



St. PETER'S INSTITUTE OF HIGHER EDUCATION AND RESEARCH
(Deemed to be University U/S 3 of the UGC Act, 1956)
AVADI, Chennai – 600 054. Tamil Nadu.

Phone: 044-26558080-84
E-mail: registrar@spiher.ac.in
Website: www.spiher.ac.in

B.Tech., (COMPUTER SCIENCE AND ENGINEERING)

(Approved by AICTE)

I to VIII Semester

**CURRICULUM AND SYLLABI
UNDER CHOICE BASED CREDIT SYSTEM**

Effective from the Academic Year 2025-2026

FACULTY OF ENGINEERING AND TECHNOLOGY

UNDER GRADUATE PROGRAMMES

REGULATIONS - 2025

CHOICE BASED CREDIT SYSTEM (CBCS)

Effective from the Academic Year 2025-2026



St. PETER'S INSTITUTE OF HIGHER EDUCATION AND RESEARCH

(Deemed to be university U/S 3 of UGC Act 1956)

Accredited with Grade "A+" by NAAC | ISO 9001:2015 Certified| Approved by AICTE

AVADI, Chennai - 600054

TABLES OF CONTENTS

Sl. No.	TITLE	PAGE No.
1	VISION AND MISSION OF THE INSTITUTION	i
2	ADMISSION	i
3	PROGRAMMES OF STUDY	ii
4	STRUCTURE OF THE PROGRAMME	ii
5	REGISTRATION AND ENROLLMENT	viii
6	REQUIREMENTS FOR APPEARING FOR THE END SEMESTER EXAMINATION OF A COURSE	xi
7	STUDENT COUNSELLING	xiii
8	CLASS COMMITTEE	xiv
9	EXAMINATIONS AND ASSESSMENT	xiv
10	EXAMINATIONS	xix
11	REQUIREMENTS FOR APPEARING FOR UNIVERSITY EXAMINATIONS	xxi
12	PASSING REQUIREMENTS FOR COMPLETION OF A COURSE	xxi
13	WITHDRAWAL FROM EXAMINATIONS	xxii
14	AUTHORIZED BREAK OF STUDY	xxiii
15	PURSUING COURSES IN OTHER INDIAN INSTITUTIONS AND ABROAD	xxiv
16	AWARD OF LETTER GRADES	xxiv
17	ELIGIBILITY FOR THE AWARD OF DEGREE	xxvii
18	RANKING	xxviii
19	DISCIPLINE	xxviii
20	STUDENT APPRAISAL	xxviii
21	DECLARATION OF RESULTS	xxviii
22	ACADEMIC BANK OF CREDITS (ABC)	xxviii
23	REVISION OF REGULATIONS / POWER TO MODIFY	xxix

I. PREAMBLE

As per the recommendations of UGC, St. Peter's Institute of Higher Education and Research (SPIHER) has introduced Choice Based Credit System (CBCS) from the academic year 2015-16. Along with Choice Based Credit System the institution also adopted Outcome based Education (OBE) from 2015-16 academic year, with more emphasis on modified academic curriculum to meet corporate needs. Open electives, credits for internship, and semester abroad program are the measures taken to induce prolific quality component into the system. Continuous evaluation system is further strengthened with 40-60 percentage weightage that is in place for internal and external examinations respectively.

SPIHER has always strived to be a pioneer in delivering quality education. SPIHER has taken incremental steps in the right direction to provide holistic development to students through its academic curriculum. The four verticals namely knowledge, skill, self-development and learning to learn are considered while designing the curriculum. The curriculum is designed to facilitate multi-disciplinary learning, experiential learning through Project Based Learning as part of the learning process.

II. DEFINITIONS AND NOMENCLATURE

PRELIMINARY DEFINITIONS & NOMENCLATURE

- i. **Degree:** Refers to the academic award conferred upon a student after the successful completion of the program within the stipulated period, fulfilling the required credits and prescribed procedures. The degree is an undergraduate program **Bachelor of Technology**, commonly referred to as **B.Tech**.
- ii. **Programme:** Refers to the undergraduate degree program in engineering or technology.
- iii. **Branch:** Denotes the specialization or discipline within the undergraduate degree program, such as Civil Engineering, Mechanical Engineering, etc.
- iv. **Course:** Represents a unit of study within a semester, including theory, practical, laboratory-integrated theory, seminar, internship, or project work. Examples include subjects like English, Mathematics, Environmental Science, Engineering Graphics, and Electronic Devices.

- v. **Institution:** Refers to **St. Peter's Institute of Higher Education and Research, Avadi, Chennai.**
- vi. **Academic Council:** The apex body responsible for all academic matters within the institution.
- vii. **Director (Academic Affairs):** The official responsible for implementing relevant academic rules and regulations across all academic activities.
- viii. **Controller of Examinations (CoE):** The official responsible for conducting examinations and declaring results.
- ix. **Head of the Department (HoD):** The head of the respective academic department.
- x. **Minor:** A discipline outside the student's major field of study, chosen for secondary specialization.
- xi. **UGC:** University Grants Commission.
- xii. **AICTE:** All India Council for Technical Education.
- xiii. **SWAYAM:** Study Webs of Active-Learning for Young Aspiring Minds—an Indian **Massive Open Online Course (MOOC)** platform.

ACADEMIC REGULATIONS 2025

Under Choice Based Credit System (CBCS)

1.0 VISION AND MISSION OF THE INSTITUTION

1.1 Vision:

To be a globally renowned institution in academic excellence, research and innovation by providing inspirational learning to produce socially conscious leaders capable of addressing future challenges with ethical values.

1.2 Mission:

- To provide a vibrant learning environment, fostering innovation and creativity inspired by cutting edge research.
- To instill ethical values, imbibe a sense of social responsibility and strive for societal wellbeing.
- To promote National and International alliances and collaborative initiatives to achieve global excellence.

2.0 ADMISSION

2.1 Candidates for admission to the first semester of the eight semester B. Tech. degree programme shall be required to have passed the Higher Secondary Examination of the 10+2 curriculum (Academic stream) prescribed by the appropriate authority or any other examination of any University or authority accepted by the Institution as equivalent thereto.

2.2 Candidate shall also write an entrance examination prescribed by the Institution for admission. The entrance examination shall test the proficiency of the candidate in the courses considered eligible for admission on the standards prescribed for 10+2 academic stream.

2.3 Candidates for admission to the third semester of the eight semester B.Tech. programme under lateral entry category shall be required to have passed minimum Three years / Two years (Lateral Entry) Diploma examination in any branch of Engineering / Technology or passed B.Sc. Degree from a recognized University as defined by UGC and passed 10+2 examination with Mathematics as a subject or

Passed three year Diploma of Vocation Stream (D.Voc) in the same or allied sector or any other examination of any other authority accepted by the Institution as equivalent thereto.

- 2.4 Multiple Entry options (Credit transfer through ABC), credit exemptions as per the direction of duly appointed expert committee in the respective department
- 2.5 The Institution shall offer suitable bridge courses in Mathematics, Physics, Engineering drawing, etc., for the students of diverse backgrounds.
- 2.6 The eligibility criteria such as marks, number of attempts and physical fitness shall be as prescribed by the Institution in adherence to the guidelines of regulatory authorities from time to time.
- 2.7 The duration of the programme for the Degree of Bachelor of Technology will be four academic years, with two semesters in each year. The duration of each semester will normally be 90 working days. However, a student may complete the programme at a slower pace by taking more time, but not more than seven years.

3.0 PROGRAMMES OF STUDY

Regulations are applicable to the following B.Tech. programmes in various branches of Engineering and Technology, each distributed over eight semesters, with two semesters per academic year.

S. No.	Programme	Discipline
1.	B.Tech.	Artificial Intelligence and Data Science
2.	B.Tech.	Biomedical Engineering
3.	B.Tech.	Civil Engineering
4.	B.Tech.	Computer Science and Engineering
5.	B.Tech.	Cyber Security
6.	B.Tech.	Electrical and Electronics Engineering
7.	B.Tech.	Electronics and Communication Engineering
8.	B.Tech.	Information Technology
9.	B.Tech.	Mechanical Engineering

4.0 STRUCTURE OF THE PROGRAMME

The detailed courses of study for a programme will be decided by the respective department's Board of Studies. As per NEP 2020, the structure and lengths of degree programmes are adjustable. The undergraduate degree will be of 4-year duration, with multiple entries/exit options as per AICTE/UGC guidelines.

The students are allowed to exit the programme after I or II or III or IV year with Undergraduate Certificate, Undergraduate Diploma, Undergraduate Degree (B.Sc) and Undergraduate B.Tech. respectively as per the regulations of NEP 2020, Government of India. Similarly, the students from other institutions can join SPIHER institution in the 3rd or 5th or 7th semester with an appropriate Undergraduate Certificate or Undergraduate Diploma or Undergraduate Degree Certificates respectively.

The 4-year multidisciplinary Bachelor's programme, however, shall be the preferred option since it allows the opportunity to experience the full range of holistic and multidisciplinary education in addition with the focus on the chosen major and minors as per the choices of the student. Every programme will have a curriculum with a syllabus consisting of theory, practical, Theory based practical, Project based theory, internship, project work, etc. for 161 credits.

4.1 Categorization of Courses

- i. **Humanities, Social Sciences and Management Courses (HSC)** include English for communication, Employability Skills, Engineering Ethics and Human Values and Management courses.
- ii. **Basic Science Courses (BSC)** include Mathematics, Physics, Chemistry, Biology, Environmental Science and Sustainability, etc.
- iii. **Basic Engineering Courses (BEC)** include Engineering Practices, Engineering Drawing, Basics of Civil / Electrical / Electronics / Mechanical / Computer Engineering, Instrumentation etc.
- iv. **Professional Core Courses (PCC)** include the core courses relevant to the chosen specialization/branch.
- v. **Professional Elective Courses (PEC)** include the verticals with elective courses and elective courses relevant to the chosen specialization/ branch.
- vi. **Open Elective Courses (OEC)** are Multidisciplinary courses that include the courses from Humanities and other disciplines of Engineering and Technology. Students can choose these courses from the list of Open Elective courses specified in the respective curriculum. Students may also choose courses from other disciplines from Swayam/NPTEL platform, including non-engineering courses.

- vii. **Project Courses (PC)** Includes Project Work and/or Internship, Career Development Skills, Creative and Innovative Project, Seminar, Professional Practices, Case Study and Industrial/Practical Training
- viii. **Mandatory Audit Courses (MAC)** Mandatory Audit Courses like Indian Constitution and Humanity Rights, NCC/YRC/NSS/Rotary Club, Yoga Behavioral Science and entrepreneurship and Startups are offered to all engineering programs of the Institution.
- The activities will include Practical / Field activities / Extension lectures. The activities shall be beyond class hours. The student participation shall be for a minimum period of 2 hours per week during the respective semester and the activities will be monitored by the respective faculty in charge.
 - Grades will be awarded on the basis of participation, attendance, performance and behavior. Grades shall be entered in the Grade statement as given below:
Very Good, Good, Satisfactory and Unsatisfactory
 - The Grades awarded by the faculty in-charge shall be entered in the respective semester Grade Sheet. If a student gets an unsatisfactory Grade, he/she has to repeat the above activity in the subsequent years.

4.1.2 **Online Courses for Credit Transfer**

The department shall approve the list of online courses offered by approved external agencies such as SWAYAM / NPTEL / MOOC. While listing the courses, the department shall consider the following points:

- a. The course evaluation is carried out by the same external agency
- b. Equivalent grading mechanism to be arrived at by the department

A student can register up to a maximum of 32 credits (total) as online courses during the entire programme of study. These shall be treated as Elective courses (programme elective or open elective). Students may be allowed to register for one course per semester. The student shall produce a Pass Certificate from the respective agencies. The credits(s) earned by the students will be transferred to the concerned course in the Grade Sheet.

4.1.3 Value Added Courses

The students are permitted to pursue department approved online courses (excluding courses registered for credit transfer) or courses offered / approved by the department as value added courses. The details of the value-added course viz., syllabus, schedule of classes and the course faculty shall be sent to the Director (Academic) for approval. The students may also undergo the valued added courses offered by other departments with the consent of the Head of the Department offering the course. These value-added courses shall be specified in the consolidated mark sheet as additional courses pursued by the student over and above the curriculum during the period of study

4.1.4 Industry Internship

The students shall undergo training for a period as specified in the curriculum during the summer vacation in any industry relevant to the field of study. The students are also permitted to undergo internship at research organizations / eminent academic institutions for the period prescribed in the curriculum during the summer vacation, in lieu of Industrial training. In any case, the student shall obtain necessary approval from the Head of the Department / Dean Academic and the training has to be taken up at a stretch.

4.1.5 Industrial Visit

The student shall undergo at least one industrial visit every year from the second year of the programme. The Heads of Departments / Dean Academic shall ensure the same.

4.2 CREDIT ASSIGNMENT FOR SEMESTER PROGRAM OF 15 WEEKS

Each course is normally assigned certain number of credits:

Lecture Hours (Theory)	1 Credit / 1 Lecture hour / week
Practical Hours	1 Credit / 2 Practical hours / week, 2 Credits / 4 Practical hours / week
Tutorial	1 Credit / 1 hour / week.
Courses with Project Based Learning Approach (PBLA)	1 Credit / 1 Lecture Hour / week
Project Work Phase I	6 Credits / 6 hours of project work (Phase-I) / week
Project Work Phase II	12 Credits / 18 hours of project work (Phase - II) / week
Internship/Entrepreneurship/ Consultancy/In plant training/	1 Credit / minimum 2 weeks during vacation

- 4.3** Each semester curriculum shall normally have a blend of lecture courses, laboratory courses, laboratory integrated theory courses, Project integrated theory courses, skill based courses etc.

4.3.1 Course Coordinator for Common Course

Each common theory course offered to more than one class or branch or group of branches, shall have a “course coordinator”. The course coordinator will be nominated by the Dean in consultation with respective Head of the Department. The Course Coordinator will be normally a senior faculty member who is one among the teachers teaching the course.

The “Course Coordinator” shall meet the teachers handling the course, as often as possible and ensure

- A common teaching methodology is followed for the course.
- The study materials are prepared by the staff members and communicated to the students periodically.
- The involvement of students in course-based projects and assignments.
- To prepare common question paper for continuous internal assessment tests.
- For uniform evaluation of continuous internal assessments answer sheets by arriving at a common scheme of evaluation.

The course coordinator is responsible for evaluating the performance of the students in the continuous internal assessments and end semester examinations and analyse them to find suitable methodologies for improvement in the performance. The analysis should be submitted to the HoD and Dean for suitable action.

- 4.4** The medium of instruction, examinations and project report shall be in English, except for courses in languages other than English.

- 4.5 ENROLLMENT FOR B.Tech. (Honours) and B.Tech. minor with specialization in another discipline. (OPTIONAL)**

4.5.1 B.Tech. (Hons.)

- a) The students should have taken additional courses from a specified group of Professional Electives (vertical) or from any of the verticals of the same programme and earned a minimum of 18 credits.
- b) Should have passed all the courses prescribed in the curriculum and additional courses in the first attempt.
- c) Should have earned a minimum of 7.50 CGPA taking into account of all the courses prescribed in the curriculum and additional courses.
- d) Lateral Entry students shall be permitted to register for the courses from Semester V onwards provided the students have earned a minimum CGPA of 7.50 until Semester III and have cleared all the courses in the first attempt.
- e) If a student decides not to opt for Honor's, after completing certain number of additional courses, such additional courses studied shall be considered instead of the Professional Elective courses which are part of the curriculum.

If the student has studied more number of such courses than the number of Professional Elective courses required as per the curriculum, the courses with higher grades shall be considered for the calculation of CGPA. Remaining courses shall be printed in the grade sheet, however, they will not be considered for calculation of CGPA and the same shall be indicated in a foot note appropriately.

If the student has failed in the additional courses or faced shortage of attendance, they will not be printed in the grade sheet and will not be considered for CPGA calculation and classification of degree.

4.5.2 B.Tech. Minor with specialisation in another discipline:

The student should have earned additionally a minimum of 18 credits in any one of the verticals offered from other Engineering Disciplines / Science and Humanities / Management.

- a) For these 18 credits students can optionally enroll and study a maximum of 6 credits in online mode from SWAYAM-NPTEL platform (in addition to the three online courses permitted for courses of curriculum), as approved by Head of the Department / Director Academic

- b) B.Tech. (Hons.) and B.Tech. minor with specialization in another discipline will be optional for students and the students shall be permitted to select any one of them only.
- c) For the category 4.5.2, the students, including Lateral Entry, will be permitted to register the courses from Semester V onwards provided the marks earned by the students until Semester III is CGPA 7.50 and above.
- d) B.Tech. (Hons.) or B.Tech. Minor shall be offered by the Department irrespective of the number of students enrolled.
- e) If a student decides not to opt for Minor, after completing certain number of courses, the additional courses studied shall be considered instead of Open Elective courses which are part of the curriculum.

If the student has studied more number of such courses than the number of open electives required as per the curriculum, the courses with higher grades shall be considered for calculation of CGPA. Remaining courses shall be printed in the grade sheet, however, they will not be considered for calculation of CGPA and the same shall be indicated in a foot note appropriately.

If the student has failed in the additional courses or faced shortage of attendance, they will not be printed in the grade sheet and will not be considered for CGPA calculation and classification of degree.

The student has to enroll for these additional courses separately and pay a tuition fee for studying these six additional courses and pay additional exam fee.

5.0 REGISTRATION AND ENROLLMENT

- 5.1** Each student, on admission, shall be assigned to a Mentor, who shall advise and counsel the student about the details of the academic programme and the choice of courses, considering the student's academic background and career objectives.
- 5.2** After registering for a course, a student shall attend the classes, satisfy the attendance requirements, earn continuous assessment marks and appear for the end semester examinations.

5.3 Each student on admission shall register for all the courses prescribed in the curriculum in the student's first Semester of study.

The enrollment for all the courses of curriculum from the Semesters II to VIII and additional courses for Honours and Minor from the semesters V and VIII will commence 5 working days prior to the commencement of the succeeding semester. The courses for Honours and Minor shall be registered separately under additional courses. The student shall enroll for the courses with the guidance of the student's Mentor. If the student wishes, the student may drop or add courses within 10 working days after the commencement of the concerned semester and complete the registration process duly authorized by the faculty in - charge within 30 days from the commencement of concerned semester. The list of students approved by the respective faculty-in-charge shall be final and would be considered for attendance, grades and calculation of CGPA and no changes shall be made thereafter.

5.4 For enrollment, a student MUST have

- I. Cleared all the Institute and Hostel dues of the previous semesters and the current semester fees.
- II. Not been debarred from registering for a specified period on disciplinary or any other ground.

5.5 Flexibility to Add or Drop courses:

5.5.1 A student has to earn the total number of credits specified in the curriculum of the respective programme of study in order to be eligible to obtain the degree. From the II to VII semesters, the student has the option of registering for additional courses or dropping existing courses in a semester. The total number of credits that a student can add or drop in a semester is limited to 8, subject to a maximum of 2 courses. Maximum number of credits enrolled in a semester (including Shortage of Attendance (SA), Honours and Minor) shall not exceed 30. The online courses registered shall be over and above this 30 credits.

5.5.2 If the student wishes to earn more than the total number of credits prescribed in the curriculum of the student's programme within the minimum duration of the programme, then he/she can enroll for such additional courses in any programme with the permission of Head of the Department to which student belongs and Head of the Department in which the course is offered by paying the examination fee. The credits earned will be neither considered for the computation of CGPA nor for the classification of the degree. The courses successfully completed will be printed in the grade sheet, however if there is shortage of attendance or failure, it shall neither be reflected in the grade sheet nor be considered for classification.

5.6 Choice of Professional Elective Courses

The professional Elective Courses are listed in the Curriculum in Table format as verticals (Specialisation groups). A student can choose all the Professional Elective Courses either from one of the verticals or a combination of courses from all verticals in a semester. However, students irrespective of enrolling for additional courses for B.Tech. (Hons.) are not permitted to choose more than one course from a row. Students are permitted to enroll more than one elective course from the same vertical in a semester. In the subsequent semesters students are permitted to enroll one more course in a row, provided if he/she has cleared the earlier course of the same row. For a professional elective course and open elective course, minimum number of students enrolment permitted shall be 10. However, the minimum number is not applicable for students enrolling B.Tech. (Hons) and B.Tech. Minor. For each professional elective course at least two choices shall be offered.

5.7 Redoing a Course

Redoing a Course refers to the process of re-registering for a course, attending all classes, meeting the attendance requirements as per Clause 6, obtaining fresh Continuous Assessment marks, and appearing for the End Semester Examinations. A student is required to redo a course under the following conditions.

- 5.7.1** If a student is prevented from writing end semester examination of any core course due to lack of attendance, the student has to register for that course again when offered next and redo the course.
- 5.7.2** If a student is prevented from writing the end semester examination of any professional/open elective course due to lack of attendance, the student can opt to register for the same course again when offered next and redo the course, or he/she can opt to register for a different professional/open elective course when it is offered, attend the classes, fulfill the attendance requirements as per clause 6, secure Continuous Assessment marks and appear for the End Semester Examinations.
- 5.7.3** If the course in which a student fails to secure a pass is a professional/open elective course, then the student can opt for a different professional/ open elective course, register for the same when it is offered, attend classes, fulfill the attendance requirements as per clause 6, secure Continuous Assessment marks and appear for End Semester Examinations.
- 5.7.4** A student who fails in Project work shall register for the course again, when offered next, and redo the course. In this case, the student shall attend the reviews and fulfill the attendance requirements as per clause 6.
- 5.7.5** A student who fails in Seminar / Case Study and Creative and Innovative project, where such other courses are evaluated through 100% continuous assessment, shall register for the same in the subsequent semester and redo the course. In this case, the student shall attend the classes and fulfill the attendance requirements as per clause 7 and earn continuous assessment marks.
- The student who fails in summer industrial training / internship shall attend the training / internship again and redo the course with the same organization or different organization with the approval of the HOD.

6.0 REQUIREMENTS FOR APPEARING THE END SEMESTER EXAMINATION OF A COURSE

A student who has fulfilled the following conditions (vide clause 6.1 and 6.2) shall be deemed to have satisfied the attendance requirements for appearing for the end semester examination of a particular course.

- 6.1** Ideally every student is expected to attend all periods and earn 100% attendance. However, the student shall secure not less than 75% attendance, course wise, taking into account the number of periods required for that course, as specified in the curriculum.
- 6.2** If a student secures attendance between 65% and less than 75% in any course in the current semester, due to medical reasons (hospitalization / accident / specific illness) or due to participation in the College / University / State / National / International level Sports events, with prior permission from the Chairman of Sports Board and Head of the Department concerned, the student shall be given exemption from the prescribed attendance requirement (75%) and the student shall be permitted to appear for the end semester examination of that course. A maximum of 10% shall be allowed under On Duty (OD) / Medical leave category. In all such cases, the students should submit the required documents on joining after the absence to the Head of the Department through the Faculty Coordinator. The HOD shall inform the course instructor to provide necessary attendance at the end of semester before finalizing attendance. Producing such documents while finalizing attendance at the end of semester shall not be accepted.
- 6.3** A student shall normally be permitted to appear for the end semester examination of the course if the student has satisfied the attendance requirements (vide Clause 6.1 – 6.2) and has registered for the examination in those courses of that semester by paying the prescribed fee.
- 6.4** Students who do not satisfy clause 6.1 and 6.2 and who secure **less than 65%** attendance in a course will not be permitted to write the end semester examination of that course. The student has to register and redo the course when it is offered next as per Clause 5.4. If the course in which the student has been prevented is a professional/ open elective, the student can opt to redo the same course or opt for different professional/ open elective course as per Clause 5.7.2.
- 6.5** If a student has shortage of attendance in all the registered courses of the current semester as per curriculum, he/she would not be permitted to move to the higher semester and has to repeat the current semester in the subsequent year.

- 6.6** In the case of reappearance (Arrear) registration for a course (the courses for which redo is not required), the attendance requirement as mentioned in Clauses 6.1 - 6.3 is not applicable. However, the student has to register for the examination in that course by paying the prescribed fee.
- 6.7** A student who has already appeared for a course in a semester and passed the examination is not entitled to reappear for the same course for improvement of letter grades / marks.

7.0 STUDENT COUNSELLING

To help students in planning their courses of study and for general advice on the academic programme and personal counselling, Faculty members are assigned.

7.1 MENTOR

To help the students in planning their courses of study and to render general advice regarding either the academic programme or any other activity, the Head of the Department concerned, will assign every year, a certain number of students from the first semester to a faculty member who will be called as Mentor. The set of students thus assigned will continue to be under the guidance of the Mentor till they complete the programme. Mentors will help the students on multiple exits, and also assess the proficiency of the student. Each student should have one-one interaction with the mentor at least once in a month.

7.2 FACULTY COORDINATOR

There is a Faculty Coordinator who will be the in-charge for a particular batch. He will coordinate with the mentors for assessing the proficiency of the batch and report to the Head of the Department. He will also collect the course registration forms from the students. He also ensures whether the student submitted feedback at the end of the semester for the courses he/she has taken.

8.0 CLASS COMMITTEE

a) Constitution of the Class Committee

For every class, a class committee shall be constituted by the Head of Department, as given below:

Chairman	A faculty member not teaching that particular class
Members	<ul style="list-style-type: none">• Faculty of all the courses of study• Four student members from the class to be nominated by the Head of the Department.

b) Functions of the Class Committee

- (i) The class committee shall meet thrice during the semester. The first meeting will be held within two weeks from the date of commencement of the semester in which the nature of the broad assessment procedure for the different courses will be discussed. The second and third meetings will be held six weeks and ten weeks respectively from the commencement of a semester to meaningfully interact and express opinions and suggestions to improve the effectiveness of teaching - learning process and analyze the performance of the students in the assessments. The chairperson of the class committee should send the minutes of the class committee meetings to the Dean through the Head of the Department, immediately after the meetings is over.
- (ii) During the first meeting of the class committee, all the faculty members shall give their course plan to the class committee chairperson/chairman for approval and uploading into the ERP.
- (iii) Any innovation in any course plan not agreed by the class committee or the HoD will be referred to the Dean for approval.

9.0 EXAMINATIONS AND ASSESSMENT

9.1 ASSESSMENTS

Continuous Internal Assessment

Continuous evaluation system is strengthened with 40-60 percentage weightage system in place for internal and external examinations. Three Continuous internal assessment will be conducted as per the academic calendar posted

in our institution website. Internal mark for every course is awarded based on the performance in Continuous Internal Assessment and the assignments submitted.

9.1.1 Theory Courses

- There will be a minimum of 2 Continuous Internal Assessments and 1 Model Test for each theory course.

DISTRIBUTION OF CONTINUOUS INTERNAL ASSESSMENT (CIA) MARKS FOR A THEORY COURSE			
Evaluation Component	Syllabus coverage	Duration of the Test	Max. Weightage (40 Marks)
CIA-1	First 1.5 Units of the syllabus	2.0 Hours	7.5 Marks
CIA-2	Second 1.5 Units of the syllabus	2.0 Hours	7.5 Marks
Model Test	Full syllabus	3 Hours	15 Marks
Assignment/ Mini Project (or) Group Presentation	Two written assignments for each course / Written quiz (or) Presentation of a written Report (or) Case study / Multiple choice Objective Type Test or Technical Project involving not more than 3 students (or) any other Group Presentation related to the course.		5 Marks
Attendance			5 Marks

- The continuous assessment marks obtained by the candidate in the first appearance shall be retained, considered and valid for all subsequent attempts, till the candidate secures a pass.

9.1.2 Practical Courses

S. No.	Category	Maximum Marks
1.	Record	15
2.	Observation work	10
3.	Model Examination	15
	Total	40

- For practical courses, the student will be evaluated on a continuous basis for 25 Marks (which will include performing all experiments, submitting observation and record note book in scheduled format and time), 15 marks for model exam at the end of the semester.

- For practical courses, if a student has been absent for some practical classes or has performed poorly, then the student will have to get permission from the lab in-charge and year coordinator to do the experiments, so that he/she meets all the requirements for the course and thereby allowed to appear for model and end semester practical exams
- If a student has not done all the experiments assigned for that lab, before the scheduled date will not be allowed to appear for the model and end semester practical exam. Such students will have to register the course again by doing all the experiments in the next semester when the course is offered.

9.1.3 End Semester Examinations (ESE)

- The end semester examinations shall be conducted at the end of the odd and even semester of the Academic year.
- End semester examinations will be conducted for a maximum of 100 marks. The marks secured in end semester exams will be converted to 60 marks.
- The evaluation of training will be made by a three member committee constituted by Head of the Department in consultation with Faculty Advisor and respective Training Coordinator. A presentation should be made by the student before the Committee, based on the Industrial Training or Professional Enrichment undergone.

Pattern of Question Paper (Theory) for Model and ESE

Particulars	Remarks
Maximum Marks	60 Marks
Duration	3 Hours
Part – A (Q.No. 1 to 10)	MCQ (10x1=10)
Part – B (Q.No. 11 to 15)	Short Answers (Either or Type) (5x10=50)

9.1.4 Project Work/ Semester long Internship

The student shall register for Project Work-I in pre-final semester and Project Work-II in final semester. Project work may be allotted to a single student or to a group of students not exceeding 4 per group. Project Work-II may/may not be a continuation of Project Work-I. If Project Work II is not a continuation of Project Work I, then the topic and constitution of the project team members need not be the same.

- The project review would be conducted by a review committee where the student/ team shall make a presentation on the progress made, before the

committee. The Head of the department shall constitute the review committee for each branch in consultation with Director Academic, approved by CoE. The members of the review committee will evaluate the progress of the project and award marks.

- The guides would evaluate the students based on their performance and follow up.
- For Project work out of 100 marks, the maximum marks for Continuous Internal Assessment are fixed as 40 and the End Semester Examination (project report evaluation and viva-voce examination) carries 60 marks.

There shall be **Three Continuous Internal Assessments** (each 100 marks) during the semester by a review committee. The student shall make presentation on the progress made before the committee. The Head of the Department shall constitute a review committee for each programme. There shall be a minimum of three members in the review committee. The committee shall consist of the supervisor, expert member from the department and a project co-ordinator from another department. The total marks obtained in the three Reviews shall be reduced to 40 marks.

Continuous Internal Assessment (40 Marks)				End Semester Examinations (60 Marks)			
Review 0	Review I	Review II	Review III	Project Report & Presentation		Viva-Voce Examination	
10	10	10	10	Supervisor	External	Internal	External
				20	20	10	10

- A student is expected to attend all the project reviews conducted by the institution on the scheduled dates. It is mandatory for every student to attend the reviews, even if they are working on a project in an industry, which is outside Chennai city. If a student does not attend any of the project reviews, he / she shall not be allowed for the successive reviews and thereby not allowed to appear for the final viva voce.
- The candidate is expected to submit the project report as per the guidelines of the institution on or before the last day of submission. If a candidate fails to submit the project report on or before the specified deadline, he/she can be granted an extension of time up to a maximum limit of 5 days for the submission of project work, by the Head of the Department.

- If he/she fails to submit the project report, even beyond the extended time, then he/she is deemed to have failed in the project work and shall register for the same in the subsequent semester and re-do the project after obtaining permission from the HoD and the respective Deans.

9.1.5 Assessment for Summer internship:

The summer Industrial / Practical Training/ summer internship/ summer project shall carry 100 marks and shall be evaluated through continuous assessment only. At the end of the summer Industrial / Practical Training/ summer internship/ summer project, the student shall submit a certificate from the organization where the student has undergone training and a brief report about the training. The evaluation will be made based on this report, presentation and a Viva-Voce Examination conducted by a three-member Departmental Committee constituted by the Head of the Department consisting of one co-ordinator and two faculty members. Certificates (issued by the Organization) submitted by the student shall be attached to the mark list and sent to the Controller of Examinations by the Head of the Department. The evaluation shall be carried out as per the procedure shown below.

