scripts to test login and navigation
Activity: Validate elements and capture screenshots

Lab 10: Performance Testing using Apache JMeter
Objective: Simulate and evaluate system load
Task: Run load tests for concurrent users
Activity: Generate reports and interpret performance metrics

Lab 11: Bug Reporting using Bugzilla
Objective: Track software defects
Task: Report and manage bugs in Bugzilla
Activity: Update bug status and assign severity

Lab 12: Test Management using TestLink
Objective: Manage and document testing activities
Task: Organize test plans, cases, and execution
Activity: Report test coverage and result summaries

Lab 13: Code Coverage Analysis
Objective: Measure code coverage using tools
Task: Use tools like JaCoCo or Coverage.py
Activity: Analyze coverage reports and improve test suite

Lab 14: Software Quality Metrics Calculation
Objective: Compute quality assurance metrics
Task: Calculate defect density, defect leakage, test effectiveness
Activity: Analyze metrics for project quality reporting

Lab 15: Capstone Project – Test a Web Application
Objective: Apply full testing lifecycle
Task: Plan, design, execute, and report tests
Activity: Submit complete test documentation for the selected app

REFERENCE MATERIALS

TEXTBOOKS

1. **Ron Patton**, *Software Testing*, 2nd Edition, Pearson Education, 2005
2. **Paul Ammann and Jeff Offutt**, *Introduction to Software Testing*, Cambridge University Press, 2nd Edition, 2016

REFERENCE BOOKS

1. **Glenford J. Myers**, *The Art of Software Testing*, Wiley, 3rd Edition, 2011
2. **Naresh Chauhan**, *Software Testing: Principles and Practices*, Oxford University Press, 2010
3. **Cem Kaner et al.**, *Testing Computer Software*, Wiley, 2nd Edition, 1999

Course Code: ISE2502	**Course Title: Information Retrieval** **Type of Course: Theory**	**L- T-P- C**	**3**	**0**	**0**	**3**
Version No.	1.0					
Course Pre-requisites	Essentials of AI					
Anti-requisites	NIL					
Course Description	This course provides an in-depth understanding of Information Retrieval (IR) systems that form the foundation of modern search engines and text-based applications. It covers algorithms, models, and system architectures for organizing, processing, indexing, and retrieving textual information, with practical applications in web search, recommender systems, and natural language processing.					
Course Objective	To understand the core concepts of information retrieval systems and models. To explore indexing, ranking, and evaluation techniques for IR systems. To apply IR techniques in text classification, clustering, and web search. To introduce modern trends in IR such as semantic search and neural IR.					
Course Outcomes	On successful completion of the course, students will be able to: 1. Explain the fundamentals and architecture of IR systems. 2. Design and implement indexing and ranking algorithms. 3. Apply evaluation metrics to assess retrieval performance.					

	4. Use IR techniques in real-world applications like search engines and recommender systems.

Course Content:

Module 1	**Foundations of Information Retrieval**	Assignment		**[8] Sessions**

Topics:
Introduction to IR and its applications, IR system architecture and pipeline, Boolean retrieval model and query processing, Inverted index and dictionary construction, Tokenization, normalization, stopword removal, stemming.

Module 2	**Retrieval Models and Ranking**	Assignment		**[7] Sessions**

Topics:
Vector Space Model, Term Frequency and Inverse Document Frequency (TF-IDF), Probabilistic Models: BM25, Language Models for IR, Document scoring and ranking, Relevance feedback and query expansion, Evaluation metrics: Precision, Recall, F1, MAP, NDCG.

Module 3	**Indexing, Crawling, and Text Mining**	Assignment		**[7] Sessions**

Topics:

Index compression and storage techniques, Web crawling and spidering algorithms, Text classification using Naïve Bayes, k-NN, and SVM, Clustering: k-means, hierarchical, and LSA, Introduction to recommender systems and collaborative filtering.

.

Module 4	**Web and Semantic Information Retrieval**			**[8] Sessions**

Topics:

Link analysis: PageRank, HITS, Search engine architecture and advertising models, IR using deep learning: word embeddings, transformers, Question answering and neural semantic search, Current trends: Conversational search, Multilingual IR.

.

REFERENCE MATERIALS

TEXTBOOKS

1. **Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze**, *Introduction to Information Retrieval*, Cambridge University Press, 2008. https://nlp.stanford.edu/IR-book/

2. **Ricardo Baeza-Yates and Berthier Ribeiro-Neto**, *Modern Information Retrieval: The Concepts and Technology behind Search*, 2nd Edition, Addison-Wesley, 2011.