Internship / Industrial Training		
Evaluation Marks (60)		
Report	Presentation	Viva Voce
40	10	10

9.1.6 Assessment for Online Courses

Students may be permitted to credit two online courses (which are provided with certificate), subject to a maximum of six credits. The online course of 3 credits can be considered instead of one elective course. These online courses shall be chosen from the SWAYAM platform, provided the offering organization conducts regular examination and provides marks. The credits earned shall be transferred and the marks earned shall be converted into grades and transferred, provided the student has passed in the examination as per the norms of the offering organization. The details regarding online courses taken up by the student and marks/credits earned

and the approval for the course from Concerned Head of the Department shall be sent to the Controller of Examinations, in the subsequent semester(s) along with the details of the elective(s) to be dropped.

9.2 ASSESSMENT WEIGHTAGE:

There will be Continuous Internal Assessment and End Semester Examination for all courses of all programmes.

(i) Theory courses

Continuous Internal Assessment : 40 Marks

End Semester Examination : 60 Marks

(ii) Practical courses

Continuous Internal Assessment : 40 Marks

End Semester Examination : 60 Marks

(iii) Theory + Practical courses

Continuous Internal Assessment (Average of Theory and Practical) : 40 Marks

End Semester Examination (Average of Theory and Practical) : 60 Marks

10.0 EXAMINATIONS

10.1 RE-EXAMINATION

Re-examination requests shall be considered only for the Continuous Internal Assessment and the Examinations in the last instructional week of the semester based on medical reasons.

10.2 REVALUATION

A candidate can apply for revaluation of his/her End semester examination answer paper in a theory course, immediately after the declaration of results, on payment of a prescribed fee through the ERP. The Controller of Examinations will arrange for the revaluation and the result will be intimated to the candidate through website.

Revaluation is not permitted for practical courses and for project work.

10.3 SCRIBE FOR EXAMINATION

Divyangjan students or students with temporary physical disability or injury due to accident or illness can apply for a scribe (writer) with proof of disability as a medical certificate obtained from a Registered Medical Officer. The student shall be assigned a scribe by CoE to such student. The application for the scribe should be submitted in the CoE office well in advance or at least 2 days before the examination, to make necessary arrangements (Scriber, Separate Examination Hall etc.). The scribe assigned shall neither be a student nor a degree holder of any technical programme having similar competency.

Divyangjan students/ students with reading or writing disability, who can write at a slower speed as compared to a normal student would be allowed an extra time of 30 minutes to write the examination for each course. The proof of disability and application of extra time has to be submitted to the CoE office well in advance or 3 days before the start of the examination.

10.4 ACADEMIC MALPRACTICE

Academic malpractice would be strictly prohibited and any student who is found indulging in such activity would be penalized as per the recommendations of the Malpractice Committee constituted by the CoE with the approval of the Director Academic. The Committee would inquire and decide on the action based on the norms and policy listed in the Examination Manual.

10.5 SUPPLEMENTARY EXAMINATION

Supplementary examination will be conducted only for the final semester students within 10 days from the date of publication of revaluation results for students who have backlogs to complete the programme. Only such students shall apply with the prescribed fee to the Controller of Examinations within the stipulated time.

11.0 REQUIREMENTS FOR APPEARING FOR UNIVERSITY EXAMINATIONS

A student shall normally be permitted to appear for the University Examinations for all the courses registered in the current semester if he/she has satisfied the semester completion requirements. Further, examination registration by a student is mandatory for all the courses in the current semester and all arrear(s) course(s) for the University examinations failing which, the student will not be permitted to move to the higher semester. A student who has already appeared for any course in a semester and passed the examination is not entitled to reappear in the same subject for improvement of grades.

12.0 PASSING REQUIREMENTS FOR COMPLETION OF A COURSE

- A candidate who secures not less than 50% of total marks prescribed for the courses (Continuous Assessment + End semester examination) with a minimum of 40% of the marks prescribed for the end-semester Examination in theory, theory with practical components (40% individually in theory and laboratory) and practical courses (including Project work), shall be declared to have passed in the Examination. However, if a student fails in any integrated theory and practical course, he/she should register and appear for the End semester examination in both theory and practical components of this course.

- If a student fails to secure a pass in a theory course / theory with laboratory/laboratory course (except electives), the student shall register and appear only for the end semester examination in the subsequent semester. In such case, the continuous assessment marks obtained by the candidate in the first appearance shall be retained and considered valid for all subsequent attempts till the candidate secure a pass. However, from the third attempt (current semester's end semester examination is considered as the first attempt) onwards if a candidate fails to obtain pass marks (IA + End Semester Examination), then the candidate shall be declared to have passed the examination if he/she secure a minimum of 50% marks prescribed for the university end semester examinations alone.

- If a student has submitted the project report but absent in the end semester examination of project work, the student is deemed to be failed. In this case and also if a student attends and fails in the End semester examination of Project work of B.Tech, he/she shall attend end semester examination again within 60 days from the date of declaration of the results. The subsequent viva-voce examination will be considered as reappearance with payment of exam fee. In case, the student fails in the subsequent viva-voce examination also, the student shall redo the course again, when offered next.
- If a student is absent during the viva - voce examination, it would be considered as fail. If a student fails to secure a pass in Project Work-I, the student shall register for the course again in the subsequent semester and can-do Project Work-I and II together.
- The passing requirement for the courses which are assessed only through continuous assessment, shall be fixed as minimum 50%.

13.0 WITHDRAWAL FROM EXAMINATIONS

- A candidate may, for valid reasons, (medically unfit / unexpected family situations) be granted permission to withdraw from appearing for the examination in any course or courses in any one of the semester examination during the entire duration of the degree programme.
- Withdrawal application shall be valid only if the candidate is otherwise normally eligible (if he/she satisfies Attendance requirements and should not be involved in Disciplinary issues or Malpractice in Exams) to write the examination and if it is made within FIVE days before the commencement of the examination in that course or courses and also recommended by the Director Academic through HoD.
- Notwithstanding the requirement of mandatory FIVE days' notice, applications for withdrawal for special cases under extraordinary conditions will be considered based on the merit of the case.
- Withdrawal shall not be considered as an appearance for deciding the eligibility of a candidate for the purpose of Classification of Degree.
- Withdrawal is NOT permitted for arrears examinations of the previous semesters.

14.0 AUTHORIZED BREAK OF STUDY

- This shall be granted by the Institution, only once during the full duration of study, for valid reasons for a maximum of one year during the entire period of study of the degree programme.
- A candidate is normally not permitted to temporarily break the period of study. However, if a candidate would like to discontinue the programme temporarily in the middle of duration of study for valid reasons (such as accident or hospitalization due to prolonged ill health), he / she shall apply through the Director Academic in advance (Not later than the Reopening Day of that semester) through the Head of the Department stating the reasons. He /She should also mention clearly, the Joining date and Semester for Continuation of Studies after completion of break of Study. In such cases, he/she will attend classes along with the Junior Batches. A student who availed break of study has to rejoin only in the same semester from where he/she left.
- The total period for completion of the programme shall not exceed more than 10 consecutive semesters from the time of commencement of the course irrespective of the period of break of study in order that he / she may be eligible for the award of the degree.
- If any student is not allowed to appear for End Semester Examinations for not satisfying Academic requirements and Disciplinary reasons, (Except due to Lack of Attendance), the period spent in that semester shall NOT be considered as permitted 'Break of Study' and is NOT applicable for Authorized Break of Study.
- In extraordinary situations, a candidate may apply for additional break of study not exceeding another one Semester by paying prescribed fee for break of study. Such extended break of study shall be counted for the purpose of classification of First Class Degree.
- If the candidate has not reported back to the department, even after the extended Break of Study, the name of the candidate shall be deleted permanently from the institution enrolment. Such candidates are not entitled to seek readmission under any circumstances.

- This shall be granted by the Institution, only once during the full duration of study, for valid reasons for a maximum of one year during the entire period of study of the degree programme.

15.0 PURSUING COURSES IN OTHER INDIAN INSTITUTIONS AND ABROAD

- A student can be selected, to get Professional Exposure in his/her area of Expertise in any Reputed Research Organization or Educational Institution of repute or any Universities in India and abroad.
- This is possible only with the List of Research Organizations, Educational Institutions in India and abroad approved by the Academic Council.
- The student can have the option of spending not more than three to Six months in the Final year or Pre - final year of his/her Degree. During this period, the student can do his/her Project work or register for courses which will be approved by the Class Committee and Director Academic, under the Guidance of a Project Supervisor who is employed in the Organization and Co-guided by a staff member from our Institution.
- Credit Transfer can be done by the CoE on submission of certificate through the HoD and Director Academic within 15 days of completion.
- The students who undergo training outside the Institution (either in India or Abroad) is expected to abide by all Rules and Regulations to be followed as per Indian and the respective Country Laws, and also should take care of Financial, Travel and Accommodation expenses.

16.0 AWARD OF LETTER GRADES

All assessments of a course will be done on absolute marks basis. However, for the purpose of reporting the performance of a candidate, letter grades, each carrying certain number of points, will be awarded as per the range of total marks (out of 100) obtained by the candidate in each course as detailed below:

RANGE OF MARKS FOR GRADES

Range of Marks	Letter Grade	Grade Point
90 -100	O	10
80 – 89	A+	9
70 – 79	A	8
60 – 69	B+	7
50 – 59	B	6
00-49 (Reappear)	F	0

ABSENT	AAA	0
Withdrawal	W	0
Authorised Break of Study	ABS	0

16.1 CUMULATIVE GRADE POINT AVERAGE CALCULATION

The CGPA calculation on a 10 Point scale is used to describe the overall performance of a student in all courses from first semester to the last semester. RA, AAA and W grades will be excluded for calculating GPA and CGPA.

$$\text{GPA} = \frac{\sum_{i=1}^N C_i GP_i}{\sum_i C_i} \qquad \text{CGPA} = \frac{\sum_{i=1}^N C_i GP_i}{\sum_i C_i}$$

Where

C_i – Credits for the course

GP_i – Grade Point for the course

i – Sum of all courses successfully cleared during all the semesters

n - Number of all courses successfully cleared during the particular semester in the case of GPA and during all the semesters in the case of CGPA

16.2 GRADE SHEET

After revaluation results are declared in each semester, Grade Sheets will be issued to each student. At the end of programme a consolidated grade sheet also will be issued to each student. The grade sheet and consolidated grade sheet will contain the following details:

- The programme and degree in which the candidate has studied
- The list of courses enrolled during the semester and the grade secured
- The Grade Point Average (GPA) for the semester.

16.3 CLASSIFICATION OF DEGREE AWARDED

Final Degree is awarded based on the following

Range of CGPA	Classification of Degree
≥ 7.50	First Class with Distinction
$\geq 6.00 < 7.50$	First Class
$\geq 5.00 < 6.0$	Second Class

Minimum requirements for award of Degree: A student should have obtained a minimum of 5.0 CGPA.

- A candidate who qualifies for the award of the Degree having passed the examination in all the courses of all the 8 semesters in his/her first appearance within a maximum of 10 consecutive semesters securing a overall CGPA of not less than 7.5 (Calculated from 1st semester) shall be declared to have passed the examination in **First Class with Distinction**. Authorized Break of Study vide Clause 14, will be considered as an Appearance for Examinations, for award of First Class with Distinction. Withdrawal shall not be considered as an appearance for deciding the eligibility of a candidate for First Class with Distinction

- A candidate who qualifies for the award of the Degree having passed the examination in all the courses of all the 8 semesters within a maximum period of 10 consecutive semesters after his/her commencement of study securing a overall CGPA of not less than 6.0 (Calculated from 1st semester), shall be declared to have passed the examination in **First Class**. Authorized break of study vides Clause 14 (if availed of) or prevention from writing End semester examination due to lack of attendance will not be considered as Appearance in Examinations. For award of First class, the extra number of semesters than can be provided (in addition to four years for Normal UG programme) will be equal to the Number of semesters availed for Authorized Break of Study or Lack of Attendance. Withdrawal shall not be considered as an appearance for deciding the eligibility of a candidate for First Class.
- All other candidates who qualify for the award of the Degree having passed the examination in all the courses of all the 8 semesters within a maximum period of 10 consecutive semesters after his/her commencement of study securing a overall CGPA of not less than 5.0, (Calculated from 1st semester) shall be declared to have passed the examination in **Second Class**.
- A candidate who is absent in semester examination in a course/project work after having registered for the same, shall be considered to have appeared in that examination for the purpose of classification.

17.0 ELIGIBILITY FOR THE AWARD OF DEGREE

A student shall be declared to be eligible for the award of the Certificate / Diploma / UG Degree / UG Honours degree, provided the student has successfully completed all the requirements of the programme, and has passed all the prescribed examinations in all the I/II/III/IV year respectively within the maximum period specified in clause 2.7.

- i) Successfully gained the required number of total credits as specified in the curriculum corresponding to his/her programme within the stipulated time.
- ii) Successfully completed the programme requirements and has passed all the courses prescribed in all the semesters within a maximum period of 5 years reckoned from the commencement of the first semester to which the candidate was admitted.
- iii) Successfully completed any additional courses prescribed by the Institution.
- iv) has earned a CGPA of not less than 5

- v) has no dues to the Institution, Library, Hostels, etc.,
- vi) has no disciplinary action pending against him / her.

18.0 RANKING

- A candidate who qualifies for the UG degree programme passing all the examinations in the first attempt, within the minimum period prescribed for the programme of study from semester I through semester VIII to the programme shall be eligible for ranking. Such ranking will be confirmed to 10 percent of the total number of candidates qualified in that particular programme of study subject to a maximum of 10 ranks.

19.0 DISCIPLINE

- Every student is required to observe disciplined and decorous behavior both inside and outside the Institution and not to indulge in any activity which will tend to bring down the prestige of the Institution. If a student indulges in malpractice in any of the end semester theory / practical examination, continuous assessment examinations he/she shall will be liable for disciplinary action as prescribed by the Institution from time to time.

20.0 STUDENT APPRAISAL

- It is mandatory for every student to submit the feedback on each and every course, he/she has undergone, at the end of every semester.

21.0 DECLARATION OF RESULTS

- The End Semester Examination results will be declared in institution website and the same is shared with the Head of the Department. In general, the results will be declared within 15 days from the date of last examination.

22.0 ACADEMIC BANK OF CREDITS (ABC)

- All the students who admitted in any one of the above programmes are mandatory to register in the Academic Bank of Credits (ABC) portal provided by the Ministry of Education (MoE), Government of India.

23.0 REVISION OF REGULATIONS / POWER TO MODIFY

- St. Peter's Institute of Higher Education and Research (Deemed to be University) may revise, amend, or modify the regulations, examination schemes, and syllabi as deemed necessary from time to time.

Notwithstanding the provisions stated above, the Academic Council holds the authority to alter any or all of these regulations as required, subject to approval by the Executive Council.

Dean Engg.

Director Academic

Registrar



St. PETER'S INSTITUTE OF HIGHER EDUCATION AND RESEARCH

(Deemed to be University U/S 3 of the UGC Act, 1956)

AVADI, Chennai – 600 054. Tamil Nadu.

Phone: 044-26558080-84

E-mail: registrar@spiher.ac.in

Website: www.spiher.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Vision

To achieve technical excellence in Computer Science and Engineering education, imbued with moral values that will enable the students to meet the societal needs scientifically and ethically.

Mission

- To impart quality education in the domain of Computer Science and Engineering at the Undergraduate and Postgraduate levels.
- To foster Entrepreneurship and innovative skills integrated with moral values among the students to make them successful engineers.
- To encourage quality Research and Development activities that would benefit society.

Program Outcomes (POs)

PO1: Engineering Knowledge: Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop to the solution of complex engineering problems.

PO2: Problem Analysis: Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)

PO3: Design/Development of Solutions: Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WK5)

PO4: Conduct Investigations of Complex Problems: Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. (WK8).

PO5: Engineering Tool Usage: Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. (WK2 and WK6)

PO6: The Engineer and The World: Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5, and WK7)

PO7: Ethics: Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9)

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

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

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

PO11: Life-Long Learning: Recognize the need for, and have the preparation and ability for i) independent long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broad technological change. (WK8)

Program Educational Objectives (PEOs)

PEO1: Apply theoretical and practical knowledge to analyze complex real-world industry problems and offer innovative technical solutions.

PEO2: Embrace and adapt to emerging technologies and industry trends to remain effective and relevant in the professional world.

PEO3: Communicate effectively and perform responsibilities efficiently as an individual, a team member, or an entrepreneur in diverse professional settings.

PEO4: Pursue continuous learning and research to build a successful career, and actively contribute through teaching, mentoring, and knowledge sharing.

PEO5: Evaluate the societal and environmental impact of technical solutions to promote sustainable development and contribute to the nation's economic growth.(WK1 to WK4)

Program Specific Outcomes (PSOs)

PSO1: Apply theoretical and practical knowledge gained in the emerging areas of computing to develop innovative and effective solutions.

PSO2: Uphold ethical values while contributing effectively in professional settings and promote social well-being.

Contribution

1:Reasonable

2:Significant

3:Strong

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

**UG PROGRAM (CBCS) – B.Tech. Computer Science and Engineering
(2025–2026 Batch and onwards)**

Semester I

Semester	Course Code	Title of the Course	Course Category	NCrF level	Objective s and outcomes		SDG Goal	Instruction Hours / week			Credit(s)	Marks		
					PEOs	POs		L	T	P		CIA	ESE	Total
I	25 MAU108	Engineering Mathematics- I	BSC	5.5	1,2	1-4	4	3	1	0	4	40	60	100
I	25PHU121	Engineering Physics	BSC	4.5	1,2	1,2,3,5,	6	3	0	2	4	40	60	100
I	25EVS001	Environmental Science	BSC	4.5	1,2	1-6	3, 13	2	0	0	2	40	60	100
I	25ENU121	Technical English	HSC	4.5	1,2	1-5	4,8	3	0	2	4	40	60	100
I	25EEU101	Basics of Engineering	BEC	4.5	1,2	1-5	7,9,11,12	3	0	0	3	40	60	100
I	25CSU121	Programming for Problem Solving	BEC	4.5	1,2	1-5	4,5,8,9,10,17	2	0	4	4	40	60	100
I	25MEU101	Engineering Graphics	BEC	4.5	1,2	1,2,3,5,11	4,9,12	2	1	0	3	40	60	100
I	25MAC111	NCC/ YRC/ NSS/ Rotary Club	MAC	4.5	-	8,9	1-3, 5-7,10-15	0	0	2	0	-	-	-
Semester Total											280	420	700	

Semester II

Semester	Course Code	Title of the Course	Course Category	NCrF level	Objective s and outcomes		SDG Goal	Instruction Hours / week			Credit(s)	Marks		
					PEOs	POs		L	T	P		CIA	ESE	Total
II	25MAU208	Engineering Mathematics II	BSC	4.5	1,2	1-4	4	3	1	0	4	40	60	100
II	25CHU221	Engineering Chemistry	BSC	4.5	1,2	1-5	4,6	3	0	2	4	40	60	100
II	25ECU223	Digital System Design	BEC	4.5	1,2	1-5	3,4,7,9,11	2	0	2	3	40	60	100
II	25CSU221	Object Oriented Programming	PCC	4.5	1,2	1-5	4,5,8,9,10,17	3	0	2	4	40	60	100
II	25CSU222	Data Structures	PCC	4.5	1,2	1-5	4,8,9	3	0	2	4	40	60	100
II	25IKS001	Introduction to Indian Knowledge System	IKS	4.5	1,2	7,9	3-5,10,16	2	0	0	2	40	60	100
II	25MEU211	Design Thinking and Innovations Laboratory	BEC	4.5	1,2	1-6,11	4,8,9,11	0	0	4	2	40	60	100
II	25MAC201	Indian Constitution and Human Rights	MAC	4.5	1,2	7,8,9	4,16	2	0	0	2	40	60	100
Semester Total											320	480	800	

Semester III

Semester	Course Code	Title of the Course	Course Category	NCrF level	Objectives and outcomes		SDG Goal	Instruction Hours / week			Credit(s)	Marks		
					PEOs	POs		L	T	P		CIA	ESE	Total
III	25MAU306	Discrete Mathematics	BSC	5.0	1,2	1-4	4	3	1	0	4	40	60	100
III	25UHV001	Universal Human Values and Ethics	HSC	5.0	1,2	7,8,9,11	3,4,5,10,16	2	0	0	2	40	60	100
III	25CSU301	Computer Architecture	PCC	5.0	1,2	1-5,12	4, 8, 9, 12, 13	3	0	0	3	40	60	100
III	25CSU302	Embedded Systems and IoT	PCC	5.0	1,2	1-5,12	4,7,8,9, 11, 13	3	0	0	3	40	60	100
III	25CSU321	Operating Systems	PCC	5.0	1,2	1-5	4, 8, 9	3	0	2	4	40	60	100
III	25CSU322	Database Management Systems	PCC	5.0	1,2	1-5	4, 8, 9	3	0	2	4	40	60	100
III	25CSU323	Design and Analysis of Algorithms	PCC	5.0	1,2	1-5	4, 8, 9	3	0	2	4	40	60	100
III	25MAC311	Yoga	MAC	5.0	1,2	7,8,9,11	3,4,5,10,16	0	0	1	0	-	-	-
Semester Total											280	420	700	

Semester IV

Semester	Course Code	Title of the Course	Course Category	NCrF level	Objectives and outcomes		SDG Goal	Instruction Hours/week			Credit(s)	Marks		
					PEOs	POs		L	T	P		CIA	ESE	Total
IV	25CSU401	Artificial Intelligence and Machine Learning	PCC	5.0	1,2	1-5, 12	4, 8, 9, 11,13, 16	3	0	0	3	40	60	100
IV	25CSU402	Fundamentals of Data Science	PCC	5.0	1,2	1-5, 12	4, 8, 9, 11, 13, 16	3	0	0	3	40	60	100
IV	25CSU421	Computer Graphics and Multimedia	PCC	5.0	1,2	1,2, 3,5	4,8,9, 17	3	0	2	4	40	60	100
IV	25CSU422	Computer Networks	PCC	5.0	1,2	1-5	4, 8, 9, 11	3	0	2	4	40	60	100
IV	25CSU411	Artificial Intelligence and Data Science Laboratory	PCC	5.0	1,2	1-5, 12	4, 8, 9, 11, 13, 16	0	0	4	2	40	60	100
IV	25CSU491	Mini Project	PRO	5.0	1,2	1-5, 9-12	4,8,9, 11,12, 13,17	0	0	4	2	40	60	100
IV	25IKS002	Vedic Mathematics and Critical Thinking	IKS	5.0	1,2	1-4	4	2	0	0	2	40	60	100
Semester Total											280	420	700	

Semester V

Semester	Course Code	Title of the Course	Course Category	NCrF level	Objectives and outcomes		SDG Goal	Instruction Hours / week			Credit(s)	Marks		
					PEOs	POs		L	T	P		CIA	ESE	Total
V	25CSU521	Object Oriented Software Engineering	PCC	5.5	1,2	1-5, 9,12	4, 8, 9, 12, 17	2	0	4	4	40	60	100
V	25CSU522	Compiler Design	PCC	5.5	1,2	1-5	4, 8, 9	3	0	2	4	40	60	100
V	25CSU523	Cloud Computing	PCC	5.5	1,2	1-6, 11	4, 8, 9, 11,12, 13	3	0	2	4	40	60	100
V		Professional Elective-1	PEC	5.5				3	0	0	3	40	60	100
V		Professional Elective-2	PEC	5.5				3	0	0	3	40	60	100
V		Open Elective-1	OEC	5.5				3	0	0	3	40	60	100
V	25CSU581	Internship-1	IAS	5.5		1-3,5, 8-11	4,8,9, 11,17	0	0	0	1	40	60	100
V	25MAC501	Entrepreneurship and Startups	MAC	5.5	1,2	1-11	4,5,8-11,17	3	0	0	3	40	60	100
Semester Total											320	480	800	

Semester VI

Semester	Course Code	Title of the Course	Course Category	NCrF level	Objective s and outcomes		SDG Goal	Instruction Hours / week			Credit(s)	Marks		
					PEOs	Pos		L	T	P		CIA	ESE	Total
VI	25CSU621	Internet Programming	PCC	5.5	1,2	1-5,9,11	4, 8, 9, 11, 17	3	0	2	4	40	60	100
VI	25CSU622	Mobile Application Development	PCC	5.5	1,2	1-5,9,11	4,8,9,11,17	3	0	2	4	40	60	100
VI	25CSU623	Full Stack Development	PCC	5.5	1,2	1-5,9,10,11	4, 8, 9, 11, 17	3	0	2	4	40	60	100
VI	25CSU611	Competitive Coding	PCC	5.5	1,2	1-5,11	4,8,9	0	0	2	1	40	60	100
VI		Professional Elective-3	PEC	5.5				3	0	0	3	40	60	100
VI		Professional Elective-4	PEC	5.5				3	0	0	3	40	60	100
VI		Open Elective 2 (Online)	OEC	5.5				3	0	0	3	40	60	100
Semester Total												280	420	700

Semester VII

Semester	Course Code	Title of the Course	Course Category	NCrF level	Objectives and outcome		SDG Goal	Instruction Hours / week			Credit(s)	Marks		
					PEOs	POs		L	T	P		CIA	ESE	Total
VII	25CSU721	Cryptography and Cyber Security	PCC	6.0	1, 2	1-7, 11	4,8,9, 11,16	3	0	2	4	40	60	100
VII		Professional Elective 5	PEC	6.0				3	0	0	3	40	60	100
VI		Professional Elective 6	PEC	6.0				3	0	0	3	40	60	100
VII		Open Elective 3	OEC	6.0				3	0	0	3	40	60	100
VII	25CSU791	Project Work - Phase I	PRO	6.0	1, 2	1-11	4,8,9, 11, 12, 13, 16, 17	0	0	12	6	40	60	100
VII	25CSU781	Internship-2	IAS	6.0	1, 2	1-3,5, 8-11	4,8,9, 11, 17	0	0	0	1	40	60	100
Semester Total											240	360	600	

Semester VIII

Semester	Course Code	Title of the Course	Course Category	NCrF level	Objective s and outcome		SDG Goal	Instruction Hours / week			Credit(s)	Marks		
					PEOs	POs		L	T	P		CIA	ESE	Total
VIII		Open Elective 4 (Online)	OEC	6.0	1,2			3	0	0	3	40	60	100
VIII	25CSU891	Project Work - Phase II	PRO	6.0	1,2	1-11	4,8,9,11,12,13,16,17	0	0	24	12	40	60	100
Semester Total												80	120	200

Program Elective Courses (PE)

Programme Electives				
Sl. No.	Course Code	Course Title	Prerequisite Courses	Credits
Program Elective – I (V Semester)				
1	25CSU501A	Social Network Security	None	3
2	25CSU501B	Data Mining and Warehousing	DBMS	3
3	25CSU501C	2 D and 3 D Animation	Computer Graphics and Multimedia	3
4	25CSU501D	Principles of Programming Languages	Programming for Problem Solving, Data Structures	3
5	25CSU501E	Nano Science and Technology	Engineering Physics, Engineering Chemistry	3
Program Elective – II (V Semester)				
6	25CSU502A	Ethical Hacking	Computer Networks	3
7	25CSU502B	Software Defined Networks	Computer Networks, Operating Systems	3
8	25CSU502C	Immersive Technologies	Computer Graphics and Multimedia	3
9	25CSU502D	Software Engineering with Agile and Scrum	Object Oriented Programming, Data Structures	3
10	25CSU502E	Robotics	Engineering Mathematics, Basics of Engineering	3
Program Elective – III (VI Semester)				
11	25CSU601A	Digital Forensics	None	3
12	25CSU601B	Storage Technologies	Computer Architecture, Operating System	3
13	25CSU601C	Video and Audio Editing	Computer Graphics	3
14	25CSU601D	Automation and Testing	Object Oriented Software Engineering	3
15	25CSU601E	Big Data Analytics	DBMS, Data Structures	3
Program Elective – IV (VI Semester)				
16	25CSU602A	Intellectual Property Rights	None	3
17	25CSU602B	Security and Privacy in Cloud	Cloud Computing	3
18	25CSU602C	UI and UX Design	None	3

19	25CSU602D	Application Development using Devops	DBMS	3
20	25CSU602E	Linux Technologies	None	3
Program Elective – V (VII Semester)				
21	25CSU701A	Enhanced Security for Cloud	Cloud Computing	3
22	25CSU701B	Edge and Fog Computing	Cloud Computing	3
23	25CSU701C	Game Development	None	3
24	25CSU701D	Full Stack Web Development using GenAI	Full stack development	3
25	25CSU701E	Digital Technologies for Marketing	None	3
Program Elective – VI (VII Semester)				
26	25CSU702A	Cryptocurrency and Blockchain Technologies	None	3
27	25CSU702B	Multi-Cloud Strategies	Cloud Computing	3
28	25CSU702C	AI for Game Programming	Artificial Intelligence	3
29	25CSU702D	Web Frameworks	Internet Programming	3
30	25CSU702E	Quantum Computing	Discrete Mathematics	3

Open Elective Courses (OE)

Open Electives				
Sl. No.	Course Code	Course Title	Prerequisite Courses	Credits
1	25CSU041	Programming in Core Java	None	3
2	25CSU042	Web Designing Technologies	None	3
3	25CSU043	Mobile Computing Technologies	None	3
4	25CSU044	E – Commerce	None	3
5	25CSU045	Applications of Immersive Technologies	None	3

Guidelines for framing the Syllabus

1. Regulation / Scheme of evaluation to be included at the beginning of the curriculum book
2. Program Outcome / Programme Specific Outcome to be placed before the curriculum page
3. Course Objective and Course Outcome to be included at the top of each course
4. Course outcome must start with students will be able to ...
5. Course objective and outcome to be framed using Bloom's Taxonomy of Educational Objectives
6. All the units should have equal weightage with subtopics in detail
7. Care must be taken to include recent topics in the relevant course

Instruction Hours / week: L: 3 T: 1 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- Develop the uses of matrix algebra techniques that engineers need for practical applications.
- Differentiate continuity and differentiability under differential calculus.
- Identify functions of several variables. This is required in many branches of engineering.
- Solve the problems under integral calculus.
- Acquaint the student with mathematical tools needed in evaluating multiple integrals and their applications.

Course Outcomes (COs)

At the completion of the course, the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Use the matrix algebra methods for solving practical problems	Apply
CO2	Use differential calculus ideas on several variable functions	Apply
CO3	Apply the concept of several variable functions in calculus	Understand
CO4	Apply the concept of integral calculus	Apply
CO5	Apply multiple integral ideas in solving areas, volumes, and other applications	Apply

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	--	--	--	--	--	--	--	--	--	2	--	--
CO2	3	2	--	--	--	--	--	--	--	--	--	2	--	--
CO3	3	2	--	--	--	--	--	--	--	--	--	2	--	--
CO4	3	2	--	--	--	--	--	--	--	--	--	2	--	--
CO5	3	2	--	--	--	--	--	--	--	--	--	2	--	--

1 - low, 2 - medium, 3 - high

Unit I – MATRICES

Eigenvalues and Eigenvectors of a real matrix – Characteristic equation – Properties of Eigen values and Eigen vectors – Cayley-Hamilton theorem – Diagonalization of matrices by orthogonal transformation – Reduction of a quadratic form to canonical form by orthogonal transformation – Nature of quadratic forms – Applications: Stretching of an elastic membrane.

Unit II – DIFFERENTIAL CALCULUS

Representation of functions – Limit of a function – Continuity – Derivatives – Differentiation rules (sum, product, quotient, chain rules) – Implicit differentiation – Logarithmic differentiation – Applications: Maxima and Minima of functions of one variable.

Unit III – FUNCTIONS OF SEVERAL VARIABLES

Partial differentiation – 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 – Applications: Maxima and minima of functions of two variables, Lagrange's method of undetermined multipliers.

Unit IV – INTEGRAL CALCULUS

Definite and Indefinite integrals — Substitution rule — Techniques of Integration — Integration by parts, Trigonometric integrals, Trigonometric substitutions, Integration of rational functions by partial fraction, Integration of irrational functions — Improper integrals.

Unit V – MULTIPLE INTEGRALS

Double integrals – Change of order of integration – Double integrals in polar coordinates – Area enclosed by plane curves – Triple integrals – Volume of solids – Change of variables in double and triple integrals – Applications: Moments and centres of mass, moment of inertia.

SUGGESTED READINGS

1. Kreyszig, E. (2016). Advanced Engineering Mathematics. 10th Edition, John Wiley and Sons.
2. Grewal, B.S. (2018). Higher Engineering Mathematics. 44th Edition, Khanna Publishers.
3. Bali, N., Goyal, M., & Watkins, C. (2009). Advanced Engineering Mathematics. 7th Edition, Firewall Media.
4. Jain, R.K. & Iyengar, S.R.K. (2016). Advanced Engineering Mathematics. 5th Edition, Narosa Publications.
5. Narayanan, S. & Manicavachagom Pillai, T.K. (2009). Calculus, Volume I and II, S. Viswanathan Publishers.

25PHU121

ENGINEERING PHYSICS

Semester – I
5H – 4C

Instruction Hours / week: L: 3 T: 0 P: 2

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- Provide a foundational understanding of the electrical properties of materials
- Introduce the fundamental concepts and behaviour of semiconductor materials
- Develop a conceptual and mathematical understanding of elasticity
- Explain the thermal properties of engineering materials
- Examine experimental evidence such as electron diffraction.

Course Outcomes (COs)

At the completion of the course, the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Discuss the basic electrical properties of materials and classify materials based on band theory.	Understand, Apply
CO2	Explain the properties of semiconductor materials and determine the band gap using appropriate experimental methods.	Understand, Apply
CO3	Calculate different moduli of elasticity and explain their applications in engineering and materials science.	Apply
CO4	Describe the thermal properties of materials and their applications, such as thermal expansion in joints and the functioning of heat exchangers.	Remember, Apply
CO5	Interpret the concept of wave-particle duality and describe experimental evidence, such as electron diffraction, that supports this duality.	Understand

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	3	3	3	3	3	3	2	3	3	3	3	2
CO2	3	3	3	3	3	3	3	2	3	3	3	3	2
CO3	3	3	3	3	3	3	3	2	3	3	3	3	2
CO4	3	3	3	3	3	3	3	2	3	3	3	3	2
CO5	3	3	3	3	3	3	3	2	1	2	3	3	2

1 - low, 2 - medium, 3 - high

Unit-I ELECTRICAL PROPERTIES OF MATERIALS

Classical free electron theory-Expression for Electrical conductivity-thermal conductivity-expression-Wiedmann Franz law- success and failure-electrons in metals-Particle in three dimensional box-degenerate state- Fermi Dirac Statistics-Density of Energy states-Electron in periodic potential-Bloch Theorem- Metals and Insulators-Energy bands in solids-Effective mass of electron- Concept of holes.

Unit II SEMICONDUCTORS AND TRANSPORT PHYSICS

Intrinsic semiconductors- Carrier concentration derivation– Fermi level – variation of Fermi level with temperature –electrical conductivity – band gap determination -extrinsic semiconductors - Carrier concentration in N-type & P-type semiconductors – Variation of fermi level with temperature and impurity concentration.

Unit III PROPERTIES OF MATTER

Elasticity – Poisson’s ratio and relationship between moduli (qualitative) - stress-strain diagram for ductile and brittle materials, uses - factors affecting elastic modulus and tensile strength - bending of beams - cantilever - bending moment - Young’s modulus determination - theory and experiment - uniform and non-uniform bending - I shaped girders - twisting couple torsion pendulum - determination of rigidity modulus- moment of inertia of a body .

Unit IV THERMAL PHYSICS

Transfer of heat energy – thermal expansion of solids and liquids – expansion joints - bimetallic strips - thermal conduction, convection and radiation – heat conductions in solids – thermal conductivity - Forbe’s and Lee’s disc method: theory and experiment - conduction through compound media (series and parallel) – thermal insulation – applications: heat exchangers, refrigerators, ovens and solar water heaters.

Unit V QUANTUM PHYSICS

Black body radiation – Planck’s theory (derivation) – Compton effect: theory and experimental verification – wave particle duality – electron diffraction – concept of wave function and its physical significance – Schrödinger’s wave equation – time independent and time dependent equations – particle in a one-dimensional rigid box – tunnelling (qualitative) - scanning tunneling microscope.

SUGGESTED READINGS

- 1.Charles Kittel – Introduction to Solid State Physics, 8th Edition (2018) Publisher: Wiley
- 2.Brij Lal and N.Subramaniam, Properties of Matter S. Chand & Co., New Delhi (1994)
- 3.G. Aruldas's Quantum Mechanics is the Second Edition, published by PHI Learning in 2008.
- 4,Donald A. Neamen's Semiconductor Physics and Devices: Basic Principles is the 4th Edition, published in 2012 by McGraw-Hill.
- 5.Halliday, D., Resnick, R. & Walker, J. "Principles of Physics". Wiley, 2015.
- 6.R. Shankar's Principles of Quantum Mechanics is the Second Edition, published in 1994 by Plenum Press
- 7.Dr. S. Stella Mary, 'Practical Engineering Physics' R. K. Publications, 2013
- 8.C.C. Ouseph, U.J. Rao, V. Vijayendran, 'Practical Physics and Electronics', S. Viswanathan Printers and Publishers Pvt. Ltd., 2011

25EVS001

ENVIRONMENTAL SCIENCE

Semester – I
2H – 2C

Instruction Hours / week: L: 2 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- Understand the scope and significance of the environment, raise public awareness about various environmental hazards and the structure and function of ecosystems
- Introduce the concept of biodiversity, its different types and the importance of its conservation at global, national, and local levels.
- Understand the causes, effects, and control measures of various environmental hazards, solid waste and disaster management, role of individuals in pollution prevention.
- Understand the need for new and renewable energy sources, focusing on energy management and conservation, and their applications.
- Understand the concepts of global and local environmental issues, various environmental protection laws

Course Outcomes (COs)

At the completion of the course, the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Define the environment and its significance, different environmental hazards, and the roles of producers, consumers, and decomposers in ecosystems, energy flow, and the structure of food chains, food webs, and ecological pyramids in various ecosystems.	Understand
CO2	Define biodiversity and its various levels, biodiversity hotspots, threats to biodiversity, and the importance of conserving endangered and endemic species in India using in-situ and ex-situ methods.	Understand
CO3	Identify the causes, effects control of different environmental hazards (air, water, marine, soil, noise, thermal, and nuclear pollution), importance of solid waste management and disaster management (floods, earthquakes, cyclones, and landslides), the role of individuals in preventing pollution, and pollution case studies.	Remember
CO4	Explore the role and potential of new and renewable energy sources, different types of renewable energy and their applications, particularly hydrogen, ocean, tidal, and geothermal energy, the concepts and technology behind energy management and conservation.	Understand
CO5	Identify key environmental issues and the role of environmental protection laws in safeguarding ecosystems, wildlife, and forests.	Understand

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	1	3	1	2	2	2	1	1	1	1	2	2	2
CO2	3	2	3	1	2	2	2	1	1	1	1	3	2	3
CO3	2	2	3	1	2	1	2	1	1	1	1	2	3	2
CO4	3	1	2	1	2	2	2	1	1	1	1	3	2	3
CO5	3	2	3	1	1	1	2	1	1	1	1	2	3	3

1 - low, 2 - medium, 3 - high

Unit I – ENVIRONMENT AND ECOSYSTEM

Environment – Definition, scope and significance - Public awareness: Risk and hazards - Chemical hazards, Physical hazards, biological hazards in the environment. Ecosystem - concept -structure and function - producers, consumers and decomposers - Food chain - Food web - Ecological pyramids - Energy flow - Forest, Grassland, desert and aquatic ecosystem

Unit II – BIODIVERSITY AND ITS CONSERVATION

Introduction to Biodiversity - Definition - genetic, species and ecosystem diversity - Values and uses of biodiversity - biodiversity at global, national (India) and local levels - Hotspots, threats to biodiversity - Endangered and endemic species of India - conservation of biodiversity - *In-situ* & *Ex-situ*.

Unit III – ENVIRONMENTAL POLLUTION AND MANAGEMENT

Definition, Causes - Effects and control measures of Air, Water, Marine, soil, Noise, thermal and nuclear hazards, Solid waste Management: Causes, effects and control measures of urban and industrial wastes- Role of an individual in prevention of pollution- Pollution case studies- Disaster management: floods, earthquake, cyclone and landslides

Unit IV – RENEWABLE SOURCES OF ENERGY

Role and potential of new and renewable sources- Energy management and conservation, New Energy Sources: Need of new sources. Different types of new energy sources. Applications of- Hydrogen energy, Ocean energy resources, Tidal energy conversion. Concept, origin, and power plants of geothermal energy

Unit V – ENVIRONMENTAL PROTECTION

Climate change- Global, Regional and local environmental issues. Environmental Impact Assessment. Environment protection act, wildlife protection act., and Forest Conservation Act.

SUGGESTED READINGS

1. Gilbert M. Masters "Introduction to Environmental Engineering and Science", 2nd edition, Pearson Education (2004).
2. Benny Joseph, "Environmental Science and Engineering", Tata McGraw-Hill, New Delhi (2006).
3. Trivedi. R.K., "Handbook of Environmental Laws, Rules, Guidelines, Compliances and Standards", Vol. I and II, Enviro Media, 3rd edition, BPB publication (2010).
4. Anubha Kaushik and C. P. Kaushik's "Perspectives in Environmental Studies", 6th Edition, New Age International Publishers, 2018.
5. Allen, D. T. and Shonnard, D. R., Sustainability Engineering: Concepts, Design and Case Studies, Prentice Hall.
6. Bradley. A.S; Adebayo, A.O., Maria, P. Engineering applications in sustainable design and development, Cengage learning
7. Environment Impact Assessment Guidelines, Notification of Government of India, 2006.
8. Mackenthun, K.M., Basic Concepts in Environmental Management, Lewis Publication, London, 1998.
9. Dharmendra S. Sengar, 'Environmental law', Prentice hall of India PVT. LTD, New Delhi, 2007.
10. Rajagopalan, R, 'Environmental Studies-From Crisis to Cure', Oxford University Press, 2005
11. Erach Bharucha "Textbook of Environmental Studies for Undergraduate Courses" Orient Blackswan Pvt. Ltd. 2013

25ENU121

TECHNICAL ENGLISH

Semester – I
5H – 4C

Instruction Hours / week: L: 3 T: 0 P: 2

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- Enhance the communicative competence of learners.
- Assist learners in using language effectively in academic/work contexts.
- Strengthen students' English language skills by engaging them in listening, speaking, and grammar learning activities that are relevant to authentic contexts.
- Develop analytical thinking skills for problem-solving in communicative contexts
- Equip them with writing skills needed for academic as well as workplace contexts.

Course Outcomes (COs)

At the completion of the course, the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Listen and comprehend complex academic texts.	Remember
CO2	Read and infer the denotative and connotative meanings of technical texts.	Apply
CO3	Write definitions, descriptions, narrations, and essays on various topics.	Apply
CO4	Speak fluently and accurately, and informal communicative contexts.	Apply
CO5	Express their opinions effectively in both oral and written medium of communication.	Create

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	3	3	3	3	3	3	2	3	3	3	3	3
CO2	3	3	3	3	3	3	3	2	3	3	3	3	3
CO3	3	3	3	3	3	3	3	2	3	3	3	3	3
CO4	3	3	3	3	3	3	3	2	3	3	3	3	3
CO5	3	3	3	3	3	3	3	2	3	1	2	3	3

1 - low, 2 - medium, 3 - high

Unit I INTRODUCTION TO COMMUNICATION SKILLS

Listening—for general information, specific details, conversation: Introduction to classmates. Speaking - Self Introduction; Introducing a friend; Conversation - politeness strategies; Telephone conversation. Reading - Reading brochures (technical context). Writing-Writing emails/letters introducing oneself, Paragraph Writing, Reading Comprehension. Grammar – Parts of Speech, Sentence kinds. Wh-Questions forms and Tags. Vocabulary-Synonyms; One word substitution; Abbreviations & Acronyms (as used in technical contexts).

Unit II: REPORTING AND NARRATIONS

Listening- Listening to podcasts, anecdotes/stories/event narration; documentaries and interviews. Speaking- Narrating personal experiences/events; Interviewing a celebrity; Reporting and summarizing of documentaries/podcasts/interviews. Reading- Reading biographies, travelogues, newspaper reports, Excerpts from literature, travel, and technical blogs. Writing – Report Writing - Short Report on an event. Grammar- Sentence Structures, Tenses. Vocabulary– Antonyms, Word Formation (prefixes & suffixes).

Unit III: ACADEMIC DEVELOPMENT AND COMMERCIAL REVIEWS

Listening- Listen to a classroom lecture. Speaking–Picture description; Instructing to use the product; Presenting a product and summarizing a lecture. Reading – Reading advertisements, gadget reviews, user manuals. Writing - Writing definitions; Instructions. Grammar-Active & Passive Voice, The Impersonal Passive., Subject-Verb Agreement; Infinitive and Gerunds. Vocabulary -Compound Words, Homonyms; and Homophones.

Unit IV: SCIENTIFIC REPORTS AND PRESENTATION TECHNIQUES

Listening – Listening to TED Talks; Scientific lectures and educational videos. Speaking – Small Talk; Mini presentations and making recommendations. Reading–Newspaper articles; Journal reports–and Non-Verbal Communication (tables, pie charts, etc.). Writing–Writing recommendations; Transferring information from non-verbal (chart, graph, etc), to verbal mode, Checklists. Grammar–Error correction; If conditional sentences., Vocabulary- Discourse markers, Connectives, Articles.

Unit V: POINT OF VIEW AND PLACEMENTS.

Listening–Listening to debates/discussions; different viewpoints on an issue; and panel discussions. Speaking–Group discussions, Debates, and Expressing opinions through Simulations & Role play. Reading – Reading Editorials and Opinion Blogs. Writing–Job/ application–Cover letter & Resume. Grammar–Numerical adjectives, Punctuation. Vocabulary- Cause & Effect Expressions

PRACTICAL EXERCISES

1. Group Discussion: Practical based on Accurate and Current Grammatical Patterns.
2. Conversational Skills for Interviews under suitable Professional Communication Lab conditions with emphasis on Kinesics
3. Communication Skills for Seminars/Conferences/Workshops with emphasis on Paralinguistics/ Kinesics. Presentation Skills for Technical Paper/Project Reports/ Professional Reports based on proper Stress and Intonation Mechanics.
4. Official/Public Speaking based on suitable Rhythmic Patterns.
5. Argumentative Skills/Role Play Presentation with Stress and Intonation

SUGGESTED READINGS

1. English for Engineers & Technologists Orient Blackswan Private Ltd. Department of English, Anna University, (2020 edition)
2. English for Science & Technology Cambridge University Press, 2021.
3. Technical Communication–Principles and Practices by Meenakshi Raman & Sangeeta Sharma
4. Dr.S.Uma Maheswari. English Workbook for Engineers and Technologists
5. Lakshmi Narayanan, Course Book on Technical English

25EEU101

BASICS OF ENGINEERING

Semester – I
3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100

End Semester Exam: 3 Hours

Course Objectives

- To understand the basic calculations and measurements in DC circuits.
- To familiarize with working and characteristics of different DC and AC machines.
- To impart knowledge on the fundamentals of measuring electrical and electronic quantities, various sensors and transducers to measure non-electrical quantities.
- Demonstrate the fundamentals and scope of Mechanical Engineering, covering its core principles, key domains, and emerging technologies
- Identify basic and modern construction materials, explain their engineering properties,

Course Outcomes (COs)

At the completion of the course the student will be able to

Cos	Course Outcomes	Blooms Level
CO1	Perform the basic calculations in DC circuits and measure the various quantities associated with DC circuits.	Understand
CO2	Choose appropriate motor for specific applications based on the motor characteristics	Analyze
CO3	Analyze the functional blocks of a measurement system and the principles of various electrical and electronic instruments,	Analyze
CO4	Describe the scope of Civil Engineering and identify basic and modern construction materials along with their properties	Understand
CO5	Distinguish between different Steams of Mechanical Engineering and to gain foundational knowledge of mechanical systems, tools, and applications.	Understand

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	–	2	2	–	–	–	–	–	2	3	1	3
CO2	3	3	–	2	2	–	–	–	–	–	2	3	1	3
CO3	3	3	–	3	3	–	–	–	–	–	2	3	2	3
CO4	2	1	–	–	–	2	2	–	–	–	1	–	3	2
CO5	2	1	–	–	–	–	2	–	–	–	1	–	3	2

1 - low, 2 - medium, 3 - high

Unit I: DC CIRCUITS AND MEASUREMENTS

The concept of voltage and current-Electric circuit elements: R, L, C – Independent and dependent sources – Ohm's law- Kirchhoff's law- series and parallel resistive circuits – Voltage and current division – Star-delta transformation - Mesh and nodal analysis of resistive circuits – simple problems - Measurement of voltage, current, and power in DC circuits.

Unit II: ELECTRICAL MACHINES

Construction, principle of operation, basic equations, characteristics and applications of DC generators, DC motors, single-phase transformers, and single-phase induction motors. Working principle of BLDC Motor and its applications in home appliances.

Unit III: ELECTRICAL AND ELECTRONIC INSTRUMENTATION

Functional blocks of a measurement system - types of measurements - Direct and indirect measurements – Classification of instruments – Induction type – dynamometer type wattmeters- Types of indicating Instruments Principles of Electrical Instruments – Multimeters, Oscilloscopes - Static and Dynamic characteristics of an instrumentation system – Errors in Measurement – Calibration and Standards. Classification of Transducers: Resistive, Inductive, Capacitive, Thermoelectric, piezoelectric, photoelectric, Hall Effect – electromagnetic flow transducers

Unit IV: INTRODUCTION TO CIVIL ENGINEERING AND MATERIALS

Introduction to Civil Engineering- Basic Construction Materials- Properties of Engineering Materials- Selection of Materials for Construction- Modern Materials in Construction

Unit V SCOPE AND CORE PRINCIPLES OF MECHANICAL ENGINEERING

Design, Manufacturing, Materials, Energy and Power Systems, Kinematics and Robotics, Instrumentation and Control, Emerging Trends, and Smart Applications.

SUGGESTED READINGS

- 1.D P Kothari and I.J Nagarath, "Basic Electrical and Electronics Engineering", McGraw Hill Education (India) Private Limited, Third Reprint, 2016.
- 2.Giorgio Rizzoni, "Principles and Applications of Electrical Engineering", McGraw Hill Education (India) Private Limited, 2010.
- 3.S.K.Bhattacharya, "Basic Electrical and Electronics Engineering", Pearson India, 2011.
- 4.Del Toro, "Electrical Engineering Fundamentals", Pearson Education, New Delhi, 2015.
- 5.Leonard S Bobrow, "Foundations of Electrical Engineering", Oxford University Press, 2013.
- 6.Rajendra Prasad, "Fundamentals of Electrical engineering", Prentice Hall of India, 2006.
- 7.Mittle N., "Basic Electrical Engineering", Tata McGraw Hill Edition, 24th reprint 2016.
- 8.Sawhney, A. K., and Puneet Sawhney "A Course in Electrical and Electronic Measurements and Instrumentation" Dhanpat Rai & Company, 2016.
9. Jonathan Wickert & Kemper Lewis, An Introduction to Mechanical Engineering Cengage Learning Publication (Enhanced SI Edition, 4th Edition), 2021.
10. Author: S.K. Duggal Publication : New Age International Publishers Edition: 4th or 5th edition

25CSU121

PROGRAMMING FOR PROBLEM SOLVING

Semester – I
6H – 4C

Instruction Hours / week: L: 2 T: 0 P: 4

Marks: Internal: 40 External: 60 Total: 100

End Semester Exam: 3 Hours

Course Objectives

- Develop the foundational understanding of problem-solving techniques, algorithm design, and programming basics using C and Python.
- Apply conditional and iterative constructs effectively for developing logical, flow-controlled programs in both C and Python.
- Impart knowledge of function-based and modular programming approaches for creating structured, maintainable, and reusable code.
- Equip students with the ability to manipulate arrays, strings, and lists, and apply fundamental searching and sorting algorithms in C and Python.
- Introduce memory management concepts through pointers in C, and provide practical skills in file handling and understanding Python's memory model.

Course Outcomes (COs)

At the completion of the course, the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Construct basic programs using variables, operators, and input/output functions in C and Python	Remember
CO2	Execute decision-making and looping structures to solve common computational problems	Apply
CO3	Assemble modular programs by defining reusable functions with appropriate parameter usage and scope control	Analyze
CO4	Implement basic searching and sorting algorithms to process structured data arrays, strings, and lists	Create
CO5	Demonstrate the use of pointers and dynamic memory in C, and operate file handling and memory reference concepts in C and Python	Evaluate

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	2	2	2	2	2	2	1	2	1	3	2
CO2	3	2	2	2	2	2	3	2	1	2	1	3	2
CO3	3	2	2	2	2	2	3	2	1	1	1	3	2
CO4	3	2	2	2	2	2	3	2	1	2	1	3	2
CO5	3	2	2	2	2	2	2	2	1	1	1	3	2

1 - low, 2 - medium, 3 - high

Unit I: INTRODUCTION TO PROGRAMMING

Introduction to problem solving, algorithms, and flowcharts, Programming structure in C and Python. Data types, variables, constants, Operators and expressions (arithmetic, relational, logical, assignment, Input/output functions: scanf, printf (C); input(), print() (Python). Type conversion and casting

Unit II: CONTROL FLOW AND ITERATIVE STATEMENTS

Decision-making: if, if-else, nested if, switch-case (C); if-elif-else (Python). Looping constructs: while, for, do-while (C); while, for-in (Python), Loop control: break, continue, pass. Problem-solving using loops (e.g., sum of digits, reverse number, pattern printing)

Unit III: FUNCTIONS AND MODULAR PROGRAMMING

Defining and calling functions in C and Python, Function parameters, return types, recursion, Python-specific: default arguments, keyword arguments, lambda functions, Variable scope and storage classes, Modular programming: creating reusable code blocks

Unit IV: ARRAYS, STRINGS, LISTS

Arrays in C: 1D and 2D, basic operations, Strings in C: declaration, input/output, string.h functions, Python lists: indexing, slicing, built-in methods, list comprehensions, Python strings: methods, slicing, immutability, Searching and sorting algorithms (linear, binary search; bubble, selection sort)

Unit V: POINTERS (C) AND FILE HANDLING IN C AND PYTHON

Pointers in C-declaration and initialization- Pointers and arrays-Pointers and functions (call by reference) Pointers and structures, Dynamic memory allocation: malloc(), calloc(), realloc(), free(), File Handling in C and Python, Python memory model, Mutable vs immutable objects, Function argument passing (by object reference) using id() to understand memory behavior

PRACTICAL EXERCISES (C and Python):

1. Write a Program to convert Celsius to Fahrenheit and vice versa. (Practice: I/O, arithmetic operators, conditionals)
2. Write a Program to simulate Simple Calculator that Perform addition, subtraction, multiplication, and division based on user input. (Practice: switch-case or if-elif-else.)
3. Write a program to check whether a given number is Odd or Even (Practice: conditionals, modulo operator).
4. Write a program to find Factorial of a Number using both iterative and recursive methods. (Practice: loops, recursion).
5. Write a program to Print Fibonacci series up to n terms. (Practice: loop/recursion logic.)
6. Write a program to find the GCD of two numbers. (Practice: functions, logic)
7. Write a program to check if a number is prime. (Practice: loops, conditionals, modularity.)
8. Write a program to reverse a 1D array (C) or list (Python). (Practice: arrays/lists, loops)
9. Write a program to check if a given string is a palindrome. (Practice: string manipulation.)
10. Write a program to sort a list/array using bubble sort or selection sort. (Practice: sorting logic).
11. Write a program to define a structure for storing student data and display it. (Practice: structs, functions).
12. Write a program to implement a stack using List (Python). Implement push and pop operations. (Practice: lists, stack logic).
13. Write a program to read from and write to a text file. Practice: file I/O basics.
14. Write a program to read a file and count the number of words. (Practice: string handling, files)

SUGGESTED READINGS

1. Paul Deitel and Harvey Deitel, "C How to Program", 9th Edition, Pearson Education, 2022
2. John Zelle, "Python Programming: An Introduction to Computer Science", 3rd Edition, Franklin, Beedle & Associates, 2016. ISBN: 978-1590282755..
3. Mark Lutz, "Learning Python", 5th Edition, O'Reilly Media, 2013.
4. Eric Matthes, "Python Crash Course", 2nd Edition, No Starch Press, 2019.

25MEU101

ENGINEERING GRAPHICS

Semester – I
3H – 3C

Instruction Hours / week: L: 1 T: 2 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- Communicate the concepts, ideas, and design of Engineering products through graphic skills.
- Acquaint the national standards related to technical drawings.
- Comprehend Orthographic, Isometric, and perspective projection to represent the objects in two and three dimensions.

Course Outcomes (COs)

At the completion of the course, the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Sketch and distinguish between conic curves, cycloids, and involutes, and construct appropriate scales for engineering applications.	Apply
CO2	Identify and apply projection techniques to represent points, lines, and plane surfaces in first angle orthographic views.	Remember
CO3	Assemble and design accurate projections of solid geometries and truncated forms using appropriate methods.	Analysis
CO4	Prepare developments and sections of solids with holes and cut-outs, and evaluate the true shape of these sections.	Evaluate
CO5	Create isometric and perspective projections of simple and compound solids, and utilize CAD tools for visualization.	Create

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	3	1	2	1	1	1	2	2	2	2	2	3
CO2	3	2	2	1	2	1	1	1	2	2	2	2	2	3
CO3	3	2	2	1	2	1	1	1	2	2	2	2	2	3
CO4	3	2	2	1	2	1	1	1	2	2	2	2	2	3
CO5	3	2	2	1	2	1	1	1	2	2	2	2	2	3

1 - low, 2 - medium, 3 - high

Unit I PLANE CURVES AND FREE HAND SKETCHING

Basic Geometrical constructions, Curves used in engineering practices: Conics – Construction of ellipse, parabola and hyperbola by eccentricity method – Construction of cycloid – Construction of involutes of square and circle – Drawing of tangents and normal to the above curves, Scales: Construction of Diagonal and Vernier scales. Visualization concepts and Free Hand sketching: Visualization principles – Representation of Three Dimensional objects – Layout of views- Free hand sketching of multiple views from pictorial views of objects.

Unit II: PROJECTION OF POINTS, LINES AND PLANE SURFACES

Orthographic projection- principles-Principal planes-First angle projection-projection of points. Projection of straight lines (only First angle projections) inclined to both the principal planes - Determination of true lengths and true inclinations by rotating line method and traces. Projection of planes (polygonal and circular surfaces) inclined to both the principal planes by rotating object method.

Unit III: PROJECTION OF SOLIDS

Projection of simple solids like prisms, pyramids, cylinders, cones, and truncated solids when the axis is inclined to one of the principal planes by rotating object method and auxiliary plane method.

Unit IV: PROJECTION OF SECTIONED SOLIDS AND DEVELOPMENT OF SURFACES

Sectioning of above solids in simple vertical position when the cutting plane is inclined to the one of the principal planes and perpendicular to the other – obtaining true shape of section. Development of lateral surfaces of simple and sectioned solids – Prisms, pyramids cylinders and cones. Development of lateral surfaces of solids with cut-outs and holes.

Unit V: ISOMETRIC AND PERSPECTIVE PROJECTIONS

Principles of isometric projection – isometric scale –Isometric projections of simple solids and truncated solids - Prisms, pyramids, cylinders, cones- combination of two solid objects in simple vertical positions and miscellaneous problems. Perspective projection of simple solids-Prisms, pyramids and cylinders by visual ray method.

COMPUTER AIDED DRAFTING (Demonstration Only)

Introduction to drafting packages and demonstration of their use.

SUGGESTED READINGS

- 1.Parthasarathy, N.S.and Vela Murali, "Engineering Drawing", Oxford University Press, 2015.
- 2.Bhatt N.D. and Panchal V.M., "Engineering Drawing", Charotar Publishing House, 53rd Edition,2014.
- 3.Gopalakrishna K.R., "Engineering Drawing" (Vol. I&II combined), Subhas Stores, Bangalore,(2017).
- 4.Venugopal K. and Prabhu Raja V., "Engineering graphics", New Age International (P) Limited,(2008).
- 5.Natrajan K.V., "A text book of Engineering Graphics", Dhanalakshmi Publishers, Chennai, (2012).

25MAU208

ENGINEERING MATHEMATICS -II

Semester – II
4H – 4C

Instruction Hours / week: L: 3 T: 1 P: 0

Marks: Internal: 40 External: 60 Total: 100

End Semester Exam: 3 Hours

Course Objectives

- Define and differentiate the Linear partial differential equations of second and higher order with constant coefficients of both homogeneous and non-homogeneous types.
- Identify Fourier and half range Fourier transform techniques used in wide variety of situations.
- Apply the effective mathematical tools for the solutions of partial differential equations that model several physical processes.
- Evaluate Fourier transform techniques for different functions.
- Identify Z-transforms and Elementary properties of several functions

Course Outcomes (COs)

At the completion of the course the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Solve the methods of solving Partial differential equations.	Apply
CO2	Apply the concepts in Fourier series.	Apply
CO3	Apply the Partial derivative one-two dimensional concept in solving the Heat flow equations.	Apply
CO4	Solve the problems under Fourier transforms.	Apply
CO5	Identify and apply Z-transform concepts in Problem solving.	Apply

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	--	--	--	--	--	--	--	--	2	--	--	3
CO2	3	2	--	--	--	--	--	--	--	--	2	--	--	3
CO3	3	2	--	--	--	--	--	--	--	--	2	--	--	3
CO4	3	2	--	--	--	--	--	--	--	--	2	--	--	3
CO5	3	2	--	--	--	--	--	--	--	--	2	--	--	3

1 - low, 2 - medium, 3 - high

Unit I -PARTIAL DIFFERENTIAL EQUATIONS

Formation of partial differential equations. Solutions of standard types of first-order partial differential equations – Lagrange's linear equation. Linear partial differential equations of second and higher order with constant coefficients of both homogeneous and non-homogeneous types.

Unit II FOURIER SERIES

Dirichlet's conditions – General Fourier series – Odd and even functions – Half range sine series – Half range cosine series – Root mean square value – Parseval's identity – Harmonic analysis

Unit III APPLICATIONS OF PARTIAL DIFFERENTIAL EQUATIONS

Classification of PDE – Method of separation of variables – Fourier Series – Solutions of one-dimensional wave equation – One-dimensional equation of heat conduction – Steady state solution of two-dimensional equation of heat conduction (excluding insulated edges).

Unit IV FOURIER TRANSFORMS

Statement of Fourier integral theorem – Fourier transform pair – Fourier sine and cosine transforms – Properties – Transforms of simple functions – Convolution theorem – Parseval's identity.

Unit V- Z TRANSFORMS

Z-transforms – Elementary properties – Convergence of Z-transform – Initial and final value theorem – Inverse Z-transform using partial fraction and residues – Formation of difference equations.