REFERENCE BOOKS

1. **Stefan Büttcher, Charles L. A. Clarke, Gordon V. Cormack**, *Information Retrieval: Implementing and Evaluating Search Engines*, MIT Press, 2010.
2. **Bruce Croft, Donald Metzler, Trevor Strohman**, *Search Engines: Information Retrieval in Practice*, Pearson, 2015.
3. **Chakrabarti S.**, *Mining the Web: Discovering Knowledge from Hypertext Data*, Morgan Kaufmann, 2002.
4. **Tony Russell-Rose, Tyler Tate**, *Designing the Search Experience*, Morgan Kaufmann, 2012.

JOURNALS / MAGAZINES

1. **ACM Transactions on Information Systems (TOIS)** https://dl.acm.org/journal/tois
2. **Information Retrieval Journal (Springer)** https://www.springer.com/journal/10791
3. **Journal of the Association for Information Science and Technology (Wiley)** https://asistdl.onlinelibrary.wiley.com/journal/23301643

SWAYAM / NPTEL / MOOCs

1. **NPTEL – Information Retrieval (IIT Kharagpur)** https://nptel.ac.in/courses/106105158
2. **Coursera – Text Retrieval and Search Engines (University of Illinois)** https://www.coursera.org/learn/text-retrieval
3. **edX – Big Data and Information Retrieval (TU Delft)** https://www.edx.org/course/big-data-and-information-retrieval

Course Code: ISE2503	Course Title: Information Theory and Coding Type of Course: Theory	L- T-P- C	3	0	0	3
Version No.	1.0					
Course Pre-requisites	Essentials of AI					
Anti-requisites	NIL					

Course Description	This course introduces the fundamental concepts of information theory and error-correcting codes. It explores the mathematical basis of information measurement, source and channel coding, and their applications in communication and data storage systems. The course aims to provide a foundation for designing robust, efficient communication systems capable of withstanding noise and ensuring data integrity.			
Course Objective	To understand the quantitative measure of information and entropy. To explore source coding and lossless data compression techniques. To study error detection and correction codes for reliable communication. To analyze channel capacity and the trade-offs in coding strategies.			
Course Outcomes	On successful completion of the course, students will be able to: 1. Compute entropy, mutual information, and related information measures. 2. Apply source coding techniques for data compression. 3. Design and analyze error-correcting codes. 4. Evaluate the capacity of noisy channels and optimize coding schemes.			
Course Content:				
Module 1	**Introduction to Information Theory**	Assignment		**[8] Sessions**
Topics: Definition and properties of information, Entropy, joint and conditional entropy, Mutual information and relative entropy, Source models and the asymptotic equipartition property (AEP), Shannon's source coding theorem.				
Module 2	**Source Coding Techniques**	Assignment		**[7] Sessions**
Topics: Prefix-free codes and Kraft's inequality, Huffman coding, Shannon-Fano coding, Arithmetic coding and run-length encoding, Universal coding: Lempel-Ziv algorithms, Lossy compression: Rate-distortion theory.				
Module 3	**Channel Capacity and Noisy Channels**	Assignment		**[8] Sessions**
Topics: Discrete memoryless channels (DMC), Channel capacity and Shannon's channel coding theorem, Binary Symmetric Channel (BSC), Binary Erasure Channel (BEC), Capacity of Gaussian channels, Trade-offs between rate, bandwidth, and error probability.				

Module 4	Error Control Coding	Assignment		[7] Sessions
Topics: Linear block codes, generator and parity-check matrices, Hamming codes, cyclic redundancy check (CRC), Convolutional codes and Viterbi decoding, BCH and Reed-Solomon codes, Low-Density Parity-Check (LDPC) and Turbo Codes.				

REFERENCE MATERIALS

TEXTBOOKS

1. **Thomas M. Cover and Joy A. Thomas**, *Elements of Information Theory*, Wiley-Interscience, 2nd Edition, 2006.
2. **Simon Haykin**, *Digital Communication*, Wiley, 2001 – selected chapters.

REFERENCE BOOKS

1. **R. Bose**, *Information Theory, Coding and Cryptography*, McGraw Hill, 2nd Edition, 2007.
2. **Todd K. Moon**, *Error Correction Coding: Mathematical Methods and Algorithms*, Wiley, 2005.
3. **S. Lin and D.J. Costello**, *Error Control Coding*, Pearson Education, 2nd Edition, 2004.
4. **Robert G. Gallager**, *Information Theory and Reliable Communication*, Wiley, 1968.