SUGGESTED READINGS

1. Kreyszig, E., 'Advanced Engineering Mathematics', John Wiley and Sons, 10th Edition, New Delhi, 2016.
2. Grewal, B.S., 'Higher Engineering Mathematics', Khanna Publishers, New Delhi, 44th Edition, 2018.
3. Bali, N., Goyal, M., and Watkins, C., 'Advanced Engineering Mathematics', Firewall Media, New Delhi, 7th Edition, 2009.
4. L.C. Andrews and B. Shivamoggi, 'Integral Transforms for Engineers', SPIE Press, 1999.
5. Narayanan, S., Manicavachagom Pillay, T.K., and Ramanaiah, G., 'Advanced Mathematics for Engineering Students', Vol. II & III, S. Viswanathan Publishers Pvt. Ltd, Chennai, 1998

25CHU221

ENGINEERING CHEMISTRY

Semester – II
5H – 4C

Instruction Hours / week: L: 3 T: 0 P: 2

Marks: Internal: 40 External: 60 Total: 100

End Semester Exam: 3 Hours

Course Objectives

- To inculcate a sound understanding of water quality parameters and water treatment techniques.
- To impart knowledge on the basic principles and preparatory methods of nanomaterials.
- To introduce the basic concepts and applications of the phase rule and composites.
- To facilitate the understanding of different types of fuels, their preparation, properties, and combustion characteristics.
- To familiarize the students with the operating principles, working processes, and applications of energy conversion and storage devices.

Course Outcomes (COs)

At the completion of the course the student will be able to

Cos	Course Outcomes	Blooms Level
CO1	Interpret water quality parameters and treatment methods for domestic and industrial use.	Understand
CO2	Differentiate nanomaterials based on their properties, types, and synthesis techniques.	Analyse
CO3	Analyze phase diagrams and composite material systems with respect to their components and applications.	Apply
CO4	Evaluate fuel types, combustion characteristics, and emission parameters for energy efficiency.	Evaluate
CO5	Compare various energy sources and storage systems based on their principles and applications.	Assess

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	2	2	1	-	-	-	-	-	-	1	-	2	3	2
CO2	2	2	1	-	-	-	-	-	-	-	-	2	3	2
CO3	2	2	1	-	-	-	-	-	-	3	-	2	3	2
CO4	2	2	1	-	-	-	-	-	-	-	-	2	3	2
CO5	2	2	1	-	-	-	-	-	-	-	-	2	3	2

1 - low, 2 - medium, 3 - high

Unit I – WATER AND ITS TREATMENT

Water: Sources and impurities, Water quality parameters: Definition and significance of color, odour, turbidity, pH, hardness, alkalinity, TDS, COD and BOD, fluoride and arsenic. Domestic water treatment: Steps involved -primary treatment and disinfection (UV, Ozonation, breakpoint chlorination). Desalination of brackish water: Electro dialysis- Reverse Osmosis. Boiler troubles: Scale and sludge, Boiler corrosion, Caustic embrittlement, Priming and foaming. Treatment of boiler feed water: Internal treatment (phosphate, colloidal, sodium aluminate, and Calgon conditioning) and External treatment – Ion exchange demineralization process and zeolite process.

Unit II – NANOCHEMISTRY

Basics: Distinction between molecules, nanomaterials and bulk materials; Size-dependent properties (optical, electrical, mechanical and magnetic); Types of nanomaterials: Definition, properties and uses of – nanoparticle, nanocluster, nanorod, nanowire and nanotube-Single walled and Multiwalled Nanotubes-Preparation of nanomaterials: sol-gel, solvothermal, laser ablation, chemical vapour deposition, electrochemical deposition and electro spinning. Applications of nanomaterials in medicine, agriculture, energy, electronics, and catalysis.

Unit III – PHASE RULE AND COMPOSITES

Phase rule: Introduction, definition of terms with examples. One component system – water system; Reduced phase rule; Construction of a simple eutectic phase diagram – Thermal analysis; Two component system: lead-silver system – Pattinson process. Composites: Introduction: Definition & Need for composites; Constitution: Matrix materials (Polymer matrix, metal matrix, and ceramic matrix) and Reinforcement (fiber, particulates, flakes, and whiskers). Properties and applications of Metal matrix composites (MMC), Ceramic matrix composites (CMC), and Polymer matrix composites (PMC). Hybrid composites – definition and examples.

Unit IV – FUELS AND COMBUSTION

Fuels: Introduction: Classification of fuels; Coal and coke: Analysis of coal (proximate and ultimate), Carbonization, Manufacture of metallurgical coke (Otto Hoffmann method). Petroleum and Diesel: Fractional distillation of Petroleum- Manufacture of synthetic petrol (Fischer–Tropsch and Bergius process), Knocking – octane number, diesel oil – cetane number; Power alcohol and biodiesel. Combustion of fuels: Introduction: Calorific value – higher and lower calorific values, Theoretical calculation of calorific value; Ignition temperature: spontaneous ignition temperature, Explosive range; Flue gas analysis – ORSAT Method. CO₂ emission and carbon footprint.

Unit V – ENERGY SOURCES AND STORAGE DEVICES

Stability of nucleus: mass defect (problems), binding energy; Nuclear energy: light water nuclear power plant, breeder reactor. Solar energy conversion: Principle, working, and applications of solar cells; Recent developments in solar cell materials. Wind energy; Geothermal energy; Batteries: Types of batteries, Primary battery – dry cell, Secondary battery – NICAD battery, lead acid battery, and lithium-ion battery; Electric vehicles – working principles; Fuel cells: H₂-O₂ fuel cell, microbial fuel cell; Super capacitors: Storage principle, types and examples.

SUGGESTED READINGS

1. P. C. Jain and Monica Jain. (2018). Engineering Chemistry, 17th Edition, Dhanpat Rai Publishing Company (P) Ltd, New Delhi.
2. Sivasankar B. (2008). Engineering Chemistry, Tata McGraw-Hill Publishing Company Ltd, New Delhi.
3. S.S. Dara. (2018). A Textbook of Engineering Chemistry, S. Chand Publishing, 12th Edition.
4. Dr. Sayeeda Sultana (2016). Engineering Chemistry, R.K. Publishers, Coimbatore.
5. B. S. Murty, P. Shankar, Baldev Raj, B. B. Rath and James Murday. (2018). Textbook of Nanoscience and Nanotechnology, Universities Press-IIM Series in Metallurgy and Materials Science.
6. Dr. Sayeeda Sultana, (2016). Practical Engineering Chemistry laboratory manual, R.K. Publishers, Coimbatore.

25ECU223

DIGITAL SYSTEM DESIGN

Semester –II
4H – 3C

Instruction Hours / week: L: 2 T: 0 P: 2

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- To learn basic techniques for the design of digital circuits and fundamental concepts used in the design of digital systems.
- To understand common forms of number representation in digital electronic circuits and to be able to convert between different representations.
- To implement simple logical operations using combinational logic circuits
- To design combinational logic circuits, sequential logic circuits.
- To impart to student the concepts of sequential circuits, enabling them to analyze sequential systems in terms of state machines
- To implement synchronous state machines using flip-flops.

Course Outcomes (COs)

At the completion of the course the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Formulate canonical expressions for switching functions.	Understand
CO2	Apply various Boolean function minimization techniques including Karnaugh Map and Quine-McCluskey method to optimize logic circuits	Apply/ Analyze
CO3	Design and implement combinational circuits such as adders, subtractors, multiplexers, encoders, decoders, and code converters using HDL	Apply /create
CO4	Analyze, design, and simulate basic sequential circuits including flip-flops, counters, and shift registers using HDL	Analyze / Create
CO5	Explain the organization and operation of memory devices and apply programmable logic devices for implementing combinational logic.	Understand / Apply

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	1	-	1	-	-	-	-	-	-	-	1	2
CO2	3	3	2	1	1	-	-	-	-	-	-	-	1	2
CO3	3	3	3	2	3	-	-	-	-	1	1	-	1	2
CO4	3	2	3	2	2	-	-	-	-	1	1	-	2	1
CO5	3	2	2	1	1	-	-	-	-	-	-	-	2	1

1 - low, 2 - medium, 3 - high

Unit I NUMBER SYSTEM AND BOOLEAN ALGEBRA

Number Systems, Base Conversion Methods, Complements of Numbers, Codes- Binary Codes, Binary Coded Decimal Code and its Properties, Unit Distance Codes, Error Detecting and Correcting Codes. Digital Logic Gates(AND,NAND,OR,NOR,EX-OR,EX-NOR), Properties of XOR Gates, Universal Gates, Basic Theorems and Properties, Switching Functions, Canonical and Standard

Unit II MINIMIZATION TECHNIQUES

Introduction, The minimization with theorems, The Karnaugh Map Method, Three, Four and Five variable K- Maps, Prime and Essential Implications, Don't Care Map Entries, Using the Maps for Simplifying, Quine-McCluskey Method, Multilevel NAND/NOR realizations.

Unit III COMBINATIONAL CIRCUITS

Design Procedure – Half Adder, Full Adder, Half Subtractor, Full Subtractor, Parallel Binary Adder, Parallel binary subtractor, Binary Multiplier, Multiplexers/ De-Multiplexers, decoder, Encoder, Code Converters, Magnitude Comparator. Introduction to HDL — HDL Models of Combinational circuits.

Unit IV SEQUENTIAL CIRCUITS

Introduction, Basic Architectural Distinctions between Combinational and Sequential circuits, Latches, Flip-Flops, SR,JK,D,T and Master slave, characteristic Tables and equations, Conversion from one type of Flip-Flop to another, Counters - Design of Single Mode Counter, Ripple Counter, Ring Counter, Shift Register, Ring counter using Shift Register - HDL Models of Sequential Circuits

Unit V MEMORY DEVICES

Classification of memories – ROM : ROM organization, PROM, EPROM,EEPROM, RAM: RAM organization, Write operation, Read operation, Static RAM , Programmable Logic Devices: Programmable Logic Array(PLA),Programmable Array Logic, Implementation of Combinational Logic circuits using ROM,PLA,PAL

PRACTICAL EXERCISES

1. Experimental Verification of Logic Gates
2. Design and Experimental verification of Boolean function
3. Design of adders,subtractors & code converters
4. Design of Multiplexers & Demultiplexers.
5. Design of Encoders and Decoders
6. Design of Magnitude Comparators
7. Design and implementation of counters using flip-flops
8. Design and implementation of shift registers.
9. Coding combinational circuits using HDL
10. Coding sequential circuits using HDL

SUGGESTED READINGS

1. Digital Design- Morris Mano, PHI, 3rd Edition.
2. Switching Theory and Logic Design-A. Anand Kumar, PHI, 2nd Edition.
3. Switching and Finite Automata Theory- Zvi Kohavi & Niraj K. Jha, 3rd Edition, Cambridge.
4. Introduction to Switching Theory and Logic Design – Fredriac J. Hill, Gerald R. Peterson, 3rd Ed,John Wiley & Sons Inc.
5. Digital Fundamentals – A Systems Approach – Thomas L. Floyd, Pearson, 2013.
6. Switching Theory and Logic Design – Bhanu Bhaskara –Tata McGraw Hill Publication, 2012
7. Fundamentals of Logic Design- Charles H. Roth, Cengage LEarning, 5th, Edition, 2004.
8. Digital Logic Applications and Design- John M. Yarbrough, Thomson Publications, 2006.
9. Digital Logic and State Machine Design – Comer, 3rd, Oxford, 2013

25CSU221

OBJECT ORIENTED PROGRAMMING

Semester –II
5H – 4C

Instruction Hours / week: L: 3 T: 0 P: 2

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- Understand and apply the fundamental concepts of object-oriented programming including classes, objects, encapsulation, inheritance, polymorphism, and abstraction.
- Analyze and implement advanced object-oriented features such as function overloading, operator overloading, templates, and exception handling.
- Design and develop programs using inheritance hierarchies, virtual functions, and abstract classes to solve complex programming problems.
- Implement Java-based object-oriented solutions utilizing packages, interfaces, multithreading, and string handling mechanisms.
- Develop interactive applications with graphical user interfaces using JavaFX controls, components, layouts, and event handling techniques.

Course Outcomes (COs)

At the completion of the course, the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Analyze programming problems and implement object-oriented solutions using classes, objects, and appropriate OOP constructs.	Create
CO2	Apply inheritance, polymorphism, and exception handling techniques to develop robust applications	Apply
CO3	Implement appropriate Java programming constructs, including packages, interfaces, and multithreading	Create
CO4	Demonstrate the concepts of exception handling and file management in both C++ and Java	Evaluate
CO5	Evaluate and create interactive GUI applications using JavaFX components and event handling mechanisms	Create

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	2	2	2	2	2	2	2	2	2	2	1	3	2
CO2	2	2	2	2	2	2	2	2	2	2	1	3	2
CO3	2	2	2	2	2	2	2	2	2	2	1	3	2
CO4	2	2	2	2	2	2	2	2	2	2	1	3	2
CO5	2	2	2	2	2	2	2	2	2	2	1	3	2

1 - low, 2 - medium, 3 - high

Unit I – FUNDAMENTAL CONCEPTS OF OOP

Object-Oriented Paradigm - Elements of Object-Oriented Programming – Structure of C++ program – Classes and Objects - Defining member functions - Passing and returning objects – Array of objects - Inline functions - Constructors - Parameterized Constructors - Constructor Overloading. Copy Constructor, Destructors, Default arguments - new, delete operators - “this” pointer, Friend classes and Friend functions - Function Overloading – Operator Overloading - Generic programming with templates- Function templates - Class templates

Unit II – INHERITANCE, VIRTUAL FUNCTIONS AND EXCEPTION HANDLING

Inheritance - Base class and derived class relationship - Forms of inheritance - Inheritance and member accessibility - constructors in derived class, abstract class, virtual functions, pure virtual function - Files and Streams - Opening and Closing a file- file modes- file pointers and their manipulation, sequential access to a file-random access to a file-Reading and Writing – Exception handling.

Unit III – INHERITANCE, PACKAGES AND INTERFACES IN JAVA

Java Buzzwords – Overview of Java – Programming Structures in Java – Defining classes in Java - Overloading Methods – Objects as Parameters – Returning Objects –Static, Nested and Inner Classes. Types of Inheritance -Super keyword -Method Overriding – Dynamic Method Dispatch –Abstract Classes – final with Inheritance. Packages and Interfaces: Packages – Packages and Member Access –Importing Packages – Interfaces -

Unit IV – EXCEPTION HANDLING, MULTITHREADING AND STRING HANDLING

Exception Handling basics – Multiple catch Clauses – Nested try Statements – Java’s Built-in Exceptions – User defined Exception. Multithreaded Programming: Java Thread Model–Creating a Thread and Multiple Threads – Priorities – Synchronization – Inter Thread Communication- Suspending –Resuming, and Stopping Threads –Multithreading. Wrappers – Auto boxing - Strings: Basic String class, methods and String Buffer Class.

Unit V – JAVA FX EVENT HANDLING, CONTROLS AND COMPONENTS

JAVAFX Events and Controls: Event Basics – Handling Key and Mouse Events. Controls: Checkbox, ToggleButton – RadioButtons – ListView – ComboBox – ChoiceBox – Text Controls – ScrollPane. Layouts – FlowPane – HBox and VBox – BorderPane – StackPane – GridPane. Menus – Basics – Menu – Menu bars – MenuItem.

SUGGESTED READINGS

1. Herbert Schildt, "The Complete Reference C++", Tata McGraw-Hill, 2017.
2. E. Balagurusamy, "Object-oriented programming with C++", Tata McGraw-Hill, 2017.
3. Paul Deitel and Harvey Deitel, "C How to Program with an Introduction to C++", Eighth edition, Pearson Education, 2018.
4. Herbert Schildt, "Java: The Complete Reference", 12th Edition (2021), McGraw-Hill Education.
5. Cay S. Horstmann, "Core Java Volume I – Fundamentals", 12th Edition (2022), Oracle Press.
6. Cay S. Horstmann, "Core Java Volume II – Advanced Features", 12th Edition (2022), Oracle Press.

25CSU222

DATA STRUCTURES

Semester –II
5H – 4C

Instruction Hours / week: L: 3 T: 0 P: 2

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- Familiarize the concepts of ADT Linear Data Structure
- Acquaint knowledge in Linear and Non-Linear Data Structure
- Gain knowledge on the fundamentals of Trees
- Describe the various Graph operations on real-world problems
- Apply the Hashing techniques

Course Outcomes (COs)

At the completion of the course, the student will be able to

COs	Course Outcomes	Blooms Level
C01	Define linear data structures and apply them in real problems	Remember
C02	Implement linear Data structures like Stack, queue, and deque	Create
C03	Use appropriate Non-Linear data structure tree to solve a real-world problem	Analyze
C04	Decide appropriate graph algorithm for graph applications	Evaluate
C05	Analyze and apply various hashing functions	Analyze

CO-PO Mapping

CO / PO	P01	P02	P03	P04	P05	P06	P07	P08	P09	P010	P011	PS01	PS02
C01	2	2	2	2	2	1	1	2	1	1	1	2	2
C02	3	2	2	3	3	2	2	2	3	2	3	2	3
C03	3	3	3	3	2	3	3	2	1	3	3	3	3
C04	2	3	2	3	2	3	2	2	1	3	3	3	2
C05	3	3	3	3	3	3	3	2	1	3	2	3	3

1 - low, 2 - medium, 3 - high

Unit I: LISTS

Basic Terminologies: Elementary Data Organizations, Data Structure Operations: insertion, deletion, traversal, etc.; Abstract Data Types (ADTs) – List ADT – Array-based implementation – Linked list implementation – Singly-linked lists – Circularly linked lists – Doubly-linked lists – Applications of lists – Polynomial ADT – Radix Sort – Multilists

Unit II: STACKS, QUEUES AND DEQUE

Stack ADT – Operations – Applications – Balancing Symbols – Evaluating arithmetic expressions- Infix to Postfix conversion – Function Calls – Queue ADT – Operations – Circular Queue – DeQueue Applications of Queues- Deque ADT – Operations- Applications

Unit III: TREES

Tree ADT – Tree Traversals - Binary Tree ADT – Expression trees – Binary Search Tree ADT – AVL Trees – Red Black Trees- B+ Tree – Priority Queue (Heaps) – Binary Heap.

Unit IV: GRAPHS

Graph Definition – Representation of Graphs – Types of Graphs - Breadth-first traversal – Depth-first traversal — Bi-connectivity – Euler circuits – Topological Sort – Dijkstra's algorithm – Bellman Ford Algorithm – Minimum Spanning Tree – Prim's algorithm – Kruskal's algorithm

Unit V: HASHING TECHNIQUES

Hashing – Hash Functions – Separate Chaining – Open Addressing – Rehashing – Extendible Hashing- Double Hashing – Load Factor and Rehashing

PRACTICAL EXERCISES:

1. Implementation of Singly Linked List
2. Implementation of Doubly Linked List
3. Implementation of Circular Linked List
4. Stack, Queue and Circular Queue ADTs implementation using Arrays
5. Stack and Linear Queue ADT implementation using Linked list
6. Deque implementation using Linked list
7. Implementation of Evaluating Postfix Expressions, Infix to Postfix conversion
8. Implementation of Binary Search Trees
9. Implementation of AVL Trees
10. Implementation of Heaps using Priority Queues
11. Implementation of Dijkstra's Algorithm,
12. Implementation of Prim's and Kruskals
13. Implementation of Open Addressing (Linear Probing and Quadratic Probing)
14. Implementation of intersection of two array
15. Implementation of longest subarray with equal number of 0's and 1's in a binary array

SUGGESTED READINGS

1. Michael T. Goodrich, Roberto Tamassia, and Michael H. Goldwasser Data Structures and Algorithms in Python Wiley 2013
2. Liang D Y, Introduction To Python Programming And Data Structures 3/Ed Pearson Education, 2022
3. Shriram K Vasudevan, Abhishek S Nagarajan , Karthick Nanmaran,Data Structures Using Python Oxford university Press 2021
4. Kent D. Lee, Data Structures and Algorithms with Python: With an Introduction to Multiprocessing Springer 2024
5. https://www.w3schools.com/dsa/dsa_intro.php

25IKS001

INTRODUCTION TO INDIAN KNOWLEDGE SYSTEM

Semester – II
2H – 2C

Instruction Hours / week: L: 2 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100

End Semester Exam: 3 Hours

Course Objectives

- To introduce students to the foundational concepts of the Indian Knowledge System (IKS)
- To explore the relevance and applications of IKS in contemporary times.
- To promote interdisciplinary learning through the integration of traditional Indian knowledge and modern education.

Course Outcomes (COs)

At the completion of the course, the student will be able to

COs	Course Outcomes	Blooms Level
C01	Describe the meaning, scope, and philosophical foundations of IKS	Remembering
C02	Summarize the features of Indian education, language, and literary contributions	Understanding
C03	Illustrate traditional Indian scientific and technological advancements	Applying
C04	Examine the impact of Indian art, aesthetics, and socio-cultural practices	Analyzing
C05	Evaluate the relevance and application of IKS in contemporary society	Evaluating

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
C01	3	2	2	2	1	1	2	0	0	0	0	1	1
C02	2	3	2	1	2	0	1	0	0	0	0	2	1
C03	2	2	3	2	2	1	2	1	0	0	0	1	2
C04	1	1	2	3	2	2	2	1	0	0	0	2	2
C05	1	1	1	2	3	3	2	2	0	0	0	3	3

1 - low, 2 - medium, 3 - high

Unit I: FOUNDATIONS OF INDIAN KNOWLEDGE SYSTEM

Meaning and Scope of IKS-Historical evolution and literary sources: Vedas, Upanishads, Puranas-Philosophical foundations: Darshanas (Nyaya, Vaisheshika, Samkhya, Yoga, Mimamsa, Vedanta)-Interdisciplinary nature of IKS.

Unit II: EDUCATION, LANGUAGE, AND LITERATURE

Traditional education systems: Gurukula, Pathashalas-Higher education: Nalanda, Takshashila-Role of Sanskrit and regional languages-Contributions of Panini, Bhartrihari-Epics and classical literature.

Unit III: SCIENCE AND TECHNOLOGY IN IKS

Mathematics: Sulbasutras, Aryabhata, Bhaskara-Astronomy: Surya Siddhanta-Ayurveda: Tridosha, healing systems-Metallurgy, Vastu Shastra, water management-

Unit IV: INDIAN ART, CULTURE, AND SOCIETY

Music, dance, painting, sculpture-Rasa theory, Natya Shastra-Festivals, rituals, socio-cultural life-Dharma, Purusharthas, social organization.

Unit V: CONTEMPORARY RELEVANCE AND APPLICATIONS OF IKS

IKS in modern education and research-Sustainable practices in agriculture, ecology, lifestyle-Yoga and meditation in wellness-Role of IKS in national identity and global relevance.

SUGGESTED READINGS

1. Kapil Kapoor (Ed.) – Encyclopedia of Hinduism, Rupa Publications, Comprehensive overview of philosophical and literary foundations of IKS.
2. Michel Danino – The Indian Mind: A Cultural and Philosophical Perspective, DK Printworld-Offers insight into Indian civilization's unique philosophical frameworks and relevance today.
3. V. Sivaramakrishnan (Ed.) – Cultural Heritage of India, Ramakrishna Mission Institute of Culture-Multi-volume work covering various aspects of Indian science, arts, literature, and education.
4. Subhash Kak, David Frawley & N.S. Rajaram – In Search of the Cradle of Civilization, Motilal Banarsidass-Discusses early Indian contributions to science, mathematics, and cosmology.
5. Bharatiya Vidya Bhavan Series – History and Culture of Indian *People*-A classic multi-volume series offering a deep dive into ancient Indian education, society, arts, and sciences.
6. Debroy, Bibek – The Bhagavad Gita, Upanishads, and the Vedas (Translations)-For primary source reading and understanding scriptural references in IKS.
7. R. Balasubramanian (Ed.) – The Bloomsbury Research Handbook of Indian Epistemology and Metaphysics
8. Yoga Sutras of Patanjali (Various commentaries) – for insights into yoga, wellness, and consciousness studies.

25MEU211

DESIGN THINKING AND INNOVATIONS LABORATORY

Semester – II
4H – 2C

Instruction Hours / week: L: 0 T: 0 P: 4

Marks: Internal: 40 External: 60 Total: 100

End Semester Exam: 3 Hours

Course Objectives

- Introduce the basic principles and phases of design thinking to solve real-world problems.
- Encourage creativity and innovation through hands-on learning and teamwork.
- Provide foundational skills in plumbing, carpentry, welding, machining, sheet metal, and basic electrical/electronic practices.
- Integrate design thinking with mechanical, electrical, and electronic fabrication techniques to develop simple and functional design solutions.
- Enable students to apply their learning in an innovation project that emphasizes planning, design, development, and presentation...

Course Outcomes (COs)

At the completion of the course the student will be able to

Cos	Course Outcomes	Blooms Level
CO1	Apply design thinking principles to identify and solve real-world problems.	Apply
CO2	Demonstrate basic plumbing and carpentry skills through simple model creation.	Apply
CO3	Develop welding, machining, and sheet metal fabrication skills.	Apply
CO4	Construct basic electrical and electronic circuits for design applications.	Apply
CO5	Execute an innovation project by integrating skills from all units to build and present a working prototype.	Create

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	1	1	2	2	3	3	3	3	2	2	3	2	3
CO2	3	3	2	2	3	2	3	2	3	3	2	3	3	3	3
CO3	3	3	3	2	3	2	2	2	3	2	1	3	3	3	3
CO4	2	2	2	2	2	2	2	2	3	3	2	2	2	2	2
CO5	3	2	2	2	2	3	3	3	3	3	2	3	3	3	3

INTRODUCTION TO DESIGN THINKING:

Definition and importance of Design Thinking- Comparison with traditional problem-solving approaches Key principles: Empathy, Experimentation, and Iteration- Problem Reframing techniques to identify innovative opportunities- Discussion and demo on Design Thinking stages and examples- Team activity to understand user needs and define problems

- **PART A: MECHANICAL PRACTICES**

1. Demonstration of pipe joints and wood connections
2. Creation of a simple model using plumbing and carpentry components
3. Arc and gas welding techniques
4. Safety practices in metalworking
5. Basic machining operations (turning, drilling)
6. Sheet metal forming: bending, joining, and component design
7. Fabrication of small metal parts like trays or holders

- **PART B: ELECTRICAL AND ELECTRONICS PRACTICES**

1. Introduction to basic residential wiring: switches, fuses, indicators, and energy meters
2. Wiring of simple lighting circuits (fluorescent, staircase wiring)
3. Study of basic electronic components: resistors, logic gates
4. Introduction to soldering and basic circuit assembly
5. Discussion and demo on simple wiring and soldering
6. Hands-on activity: build a basic electrical/electronic setup for design integration

- **PART C: INNOVATION PROJECT (Integrated Design and Development)**

Application of design thinking in an innovation project - Planning, designing, and building a functional prototype- Integration of mechanical, electrical, and electronic fabrication techniques- Emphasis on creativity, teamwork, and presentation skills

SUGGESTED READINGS

1. Rolf Faste, Design Thinking: Understanding How Designers Think and Work, Springer, 2013.
2. S. K. Hajra Choudhury, A. K. Hajra Choudhury & Nirjhar Roy, Elements of Workshop Technology – Volume 1: Manufacturing Processes, Media Promoters, 2020.
3. K. J. Vinoy & M. Gokhale, Basic Electrical and Electronics Engineering, Wiley India, 2015.
4. Tim Brown, Change by Design: How Design Thinking Creates New Alternatives for Business and Society, Harvard Business Review Press, 2009.
5. Ramesh Babu, Basic Electrical and Electronics Engineering, Scitech Publications, 2021.

25MAC201

INDIAN CONSTITUTION AND HUMAN RIGHTS

Semester – II
2H – 2C

Instruction Hours / week: L: 2 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100

End Semester Exam: 3 Hours

Course Objectives.

- To create the basic philosophical tenets of Indian Constitution and Human Rights.
- To underline the significance of our Constitution as Fundamental Law of the land and its features.
- To respect human rights, rule of law and democracy.
- To gain In-depth insight into the constitutional, statutory and institutional aspects of human rights protection in India.
- To identify the constitutional provisions dealing with human rights and special legislations dealing with protection of vulnerable and marginalized groups.

Course Outcomes (COs)

At the completion of the course, the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Demonstrate the provisions under the Constitution of India dealing with human rights	Remember
CO2	Display the nature and scope of special legislations dealing with protection of human rights of marginalized and vulnerable sections.	Understand
CO3	Apply practically human rights law to specific human rights problems in India	Apply
CO4	Analyze complex human rights problems and apply relevant provisions of human rights law in India to a hypothetical situation/case study..	Analyze
CO5	Acquainted with the theoretical knowledge of the underpinnings of the human rights framework in India, its operation and issues associated with its implementation.	Create

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	2	2	1	3	3	3	3	3	3	3	2	3	2	2
CO2	3	3	3	3	3	3	3	3	2	2	3	3	2	3
CO3	3	3	2	2	1	3	3	3	3	3	2	3	2	3
CO4	3	2	3	3	3	2	3	2	3	3	3	3	2	3
CO5	2	3	1	3	2	3	2	3	2	3	3	2	1	2

1 - low, 2 - medium, 3 - high

Unit I THE CONSTITUTION

Definition and Principles of the Constitution – Socio-Economic and Political Conditions in India at the time of Independence – Contents and Amendments to the Constitution.

Unit II FUNDAMENTAL RIGHTS

Historical Perspectives on Rights in India – Fundamental Rights in India – Provisions in Articles 14 to 32 and its implications on Human Rights – Right against unlawful detention.

Unit III - DUTIES, DIRECTIVE PRINCIPLES AND AFFIRMATIVE ACTIONS

Fundamental Duties of a Citizen in India - Directive Principles - Policy and Practices in Reservation
Affirmative Actions: Special Provisions for SCs and STs.

Unit IV- PROTECTION OF WEAKER SECTIONS OF SOCIETY

Constitutional Provisions for the Protection of women and children - Safeguard for the Labours – Minorities – Tribals.

Unit V - ENFORCEMENT MECHANISM AND EVALUATION

Protection of Human Rights Act 1993 – National and State Human Rights Commissions – Role of Judiciary in Human Rights Protection – Critical Appraisal of the Current Status of Human Rights in India – AFSPA.

SUGGESTED READINGS

1. Desai, A.R. (ed.) (1986), Violations of Democratic Rights in India, Bombay: Popular Prakashan.
2. Meghraj Kapurderiya (2013) Indian Philosophical Foundation of Human Rights, New Delhi: R.P. Publications.
3. Mishra, P.K. (2012) Human Rights: Human Rights: Acts, Statues and Constitutional Provisions, Jaipur: Ritu Publications.
4. Satish Chandra (1995) Minorities in National And International Laws, New Delhi: Deep and Deep Publications.

25MAU306

DISCRETE MATHEMATICS

Semester –III
4H – 4C

Instruction Hours / week: L: 3 T: 1 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- Define and identify the fundamentals of logic and predicate calculus used in mathematical reasoning.
- Develop skills in combinatorics, recurrence relations, and generating functions to solve counting problems.
- Gain foundational knowledge of graph theory and its applications in modeling relationships.
- Explore algebraic structures such as groups, rings, and fields relevant to abstract mathematical systems.
- Familiarize lattice theory and its role in logic and algebraic systems.
- Solve the boundary value problems.

Course Outcomes (COs)

At the completion of the course, the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Apply propositional and predicate logic to reason about statements and arguments.	Apply
CO2	Solve combinatorial problems using techniques like permutations, combinations, and recurrence relations.	Apply
CO3	Analyze and represent graphs and apply graph theory concepts in problem solving	Analyze
CO4	Apply concepts of algebraic structures such as groups and rings.	Apply
CO5	Analyze partially ordered sets and lattices, and their algebraic properties.	Analyze

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	--	2	--	--	--	--	--	--	--	3	--	--
CO2	3	3	2	-2	--	--	--	--	--	--	--	3	--	--
CO3	3	2	3	2	--	--	--	--	--	--	--	3	--	--
CO4	3	2	--	--	--	--	--	--	--	--	--	2	--	--
CO5	3	2	--	--	--	--	--	--	--	--	--	2	--	--

1 - low, 2 - medium, 3 - high

Unit I – LOGIC AND PREDICATE CALCULUS

Propositional logic – Propositional equivalences - Connectives–Tautologies - logic -Equivalence - Implications–Normal forms - Predicates and quantifiers – Nested quantifiers

Unit II – COMBINATORICS

Mathematical inductions-The basics of counting- The pigeonhole principle –Permutations and Combinations-Recurrence relations- generating functions- inclusion and exclusion and applications.

Unit III – FUNDAMENTALS OF GRAPHS

Basic Concepts of Graphs and Graph Models -Graph terminology and special types of graphs- Representing graphs and graph isomorphism -connectivity-Euler and Hamilton path

Unit IV – ALGEBRAIC STRUCTURES

Algebraic systems – Semi groups and monoids - Groups – Subgroups – Homomorphisms – Normal subgroup and Cosets – Lagrange’s theorem – Definitions and examples of Rings and Fields.

Unit V – LATTICES

Partial ordering-Posets-Lattices as Posets- Properties of lattices-Lattices as Algebraic systems –Sub lattices –Some Special lattices.

SUGGESTED READINGS

1. J.P.Tremblay. R. Manohar “Discrete Mathematical Structures with applications to Computer Science” Tata Mc-Graw-Hill Publishing company pvt. Ltd.New Delhi, 35th edition, 2008.
2. Veerajan.T, Discrete Mathematics with Graph Theory and Combinatorics”, 10th edition, Tata Mc-Graw-Hill Companies, 2010.
3. Ralph. P. Grimaldi, “Discrete and Combinatorial Mathematics: An Applied Introduction”, Fourth Edition, Pearson Education Asia, Delhi, (2002)..
4. Thomas Koshy, ”Discrete Mathematics with Applications”, Elsevier Publications, (2006).
5. Seymour Lipschutz and Mark Lipson, ”Discrete Mathematics”, Schaum’s Outlines, Tata McGraw-Hill Pub. Co. Ltd., New Delhi, Second edition, (2007).

25UHV001

UNIVERSAL HUMAN VALUES AND ETHICS

Semester –III
2H – 2C

Instruction Hours / week: L: 2 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100

End Semester Exam: 3 Hours

Course Objectives

- Understand the need, basic guidelines, content, and process of value education.
- Develop right understanding and relationship at all levels of living.
- Understand harmony in the human being, family, society, and nature.
- Relate the holistic understanding with professional ethics.
- Apply value-based principles in personal and professional life

Course Outcomes (COs)

At the completion of the course, the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Demonstrate understanding of the holistic development of a human being.	Understand
CO2	Distinguish between physical needs and the needs of the self.	Understand
CO3	Apply human values in family and societal interactions.	Apply
CO4	Analyze nature and existence in terms of mutual harmony and co-existence.	Analyze
CO5	Integrate ethical human conduct in their professional practice.	Apply

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	-	-	-	2	2	3	2	-	-	1	3
CO2	2	2	-	-	-	-	--	--	2	2	-	-	1	2
CO3	-	3	-	-	-	-	3	3	-	3	1	-	1	-
CO4	-	3	-	3	2	-	3	3	-	3	1	-	2	-
CO5	3	2	2	-	1	2	3	3	3	2	-	-	2	3

1 - low, 2 - medium, 3 - high

Unit I INTRODUCTION TO VALUE EDUCATION

Right Understanding, Relationship and Physical Facility (Holistic Development and the Role of Education) Understanding Value Education, Self-exploration as the Process for Value Education, Continuous Happiness and Prosperity – the Basic Human Aspirations, Happiness and Prosperity – Current Scenario, Method to Fulfil the Basic Human Aspirations.

Unit II HARMONY IN THE HUMAN BEING

Understanding Human being as the Co-existence of the Self and the Body, Distinguishing between the Needs of the Self and the Body, The Body as an Instrument of the Self, Understanding Harmony in the Self, Harmony of the Self with the Body, Programme to ensure self-regulation and Health

Unit III HARMONY IN THE FAMILY AND SOCIETY

Harmony in the Family – the Basic Unit of Human Interaction, 'Trust' – the Foundational Value in Relationship, 'Respect' – as the Right Evaluation, Other Feelings, Justice in Human-to-Human Relationship, Understanding Harmony in the Society, Vision for the Universal Human Order.

Unit IV HARMONY IN THE NATURE

Understanding Harmony in the Nature, Interconnectedness, self-regulation and Mutual Fulfilment among the Four Orders of Nature, Realizing Existence as Co-existence at All Levels, The Holistic Perception of Harmony in Existence.

Unit V IMPLICATIONS OF THE HOLISTIC UNDERSTANDING – A LOOK AT PROFESSIONAL ETHICS

Natural Acceptance of Human Values, Definitiveness of (Ethical) Human Conduct, A Basis for Humanistic Education, Humanistic Constitution and Universal Human Order, Competence in Professional Ethics Holistic Technologies, Production Systems and Management Models-Typical Case Studies, Strategies for Transition towards Value-based Life and Profession

SUGGESTED READINGS

1. Gaur, R.R., Sangal, R., & Bagaria, G.P., A Foundation Course in Human Values and Professional Ethics, Publisher: Excel Books, New Delhi, 2022.
2. Sharma, S. B., Education for Values, Environment and Human Rights, Publisher: Lotus Press, New Delhi, 2022
3. Pathania, A. , Value Education: Perspectives and Practices, Pearson Education India, 2023
4. The Textbook A Foundation Course in Human Values and Professional Ethics, R R Gaur, R Asthana, G P Bagaria, 2nd Revised Edition, Excel Books, New Delhi, 2019. ISBN 978-93-87034- 47-1
5. Schumacher, E.F. , Small is Beautiful: A Study of Economics as if People Mattered, Random House UK / Vintage publisher, 2022
6. Dhar, P.L. & Gaur, R.R, Science and Humanism, Publisher: Commonwealth Publishers, New Delhi, 2023
7. Sharma, R.N., Human Values and Professional Ethics, Kalyani Publishers, 2022
8. <http://www.mhrd.gov.in> – Ministry of Education, Govt. of India.
9. <http://www.uhv.org.in> – Universal Human Values initiative.
10. <https://www.aiu.ac.in> – Association of Indian Universities: Value Education resources

25CSU301

COMPUTER ARCHITECTURE

Semester –III
3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours

Course Objectives

- Identify the components of a computer system and outline the principles of Von Neumann architecture.
- Demonstrate how various number systems and arithmetic units contribute to ALU operations.
- Investigate control signal generation and instruction flow through pipelining.
- Assess memory structures and techniques used to improve system efficiency.
- Formulate strategies for integrating parallel processing and efficient I/O handling in modern architectures.

Course Outcomes (COs)

At the completion of the course, the student will be able to

COs	Course Outcomes	Blooms Level
C01	Recall key elements of instruction formats, addressing modes, and performance metrics.	Remember
C02	Use binary and floating-point formats to perform arithmetic operations.	Apply
C03	Analyze the stages of instruction execution and identify potential hazards in pipelined systems.	Analyze
C04	Evaluate cache performance using hit/miss rates and access times.	Evaluate
C05	Construct a conceptual model that incorporates multi-core processing and GPU acceleration.	Create

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
C01	3	2	2	1	2	2	2	2	2	1	1	2	1
C02	3	3	2	1	2	2	2	2	2	1	1	3	1
C03	3	3	2	2	2	2	2	2	2	1	1	3	2
C04	3	2	2	1	3	2	2	2	2	1	1	3	2
C05	2	3	3	2	3	1	1	1	1	1	2	3	3

1 - low, 2 - medium, 3 - high

Unit I: INTRODUCTION TO COMPUTER ARCHITECTURE

Overview of computer systems and digital components - Von-Neumann architecture - Instruction set architecture (ISA): types, formats, and addressing modes - Performance metrics: MIPS, CPI, execution time - Introduction to assembly language (e.g., MIPS/RISC-V basics)

Unit II: ARITHMETIC AND LOGIC UNIT (ALU)

Number representations: signed, unsigned, BCD, floating point - Addition, subtraction, multiplication, division algorithms - ALU design and bit-slice architecture - IEEE 754 floating-point standard

Unit III: CONTROL UNIT AND INSTRUCTION EXECUTION

Instruction cycle, control signals, micro-operations - Hardwired and micro programmed control units
Instruction pipe-lining: stages, hazards, forwarding, and stalling - Branch prediction and out-of-order execution basics

Unit IV MEMORY HIERARCHY

Types of memory: RAM, ROM, Cache, Virtual Memory - Memory hierarchy: locality of reference – Cache organization: direct, associative, set-associative - Memory access time, hit/miss rate, and performance
Introduction to paging and segmentation

Unit V INPUT/OUTPUT AND PARALLELISM

I/O devices and I/O interfacing techniques - Interrupts and DMA - Buses and bus arbitration - Basics of parallel architectures: SIMD, MIMD - Introduction to multi-core processors and GPU's.

SUGGESTED READINGS

1. Computer Organization and Design: The Hardware/Software Interface, David A. Patterson & John L. Hennessy, 5th Edition, Morgan Kaufmann Publishers, 2014.
2. Computer Architecture: A Quantitative Approach, John L. Hennessy & David A. Patterson, 6th Edition, Morgan Kaufmann Publishers, 2017.
3. Computer Organization and Architecture: Designing for Performance, William Stallings, 10th Edition, Pearson Education, 2016.
4. Structured Computer Organization, Andrew S. Tanenbaum & Todd Austin, 6th Edition, Pearson Education, 2013.
5. NPTEL–Computer Architecture by IIT Professors <https://nptel.ac.in/> Free video lectures on computer organization and architecture fundamentals.
6. Coursera – Computer Architecture (Princeton University <https://www.coursera.org/learn/comparch>)
Online course covering ISA, pipelining, memory, and performance

25CSU302

EMBEDDED SYSTEMS AND IOT

Semester –III
3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- To learn the internal architecture and programming of an embedded processor.
- To introduce interfacing I/O devices to the processor.
- To introduce the evolution of the Internet of Things (IoT).
- To build a small low-cost embedded and IoT system using Arduino/Raspberry Pi/ open platform.
- To apply the concept of Internet of Things in real real-world scenario.

Course Outcomes (COs)

At the completion of the course, the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Explain the architecture of embedded processors.	Remember
CO2	Write embedded C programs.	Create
CO3	Design simple embedded applications.	Analyze
CO4	Compare the communication models in IOT	Evaluate
CO5	Design IoT applications using Arduino/Raspberry Pi /open platform.	Create

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	3	3	3	3	2	2	2	2	2	3	3	3
CO2	2	2	3	2	2	0	0	0	3	3	1	3	2
CO3	3	3	3	3	2	2	2	1	2	2	2	2	3
CO4	2	1	3	2	2	0	0	0	3	2	2	2	2
CO5	2	2	3	2	2	2	1	2	2	3	3	2	2

1 - low, 2 - medium, 3 - high

Unit I -8 BIT EMBEDDED PROCESSORS

8-Bit Microcontroller – Architecture – Instruction Set and Programming – Programming Parallel Ports– Timers and Serial Port – Interrupt Handling

Unit II-EMBEDDED C PROGRAMMING

Memory And I/O Devices Interfacing – Programming Embedded Systems in C – Need For RTOS –Multiple Tasks and Processes – Context Switching – Priority Based Scheduling Policies.

UNIT III- IOT AND ARDUINO PROGRAMMING

Introduction to the Concept of IoT Devices – IoT Devices Versus Computers – IoT Configurations – Basic Components – Introduction to Arduino – Types of Arduino– Arduino Toolchain – Arduino Programming Structure – Sketches – Pins – Input/Output From Pins Using Sketches – Introduction to Arduino Shields – Integration of Sensors and Actuators with Arduino.

UNIT IV-IOT COMMUNICATION AND OPEN PLATFORMS

IoT Communication Models and APIs – IoT Communication Protocols – Bluetooth – WiFi – ZigBee– GPS – GSM modules – Open Platform (like Raspberry Pi) – Architecture – Programming – Interfacing – Accessing GPIO Pins – Sending and Receiving Signals Using GPIO Pins – Connecting to the Cloud.

UNIT V-APPLICATIONS DEVELOPMENT

Complete Design of Embedded Systems – Development of IoT Applications – Home Automation –Smart Agriculture – Smart Cities – Smart Healthcare.

SUGGESTED READINGS

- 1.Muhammed Ali Mazidi, Janice Gillispie Mazidi, Rolin D. McKinlay, "The 8051
- 2.Microcontroller and Embedded Systems", Pearson Education, Second Edition, 2014
- 3.Robert Barton, Patrick Grossetete, David Hanes, Jerome Henry, Gonzalo Salgueiro, "IoT
- 4,Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things", CISCO Press, 2017.
- 5.Michael J. Pont, "Embedded C", Pearson Education, 2007.
- 6.Wayne Wolf, "Computers as Components: Principles of Embedded Computer System Design", Elsevier, 2006.
- 7.Andrew N Sloss, D. Symes, C. Wright, "Arm System Developer's Guide", Morgan Kauffman/ Elsevier, 2006.
- 8.Arshdeep Bahga, Vijay Madiseti, "Internet of Things – A hands-on approach", Universities Press, 2015

25CSU321

OPERATING SYSTEMS

Semester –III
5H – 4C

Instruction Hours / week: L: 3 T: 0 P: 2

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours

Course Objectives

- Understand the core concepts and principles of operating systems, including process management, memory management, and file systems.
- Explore synchronization, inter-process communication, and resource allocation mechanisms.
- Develop practical skills in working with operating system functionalities and tools.
- Evaluate and compare different types of operating systems.
- Grasp the role of the operating system in modern computing systems, with a focus on emerging technologies.

Course Outcomes (COs)

At the completion of the course the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Explain the fundamental concepts of operating systems, including process and memory management	Understand
CO2	Analyze the performance of operating system components such as CPU scheduling, deadlock detection, and synchronization	Analyze
CO3	Develop solutions to common operating system challenges like process synchronization and memory allocation	Create
CO4	Evaluate basic system programs related to file management and resource allocation.	Evaluate
CO5	Apply theoretical knowledge to real-world scenarios involving OS design and troubleshooting	Apply

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	3	2	3	2	2	1	1	2	2	3	3
CO2	3	3	2	2	3	2	1	1	1	3	3	2	3
CO3	2	2	3	2	3	2	2	1	1	2	2	3	2
CO4	2	2	3	3	3	1	2	1	1	3	2	2	2
CO5	3	3	3	2	3	2	1	2	1	2	3	2	2

1 - low, 2 - medium, 3 - high

Unit I-INTRODUCTION TO OPERATING SYSTEMS

Overview of operating systems-Functions and types of operating systems-System calls and OS services-Structure of operating systems-Operating system design goals

Unit II-PROCESS MANAGEMENT

Processes and process states-Process control block-Process scheduling algorithms (FCFS, SJF, Round Robin)-Inter-process communication (IPC)-Process synchronization and mutual exclusion-Deadlock detection and prevention

Unit III-MEMORY MANAGEMENT

Memory management concepts-Contiguous memory allocation-Paging and segmentation-Virtual memory-Page replacement algorithms (FIFO, LRU, Optimal)

Unit IV-FILE SYSTEMS AND STORAGE MANAGEMENT

File system structure and organization-File allocation methods-Disk scheduling algorithms (FCFS, SSTF, C-SCAN)-File system security and protection-Secondary storage management

Unit V-ADVANCED TOPICS IN OPERATING SYSTEMS

Distributed systems and multi-core OS-Virtualization and cloud computing-Mobile operating systems-Security in operating systems-Case studies: Linux, Windows, and UNIX

PRACTICAL EXERCISES:

1. Basic Shell Commands: Navigating and managing files and processes in Linux
2. Shell Scripting: Write shell scripts for file handling, user management, and automation
3. CPU Scheduling Algorithms: Simulate FCFS, SJF, Round Robin, and Priority Scheduling
4. Thread Creation and Management: Create and manage threads using pthreads
5. Process Synchronization: Use semaphores/mutexes for producer-consumer and reader-writer problems
6. Banker's Algorithm: Simulate deadlock avoidance algorithm
7. Memory Allocation Techniques: Simulate First Fit, Best Fit, and Worst Fit strategies
8. Page Replacement Algorithms: Implement FIFO, LRU, and Optimal page replacement
9. File Allocation Techniques: Simulate contiguous, linked, and indexed file allocation
10. System Calls: Demonstrate use of fork(), exec(), wait(), open(), read(), write()

SOFTWARE/PLATFORM REQUIREMENTS:

- Ubuntu/Linux OS
- GCC Compiler
- Python (optional)
- Shell (bash)
- POSIX Libraries for process and thread handling

SUGGESTED READINGS

1. Silberschatz A., Galvin P. B., Gagne G., Operating System Concepts, 10th Edition, Wiley, 2018
2. Stallings W., Operating Systems: Internals and Design Principles, 9th Edition, Pearson Education, 2018.
3. Tanenbaum A. S., Bos H., Modern Operating Systems, 4th Edition, Pearson, 2014
4. Dhamdhere D. M., Operating Systems: A Concept-Based Approach, 3rd Edition, McGraw-Hill Education, 2017.
5. Milenkovic M., Operating Systems: Concepts and Design, 2nd Edition, McGraw-Hill, 1992

25CSU322

DATABASE MANAGEMENT SYSTEMS

Semester –III
5H – 4C

Instruction Hours / week: L: 3 T: 0 P: 2

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- Understand database architecture, data models, and ER modeling.
- Develop skills in SQL and relational algebra for querying databases.
- Apply normalization and manage transactions for data consistency.
- Implement concurrency control, indexing, and query optimization.
- Explore advanced databases like NoSQL, distributed, and cloud databases.

Course Outcomes (COs)

At the completion of the course, the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Recall the fundamental concepts of database systems and data models	Understand
CO2	Design relational databases using ER modelling and normalization techniques	Apply
CO3	Use optimized SQL queries for data definition and manipulation	Apply
CO4	Explain transaction concepts, concurrency control, and recovery techniques.	Understand
CO5	Analyze storage structures and indexing mechanisms used for efficient data retrieval.	Analyze

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	1	2	2	1-	1-	1-	1	1	1	3	2
CO2	3	3	2	2	2	1-	1-	1-	1	1	1	3	3
CO3	3	3	2	1	2	1-	1-	1-	1	2	1	3	2
CO4	3	3	2	2	3	1-	1-	1-	1	1	1	2	2
CO5	3	2	2	2	2	1-	1-	1-	1	1	1	2	2

1 - low, 2 - medium, 3 - high

Unit I- INTRODUCTION TO DATABASES

Purpose of database systems - Database system architecture - Data models (Relational, Hierarchical, Network) - Introduction to relational model - ER model: Entities, attributes, relationships - ER relational mapping

Unit II- RELATIONAL ALGEBRA AND SQL

Relational algebra operations - Tuple relational calculus - SQL: DDL, DML, DCL, TCL -Aggregate functions, joins, subqueries, views - Indexes, constraints, and triggers

Unit III-DATABASE NORMALIZATION AND TRANSACTION MANAGEMENT

Functional dependencies - Normal forms (1NF to BCNF) - Multi valued dependencies and 4NF – Lossless decomposition, dependency preservation, Transactions and ACID properties

Unit IV-CONCURRENCY CONTROL, QUERY PROCESSING

Serializability and concurrency control - Lock-based and timestamp-based protocol - Deadlocks and recovery. File organization (heap, sorted, hashed files) - Indexing: single-level and multi-level indexes B+ trees, hashing - Query processing and optimization basics

Unit V-ADVANCED TOPICS

Distributed databases-Temporal, Spatial and Multimedia Databases, Semantic Web Graph databases. Cloud Databases, NoSQL databases

PRACTICAL EXERCISES:

1. ER Diagram Creation and conversion to relational model
2. Creating databases and tables using SQL (DDL commands)
3. Data manipulation using DML commands (INSERT, UPDATE, DELETE)
4. Querying the database using SELECT with WHERE, ORDER BY, GROUP BY, HAVING
5. Implementing joins: INNER, LEFT, RIGHT, FULL OUTER
6. Nested queries and sub-queries
7. Views and Indexes
8. Constraints and Triggers
9. Stored procedures and functions
- 10.PL/SQL block using a cursor to display all employee names and salaries.

SUGGESTED READINGS

1. Silberschatz, Korth, Sudarshan, Database System Concepts, McGraw-Hill Education,7th Edition (2019)
2. Elmasri, Navathe, Fundamentals of Database Systems, Pearson Education,7th Edition (2016)
3. Ramakrishnan, Gehrke, Database Management Systems, McGraw-Hill Education,3rd Edition (2003)
4. Date, C. J. An Introduction to Database Systems, Pearson Education, 8th Edition (2003)
5. Feuerstein, Steven; Pribyl, zBill, Oracle PL/SQL Programming, O'Reilly Media,6th Edition (2014)

25CSU323

DESIGN AND ANALYSIS OF ALGORITHM

Semester –III
5H – 4C

Instruction Hours / week: L: 3 T: 0 P: 2

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours

Course Objectives

- Apply the algorithm analysis techniques in searching and sorting algorithms
- Critically analyze the efficiency of graph algorithms
- Understand different algorithm design techniques
- Solve programming problems using a state space tree
- Grasp the concepts behind NP-completeness, Approximation algorithms and randomized algorithms

Course Outcomes (COs)

At the completion of the course, the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Apply the algorithm analysis techniques in searching and sorting algorithms	Apply
CO2	Critically analyze the efficiency of graph algorithms	Analyze
CO3	Understand different algorithm design techniques	Understand
CO4	Solve programming problems using a state space tree	Apply
CO5	Grasp the concepts behind NP-completeness, Approximation algorithms and randomized algorithms..	Understand

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	2	2	2	2	2	2	2	2	2	2	3
CO2	3	3	3	2	3	2	2	2	2	2	3	3	3
CO3	3	3	2	2	3	2	2	2	2	1	2	2	3
CO4	3	3	2	3	2	2	2	2	2	2	2	3	3
CO5	2	2	1	2	1	3	2	3	3	2	2	1	2

1 - low, 2 - medium, 3 - high

Unit I-INTRODUCTION

Algorithm analysis: Time and space complexity - Asymptotic Notations and their properties. Best case, Worst case, and average case analysis – Recurrence relation: substitution method - Lower bounds – searching: linear search, binary search, and Interpolation Search, Pattern search: The naïve string-matching algorithm - Rabin-Karp algorithm - Knuth-Morris-Pratt algorithm. Sorting: Insertion sort – heap sort

Unit II-GRAPH ALGORITHMS

Graph algorithms: Representations of graphs - Graph traversal: DFS – BFS - applications - Connectivity, strong connectivity, bi-connectivity - Minimum spanning tree: Kruskal's and Prim's algorithm- shortest path: Bellman-Ford algorithm - Dijkstra's algorithm - Floyd-Warshall algorithm Network flow: Flow networks - Ford-Fulkerson method – Matching: Maximum bipartite matching

Unit III-ALGORITHM DESIGN TECHNIQUES

Divide and Conquer methodology: Finding maximum and minimum - Merge sort - Quick sort. Dynamic programming: Elements of dynamic programming — Matrix-chain multiplication - Multi-stage graph — Optimal Binary Search Trees. Greedy Technique: Elements of the greedy strategy- Activity-selection problem — Optimal Merge pattern — Huffman Trees.

Unit IV-STATE SPACE SEARCH ALGORITHMS

Backtracking: n-Queens problem - Hamiltonian Circuit Problem - Subset Sum Problem – Graph coloring problem Branch and Bound: Solving a 15-Puzzle problem - Assignment problem - Knapsack Problem - Travelling Salesman Problem

Unit V-NP-COMPLETE AND APPROXIMATION ALGORITHM

Tractable and intractable problems: Polynomial-time algorithms – Venn diagram representation - NP-algorithms - NP-hardness and NP-completeness – Bin Packing problem - Problem reduction: TSP – 3-CNF problem. Approximation Algorithms: TSP - Randomized Algorithms: concept and application - primality testing - randomized quick sort - Finding kth smallest number

PRACTICAL EXERCISES

1. Compare the performance of Linear Search, Binary Search, and Interpolation Search on different input sizes.
2. Design and implement the Naïve, Rabin-Karp, and Knuth-Morris-Pratt string matching algorithms.
3. Analyze the efficiency of Insertion Sort and Heap Sort through experimental evaluation.
4. Construct DFS and BFS algorithms to traverse graphs and record traversal order.
5. Develop programs to compute Minimum Spanning Trees using Kruskal's and Prim's algorithms.
6. Implement Dijkstra's and Bellman-Ford algorithms for finding the shortest path in weighted graphs.
7. Apply Floyd-Warshall algorithm to compute all-pairs shortest paths in a graph.
8. Solve the Maximum Bipartite Matching problem using appropriate graph-based approaches.
9. Demonstrate Matrix Chain Multiplication using Dynamic Programming to find optimal cost.
10. Generate Huffman Codes for a given set of characters with assigned frequencies.
11. Explore the solution space of the N-Queens problem using Backtracking.
12. Estimate a near-optimal solution for the Travelling Salesman Problem using an approximation

SUGGESTED READINGS

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, Introduction to Algorithms, MIT Press, 3rd Edition (2009).
2. Ellis Horowitz, Sartaj Sahni, and Sanguthevar Rajasekaran, Fundamentals of Computer Algorithms, Universities Press, 2nd Edition (2008).
3. Jon Kleinberg and Éva Tardos, Algorithm Design, Pearson Education, 1st Edition (2006).
4. Steven S. Skiena, The Algorithm Design Manual, Springer, 2nd Edition (2008).
5. Sara Baase and Allen Van Gelder, Computer Algorithms: Introduction to Design and Analysis, Pearson, 3rd Edition (2000).

25MAC311

YOGA

Semester –III
1H – 0C

Instruction Hours / week: L: 0 T: 0 P: 1

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- Articulate and Focus on goals and move away from demotivating and corrosive environment.
- Explain the concept and dimensions of wellness which help in pursuing their goals.
- Demonstrate the practices that can promote wellness.
- Plan and follow a daily routine that includes Physical Exercises, Yoga, and Meditation.
- Demonstrate behavior's aligned to universal Human Values

Course Outcomes (COs)

At the completion of the course, the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Demonstrate understanding of the holistic development of a human being.	Understand
CO2	Distinguish between physical needs and the needs of the self.	Understand
CO3	Apply human values in family and societal interactions.	Apply
CO4	Analyze nature and existence in terms of mutual harmony and co-existence.	Analyze
CO5	Integrate ethical human conduct in their professional practice.	Apply

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	-	-	-	2	2	3	2	-	-	1	3
CO2	2	2	-	-	-	-	--	--	2	2	-	-	1	2
CO3	-	3	-	-	-	-	3	3	-	3	1	-	1	-
CO4	-	3	-	3	2	-	3	3	-	3	1	-	2	-
CO5	3	2	2	-	1	2	3	3	3	2	-	-	2	3

1 - low, 2 - medium, 3 - high

Unit I PHYSICAL HEALTH

Introduction to SKY -Education as a means of Youth Empowerment-Simplified Physical exercises- Yog asanas (Rules- Sun Salutation-Dandasana-Chakrasana-Vrichasana-Trikonasana-Vajrasana-Pranayama-Nadi Suddhi-Clearance Practice)

Unit II STRENGTHENING THE LIFE FORCE

Reasons for Diseases-Philosophy of Kaya Kalpa -Maintaining Youthfulness & Postponing Aging – Transformation of Food into seven Body Constituents -Greatness of Seminal Fluid -Limit and Method in Five Factors- Kaya Kalpa Practice

Unit III WELLNESS OF MIND

Classification of Mind Waves-Agna Meditation- Shanthi Meditation- Thuriya Meditation-Blessing and Benefits-Virtues: Individual Virtues and Societal Virtues -Morals (Importance of Introspection, Six Temperaments and Manoevering, Benefits of Meditation)

Unit IV PROSPERITY OF MIND- PART I

Philosophy of Life (Purpose of Life, Philosophy of Life, Five Duties-Safeguarding Natural Resources)- Analysis of Thoughts (Ten stages of the Mind-The Five Kosas-Thoughts-Analysis of thoughts and practice)- Moralization of Desires (Desires-Explanation, Nature, Reasons, Moralisation Practice).

Unit V PROSPERITY OF MIND-PART II

Neutralization of Anger (Anger-Reasons, Effects, Peace, Tolerance and Forgiving, Neutralisation) - Eradication of Worries (Reasons, Effects, Corrective measures, Eradication)- Diversity in Men- Love and Compassion

25CSU401

ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

Semester –IV
3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100

End Semester Exam: 3 Hours

Course Objectives

- Study about uninformed and Heuristic search techniques.
- Learn techniques for reasoning under uncertainty
- Introduce Machine Learning and supervised learning algorithms
- Study about ensembling and unsupervised learning algorithms
- Learn the basics of deep learning using neural networks

Course Outcomes (COs)

At the completion of the course, the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Use appropriate search algorithms for problem solving	Understand
CO2	Apply reasoning under uncertainty	Apply
CO3	Build supervised learning models	Create
CO4	Build ensembling and unsupervised models	Create
CO5	Build deep learning neural network models	Create

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	2	2	2	2	2	2	1	2	1	3	2
CO2	3	3	3	2	3	1	1	1	2	2	2	3	3
CO3	3	3	2	2	3	1	1	1	1	2	1	3	2
CO4	3	3	2	3	2	1	1	1	1	2	1	3	3
CO5	2	2	1	2	1	1	2	3	1	3	2	2	1

1 - low, 2 - medium, 3 - high

Unit I-PROBLEM SOLVING

Introduction to AI - AI Applications - Problem solving agents – search algorithms –uninformed search strategies – Heuristic search strategies – Local search and optimization problems – adversarial search – constraint satisfaction problems (CSP)

Unit II-PROBABILISTIC REASONING

Acting under uncertainty – Bayesian inference – naïve bayes models. Probabilistic reasoning – Bayesian networks – exact inference in BN – approximate inference in BN – causal networks.

Unit III-SUPERVISED LEARNING

Introduction to machine learning – Linear Regression Models: Least squares, single & multiple variables, Bayesian linear regression, gradient descent, Linear Classification Models: Discriminant function – Probabilistic discriminative model - Logistic regression, Probabilistic generative model – Naive Bayes, Maximum margin classifier, Support vector machine, Decision Tree, Random forests

Unit IV-ENSEMBLE TECHNIQUES AND UNSUPERVISED LEARNING

Combining multiple learners: Model combination schemes, Voting, Ensemble Learning - bagging, boosting, stacking, Unsupervised learning: K-means, Instance Based Learning: KNN, Gaussian mixture models, and Expectation maximization

Unit V-NEURAL NETWORKS

Perceptron - Multilayer perceptron, activation functions, network training – gradient descent optimization – stochastic gradient descent, error backpropagation, from shallow networks to deep networks –Unit saturation (aka the vanishing gradient problem), ReLU, hyperparameter tuning, batch normalization, regularization, dropout.

SUGGESTED READINGS

1. Stuart Russell and Peter Norvig, "Artificial Intelligence – A Modern Approach", Fourth Edition, Pearson Education, 2021.
2. Ethem Alpaydin, "Introduction to Machine Learning", MIT Press, Fourth Edition, 2020.
3. Dan W. Patterson, "Introduction to Artificial Intelligence and Expert Systems", Pearson Education, 2007
4. Kevin Night, Elaine Rich, and Nair B., "Artificial Intelligence", McGraw Hill, 2008
5. Patrick H. Winston, "Artificial Intelligence", Third Edition, Pearson Education, 2006
6. Deepak Khemani, "Artificial Intelligence", Tata McGraw Hill Education, 2013
7. Bishop, C. M. *Pattern Recognition and Machine Learning*, Springer, 2006

25CSU402

FUNDAMENTALS OF DATA SCIENCE**Semester –IV**
3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100

End Semester Exam: 3 Hours

Course Objectives

- Understand the data science life-cycle and its real-world applications
- Gain proficiency in understanding the data science life-cycle and its real-world applications
- Gain proficiency in data collection, cleaning, and transformation techniques
- Develop skills in exploratory data analysis (EDA) and data visualization
- Learn basic concepts in statistical analysis and machine learning
- Use the ethical, privacy, and fairness considerations in data science and gain awareness of emerging trends such as big data, deep learning, and Auto-ML shaping the future of data science.