JOURNALS / MAGAZINES

1. **IEEE Transactions on Information Theory** https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=18
2. **IEEE Transactions on Communications** https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=26
3. **Information Sciences (Elsevier)** https://www.sciencedirect.com/journal/information-sciences

---

SWAYAM / NPTEL / MOOCs

1. **NPTEL – Information Theory (IIT Madras)** – Prof. Andrew Thangaraj https://nptel.ac.in/courses/108106097

2. **NPTEL – Coding Theory (IIT Kanpur)** – Prof. R.K. Ghosh https://nptel.ac.in/courses/106104156
3. **edX – Information Theory (MIT)** https://ocw.mit.edu/courses/6-450-principles-of-digital-communications-i-fall-2006/
4. **Coursera – Introduction to Information Theory** https://www.coursera.org/learn/information-theory

Course Code: ISE2504	**Course Title: Image Processing and Computer Vision** **Type of Course: Theory**	**L- T-P- C**	**3**	**0**	**0**	**3**
Version No.	1.0					
Course Pre-requisites	Essentials of AI					
Anti-requisites	NIL					
Course Description	This course provides an introduction to the fundamental principles and techniques of digital image processing and computer vision. It emphasizes both theoretical foundations and practical applications including image enhancement, segmentation, feature extraction, object detection, and recognition, preparing students for real-world computer vision challenges in areas like robotics, medical imaging, and autonomous systems.					
Course Objective	To understand the fundamentals of digital images and image processing techniques. To explore computer vision algorithms for object detection, tracking, and classification. To implement practical solutions using modern image processing libraries and tools. To apply image and video analytics for real-time intelligent systems.					
Course Outcomes	On successful completion of the course, students will be able to: 1. Apply image enhancement and transformation techniques. 2. Analyze and segment images using appropriate algorithms. 3. Extract, match, and classify features from image data. 4. Implement computer vision models for detection, tracking, and recognition.					
Course Content:						

Module 1	**Fundamentals of Digital Image Processing**	Assignment		**[8] Sessions**

Topics:
Image formation and perception, ampling, quantization, and pixel relationships, Intensity transformations and histogram processing, Spatial filtering: smoothing and sharpening, Frequency domain processing using Fourier transforms.

Module 2	**Image Segmentation and Morphological Processing**	Assignment		**[7] Sessions**

Topics:
Thresholding, region growing, and watershed algorithms, Edge detection: Sobel, Prewitt, Canny, Morphological operations: dilation, erosion, opening, closing, Connected components labelling, Segmentation evaluation techniques.

Module 3	**Feature Extraction and Matching**	Assignment		**[7] Sessions**

Topics:

Texture and color features, Corner and blob detection: Harris, FAST, DoG, SIFT, SURF, Feature descriptors and matching (BRIEF, ORB, FLANN), Hough transform for lines, circles, Motion detection and optical flow.

Module 4	**Object Detection, Recognition, and Vision Applications**	Assignment		**[8] Sessions**

Topics:

Object detection using Haar cascades and HOG, Face detection and facial recognition, Deep learning in vision: CNNs and YOLO, Video surveillance and real-time tracking, Applications: Medical imaging, AR/VR, Robotics, Self-driving vehicles.

REFERENCE MATERIALS

TEXTBOOKS

1. **Rafael C. Gonzalez and Richard E. Woods**, *Digital Image Processing*, 4th Edition, Pearson Education, 2018.
2. **Richard Szeliski**, *Computer Vision: Algorithms and Applications*, Springer, 2nd Edition, 2022.
   https://szeliski.org/Book/

REFERENCE BOOKS

1. **Sonka, Hlavac, and Boyle**, *Image Processing, Analysis, and Machine Vision*, Cengage Learning, 4th Edition, 2014.
2. **Mark Nixon and Alberto Aguado**, *Feature Extraction and Image Processing for Computer Vision*, Academic Press, 3rd Edition, 2012.
3. **Adrian Rosebrock**, *Practical Python and OpenCV*, PyImageSearch Press, 2016.
4. **Gary Bradski and Adrian Kaehler**, *Learning OpenCV: Computer Vision with the OpenCV Library*, O'Reilly Media, 2nd Edition, 2016.

JOURNALS / MAGAZINES

1. **IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)**
   https://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=34
2. **Computer Vision and Image Understanding (Elsevier)**
   https://www.sciencedirect.com/journal/computer-vision-and-image-understanding
3. **International Journal of Computer Vision (Springer)**
   https://www.springer.com/journal/11263

SWAYAM / NPTEL / MOOCs

1. **NPTEL – Computer Vision (IIT Hyderabad)** – Prof. P.J. Narayanan
   https://nptel.ac.in/courses/106106210
2. **Coursera – Introduction to Computer Vision (Georgia Tech)**
   https://www.coursera.org/learn/computer-vision-basics
3. **edX – Fundamentals of Digital Image and Video Processing (Northwestern University)**
   https://www.edx.org/course/digital-image-and-video-processing
4. **Udacity – Intro to Computer Vision with OpenCV and Python**
   https://www.udacity.com/course/introduction-to-computer-vision--ud810