Course Outcomes (COs)

At the completion of the course, the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Recall the key concepts, tools, and roles involved in the Data Science life-cycle.	Remember
CO2	Apply techniques for collecting, cleaning, and preparing data for analysis.	Apply
CO3	Apply exploratory data analysis and visualize data insights using Python libraries.	Apply
CO4	Analyse basic statistical inference and machine learning algorithms to solve real problems.	Analyse
CO5	Explain ethical and social issues in data science and understand upcoming trends.	Understand

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	2	1	2	2	2	2	1	2	1	3	2
CO2	3	3	3	2	3	2	2	1	2	2	2	3	3
CO3	3	3	2	2	3	2	2	1	1	2	1	3	2
CO4	3	3	2	3	2	2	2	1	1	2	1	3	3
CO5	2	2	1	1	1	3	2	3	1	3	2	2	1

1 - low, 2 - medium, 3 - high

Unit I-INTRODUCTION TO DATA SCIENCE

What is Data Science - Applications and lifecycle of Data Science - Roles: Data Scientist, Data Engineer, Analyst - Tools: Python, Jupyter, Git, Pandas, Numpy

Unit II: DATA COLLECTION, WRANGLING, AND PREPROCESSING

Types of data (structured, unstructured, semi-structured) - Data collection methods: APIs, Web scraping, CSV, JSON - Data cleaning: missing values, outliers, duplicates - Data transformation and feature engineering

Unit III: EXPLORATORY DATA ANALYSIS (EDA) AND VISUALIZATION

Descriptive statistics and distributions - Correlation and covariance - Data visualization using Matplotlib, Seaborn - Dashboards with Plotly or Tableau (introduction)

Unit IV STATISTICAL INFERENCE AND BASIC MACHINE LEARNING

Probability, hypothesis testing, confidence intervals - Introduction to supervised vs. unsupervised learning - Algorithms: Linear regression, KNN, Decision Trees, K-Means - Model evaluation: accuracy, confusion matrix, precision, recall, F1-score

Unit V ETHICS, PRIVACY, AND FUTURE TRENDS IN DATA SCIENCE

Data privacy, fairness, and bias in algorithms - Ethics in data handling and AI usage - Overview of big data, deep learning, and AutoML trends

SUGGESTED READINGS

1. Aurélien Géron, "Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow, O'Reilly Media, 3rd Edition
2. Joel Grus, "Data Science from Scratch: First Principles with Python, O'Reilly Media, 2nd Edition
3. "Python Data Science Handbook: Essential Tools for Working with Data"– O'Reilly Media
4. Cathy O'Neil and Rachel Schutt, "Doing Data Science: Straight Talk from the Frontline", O'Reilly Media

25CSU421

COMPUTER GRAPHICS AND MULTIMEDIA

Semester –IV
5H – 4C

Instruction Hours / week: L: 3 T: 0 P: 2

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- Introduce the fundamentals of illumination models, color theory, and basic output primitives in computer graphics.
- Provide knowledge of 2D transformations, viewing systems, and clipping techniques.
- Develop understanding of 3D object modeling, transformations, and visible surface detection methods.
- Explore multimedia system components, standards, and file handling techniques.
- Enable creation of hypermedia content and gain practical experience with Blender for 3D graphics..

Course Outcomes (COs)

At the completion of the course, the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Demonstrate the principles of illumination, color models, and basic graphic primitives.	Understand
CO2	Construct two-dimensional transformations and clipping operations for graphical rendering.	Apply
CO3	Design three-dimensional models and apply transformation and viewing techniques.	Apply
CO4	Organize multimedia data and formats using standard compression and file handling methods.	Analyse
CO5	Create hypermedia content and 3D visual elements using multimedia authoring tools..	Create

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	1	1	2	3	2	2	1	2	2	1	3	2
CO2	3	2	2	2	3	2	3	2	2	2	2	3	2
CO3	3	2	2	2	3	2	3	2	1	1	2	3	2
CO4	3	2	2	2	3	2	3	2	2	2	2	3	2
CO5	3	1	1	2	3	2	2	2	2	1	1	3	2

1 - low, 2 - medium, 3 - high

Unit I ILLUMINATION AND COLOR MODELS

Light sources — basic illumination models — halftone patterns and dithering techniques; Properties of light — Standard primaries and chromaticity diagram; Intuitive colour concepts — RGB colour model — YIQ colour model — CMY colour model — HSV colour model — HLS colour model; Colour selection. Output primitives — points and lines, line drawing algorithms, loading the frame buffer, line function; circle and ellipse generating algorithms; Pixel addressing and object geometry, filled area primitives.

Unit II TWO-DIMENSIONAL GRAPHICS

Two dimensional geometric transformations — Matrix representations and homogeneous coordinates, composite transformations; Two-dimensional viewing — viewing pipeline, viewing coordinate reference frame; window-to-viewport coordinate transformation, Two dimensional viewing functions; clipping operations — point, line, and polygon clipping algorithms.

Unit III THREE-DIMENSIONAL GRAPHICS

Three dimensional concepts; Three-dimensional object representations — Polygon surfaces- Polygon tables- Plane equations — Polygon meshes; Curved Lines and surfaces, Quadratic surfaces; Blobby objects; Spline representations — Bezier curves and surfaces -B-Spline curves and surfaces. TRANSFORMATION AND VIEWING: Three dimensional geometric and modeling transformations — Translation, Rotation, Scaling, composite transformations; Three dimensional viewing — viewing pipeline, viewing coordinates, Projections, Clipping; Visible surface detection methods

Unit IV MULTIMEDIA SYSTEM DESIGN & MULTIMEDIA FILE HANDLING

Multimedia basics – Multimedia applications – Multimedia system architecture – Evolving technologies for multimedia – Defining objects for multimedia systems – Multimedia data interface standards – Multimedia databases. Compression and decompression – Data and file format standards – Multimedia I/O technologies – Digital voice and audio – Video image and animation – Full motion video – Storage and retrieval technologies

Unit V HYPERMEDIA

Multimedia authoring and user interface — Hypermedia messaging -Mobile messaging – Hypermedia message component – Creating hypermedia message – Integrated multimedia message standards – Integrated document management – Distributed multimedia systems. CASE STUDY: BLENDER GRAPHICS Blender Fundamentals — Drawing Basic Shapes — Modelling — Shading & Textures

PRACTICAL EXERCISES

1. Write a program to implement the Bresenham Line Drawing algorithm for drawing straight lines in a 2D coordinate system.
2. Write a program to generate and display a circle and ellipse using Midpoint Circle and Ellipse algorithms. Allow the user to change parameters such as the center and radius
3. Write a program that takes a window with a defined area and maps it to a viewport on the screen. Implement the transformation by applying the viewport and window coordinates to graphical objects.
4. Implement a program where the user can apply translation, scaling, and rotation to a 2D shape (such as a triangle or rectangle). Allow the user to input transformation parameters.
5. Write a program to clip lines in a window using the Cohen-Sutherland line clipping algorithm. Display the clipped portions of lines that fit inside a defined rectangular window
6. Write a program to clip a polygon using the Sutherland-Hodgman algorithm. Display the resulting clipped polygon after applying the clipping window.
7. Write a program to represent 3D objects (e.g., cube, pyramid) and apply basic transformations such as scaling, translation, and rotation. Display the transformed 3D object.
8. Write a program to project a 3D object onto a 2D screen using perspective projection. Implement transformations and project 3D coordinates onto the 2D plane.
9. Write a program that uses the Z-buffer algorithm to render a 3D scene, ensuring that only visible surfaces are displayed. Implement depth comparison to hide surfaces that are behind others.
10. Write a program to implement image compression using a basic technique like Run-Length Encoding (RLE) or JPEG compression. Allow users to decompress and view the image.
11. Write a program that can display images, audio, and text in a sequence to create a multimedia presentation. Implement play, pause, and navigation functions to control the slideshow.
12. Write a program to design a multimedia message with text, images, and audio. Implement the creation of hypermedia components, allowing the user to send multimedia messages as part of a hypermedia application.

SUGGESTED READINGS

1. Donald Hearn and M. Pauline Baker, "Computer Graphics with OpenGL", 4th Edition, Pearson Education, 2018.
2. Foley, Van Dam, Feiner, Hughes, "Computer Graphics: Principles and Practice", 3rd Edition, Addison-Wesley, 2013.
3. Ralf Steinmetz and Klara Nahrstedt, "Multimedia: Computing, Communications and Applications", Pearson Education, 2nd Edition, 2004.
4. Ze-Nian Li and Mark S. Drew, "Fundamentals of Multimedia", 2nd Edition, Springer, 2014.
5. Isaac V. Kerlow, "The Art of 3D Computer Animation and Effects", 4th Edition, Wiley India, 2009.

25CSU422

COMPUTER NETWORKS

Semester –IV
5H – 4C

Instruction Hours / week: L: 3 T: 0 P: 2

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours

Course Objectives

- Summarize network types, architecture layers, and switching techniques with key components.
- Explore framing techniques, error control mechanisms, and MAC protocols for reliable transmission.
- Construct network addressing schemes and routing paths using IPv4/IPv6 and routing algorithms.
- Examine reliable transmission principles using TCP/UDP, congestion control, and QoS techniques.
- Develop network applications using client-server architecture, security protocols, and socket programming.

Course Outcomes (COs)

At the completion of the course, the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Describe the functions of OSI/TCP-IP layers and identify network devices and switching techniques.	Remember
CO2	Explain how framing, error detection, and MAC protocols ensure data link reliability.	Understand
CO3	Apply subnetting, routing algorithms, and address translation mechanisms in IP networks.	Apply
CO4	Analyze the working of TCP and UDP protocols with flow, congestion control, and quality of service.	Analyze
CO5	Create basic socket-based communication programs and integrate fundamental security mechanisms.	Create

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	–	–	2	–	–	–	–	–	–	3	2
CO2	3	3	2	–	2	–	–	–	–	–	–	3	3
CO3	3	3	3	2	2	–	–	–	–	–	–	3	3
CO4	2	3	2	3	2	–	–	–	–	–	–	3	2
CO5	2	3	3	2	3	1	–	2	2	2	1	3	3

1 - low, 2 - medium, 3 - high

Unit I: INTRODUCTION TO NETWORKING

Basics of Computer Networks – Need and Uses - Types of Networks: LAN, MAN, WAN - Network Typologies and Transmission Media - ISO/OSI and TCP/IP Models – Layered Architecture - Switching Techniques: Circuit, Packet, and Message Switching - Introduction to Network Devices: Hub, Switch, Router, Bridge

Unit II: DATA LINK LAYER

Framing, Error Detection and Correction - Stop-and-Wait, Go-Back-N, and Selective Repeat ARQ - MAC Protocols – ALOHA, CSMA/CD, CSMA/CA - Ethernet (IEEE 802.3), Token Ring - Switching – Store and Forward, Cut Through

Unit III NETWORK LAYER

Logical Addressing – IPv4 and IPv6 - Sub-netting and Super-netting - Routing Algorithms: Distance Vector, Link State - IP Protocol, ICMP, ARP, RARP - Packet Switching and Forwarding - Network Address Translation (NAT)

Unit IV TRANSPORT LAYER

Process-to-Process Delivery - TCP and UDP – Features and Comparisons - TCP Congestion Control and Flow Control - Connection Establishment and Termination - Ports and Sockets - Quality of Service (QoS) and Reliability

Unit V: APPLICATION LAYER AND NETWORK PROGRAMMING

DNS, HTTP, FTP, SMTP, SNMP - Client-Server Architecture - Basics of Socket Programming (TCP/UDP sockets in Python or C) - Introduction to Network Security: Cryptography, Firewalls, VPNs - Emerging Trends: IoT Networking, SDN Basics

PRACTICAL EXERCISES:

1. Simulate data transmission between client and server using TCP sockets in Python/C
2. Implement UDP-based echo client-server application
3. Write a program to demonstrate Stop-and-Wait ARQ
4. Implement CRC or Checksum for error detection
5. Simulate Distance Vector Routing Algorithm
6. Implement Subnetting calculation tool
7. Observe and analyze ARP, ICMP, DNS traffic using Wireshark
8. Implement simple HTTP client using socket programming
9. Setup a basic packet forwarding using static routing in a simulator (e.g., Cisco Packet Tracer)
10. Demonstrate network address translation (NAT) using a simulation tool

SUGGESTED READINGS

1. James F. Kurose & Keith W. Ross, Computer Networking: A Top-Down Approach, 7th Edition, Pearson Education, 2017.
2. Andrew S. Tanenbaum & David J. Wetherall, Computer Networks, 5th Edition, Pearson Education, 2010.
3. William Stallings Computer Organization and Architecture: Designing for Performance, 10th Edition, Pearson Education, 2016.
4. NPTEL – Computer Networks (IIT Lectures) <https://nptel.ac.in/courses/106/105/106105183/> - Comprehensive lecture series on networking fundamentals and protocols Coursera
5. RFC Editor (IETF Standards and Protocols) <https://www.rfc-editor.org/> – Official documents on networking protocols such as HTTP, TCP, IP, etc.

25CSU411

ARTIFICIAL INTELLIGENCE AND DATA SCIENCE LAB

Semester –IV
4H – 2C

Instruction Hours / week: L: 0 T: 0 P: 4

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours

Course Objectives

- Reinforce Python programming skills using essential libraries like NumPy, Pandas, Matplotlib, and Seaborn.
- Enable students to perform effective data preprocessing and exploratory data analysis (EDA).
- Implement core machine learning algorithms such as classification, regression, and clustering.
- Apply dimensionality reduction techniques to visualize and analyze high-dimensional data.
- Evaluate machine learning models using standard performance metrics.

Course Outcomes (COs)

At the completion of the course, the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Apply Python libraries for data manipulation and visualization tasks.	Apply
CO2	Perform data preprocessing techniques like normalization, encoding, and handling missing values.	Apply
CO3	Implement classification, regression, and clustering algorithms on real-world datasets.	Create
CO4	Analyze data using dimensionality reduction methods like PCA and t-SNE.	Analyze
CO5	Evaluate machine learning models using confusion matrix, ROC curve, precision, recall, and F1-score.	Evaluate

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	2	2	2	2	2	1	1	2	2	3	2
CO2	3	3	2	2	2	1	1	2	1	1	1	3	3
CO3	3	3	3	2	2	2	1	2	1	1	2	3	3
CO4	2	3	2	3	2	2	2	2	1	2	2	3	2
CO5	2	3	3	2	3	1	1	2	2	2	1	3	3

1 - low, 2 - medium, 3 - high

PRACTICAL EXERCISES

1. Python refresher: Numpy, Pandas, Matplotlib, Seaborn
2. Data Preprocessing: Handling missing values, normalization, encoding
3. Data Visualization: Histograms, boxplots, pair plots, heatmaps
4. Search Algorithms: Implement BFS and DFS using Python
5. Constraint Satisfaction Problem: Sudoku Solver or N-Queens
6. Classification: Build and evaluate Decision Tree and k-NN models
7. Regression: Implement Linear and Polynomial Regression
8. Clustering: K-Means, DBSCAN on real datasets
9. Dimensionality Reduction: PCA and t-SNE
10. Evaluation Techniques: Confusion matrix, precision, recall, F1-score, ROC

SOFTWARE / TOOLS REQUIRED

- Python (3.x) with Jupyter Notebook / Google Colab
- Libraries: Numpy, Pandas, Scikit-learn, Matplotlib, Seaborn
- (Optional) TensorFlow or PyTorch for advanced exploration

SUGGESTED READINGS

1. Wes McKinney, Python for Data Analysis, 2nd Edition, O'Reilly Media, 2017.
2. Jake VanderPlas, Python Data Science Handbook, 1st Edition, O'Reilly Media, 2016.
3. Aurélien Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 2nd Edition, O'Reilly Media, 2019.
4. Stuart Russell and Peter Norvig, Artificial Intelligence: A Modern Approach, 4th Edition, Pearson Education, 2020

25IKS002

VEDIC MATHEMATICS AND CRITICAL THINKING

Semester –IV
2H – 2C

Instruction Hours / week: L: 2 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours

Course Objectives

- Learn ancient Vedic techniques for faster and smarter mathematical computation.
- Develop logical reasoning, analytical thinking, and problem-solving skills.
- Bridge traditional math with modern critical thinking.
Enhance mental agility through Vedic sutras and reasoning tools

Course Outcomes (COs)

At the completion of the course the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Apply Vedic mathematical techniques for faster arithmetic computation.	Apply
CO2	Demonstrate improved numerical aptitude and mental computation.	Understand
CO3	Utilize critical thinking frameworks for logical problem-solving.	Analyze
CO4	Identify fallacies and construct coherent arguments.	Evaluate
CO5	Integrate Vedic and logical reasoning in practical, real-life scenarios.	Apply/Create

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	2	-	-	-	2	2	3	2	-	-	1	3
CO2	2	2	-	-	-	-	--	--	2	2	-	-	1	2
CO3	-	3	-	-	-	-	3	3	-	3	1	-	1	-
CO4	-	3	-	3	2	-	3	3	-	3	1	-	2	-
CO5	3	2	2	-	1	2	3	3	3	2	-	-	2	3

1 - low, 2 - medium, 3 - high

Unit I: INTRODUCTION TO VEDIC MATHEMATICS

History and origin, Overview of 16 Sutras & 13 Sub-Sutras, Mental calculation strategies, Sutras: Ekādhikena Pūrvena, Nikhīlam Navataścaramam Daśatah

Unit II: ARITHMETIC WITH VEDIC METHODS

Techniques for rapid addition, subtraction, multiplication, and division, Digit sum verification, Shortcuts for squares and square roots

Unit III: CRITICAL THINKING FOUNDATIONS

Basics of critical thinking, Deductive vs. inductive reasoning, Logical fallacies and cognitive biases, Evaluating arguments

Unit IV: PROBLEM SOLVING AND ANALYTICAL REASONING

Pattern recognition, logic puzzles, Syllogisms, Venn diagrams, Decision-making frameworks

Unit V: INTEGRATION AND APPLICATION

Real-life quantitative applications, Logic-based questions for competitive exams, Critical essays and group debates

SUGGESTED READINGS

1. Tirthaji, B. K., 'Vedic Mathematics', Motilal Banarsidass Publishers, 1st Edition, New Delhi, 1965.
2. Bathia, D., 'Vedic Mathematics Made Easy', Jaico Publishing House, 1st Edition, Mumbai, 2005.
3. Gupta, A., 'The Power of Vedic Maths', Jaico Publishing House, 1st Edition, Mumbai, 2008.
4. Tekriwal, G., 'The Art of Indian Speed Calculation', S. Chand Publishing, 1st Edition, New Delhi, 2011.
5. Bassham, G. et al., 'Critical Thinking: A Student's Introduction', McGraw-Hill Education, 5th Edition, New York, 2012.
6. Browne, M. N. & Keeley, S. M., 'Asking the Right Questions', Pearson Education, 11th Edition, Boston, 2014.
7. Kallet, M., 'Think Smarter', Wiley, 1st Edition, Hoboken, 2014.

25CSU521

OBJECT ORIENTED SOFTWARE ENGINEERING

Semester –V
6H – 4C

Instruction Hours / week: L: 2 T: 0 P: 4

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours

Course Objectives

- Develop foundational knowledge of software engineering and object-oriented programming principles.
- Understand various phases of the software development life cycle using object-oriented approaches.
- Apply UML techniques and design patterns for software modeling and analysis.
- Explore object-oriented languages and software testing methodologies.
- Investigate advanced software engineering practices including MDE, AOP, and agile methods..

Course Outcomes (COs)

At the completion of the course, the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Identify the concepts of object-oriented software engineering	Understand
CO2	Perform requirements analysis and system design	Apply
CO3	Develop software using Object oriented programming languages	Create
CO4	Analyze software testing techniques and test-driven development practices	Analyze
CO5	Evaluate modern approaches like MDE, AOP, SOA	Evaluate

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	2	2	2	1	1	1	1	1	1	3	2
CO2	3	3	3	2	2	1	1	1	2	2	1	3	3
CO3	3	2	3	2	3	1	1	1	2	2	1	3	3
CO4	2	3	3	2	3	1	1	1	2	2	1	2	3
CO5	2	2	3	2	3	1	1	2	3	2	2	3	3

1 - low, 2 - medium, 3 - high

Unit-I INTRODUCTION TO OBJECT-ORIENTED PROGRAMMING

Overview of software engineering, Introduction to Object-Oriented Programming (OOP) concepts (classes, objects, inheritance, polymorphism), Unified Modelling Language (UML) basics, Introduction to software development process and software development life cycle (SDLC).

Unit -II REQUIREMENTS ANALYSIS AND DESIGN

Requirements analysis and specification, Use cases and Scenarios, Object-oriented analysis and design (OOAD), Design patterns, UML modelling techniques (class diagrams, sequence diagrams, state machine diagrams, activity diagrams)

Unit -III SOFTWARE CONSTRUCTION AND TESTING

Software construction basics, Object-oriented design principles, Object-oriented programming languages (Java, C++, Python), Software testing basics (unit testing, integration testing, system testing), Test-driven development (TDD)

Unit -IV SOFTWARE MAINTENANCE AND EVOLUTION

Software maintenance basics, refactoring techniques Software version control, Code review and inspection, Software evolution and re-engineering

Unit -V ADVANCED TOPICS IN OBJECT-ORIENTED SOFTWARE ENGINEERING

Model-driven engineering (MDE), Aspect-oriented programming (AOP), Component-based software engineering (CBSE), Service oriented architecture (SOA), Agile software development and Scrum methodologies.

PROJECT WORK (OOSE PROJECTS)

Documentation -Oriented (Analysis & Design):

- Library Management System
- Hospital Information System
- Online Course Registration System
- Vehicle Rental System
- Inventory Management System

IMPLEMENTATION -ORIENTED PROJECTS

1. Student Information Management System
 - Manage student records, attendance, grades
 - Implements concepts of classes, inheritance, file handling
2. Hotel Booking System
 - Rooms, guests, check-in/out
 - Includes use case diagram, class diagram, and database integration
3. Online Shopping System
 - Product catalog, cart, orders, users
 - Demonstrates polymorphism, encapsulation, and GUI (Swing/JavaFX/PyQt)
4. Bank Management System
 - Customer accounts, transactions, loan processing

- Can involve design patterns like Singleton (for DB connection)
5. Crime Record Management System
- Track FIRs, cases, police stations
 - Can focus on good modular design and test-driven development

SUGGESTED READINGS

1. Craig Larman, An Introduction to Object-Oriented Analysis and Design and the Unified Process, 3rd Edition, Prentice-Hall, 2004.
2. Sachin Malhotra, Programming in Java, Oxford University Press, 2014.
3. G. Kotonya and I. Sommerville, Requirements Engineering: Processes and Techniques, Wiley, 1998.
4. Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides, Design Patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley, 1994.
5. James Rumbaugh, Ivar Jacobson, and Grady Booch, The Unified Modeling Language Reference Manual, Addison-Wesley, 2004.

25CSU522

COMPILER DESIGN

Semester –V
5H – 4C

Instruction Hours / week: L: 3 T: 0 P: 2

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- Introduce the fundamental principles and structure of compilers, including lexical, syntax, and semantic analysis phases.
- Explain the process of lexical analysis using regular expressions, finite automata, and lexical analyzers like Lex.
- Develop an understanding of parsing techniques, including top-down, bottom-up, and LR parsing methods, and grammar handling.
- Provide knowledge of syntax-directed translation, intermediate code generation, and type checking for high-level language constructs.
- Familiarize students with run-time environments, code generation, and optimization techniques, including data-flow analysis and machine-independent optimizations

Course Outcomes (COs)

At the completion of the course, the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Demonstrate the structure of a compiler and fundamentals of lexical analysis using regular expressions and finite automata.	Understand
CO2	Explore syntax analysis techniques including context-free grammars, top-down and bottom-up parsing strategies.	Apply
CO3	Apply syntax-directed translation methods for generating intermediate code representations.	Apply
CO4	Analyze run-time environments and implement efficient code generation strategies.	Analyze
CO5	Examine machine-independent optimization techniques using data-flow analysis and loop optimizations.	Evaluate

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	1	1	2	2	2	2	-	2	-	3	2
CO2	3	3	3	2	3	-	-	-	2	2	2	3	3
CO3	3	3	2	2	3	-	-	-	-	2	1	3	2
CO4	3	3	2	3	2	-	-	-	-	2	-	3	3
CO5	2	2	1	1	1	3	2	3	-	3	2	2	1

1 - low, 2 - medium, 3 - high

Unit I INTRODUCTION

The structure of a compiler, the science of building a compiler, programming language basics Lexical Analysis: The Role of the Lexical Analyzer, Input Buffering, Recognition of Tokens, The Lexical Analyzer Generator Lex, Finite Automata, From Regular Expressions to Automata, Design of a Lexical-Analyzer Generator, Optimization of DFA-Based Pattern Matchers.

Unit II: SYNTAX ANALYSIS

Syntax Analysis: Introduction, Context-Free Grammars, Writing a Grammar, Top-Down Parsing, Bottom-Up Parsing, Introduction to LR Parsing: Simple LR, More Powerful LR Parsers, Using Ambiguous Grammars and Parser Generators.

Unit III: SYNTAX-DIRECTED TRANSLATION:

Applications of Syntax-Directed Translation, Syntax-Directed Translation Schemes, Implementing L Attributed SDDs. Intermediate-Code Generation: Variants of Syntax Trees, Three-Address Code, Types and Declarations, Type Checking, Control Flow, Switch-Statements, Intermediate Code for Procedures.

Unit IV: RUN-TIME ENVIRONMENTS, CODE GENERATION

Run-Time Environments: Stack Allocation of Space, Access to Nonlocal Data on the Stack, Heap Management, Introduction to Garbage Collection, Introduction to Trace-Based Collection. Code Generation: Issues in the Design of a Code Generator, The Target Language, addresses in the Target Code, Basic Blocks and Flow Graphs, Optimization of Basic Blocks, A Simple Code Generator, Peephole Optimization, Register Allocation and Assignment, Dynamic Programming Code-Generation.

Unit V: MACHINE-INDEPENDENT OPTIMIZATION

Machine-Independent Optimization: The Principal Sources of Optimization, Introduction to Data-Flow Analysis, Foundations of Data-Flow Analysis, Constant Propagation, Partial-Redundancy Elimination, Loops in Flow Graphs.

PRACTICAL EXERCISES

1. Construct a lexical analyzer using Lex to recognize tokens such as identifiers, keywords, and operators.
2. Simulate the conversion of regular expressions to finite automata (NFA/DFA) and visualize state transitions.
3. Analyze input buffering techniques and demonstrate token recognition efficiency.
4. Design and implement a context-free grammar for a small expression-based language.
5. Develop a recursive descent parser (top-down parser) for arithmetic expressions.
6. Implement a bottom-up parser using shift-reduce technique or a parser generator tool like YACC.
7. Apply syntax-directed translation schemes to generate intermediate code (3-address code) for arithmetic expressions.
8. Demonstrate type checking and control flow generation for conditional and looping statements.
9. Simulate stack-based memory allocation for procedure calls with activation records.
10. Generate and optimize code for basic blocks using peephole optimization techniques.
11. Perform register allocation using graph coloring or dynamic programming method
12. Execute data-flow analysis to perform constant propagation and loop optimization on flow graph

SUGGESTED READINGS

1. Alfred V. Aho, Monica S. Lam, Ravi Sethi, and Jeffrey D. Ullman, Compilers: Principles, Techniques and Tools, Pearson Education, 2nd Edition, 2007.
2. Dick Grune, Henri E. Bal, Criel J.H. Jacobs, and Koen G. Langendoen, Modern Compiler Design, Springer, 2nd Edition, 2012.
3. Keith D. Cooper and Linda Torczon, Engineering a Compiler, Elsevier, 2nd Edition, 2011.
4. Andrew W. Appel, Modern Compiler Implementation in C, Cambridge University Press, 1st Edition, 2004.
5. John R. Levine, Tony Mason, and Doug Brown, Lex & Yacc, O'Reilly Media, 2nd Edition, 1992

25CSU523

CLOUD COMPUTING

Semester –V
5H – 4C

Instruction Hours / week: L: 3 T: 0 P: 2

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours

Course Objectives

1. Introduce distributed computing paradigms and cloud computing fundamentals including delivery and deployment models.
2. Enable students to use public cloud services for infrastructure management and application deployment.
3. Provide foundational understanding of virtualization, hypervisors, containers, and Docker operations.
4. Familiarize students with Kubernetes architecture and container orchestration techniques.
5. Develop awareness of cloud security architecture, identity management, and trusted computing practices

Course Outcomes (COs)

At the completion of the course, the student will be able to

COs	Course Outcomes	Blooms Level
C01	Explain various distributed computing models and identify suitable cloud deployment and delivery models.	Understand
C02	Demonstrate the use of public cloud services for deploying and managing web and containerized applications.	Apply
C03	Compare virtualization and containerization techniques and build Docker container images.	Analyze
C04	Apply Kubernetes tools for orchestrating and managing containerized applications.	Apply
C05	Analyze cloud security architecture and recommend security practices for trusted cloud environments.	Analyze

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
C01	3	3	2	3	3	2	1	2	1	2	1	3	3
C02	3	2	3	3	2	2	2	2	1	2	2	3	2
C03	3	3	3	2	2	2	1	2	1	2	2	3	2
C04	3	3	2	3	3	2	1	2	1	2	2	3	2
C05	3	2	3	2	3	2	2	2	1	2	1	3	3

1 - low, 2 - medium, 3 - high

Unit I: FUNDAMENTALS OF CLOUD COMPUTING

Distributed Computing Taxonomy – Cluster, Grid, P2P, Utility, Cloud, Edge, Fog computing paradigms; Introduction to Cloud Computing – Cloud delivery models (XaaS), Cloud deployment models (Private, Public, Hybrid); Characteristics of Cloud, Major use cases of Cloud; disadvantages and best practices; Major public cloud players in the market; Security Issues and Challenges; Cloud Native application development – Introduction to JavaScript Cloud native application development.

Unit II: PUBLIC CLOUD

Public Cloud – Using public cloud for infrastructure management (compute and storage services), Web application deployment using public cloud services, and Deploying container images in public cloud. Overview of cognitive services, Case study on architecting cloud-based solutions for a chosen scenario.

Unit III : VIRTUALIZATION

Virtualization – Basics, Cloud vs Virtualization, Types of virtualization, Hypervisor types; Containers – Introduction to Docker and containers, containerization vs virtualization, docker architecture, Use cases, Learn how to build container images, Operations on container images

Unit IV KUBERNETES

Kubernetes – Need for orchestration, container orchestration methods, Introduction to Kubernetes, Kubernetes architecture, using YAML file, Running Kubernetes via minikube.

Unit V CLOUD COMPUTING SECURITY ARCHITECTURE

Cloud Computing Security Architecture: Architectural Considerations, General Issues, Trusted Cloud Computing, Secure Execution environments and Communications, Micro architectures, Identity Management and Access Control, Autonomic Security

PRACTICAL EXERCISES

1. Installation and configuration of own Cloud
2. Study and implementation of infrastructure as Service using Open Stack.
3. Implementation of Virtualization in Cloud Computing to Learn Virtualization
4. Write a program for Web feed using PHP and HTML
5. Write a Program to Create, Manage and groups User accounts in own Cloud by Installing Administrative Features.
6. Case Study Microsoft Azure
8. Copy files from one virtual machine to another
9. Install hadoop single node cluster
10. Create, deploy and launch virtual machines in open stack

SUGGESTED READINGS

1. Rajkumar Buyya et.al. Mastering cloud computing, McGraw Hill Education; 2013.
2. Matthias K, Kane SP. Docker: Up & Running: Shipping Reliable Containers in Production. " O'Reilly Media, Inc."; 2018.
3. Kocher PS. Microservices and Containers. Addison-Wesley Professional; 2018.
4. Sarkar A, Shah A. Learning AWS: Design, build, and deploy responsive applications using AWS Cloud components. Packt Publishing Ltd; 2018.
5. Menga J. Docker on Amazon Web Services: Build, deploy, and manage your container applications at scale. Packt Publishing Ltd; 2018.

25MAC501

ENTREPRENEURSHIP AND STARTUPS

Semester – V
3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- Demonstrate an understanding of the evolution, characteristics, and role of entrepreneurship in society.
- Construct knowledge of the startup ecosystem, including key support systems and government initiatives.
- Assemble techniques for opportunity identification and early-stage startup design.
- Manipulate domain-specific tools and knowledge (Engineering/Science/Arts) to build sector-specific startup models.
- Create a complete entrepreneurial plan, integrating business model elements and innovative strategies.

Course Outcomes (COs)

At the completion of the course, the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Identify different types of entrepreneurs, their mindset, and their contributions to economic development.	Understand
CO2	Recognize key startup ecosystem stakeholders and government startup policies to support new ventures.	Apply
CO3	Create innovative business ideas using ideation tools and develop a basic startup design with MVP and financial planning.	Create
CO4	Identify and evaluate entrepreneurial opportunities in emerging areas of Computer Science and Engineering such as AI, cybersecurity, cloud computing, and IoT.	Evaluate
CO5	Design scalable tech startup models and articulate business plans integrating appropriate technology stacks, legal frameworks, and data-driven growth strategi	Analyze

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	2	2	1	3	2	2	2	3	2	2	2
CO2	2	2	2	1	2	2	2	2	3	3	2	2	3	2
CO3	2	3	3	2	3	2	2	2	2	3	3	3	3	2
CO4	3	3	3	3	3	2	2	2	3	3	3	3	3	
CO5	3	3	3	3	3	2	2	2	3	3	3	3	3	

1 - low, 2 - medium, 3 - high

Unit I FUNDAMENTALS OF ENTREPRENEURSHIP

Evolution of Entrepreneurship; concept, significance, and role in economic development and nation-building. Theories and models of entrepreneurship development. Types of entrepreneurs and the entrepreneurial mindset: characteristics, motivation, risk-taking, and leadership. Common entrepreneurial myths, realities, failures, and analysis to restart after failure. Case studies of successful Indian and global entrepreneurs with insights.

Unit II STARTUP ECOSYSTEM AND SUPPORT SYSTEMS

Overview of Local and Global Startup Ecosystem. Entrepreneurial Success in Rural Areas and establishing local entrepreneurial system. Support Institutions and Stakeholders: Incubators, accelerators, venture capitalists, angel investors, Mentors, and other stakeholders. Government initiatives and startup policies: Startup India, Atal Innovation Mission, TBI, DST & MSME support schemes, and National Innovation Startup Policy (NISP) for HEIs.

Unit III OPPORTUNITY IDENTIFICATION AND STARTUP DESIGN

Creativity and innovation techniques: brainstorming, SCAMPER, design thinking, mind mapping. Problem identification, market analysis, and opportunity evaluation. From ideation to execution: idea selection, value proposition, and pitching. Introduction to Business Model Canvas, Minimum Viable Product (MVP) development, customer validation, and financial planning. Techno-economic feasibility and preparing Project Report - Financial Requirements.

Unit IV TECH-DRIVEN ENTREPRENEURSHIP IN COMPUTING

Entrepreneurial opportunities in emerging technologies: Artificial Intelligence, Machine Learning, Blockchain, Cybersecurity, Cloud Computing, Data Analytics, IoT, and Web & Mobile App Development. Technology selection and development process: Choosing appropriate tech stacks, cloud platforms, databases, APIs, and development tools. Product lifecycle: From ideation to MVP in software startups, including Agile methodology, CI/CD, DevOps, and version control.

Platform-based and SaaS models: architecture, monetization strategies, and scalability. Domain-specific startup trends: EdTech, FinTech, HealthTech, AgriTech, and Smart City Solutions. Case studies of successful Indian and global tech startups and their technology stack.

Unit V STARTUP STRATEGY, BUSINESS MODELS, AND SCALING IN COMPUTING

Tech startup business models: Freemium, Subscription, Licensing, Ad-based, Marketplace, and B2B/B2C strategies. Investor pitching for tech ventures: crafting compelling narratives, technology differentiation, and value proposition. Startup analytics: KPIs such as CAC, LTV, MRR, churn rate, user retention, and burn rate. Funding lifecycle: seed capital, angel investors, venture capital, and bootstrapping strategies for software ventures. Legal and regulatory aspects: software IP rights, open-source licensing, data protection laws (GDPR, DPDP Act). Scaling strategies: user growth, performance optimization, hiring tech talent, and cloud infrastructure scaling. Exit strategies and startup sustainability in the tech sector.

SUGGESTED READINGS

1. Hisrich, R. D., Peters, M. P., & Shepherd, D. A. *Entrepreneurship*, 11th Edition, McGraw Hill Education, 2022.
2. S.S. Khanka, *Entrepreneurial Development*, S. Chand Publishing, Revised Edition, 2023.
3. Vasant Desai, *Dynamics of Entrepreneurial Development and Management*, Himalaya Publishing House, 2020
4. Bill Aulet, *Disciplined Entrepreneurship: 24 Steps to a Successful Startup*, Wiley, Updated Edition, 2021
5. Donald F. Kuratko, *Entrepreneurship: Theory, Process, and Practice*, 11th Edition, Cengage Learning, 2021. ISBN: 978-0357899580
6. Eric Ries, *The Lean Startup*, Penguin Random House, Special Indian Edition, 2021. ISBN: 978-0670921607
7. Duening, T. N., Hisrich, R. D., & Lechter, M. A. *Technology Entrepreneurship: A Treatise on Entrepreneurs and Entrepreneurship for and in Technology Ventures*, 3rd Edition, Academic Press, 2020. ISBN: 978-128178850
8. Hisrich, R. D., Peters, M. P., & Shepherd, D. A. *Entrepreneurship*, 11th Edition, McGraw Hill Education, 2022. ISBN: 978-9354600240
9. Ries, E. *The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses*, 1st Edition, Crown Business, 2011. ISBN: 978-0307887894

25CSU621

INTERNET PROGRAMMINGSemester –VI
5H – 4C

Instruction Hours / week: L: 3 T: 0 P: 2

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- To understand about client-server communication and protocols used during communication
- To design interactive web pages using Scripting languages.
- To learn server side programming using servlets and JSP.
- To develop web pages using XML/XSLT.
- To develop the web applications using web development framework.

Course Outcomes (COs)

At the completion of the course the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Describe the architecture of the Internet, web communication protocols and structure web content using HTML, XHTML	Understand
CO2	Apply Cascading Style Sheets and client-side scripting using JavaScript to create interactive web pages.	Apply
CO3	Construct & manipulate Structured web data using XML technologies	Create
CO4	Build server-side application using Java Servlets, and JSP	Create
CO5	Implement full-stack web applications, design and Style web pages	Create

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	2	3	3	3	1	2	1	2	1	3	3
CO2	3	2	3	2	2	2	2	2	1	2	2	3	2
CO3	3	3	3	2	2	2	1	2	1	2	2	3	2
CO4	3	3	2	3	3	2	1	2	1	2	2	3	2
CO5	3	2	3	2	3	2	2	2	1	2	1	3	3

1 - low, 2 - medium, 3 - high

Unit I: WEB SITE BASICS AND HTML

Web Essentials: Clients, Servers, and Communication. The Internet-Basic Internet Protocols -The World Wide Web-HTTP request message-response message-Web Clients Web Servers. Markup Languages: XHTML. An Introduction to HTML History-Versions-Basic XHTML Syntax and Semantics-Some Fundamental HTML Elements-Relative URLs-Lists Tables-Frames-Forms-HTML 5.0.

Unit II: CSS AND CLIENT-SIDE SCRIPTING

Style Sheets: CSS-Introduction to Cascading Style Sheets-Features-Core Syntax-Style Sheets and HTML-Style Rule Cascading and Inheritance-Text Properties-Box Model Normal Flow Box Layout-Beyond the Normal Flow-CSS3.0. Client-Side Programming: The JavaScript Language-History and Versions Introduction JavaScript in Perspective-Syntax Variables and Data Types-Statements-Operators-Literals-Functions-Objects-Arrays-BuiltinObjects-JavaScript Debuggers.

Unit III: CLIENT-SIDE TECHNOLOGIES

Model-View-Controller Paradigm, Representing Web Data: XML-Documents and Vocabularies-Versions and Declaration-Namespaces-DOM based XML processing Event oriented Parsing: SAX-Transforming XML Documents-Selecting XML Data: XPATH Template based Transformations: XSLT-Displaying XML Documents in Browsers.

Unit IV SERVER-SIDE TECHNOLOGIES

Server-Side Programming: Java Servlets-Architecture -Overview-A Servlet-Generating Dynamic Content-Life Cycle-Parameter Data-Sessions-Cookies-URL Rewriting-Other Capabilities-Data Storage Servlets and Concurrency-Databases and Java Servlets. Separating Programming and Presentation: JSP Technology Introduction-JSP and Servlets Running JSP Applications Basic JSP-JavaBeans Classes and JSP-Tag Libraries and Files Support for the Model-View-Controller Paradigm-Databases and JSP.

Unit V APPLICATION DEVELOPMENT ENVIRONMENT

Overview of MVC architecture – Java Server Faces: Features – Components – Tags – Struts: Working principle of Struts – Building model components – View components – Controller components – Forms with Struts – Presentation tags – Developing Web applications – Hibernate: Configuration Settings – Mapping persistent classes – Working with persistent objects – Concurrency – Transactions – Caching – Queries for retrieval of objects – Spring: Framework – Controllers – Developing simple applications.

PRACTICAL EXERCISES

1. Personal Portfolio Website (HTML/CSS/JavaScript)
2. To-Do List Application (JavaScript Local Storage)
3. Online Quiz System (HTML/CSS/JS with basic scoring)
4. Student Record Management System (PHP + MySQL)
5. Online Feedback Form with database storage
6. Blogging Platform (CRUD operations using PHP/MySQL or Node.js)
7. E-Commerce Website (product listing, cart, checkout, user login)
8. Online Food Ordering System
9. Event Management System
10. Real-time Chat Application (Node.js + Socket.io)
11. Online Code Editor (HTML, CSS, JS sandbox with live preview)

SUGGESTED READINGS:

1. Jeffrey C. Jackson, "Web Technologies--A Computer Science Perspective", Pearson Education, 2006.
2. Deitel, Deitel, Goldberg, "Internet & World Wide Web How to Program", Third Edition, Pearson Education, 2006.
3. Robert. W. Sebesta, "Programming the World Wide Web", Fourth Edition, Pearson Education, 2007
4. Marty Hall and Larry Brown, "Core Web Programming Second Edition", Volume I and II, Pearson Education, 2001.
5. Bates, "Developing Web Applications", Wiley, 2006.

25CSU622

MOBILE APPLICATION DEVELOPMENTSemester –VI
5H – 4C

Instruction Hours / week: L: 3 T: 0 P: 2

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- To understand the fundamentals of web and mobile application development.
- To learn client-side and server-side technologies used in modern web/mobile apps.
- To explore XML, MVC paradigms, and frameworks for dynamic app development.
- To design and develop applications using servlets, JSP, and component-based technologies.
- To build full-fledged applications using frameworks like JSF, Struts, Hibernate, and Spring.

Course Outcomes (COs)

At the completion of the course, the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Explain the fundamentals of web architecture and develop static web pages using HTML and CSS.)	Understand
CO2	Develop interactive client-side applications using JavaScript and manipulate web content dynamically.	Create
CO3	Demonstrate data representation and transformation using XML, XSLT, and related technologies.	Understand
CO4	Design and build server-side components using Servlets and JSP integrated with backend databases.	Create
CO5	Build and deploy scalable enterprise web applications using JSF, Struts, Hibernate, and Spring frameworks.	Create

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	3	2	3	3	3	1	2	1	2	1	3	3
CO2	3	2	3	3	3	2	2	2	1	2	2	3	2
CO3	3	3	2	2	2	3	1	2	1	2	2	3	2
CO4	3	3	2	3	3	2	1	2	1	2	2	3	2
CO5	3	2	3	2	3	2	2	2	1	2	1	3	3

1 - low, 2 - medium, 3 - high

Unit I: WEB SITE BASICS AND HTML

Web Essentials: Clients, Servers, and Communication – The Internet: Basic Internet Protocols – The World Wide Web – HTTP request message, response message – Web Clients and Web Servers. Markup Languages: XHTML – Introduction to HTML – History and Versions – Basic XHTML Syntax and Semantics – Some Fundamental HTML Elements – Relative URLs – Lists – Tables – Frames – Forms – HTML 5.0.

Unit II: CSS AND CLIENT-SIDE SCRIPTING

Style Sheets: CSS – Introduction to Cascading Style Sheets – Features – Core Syntax – Style Sheets and HTML – Style Rule Cascading and Inheritance – Text Properties – Box Model – Normal Flow – Box Layout – Beyond the Normal Flow – CSS3.0. Client-Side Programming: The JavaScript Language – History and Versions – Introduction – JavaScript in Perspective – Syntax – Variables and Data Types – Statements – Operators – Literals – Functions – Objects – Arrays – Built-in Objects – JavaScript Debuggers.

Unit III: CLIENT-SIDE TECHNOLOGIES

Model-View-Controller Paradigm – Representing Web Data: XML – Documents and Vocabularies – Versions and Declaration – Namespaces – DOM-based XML Processing – Event-Oriented Parsing: SAX – Transforming XML Documents – Selecting XML Data: XPATH – Template-Based Transformations: XSLT – Displaying XML Documents in Browsers.

Unit IV: SERVER-SIDE TECHNOLOGIES

Server-Side Programming: Java Servlets – Architecture Overview – A Servlet – Generating Dynamic Content – Life Cycle – Parameter Data – Sessions – Cookies – URL Rewriting – Other Capabilities – Data Storage – Servlets and Concurrency – Databases and Java Servlets. Separating Programming and Presentation: JSP Technology – Introduction – JSP and Servlets – Running JSP Applications – Basic JSP – JavaBeans Classes and JSP – Tag Libraries and Files – Support for the Model-View-Controller Paradigm – Databases and JSP.

Unit V: APPLICATION DEVELOPMENT ENVIRONMENT

Overview of MVC Architecture – Java Server Faces (JSF): Features – Components – Tags. Struts Framework: Working Principle of Struts – Building Model Components – View Components – Controller Components – Forms with Struts – Presentation Tags – Developing Web Applications. Hibernate: Configuration Settings – Mapping Persistent Classes – Working with Persistent Objects – Concurrency – Transactions – Caching – Queries for Retrieval of Objects. Spring Framework: Architecture – Controllers – Developing Simple Applications.

PRACTICAL EXERCISES

1. Design a personal profile or portfolio page using HTML/XHTML that includes lists, tables, frames, and forms with proper structure and formatting.
2. Create a user registration or feedback form using HTML5 form elements and attributes for basic client-side validation.
3. Style HTML pages using CSS, including layout control with the box model, and apply CSS3 features such as gradients, transitions, and shadows.
4. Develop JavaScript-based form validation for login or registration, including checks for empty fields, email format, and password rules.
5. Create a dynamic web application (like a to-do list) using JavaScript and DOM manipulation to add, update, and remove elements.
6. Write an XML document to store structured data and apply XSLT with XPath to transform and display the content in a web browser.
7. Develop a simple Java Servlet to handle form input from an HTML page and dynamically display the output using server-side scripting.
8. Implement user session tracking using Servlets with HttpSession, cookies, and URL rewriting to manage login and navigation flow.
9. Integrate JavaBeans with JSP pages to separate business logic from presentation in a product listing or user profile display application.
10. Develop a JSP-based CRUD application connected to a MySQL database using JDBC to perform insert, update, delete, and fetch operations.
11. Create a user login module using the Struts framework, implementing model, view, and controller components with form beans and action classes.
12. Build a full-stack application using Spring MVC and Hibernate to perform database operations like adding and retrieving records with ORM mapping.

SUGGESTED READINGS

1. Ralph Moseley and M.T. Savaliya, *Developing Web Applications*, Wiley-India, 2007.
2. Uttam K. Roy, *Web Technologies*, Oxford University Press, 2010.
3. Marty Hall and Larry Brown, *Core Servlets and Java Server Pages*, Vol.1, Pearson, 2nd Edition, 2003.
4. Bryan Basham, Kathy Sierra, Bert Bates, *Head First Servlets and JSP*, O'Reilly Media, 2nd Edition, 2008.
5. Craig Walls, *Spring in Action*, Manning Publications, 5th Edition, 2018.
6. Gavin King et al., *Hibernate in Action*, Manning Publications, 2004.

25CSU623

FULL STACK DEVELOPMENT

Semester –VI
5H – 4C

Instruction Hours / week: L: 3 T: 0 P: 2

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- Understand the essential JavaScript concepts for web development.
- Style Web applications using Bootstrap.
- Utilize React JS to build front-end User Interface.
- Understand the usage of APIs to create web applications using Express JS.
- Store and model data in a NoSQL database

Course Outcomes (COs)

At the completion of the course, the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Apply Javascript to build dynamic and interactive Web projects.	Apply
CO2	Implement user interface components for JavaScript-based Web using React.js.	Create
CO3	Apply Express/Node to build web applications on the server side	Apply
CO4	Develop data model in an open-source NoSQL database	Create
CO5	Demonstrate modularization and packing of the front-end modules	Understand

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	3	2	3	3	3	1	2	2	2	1	3	3
CO2	3	2	3	3	3	2	2	2	2	2	2	3	2
CO3	3	3	2	2	2	3	1	2	2	2	2	3	2
CO4	3	3	2	3	3	2	1	2	2	2	2	3	2
CO5	3	2	3	2	3	2	2	2	2	2	1	3	3

1 - low, 2 - medium, 3 - high

Unit I: SCRIPTS

Script Instructions, Statements, Comments, Variables, Data Types, Arrays, Strings, Functions, Methods & Objects, Decisions & Loops.

Unit II: DOCUMENT OBJECT MODEL

Document Object Model: DOM Manipulation, Selecting Elements, Working with DOM Nodes, Updating Element Content & Attributes, Events, Different Types of Events, How to Bind an Event to an Element, Event Delegation, Event Listeners

Unit III: FORMS-MERN COMPONENTS

Form enhancement and validation. Introduction to MERN: MERN components, Serverless Hello world. React Components: Issue Tracker, React Classes, Composing Components, Passing Data Using Properties, Passing Data Using Children, Dynamic Composition.

Unit IV EVENT HANDLING

React State: Initial State, Async State Initialization, Updating State, Lifting State Up, Event Handling, Stateless Components, Designing Components, State vs. Props, Component Hierarchy, Communication, Stateless Components. Express, REST API, GraphQL, Field Specification, Graph Based, Single Endpoint, Strongly Typed, Introspection, Libraries, The About API GraphQL Schema File, The List API, List API Integration, Custom Scalar types, The Create API, Create API Integration, Query Variables, Input Validations, Displaying Errors.

Unit V: MONGODB

MongoDB: Basics, Documents, Collections, Databases, Query Language, Installation, The Mongo Shell, MongoDB CRUD Operations, Create, Read, Projection, Update, Delete, Aggregate, MongoDB Node.js Driver, Schema Initialization, reading from MongoDB, Writing to MongoDB. Modularization and Webpack, Back-End Modules, Front-End Modules and Webpack Transform and Bundle, Libraries Bundle, Hot Module Replacement, Debugging Define Plugin: Build Configuration, Production Optimization

PRACTICAL EXERCISES

1. Write a script that Logs "Hello, World!" to the console. Create a script that calculates the sum of two numbers and displays the result in an alert box. b. Create an array of 5 cities and perform the following operations: Log the total number of cities. Add a new city at the end. Remove the first city. Find and log the index of a specific city
2. Read a string from the user. Find its length. Extract the word "JavaScript" using substring() or slice(). Replace one word with another word and log the new string. Write a function isPalindrome(str) that checks if a given string is a palindrome (reads the same backward).
3. Create an object student with properties: name (string), grade (number), subjects (array), displayInfo() (method to log the student's details). Write a script to dynamically add a passed property to the student object, with a value of true or false based on their grade. Create a loop to log all keys and values of the student object.
4. Create a button in your HTML with the text "Click Me". Add an event listener to log "Button clicked!" to the console when the button is clicked. Select an image and add a mouseover event listener to change its border color. Add an event listener to the document that logs the key pressed by the user.

5. Build a React application to track issues. Display a list of issues (use static data). Each issue should have a title, description, and status (e.g., Open/Closed). Render the list using a functional component.
6. Create a component Counter with A state variable count initialized to 0. Create Buttons to increment and decrement the count. Simulate fetching initial data for the Counter component using useEffect (functional component) or componentDidMount (class component). Extend the Counter component to Double the count value when a button is clicked. Reset the count to 0 using another button..
7. Install Express (npm install express). Set up a basic server that responds with "Hello, Express!" at the root endpoint (GET /). Create a REST API. Implement endpoints for a Product resource: GET: Returns a list of products. POST: Adds a new product. GET /:id: Returns details of a specific product. PUT /:id: Updates an existing product. DELETE /:id: Deletes a product. Add middleware to log requests to the console. Use express.json() to parse incoming JSON payloads.
8. Install the MongoDB driver for Node.js. Create a Node.js script to connect to the shop database. Implement insert, find, update, and delete operations using the Node.js MongoDB driver. Define a product schema using Mongoose. Insert data into the products collection using Mongoose. Create an Express API with a /products endpoint to fetch all products. Use fetch in React to call the /products endpoint and display the list of products. Add a POST /products endpoint in Express to insert a new product. Update the Product List. After adding a product, update the list of products displayed in React.

SUGGESTED READINGS

1. Robin Wieruch, "The Road to React: Your Journey to Master Plain Yet Pragmatic React.js", 2023 Edition, Independently Published.
2. Ethan Brown, "Learning JavaScript: JavaScript Essentials for Modern Application Development", 3rd Edition, O'Reilly Media, 2023.
3. Brad Dayley, Brendan Dayley, and Caleb Dayley, "Node.js, MongoDB, and Angular Web Development", 2nd Edition, Addison-Wesley, 2020.
4. Shama Hoque, "Full-Stack React Projects: Modern web development using React 18, Node.js, Express, and MongoDB", 3rd Edition, Packt Publishing, 2023.
5. Samer Buna, "Learning GraphQL: Declarative Data Fetching for Modern Web Apps", 2nd Edition, O'Reilly Media, 2022.

25CSU611

COMPETITIVE CODING

Semester –VI
2H – 1C

Instruction Hours / week: L: 0 T: 0 P: 2

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- To impart knowledge of designing solutions to print various patterns or shapes.
- Be familiar with various number-based problems and their solutions.
- To understand problem concepts based on arrays and design solutions.
- Be familiar with functions, recursion, and to design the implementation of solutions based on recursion.
- To learn and develop solutions for problems on pointers.
- To understand and apply solutions based on linked lists, stacks, and queues

Course Outcomes (COs)

At the completion of the course, the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Design solutions for problems based on various printing patterns/shapes.	Understand
CO2	Demonstrate the basic principles of various number-based problems and design solutions	Understand
CO3	Apply appropriate algorithm design technique to solve array-based application problems	Apply
CO4	Identify suitable method to solve problems based on functions and recursion	Analyze
CO5	Solve problems based on pointers	Evaluate

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	3	3	2	1	2	-	-	2	2	-	3	2
CO2	3	3	3	2	2	2	-	-	1	1	-	3	3
CO3	2	3	2	1	2	1	-	-	1	1	-	2	2
CO4	2	3	3	2	2	2	-	-	2	2	-	3	2
CO5	3	3	3	2	3	3	2	2	3	2	2	3	3

1 - low, 2 - medium, 3 - high

Unit I: BASIC PROBLEMS

Pattern-based problems, triangle pattern, diamond shapes, Pascal shape, Floyd triangle, etc. Number-based problems: Prime, Armstrong, Strong, Perfect, Palindrome, Kaprekar numbers etc.

Code the following

1. Print the multiplication table of a given number up to another specific number
2. Find m power n
3. Find out the generic root of a number by c program
4. Sum of Prime factors of a number
5. Display the factors of a given number
6. Find whether given number is Armstrong or not?
7. Find whether given number is Perfect or not?
8. Find whether given number is Strong or not?
9. Find whether given number is Palindrome or not?
10. Find whether given number is Kaperkar or not?
11. Print following patterns

a.

```
5
5 4
5 4 3
5 4 3 2
5 4 3 2 1
```

b.

```
1
2 3
4 5 6
7 8 9 10
```

c.

```
1 2 3 4 5
1 2 3 4
1 2 3
1 2
1
```

d.

```
*
* * *
* * * * *
* * * * * *
* * * * * * *
* * * * * * *
* * * * * *
* * * * *
* * * *
* * *
*
```

e.

```
3
```

44
555
6666
555
44
3

f.
1
3*2
6*5*4
10*9*8*7

12. Find nth Prime
13. Find first n primes
14. Find 1 to n primes
15. Find whether the given number is a perfect square or not.

Unit II: ARRAY BASED PROBLEMS

Array-based problems (one-dimensional and two-dimensional) – segregation-making two lists from one list making three lists from one list, list searching techniques, various sorting, special series problems, matrix printing in spiral order, matrix diagonal printing, maze problems, etc.

Character and string-based problems – removal of spaces, word reversal in sentence, reverse sentence without reversing words, making string shortest possible palindrome, etc.

Code the following:

1. Conversion from uppercase to lowercase using a C program
2. Counting different characters in a string using C program
3. Code which prints the initial of any name
4. Segregate 0's & 1's in a list. Segregate 0's, 1's & 2's in an array. Segregate positive and negative numbers in an array.
5. Find the second last index occurrence of a number in list of elements and if the given number does not occur twice, display -1.
6. Longest sub sequence in an array
7. Find the only number which occurs odd number of times in an array (where all other numbers are even) without counting.
8. Remove duplicates from sorted array
9. Print a given matrix in spiral form Given a 2d array, print it in spiral form. See the following examples.

Please comment down the code in other languages as well below –

Input:

```
1 2 3 4
5 6 7 8
9 10 11 12
13 14 15 16
```

Output:

```
1 2 3 4 8 12 16 15 14 13 9 5 6 7 11 10
```

Input:

```
1 2 3 4 5 6
7 8 9 10 11 12
13 14 15 16 17 18
```

Output:

```
1 2 3 4 5 6 12 18 17 16 15 14 13 7 8 9 10 11
```

10. Print the following pattern

Testcase 1:

```
1 1 1 1 1 1 2
3 2 2 2 2 2 2
3 3 3 3 3 3 4
```

```
5 4 4 4 4 4 4
5 5 5 5 5 5 6
7 6 6 6 6 6 6
```

Testcase 2:

```
1 1 1 2
3 2 2 2
3 3 3 4
```

11. Zigzag (or diagonal) traversal of Matrix

Given a 2D matrix, print all elements of the given matrix in diagonal order. For example, consider the following 5 X 4 input matrix.

```
1   2   3   4
5   6   7   8
9   10  11  12
13  14  15  16
17  18  19  20
```

Diagonal printing of the above matrix is

```
1   5   2   9   6   3   13  10   7   4   17  14   11
    8   18  15  12  19  16  20
```

12. Check for Matrix Symmetry

13. Removal of spaces in a string

14. Reverse words in sentence. Reverse sentence without reversing words

15. Make a string the shortest possible palindrome

Unit III: FUNCTIONS AND RECURSION

Functions and recursion, pointer-based problems, function pointers and array pointers

1. Write a C program to print Fibonacci series of given range.

2. Consider the below series: 0,0,2,1,4,2,6,3,8,4,10,5,12,6,14,7,16,8

This series is a mixture of 2 series. All the odd terms in this series form even numbers in ascending order, and every even term is derived from the previous term using the formula $(x/2)$

Write a program to find the nth term in this series.

The value n in a positive integer that should be read from stdin. The nth term that is calculated by the program should be written to stdout. You can assume that the n will not exceed 20,000.

3. Consider the following series: 1,1,2,3,4,9,8,27,16,81,32,243,64,729,128,2187...

This series is a mixture of 2 series – all the odd terms in this series form a geometric series, and all the even terms form yet another geometric series. Write a program to find the nth term in the series. The value n is a positive integer that should be read from stdin. The nth term that is calculated by the program should be written to stdout. May consider that n not greater than 30.

4. There is a colony of 8 houses represented as cells arranged in a straight line. Each day, every cell competes with its adjacent cells (neighbours). Each day, for each cell, if its neighbours are both active or both inactive, the cell becomes inactive the next day; Otherwise, it becomes active the next day.

Assumptions: the two cells on the ends have a single adjacent cell, so the other adjacent cell can be assumed to be always inactive. Even after updating the cell state, consider its previous state for updating the state of other cells. Update the cell information of all cells simultaneously.

Write a function cellcompete which takes one 8 element array of integers cells representing the current state of 8 cells and one integer days representing the number of days to simulate. An integer value of 1 represents an active cell and value of 0 represents an inactive cell.

5. The least recently used (lru) cache algorithm exits the element from the cache that was least recently used when the cache is full. After an element is requested from the cache, it should be added to the cache (if not there) and considered the most recently used element in the cache whether it is newly added or was already existing. Initially, the cache is empty. Implement the function lrucountmiss shall consist of an integer max_cache_size, an array pages and its length len and the function returns an integer indicating the number of cache misses m using the lru cache algorithm execution for the given input. Assume that the array pages always have pages numbered from 0 to 50. (a hit means the requested page is already existing in the cache and a miss means the requested page is not found in the cache). Input format: cache size s, the n pages being requested from the cache and the number of page requests n separated by a space.

6. Print palindrome from num to given number of digits when num is less than the max of given number of digits in the following pattern; otherwise, print "invalid."

Input:

3 2

[4,5,6,7,8,9,11,22,33,44,55,66,77,88,99]

7. Sandwich pattern Input:

5

Output:

1 *2 *3 *4 *5

11 *12 *13 *14 *15

21 *22 *23 *24 *25

16 *17 *18 *19 *20

6 *7 *8 *9 *10

Input: 6 Output:

1 *2 *3 *4 *5 *6

13 *14 *15 *16 *17 *18

25 *26 *27 *28 *29 *30

31 *32 *33 *34 *35 *36

19 *20 *21 *22 *23 *24

7 *8 *9 *10 *11 *12

8. TRAPEZIUM PATTERN INPUT:

4

OUTPUT:

1*2*3*4*17*18*19*20

5*6*7*14*15*16

8*9*12*13

10*11

9. Longest Increasing Subsequence using Longest Common Subsequence Algorithm
10. Number of sub arrays with negative product Array is {-1,2,-2} No. of negative products are 4.

Unit IV: LINKED LISTS

Linked lists, Queues, Stack, Graph and tree based problems

1. Merge two sorted lists into third list
2. Reverse linked list without using extra space
3. Sort the linked list without using extra space
4. Intersection of two linked lists
5. Swap pair wise nodes in Linked Lists
6. Count of triples (A, B, C) where $A*C$ is greater than $B*B$, where A, B and C are integers.'