Course Code: ISE2505	**Course Title: Image Processing and Computer Vision Lab** **Type of Course: Lab**	**L- T-P- C**	**0**	**0**	**2**	**1**
Version No.	1.0					
Course Pre-requisites	Essentials of AI					
Anti-requisites	NIL					
Course Description	This lab course provides hands-on experience in the foundational techniques and applications of digital image processing and computer vision. Students will gain practical skills in reading, manipulating, and					

	analyzing images using Python, OpenCV, and deep learning frameworks to solve real-world visual tasks.
Course Objective	To provide practical exposure to image acquisition, processing, and enhancement To understand and implement key computer vision algorithms To apply feature extraction, segmentation, and classification techniques To develop small-scale vision-based applications using Python and OpenCV
Course Outcomes	Upon successful completion, students will be able to: 9. Apply image transformation and enhancement operations 10. Implement segmentation and object detection techniques 11. Extract and match visual features from images 12. Build vision-based projects using OpenCV and CNN models

List of Laboratory Tasks

Lab 1: Read and Display Images using OpenCV
Objective: Load and visualize images using OpenCV
Task: Read image files and convert between color spaces
Activity: Display images in grayscale, RGB, and HSV formats

Lab 2: Image Enhancement Techniques
Objective: Apply contrast and brightness enhancement
Task: Use histogram equalization and contrast stretching
Activity: Compare results on different image datasets

Lab 3: Perform Image Filtering and Smoothing
Objective: Reduce noise using filters
Task: Apply Gaussian, median, and bilateral filters
Activity: Observe effects of different kernels on noisy images

---

Lab 4: Edge Detection Techniques
Objective: Detect edges using gradient-based methods
Task: Use Sobel, Prewitt, and Canny edge detectors
Activity: Tune thresholds and visualize edge maps

Lab 5: Geometric Transformations of Images
Objective: Apply image transformation techniques
Task: Perform translation, rotation, scaling, and affine transformations
Activity: Warp an image using transformation matrices

Lab 6: Image Thresholding and Binarization
Objective: Segment images using intensity values
Task: Apply global, adaptive, and Otsu's thresholding
Activity: Visualize segmentation results and contours

Lab 7: Morphological Operations
Objective: Use structuring elements for shape-based processing
Task: Apply dilation, erosion, opening, and closing
Activity: Clean up binary images using morphological filters

Lab 8: Color-based Object Detection
Objective: Segment objects based on color
Task: Use HSV color space for object detection
Activity: Detect colored objects (e.g., red ball) in real-time

---

Lab 9: Feature Detection using SIFT and ORB
Objective: Extract and match image features
Task: Use SIFT and ORB for keypoint detection
Activity: Match features between two images and draw correspondences

---

Lab 10: Object Tracking in Videos
Objective: Track moving objects in video streams
Task: Implement object tracking using meanshift or CSRT
Activity: Track objects with bounding boxes in webcam input

Lab 11: Face Detection using Haar Cascades
Objective: Detect faces in images and videos
Task: Load pre-trained Haar classifiers
Activity: Draw bounding boxes around detected faces

Lab 12: Image Classification using Pre-trained CNNs
Objective: Classify images using deep learning

Task: Use MobileNet, VGG, or ResNet for classification
Activity: Load and predict labels for input images

Lab 13: Implement Background Subtraction
Objective: Separate foreground from background in video
Task: Use frame differencing or background subtraction algorithms
Activity: Detect motion and isolate moving objects

Lab 14: Create an Image Stitching Pipeline
Objective: Stitch overlapping images into panoramas
Task: Detect and match features, compute homography
Activity: Combine images to form a wide-view panorama

Lab 15: Mini Project – Build a Real-Time Computer Vision Application
Objective: Apply techniques learned to solve a real-world problem
Task: Choose a task (e.g., face mask detection, barcode scanner)
Activity: Develop and present the complete working prototype

REFERENCE MATERIALS

TEXTBOOKS

1. **Rafael C. Gonzalez, Richard E. Woods**, *Digital Image Processing*, 4th Edition, Pearson Education, 2018
2. **Richard Szeliski**, *Computer Vision: Algorithms and Applications*, Springer, 2022
   🔗 https://szeliski.org/Book/

REFERENCE BOOKS

1. **Sonka, Hlavac, and Boyle**, *Image Processing, Analysis, and Machine Vision*, Cengage Learning, 2014
2. **Adrian Rosebrock**, *Practical Python and OpenCV*, PyImageSearch Press
3. **Gary Bradski & Adrian Kaehler**, *Learning OpenCV*, O'Reilly, 2nd Edition, 2016

Ittagalpura, Rajanukunte, Yelahanka, Bengaluru 560 119

www.presidencyuniversity.in