Unit V: QUEUES STACKS GRAPH AND TREE BASED PROBLEMS

1. Implement Queue using Stack
2. Implement Stack using Queue
3. Design an Algorithm to find articulation point of a tree.
4. Convert a graph to a tree

SUGGESTED READINGS

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, And Clifford Stein, Introduction To Algorithms, First Or Second Edition, Mcgraw Hill.
2. Data Structures And Algorithmic Thinking With Python, By Narasimha Karumanchi, Careermonk Publications
3. Dynamic Programming For Coding Interviews. A Bottom-Up Approach To Problem Solving, By Meenakshi & Kamal Rawat, Notion Press
4. Guide To Competitive Programming: Learning And Improving Algorithms Through Contests (Undergraduate Topics In Computer Science) Springer
5. <https://cses.fi/book/book.pdf>. <https://www.comp.nus.edu.sg/~stevenha/myteaching/>
6. <https://www.javatpoint.com/programs-list> <https://practice.geeksforgeeks.org/>

25CSU721

CRYPTOGRAPHY AND CYBER SECURITY

Semester –VII
5H – 4C

Instruction Hours / week: L: 3 T: 0 P: 2

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours

Course Objectives

- Understand fundamental security concepts, classical encryption techniques, and modern cryptographic foundations to build a strong theoretical base in security.
- Analyze symmetric cipher algorithms, their mathematical foundations, and implementation methods to evaluate their application in secure communication systems.
- Examine asymmetric cryptography principles and algorithms to facilitate secure key exchange and encryption in public networks.
- Evaluate various authentication algorithms, digital signatures, and trust mechanisms to implement proper verification and validation systems.
- Investigate cyber-crimes, their classifications, and security countermeasures to develop comprehensive defensive strategies for information systems

Course Outcomes (COs)

At the completion of the course the student will be able to

Cos	Course Outcomes	Blooms Level
CO1	Explain the core principles of computer security and various types of security attacks and services.	Understand
CO2	Apply classical and modern cryptographic algorithms for secure communication.	Apply
CO3	Utilize number theory and algebraic structures in designing cryptographic systems.	Apply
CO4	Analyze and implement authentication protocols and digital signature schemes.	Analyze
CO5	Identify cyber threats and apply suitable security mechanisms for web, wireless, and cloud environments.	Evaluate

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	3	2	2	1	2	1	3	2	2	2	3	2
CO2	3	3	3	2	2	2	1	3	1	1	2	3	3
CO3	2	2	2	1	2	1	1	3	1	1	1	2	2
CO4	2	2	3	2	2	2	1	3	2	2	2	3	2
CO5	3	3	3	2	3	3	2	3	3	2	2	3	3

1 - low, 2 - medium, 3 - high

Unit I: INTRODUCTION TO SECURITY

Computer Security Concepts – The OSI Security Architecture – Security Attacks – Security Services and Mechanisms – A Model for Network Security – Classical encryption techniques: Substitution techniques, Transposition techniques, Steganography – Foundations of modern cryptography: Perfect security – Information Theory – Product Cryptosystem – Cryptanalysis.

Unit II: SYMMETRIC CIPHERS

Number theory – Algebraic Structures – Modular Arithmetic – Euclid’s algorithm – Congruence and matrices – Group, Rings, Fields, Finite Fields SYMMETRIC KEY CIPHERS: SDES – Block Ciphers – DES, Strength of DES – Differential and linear cryptanalysis – Block cipher design principles – Block cipher mode of operation – Evaluation criteria for AES – Pseudorandom Number Generators – RC4 – Key distribution.

Unit III: ASYMMETRIC CRYPTOGRAPHY

Primes – Primality Testing – Factorization – Euler’s totient function, Fermat’s and Euler’s Theorem – Chinese Remainder Theorem – Exponentiation and logarithm ASYMMETRIC KEY CIPHERS: RSA Cryptosystem – Key distribution – Key management – Diffie-Hellman key exchange – Elliptic curve arithmetic – Elliptic curve cryptography.

Unit IV AUTHENTICATION ALGORITHMS

Authentication requirement – Authentication function – MAC - HMAC, CMAC – Hash function – MD - SHA – Digital signature and authentication protocols – DSS – Schnorr Digital Signature Scheme – ElGamal cryptosystem – Entity Authentication: Biometrics, Passwords, Challenge Response protocols – Authentication applications – Kerberos MUTUAL TRUST: Key management and distribution – Symmetric key distribution using symmetric and asymmetric encryption – Distribution of public keys – X.509 Certificates

Unit V CYBER CRIMES AND CYBER SECURITY

Cyber Crime and Information Security – classifications of Cyber Crimes – Tools and Methods – Password Cracking, Keyloggers, Spywares, SQL Injection – Network Access Control – Internet Security – Web Security – Wireless Security - Cloud Security.

PRACTICAL EXERCISES

1. Implement classical encryption techniques: Caesar cipher, Playfair cipher, and Vigenère cipher.
2. Demonstrate transposition ciphers and steganography techniques for hiding information.
3. Simulate the working of a substitution-permutation network (SPN) using basic block cipher principles.
4. Design and implement the Simplified DES (SDES) algorithm.
5. Analyze the strength of DES through differential and linear cryptanalysis (simulation-based).
6. Evaluate AES encryption using different modes of operation (ECB, CBC, CFB, OFB).
7. Implement RC4 stream cipher and study its vulnerabilities.
8. Write a program to compute modular inverse using Extended Euclidean Algorithm.
9. Implement the RSA algorithm for encryption and decryption.
10. Perform key exchange using the Diffie-Hellman protocol.
11. Simulate HMAC generation using SHA-256 or SHA-1 and verify message integrity.
12. Investigate SQL injection, password cracking, and keylogger simulation in a controlled environment.

SUGGESTED READINGS

1. William Stallings, “Cryptography and Network Security – Principles and Practice”, Seventh Edition, Pearson Education, 2017.
2. Nina Godbole, Sunit Belapure, “Cyber Security: Understanding Cyber-crimes, Computer Forensics and Legal Perspectives”, First Edition, Wiley India, 2011.
3. Behrouz A. Ferouzan, Debdeep Mukhopadhyay, “Cryptography and Network Security”, 3rd Edition, Tata Mc Graw Hill, 2015.
4. Charles Pfleeger, Shari Pfleeger, Jonathan Margulies, “Security in Computing”, Fifth Edition, Prentice Hall, New Delhi, 2015.

Program Elective Courses (PE)

Programme Electives				
Sl. No.	Course Code	Course Title	Prerequisite Courses	Credits
Program Elective – I (V Semester)				
1	25CSU501A	Social Network Security	None	3
2	25CSU501B	Data Mining and Warehousing	DBMS	3
3	25CSU501C	2 D and 3 D Animation	Computer Graphics and Multimedia	3
4	25CSU501D	Principles of Programming Languages	Programming for Problem Solving, Data Structures	3
5	25CSU501E	Nano Science and Technology	Engineering Physics, Engineering Chemistry	3
Program Elective – II (V Semester)				
6	25CSU502A	Ethical Hacking	Computer Networks	3
7	25CSU502B	Software Defined Networks	Computer Networks, Operating Systems	3
8	25CSU502C	Immersive Technologies	Computer Graphics and Multimedia	3
9	25CSU502D	Software Engineering with Agile and Scrum	Object Oriented Programming, Data Structures	3
10	25CSU502E	Robotics	Engineering Mathematics, Basics of Engineering	3
Program Elective – III (VI Semester)				
11	25CSU601A	Digital Forensics	None	3
12	25CSU601B	Storage Technologies	Computer Architecture, Operating System	3
13	25CSU601C	Video and Audio Editing	Computer Graphics	3
14	25CSU601D	Automation and Testing	Object Oriented Software Engineering	3
15	25CSU601E	Big Data Analytics	DBMS, Data Structures	3

Program Elective – IV (VI Semester)				
16	25CSU602A	Intellectual Property Rights	None	3
17	25CSU602B	Privacy in Cloud	Cloud Computing	3
18	25CSU602C	UI and UX Design	None	3
19	25CSU602D	Application Development using Devops	DBMS	3
20	25CSU602E	Linux Technologies	None	3
Program Elective – V (VII Semester)				
21	25CSU701A	Enhanced Security for Cloud	Cloud Computing	3
22	25CSU701B	Edge and Fog Computing	Cloud Computing	3
23	25CSU701C	Game Development	None	3
24	25CSU701D	Full Stack Web Development using GenAI	Full stack development	3
25	25CSU701E	Digital Technologies for Marketing	None	3
Program Elective – VI (VII Semester)				
26	25CSU702A	Cryptocurrency and Blockchain Technologies	None	3
27	25CSU702B	Multi-Cloud Strategies	Cloud Computing	3
28	25CSU702C	AI for Game Programming	Artificial Intelligence	3
29	25CSU702D	Web Frameworks	Internet Programming	3
30	25CSU702E	Quantum Computing	Discrete Mathematics	3

Open Elective Courses (OE)

Open Electives				
Sl. No.	Course Code	Course Title	Prerequisite Courses	Credits
1	25CSU041	Programming in Core Java	None	3
2	25CSU042	Web Designing Technologies	None	3
3	25CSU043	Mobile Computing Technologies	None	3
4	25CSU044	E – Commerce	None	3
5	25CSU045	Applications of Immersive Technologies	None	3

25CSU501A

SOCIAL NETWORK SECURITY

Semester –V
3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives.**

- To develop semantic web related simple applications
- To explain Privacy and Security issues in Social Networking
- To explain the data extraction and mining of social networks
- To discuss the prediction of human behavior in social communities
- To describe the Access Control, Privacy and Security management of social networks.

Course Outcomes (COs)

At the completion of the course the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Develop semantic web related simple applications	Create
CO2	Address Privacy and Security issues in Social Networking	Understand
CO3	Explain the data extraction and mining of social networks	Understand
CO4	Discuss the prediction of human behavior in social communities	Analyze
CO5	Describe the applications of social networks	Understand

CO-PO Mapping

CO / PO	P01	P02	P03	P04	P05	P06	P07	P08	P09	P010	P011	PS01	PS02
CO1	3	3	3	3	3	2	2	2	2	1	2	2	2
CO2	2	3	2	3	3	2	2	2	2	1	2	2	2
CO3	3	2	3	2	2	2	2	2	2	1	2	2	2
CO4	2	3	3	3	3	2	2	2	2	1	2	2	2
CO5	3	3	3	3	3	2	2	2	2	1	2	2	2

1 - low, 2 - medium, 3 - high

Unit I FUNDAMENTALS OF SOCIAL NETWORKING

Introduction to Semantic Web, Limitations of current Web, Development of Semantic Web, Emergence of the Social Web, Social Network analysis, Development of Social Network Analysis, Key concepts and measures in network analysis, Historical overview of privacy and security, Major paradigms, for understanding privacy and security

Unit II SECURITY ISSUES IN SOCIAL NETWORKS

The evolution of privacy and security concerns with networked technologies, Contextual influences on privacy attitudes and behaviors, Anonymity in a networked world

Unit III EXTRACTION AND MINING IN SOCIAL NETWORK DATA

Extracting evolution of Web Community from a Series of Web Archive, Detecting communities in social networks, Definition of community, Evaluating communities, Methods for community detection and mining, Applications of community mining algorithms, Tools for detecting communities social network infrastructures and communities, Big data and Privacy

Unit IV PREDICTING HUMAN BEHAVIOR AND PRIVACY ISSUES

Understanding and predicting human behavior for social communities, User data Management, Inference and Distribution, Enabling new human experiences, Reality mining, Context, Awareness, Privacy in online social networks, Trust in online environment, What is Neo4j, Nodes, Relationships, Properties

Unit V ACCESS CONTROL, PRIVACY AND IDENTITY MANAGEMENT

Understand the access control requirements for Social Network, Enforcing Access Control Strategies, Authentication and Authorization, Roles-based Access Control, Host, storage and network access control options, Firewalls, Authentication, and Authorization in Social Network, Identity & Access Management, Single Sign-on, Identity Federation, Identity providers and service consumers, The role of Identity provisioning

SUGGESTED READINGS

1. Peter Mika, Social Networks and the Semantic Web, First Edition, Springer 2007.
2. Borko Furht, Handbook of Social Network Technologies and Application, First Edition, Springer, 2010.
3. Learning Neo4j 3.x Second Edition By Jérôme Baton, Rik Van Bruggen, Packt publishing
4. Easley D. Kleinberg J., Networks, Crowds, and Markets – Reasoning about a Highly Connected World, Cambridge University Press, 2010.
5. Jackson, Matthew O., Social and Economic Networks, Princeton University Press, 2008.
6. Guandong Xu, Yanchun Zhang and Lin Li, –Web Mining and Social Networking – Techniques and applications, First Edition, Springer, 2011.
7. Dion Goh and Schubert Foo, Social information Retrieval Systems: Emerging Technologies and Applications for Searching the Web Effectively, IGI Global Snippet, 2008.
8. Max Chevalier, Christine Julien and Chantal Soulé-Dupuy, Collaborative and Social Information Retrieval and Access: Techniques for Improved user Modeling, IGI Global Snippet, 2009

25CSU501B

DATA MINING AND WAREHOUSINGSemester –V
3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives.**

- Understand the architecture and components of data warehousing systems.
- Apply data preprocessing techniques to prepare data for mining.
- Learn association rule mining methods and their real-world applications.
- Explore classification algorithms and evaluate predictive models.
- Analyze clustering techniques and their use in various application domains.

Course Outcomes (COs)

At the completion of the course the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Explain data warehousing concepts and architecture.	Understand
CO2	Apply preprocessing techniques to clean and prepare data.	Apply
CO3	Use association rules to discover patterns in data.	Apply
CO4	Analyze classification models for prediction.	Analyze
CO5	Evaluate clustering methods for grouping data.	Evaluate

CO-PO Mapping

CO / PO	P01	P02	P03	P04	P05	P06	P07	P08	P09	P010	P011	PS01	PS02
CO1	3	3	3	3	3	2	2	2	2	2	2	3	3
CO2	3	3	3	3	3	2	2	2	2	2	2	3	3
CO3	3	3	3	3	3	2	2	2	2	2	2	3	3
CO4	3	3	3	3	3	2	2	2	2	2	2	3	3
CO5	3	3	3	3	3	2	2	2	2	2	2	3	3

1 - low, 2 - medium, 3 - high

Unit I DATA WAREHOUSING CONCEPTS

Introduction to Decision Support Systems - Data Warehouse Architecture - Data Marts, Metadata, Star and Snowflake Schemas - ETL Process - OLAP operations (Slice, Dice, Roll-up, Drill-down)

Unit II DATA PREPROCESSING

Data cleaning, Integration, Transformation, Normalization - Data reduction, Discretization, and Binning - Handling missing values and outliers - Feature selection and extraction

Unit III ASSOCIATION RULE MINING

Market Basket Analysis - Apriori Algorithm and FP-Growth - Support, Confidence, Lift - Constraints-based Association Mining - Applications in recommendation systems

Unit IV CLASSIFICATION

Supervised learning overview - Decision Trees (ID3, C4.5) - Bayesian classifiers - k-NN and Rule-based Classification - Model Evaluation (Confusion Matrix, Precision, Recall, ROC)

Unit V CLUSTERING AND APPLICATIONS

k-Means, Hierarchical Clustering, DBSCAN - Distance metrics and validity measures (Silhouette Score) - Outlier Detection - Web Mining and Text Mining - Applications: CRM, Fraud Detection, Bioinformatics

SUGGESTED READINGS

1. Jiawei Han, Micheline Kamber, Data Mining: Concepts and Techniques, 3rd Edition, Morgan Kaufmann, 2011.
2. Ralph Kimball, The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling, 3rd Edition, Wiley, 2013.
3. Pang-Ning Tan, Michael Steinbach, Introduction to Data Mining, 2nd Edition, Pearson, 2005.
4. David Hand, Heikki Mannila, Padhraic Smyth, Principles of Data Mining, 1st Edition, MIT Press, 2001.
5. Mohammed J. Zaki, Wagner Meira Jr., Data Mining and Analysis: Fundamental Concepts and Algorithms, 1st Edition, Cambridge University Press, 2014.

25CSU501C

2D AND 3D ANIMATION

Semester –V
3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives.**

- Introduce students to the fundamentals and principles of 2D and 3D animation.
- Provide hands-on experience in using industry-standard animation tools.
- Understand the workflows involved in character design, motion, rigging, and rendering.
- Explore the use of animation in multimedia, games, simulation, and virtual environments.
- Foster creativity and critical thinking in storytelling through animated content.

Course Outcomes (COs)

At the completion of the course the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Describe the fundamental principles and techniques of 2D and 3D animation.	Understand
CO2	Apply techniques of keyframing, tweening, and rigging in animation creation..	Apply
CO3	Design and develop animated characters and scenes using animation tools..	Create
CO4	Analyze the role of timing, motion, and visual storytelling in animations.	Analyze
CO5	Evaluate and optimize animations for performance and visual quality.	Evaluate

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
CO1	3	2	3	3	3	2	3	2	2	1	2	2	2
CO2	3	3	3	3	3	2	3	2	2	1	2	2	2
CO3	3	3	2	3	2	2	3	2	2	1	2	2	2
CO4	2	3	3	3	3	2	3	2	2	1	2	2	2
CO5	3	3	3	3	3	2	3	2	2	1	2	2	2

1 - low, 2 - medium, 3 - high

Unit I INTRODUCTION TO ANIMATION AND PRINCIPLES OF ANIMATION

History and evolution of animation, Types: Traditional, 2D, 3D, Stop-motion, Motion Graphics, 12 Principles of Animation (Disney principles), Frame rate, timing, and spacing, Storyboarding and script writing basics

Unit II 2D ANIMATION TECHNIQUES AND TOOLS

Introduction to 2D animation software (e.g., Adobe Animate, Pencil2D), Keyframes, in-betweening (tweening), onion skinning, Character sketching and rigging for 2D, Lip syncing and facial expressions, Walk cycles and looping animations

Unit III 3D ANIMATION BASICS

Introduction to 3D software (e.g., Blender, Maya), Modeling: primitives, extrusion, subdivision modelling, Texturing and materials, Lighting and camera setup, Rendering basics

Unit IV ADVANCED 3D ANIMATION

Rigging and skeletons, Keyframe and procedural animation, Inverse kinematics (IK) and forward kinematics (FK), Motion capture integration, Particle systems and dynamics

Unit V ANIMATION PRODUCTION PIPELINE AND APPLICATIONS

Animation workflow: pre-production, production, post-production, Compositing and editing, Sound synchronization, Applications in gaming, film, AR/VR, e-learning, Trends in real-time animation and AI-driven animation

SUGGESTED READINGS

1. Richard Williams, *The Animator's Survival Kit*, Faber & Faber, 2009.
2. Tony White, *Animation from Pencils to Pixels: Classical Techniques for the Digital Animator*, Focal Press, 2006.
3. John Vince, *Introduction to Computer Graphics and Creative 3D Environments*, Springer, 2011.
4. Isaac V. Kerlow, *The Art of 3D Computer Animation and Effects*, Wiley, 2009.
5. Steven Withrow and Glen Vilppu, *Illustrating for Animation: A Visual Guide*, Rotovision, 2005.
6. Chris Webster, *Animation: The Mechanics of Motion*, Focal Press, 2005.
7. Blender Foundation, *Blender Documentation*, <https://docs.blender.org>, 2024.
8. Adobe Systems Inc., *Adobe Animate Help & Tutorials*, <https://helpx.adobe.com/animate/tutorials.html>, 2024.
9. Khan Academy and Pixar Animation Studios, *Pixar in a Box*, <https://www.khanacademy.org/partner-content/pixar>, 2023.

25CSU501D

PRINCIPLES OF PROGRAMMING LANGUAGES

Semester –V
3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives.**

- To understand the fundamental concepts and design principles of programming languages.
- To compare different programming paradigms such as procedural, object-oriented, functional, and logic programming.
- To learn the syntax and semantics of programming languages through formal methods.
- To explore the structure and implementation of programming constructs like control flow, types, and procedures.
- To analyze and evaluate the design and execution of programs across multiple language paradigms.

Course Outcomes (COs)

At the completion of the course the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Explain the fundamental principles, syntax, and semantics of programming languages..	Understand
CO2	Analyze and compare various programming paradigms and their features	Analyze
CO3	Apply formal grammar and parsing techniques to define language syntax.	Apply
CO4	Evaluate language constructs like data types, control structures, and scoping rules..	Evaluate
CO5	Design and implement small programs using functional and logic programming..	Create

CO-PO Mapping

CO / PO	P01	P02	P03	P04	P05	P06	P07	P08	P09	P010	P011	PS01	PS02
CO1	2	3	3	3	2	2	3	2	2	2	3	3	2
CO2	2	2	2	2	3	2	3	2	2	2	3	2	3
CO3	3	3	3	3	3	2	3	2	2	2	3	2	3
CO4	3	3	3	3	2	2	3	2	2	2	3	2	3
CO5	3	3	3	3	3	2	3	2	2	2	3	2	2

1 - low, 2 - medium, 3 - high

Unit I LANGUAGE DESIGN AND SYNTAX

Language design goals, Evolution of programming languages, Language paradigms, Syntax and semantics, BNF and context-free grammar, Parse trees, Lexical analysis, Syntax analysis, Compilers and interpreters

Unit II VARIABLES, DATA TYPES, AND SCOPING

Names and bindings, Variable scope and lifetime, Data types, Type systems, Type checking, Type conversion, Abstract data types, Memory management, Garbage collection, Parameter passing mechanisms

Unit III CONTROL STRUCTURES AND SUBPROGRAMS

Control structures, Conditional statements, Loops and iteration, Exception handling, Subprograms, Functions and procedures, Parameter passing, Recursion, Activation records, Stack management

Unit IV PROGRAMMING PARADIGMS

Programming paradigms, Imperative programming, Object-oriented programming, Functional programming, Logic programming, Syntax and semantics in various paradigms, Execution models, C/C++, Java, Scheme, Haskell, Prolog

Unit V ADVANCED TOPICS IN LANGUAGE DESIGN

Concurrency and parallelism, Event-driven programming, Scripting languages, Language interoperability, Domain-specific languages, Language security, Virtual machines, JIT compilation, Language evolution trends

SUGGESTED READINGS

1. Robert W. Sebesta, Concepts of Programming Languages, Pearson Education, 12th Edition, 2022.
2. Kenneth C. Louden and Kenneth A. Lambert, Programming Languages: Principles and Practice, Cengage Learning, 3rd Edition, 2011.
3. Ravi Sethi, Programming Languages: Concepts and Constructs, Pearson Education, 2nd Edition, 1996.
4. Terrence W. Pratt, Marvin V. Zelkowitz, and T.V.G. Kumar, Programming Languages: Design and Implementation, Pearson Education, 4th Edition, 2000.
5. Allen B. Tucker and Robert E. Noonan, Programming Languages: Principles and Paradigms, Tata McGraw-Hill, 2nd Edition, 2006.
6. Harold Abelson and Gerald Jay Sussman, Structure and Interpretation of Computer Programs, MIT Press, 2nd Edition, 1996.
7. John Mitchell, Concepts in Programming Languages, Cambridge University Press, 2002.
8. Peter Van Roy and Seif Haridi, Concepts, Techniques, and Models of Computer Programming, MIT Press, 2004.

25CSU501E

NANOSCIENCE AND TECHNOLOGY

Semester –V
3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100

End Semester Exam: 3 Hours

Course Objectives.

- To introduce the basic concepts and principles of nanoscience and nanotechnology.
- To understand the properties of materials at the nanoscale and how they differ from bulk materials.
- To learn various synthesis and fabrication techniques used in nanotechnology.
- To explore characterization tools and techniques used for nanomaterials.
- To study the applications of nanomaterials in electronics, energy, healthcare, and environment.

Course Outcomes (COs)

At the completion of the course the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Describe the fundamental concepts and scope of nanoscience and nanotechnology.	Understand
CO2	Analyze the size-dependent properties of nanomaterials and their significance.	Analyze
CO3	Explain the methods of synthesis and fabrication of nanomaterials.	Understand
CO4	Apply characterization techniques to study nanoscale materials.	Apply
CO5	Evaluate the applications and challenges of nanotechnology in various domains.	Evaluate

CO-PO Mapping

CO / PO	P01	P02	P03	P04	P05	P06	P07	P08	P09	P010	P011	PS01	PS02
CO1	3	2	3	3	3	2	2	1	1	1	2	2	3
CO2	2	3	3	3	2	2	2	1	1	1	2	2	3
CO3	3	3	2	3	3	2	2	1	1	1	2	2	3
CO4	3	3	3	2	3	2	2	1	1	1	2	2	3
CO5	3	3	3	3	3	2	2	1	1	1	2	2	3

1 - low, 2 - medium, 3 - high

Unit I INTRODUCTION TO NANOSCIENCE AND NANOTECHNOLOGY

Definition of nanoscience and nanotechnology, Historical development, Scope and interdisciplinary nature, Nanoscale vs bulk scale, Quantum confinement, Surface-to-volume ratio, Top-down and bottom-up approaches, Overview of current research trends

Unit II PROPERTIES OF NANOMATERIALS

Mechanical, optical, electrical, magnetic, and thermal properties at the nanoscale, Size-dependent effects, Quantum size effects, Band gap engineering, Surface plasmon resonance, Superparamagnetism, Carbon-based nanomaterials: Fullerenes, carbon nanotubes (CNTs), and graphene

Unit III SYNTHESIS AND FABRICATION TECHNIQUES

Physical methods: Ball milling, sputtering, laser ablation, evaporation, Chemical methods: Sol-gel, chemical vapor deposition (CVD), hydrothermal synthesis, microemulsion, Biological methods: Green synthesis using microbes and plants, Lithography: Photolithography, electron beam lithography, nanoimprint lithography

Unit IV CHARACTERIZATION OF NANOMATERIALS

Microscopy techniques: Scanning electron microscopy (SEM), transmission electron microscopy (TEM), atomic force microscopy (AFM), scanning tunneling microscopy (STM), Spectroscopy techniques: X-ray diffraction (XRD), Fourier transform infrared spectroscopy (FTIR), Raman spectroscopy, UV-Visible spectroscopy, Energy dispersive X-ray spectroscopy (EDX)

Unit V APPLICATIONS AND FUTURE TRENDS

Nanomaterials in electronics: Nanosensors, nanoelectronics, memory devices, Energy: Solar cells, hydrogen storage, fuel cells, Healthcare: Drug delivery, bioimaging, diagnostics, tissue engineering, Environment: Water purification, pollution control, Ethical, social, and regulatory issues in nanotechnology

SUGGESTED READINGS

1. Charles P. Poole Jr. and Frank J. Owens, Introduction to Nanotechnology, Wiley-Interscience, 2003.
2. T. Pradeep, A Textbook of Nanoscience and Nanotechnology, Tata McGraw-Hill Education, 2012.
3. M.S. Ramachandra Rao and Shubra Singh, Nanoscience and Nanotechnology: Fundamentals to Frontiers, Wiley India, 2015.
4. C.N.R. Rao, A. Müller, and A.K. Cheetham (Eds.), The Chemistry of Nanomaterials: Synthesis, Properties and Applications, Wiley-VCH, 2004.
5. Gabor L. Hornyak, Joydeep Dutta, Harry F. Tibbals, and Anil Rao, Introduction to Nanoscience, CRC Press, 2008.
6. B.S. Murty, P. Shankar, Baldev Raj, B.B. Rath, and James Murday, Textbook of Nanoscience and Nanotechnology, Springer, 2013.
7. K. Eric Drexler, Engines of Creation: The Coming Era of Nanotechnology, Anchor Books, 1987.
8. Mark Ratner and Daniel Ratner, Nanotechnology: A Gentle Introduction to the Next Big Idea, Pearson Education, 2003.

25CSU502A

ETHICAL HACKING

Semester –V
3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours

Course Objectives.

- To understand the role and responsibilities of ethical hackers and penetration testers.
- To study various hacking techniques used by attackers and how to counter them.
- To explore the methodologies and tools used in ethical hacking and vulnerability assessment.
- To gain hands-on experience with reconnaissance, scanning, and exploitation techniques.
- To understand legal, ethical, and professional issues in cybersecurity.

Course Outcomes (COs)

At the completion of the course the student will be able to

COs	Course Outcomes	Blooms Level
C01	Describe the concepts, tools, and techniques used in ethical hacking..	Understand
C02	Perform reconnaissance and vulnerability scanning using appropriate tools.	Apply
C03	Analyze and simulate various network and system attacks...	Analyze
C04	Demonstrate penetration testing methods to assess system security.	Understand
C05	Evaluate ethical, legal, and social issues in cybersecurity practices.	Evaluate

CO-PO Mapping

CO / PO	P01	P02	P03	P04	P05	P06	P07	P08	P09	P010	P011	PS01	PS02
C01	3	3	3	3	3	3	3	1	1	2	3	2	3
C02	2	3	2	2	3	3	3	1	1	2	3	2	3
C03	2	2	3	3	2	3	3	1	1	2	3	2	3
C04	3	3	3	3	3	3	3	1	1	2	3	2	3
C05	3	3	3	3	3	3	3	1	1	2	3	2	3

1 - low, 2 - medium, 3 - high

Unit I INTRODUCTION TO ETHICAL HACKING

Overview of cybersecurity and hacking, Types of hackers, Roles of ethical hackers, Penetration testing phases, Information security laws and regulations, Overview of TCP/IP model, IP addressing, OSI layers, Network devices and protocols

Unit II RECONNAISSANCE AND FOOTPRINTING

Footprinting techniques, Passive and active reconnaissance, Whois, NSlookup, Social engineering, Google hacking, DNS enumeration, Email and social media footprinting, Tools: Maltego, Recon-ng, theHarvester

Unit III SCANNING AND VULNERABILITY ANALYSIS

Network scanning: Nmap, Hping, Netcat, Banner grabbing, Port scanning, Service enumeration, Vulnerability assessment tools: Nessus, OpenVAS, Nikto, Identifying vulnerabilities in web servers and systems, Exploit databases

Unit IV SYSTEM HACKING AND EXPLOITATION

Password cracking techniques: Dictionary, brute force, rainbow tables, Keyloggers, Buffer overflow attacks, Privilege escalation, Rootkits and backdoors, Sniffing and spoofing attacks, Session hijacking, Exploitation frameworks: Metasploit

Unit V WEB APPLICATION AND WIRELESS SECURITY

Web application vulnerabilities: SQL injection, XSS, CSRF, File inclusion, Command injection, OWASP Top 10, Wireless attacks: WEP/WPA cracking, Rogue access points, Wireless sniffing tools: Aircrack-ng, Kismet, Best practices for securing applications and networks

SUGGESTED READINGS

1. Michael T. Simpson, Kent Backman, James Corley, Hands-On Ethical Hacking and Network Defense, Cengage Learning, 3rd Edition, 2016.
2. Kimberly Graves, CEH: Certified Ethical Hacker Study Guide, Wiley Publishing, 1st Edition, 2010.
3. Patrick Engebretson, The Basics of Hacking and Penetration Testing, Syngress (Elsevier), 2nd Edition, 2013.
4. Sean-Philip Oriyano, Ethical Hacking and Penetration Testing Guide, Syngress, 2014.
5. Jon Erickson, Hacking: The Art of Exploitation, No Starch Press, 2nd Edition, 2008.
6. Georgia Weidman, Penetration Testing: A Hands-On Introduction to Hacking, No Starch Press, 2014.
7. Kevin Mitnick and William L. Simon, The Art of Invisibility, Little, Brown and Company, 2017.

25CSU502B

SOFTWARE DEFINED NETWORKSSemester –V
3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives.**

- Introduce the fundamental principles and architecture of Software Defined Networking.
- Explain the separation of control and data planes and its significance.
- Familiarize students with SDN protocols such as OpenFlow and their real-world use.
- Analyze network virtualization, SDN controllers, and southbound/northbound APIs.
- Apply SDN concepts in data center, cloud, and network management scenarios.

Course Outcomes (COs)

At the completion of the course the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Explain the SDN architecture, key concepts, and the evolution of SDN.	Understand
CO2	Differentiate between the traditional and SDN-based network models.	Analyze
CO3	Demonstrate knowledge of SDN protocols like OpenFlow and southbound APIs.	Understand
CO4	Evaluate SDN controllers and their role in network programmability.	Evaluate
CO5	Design and implement SDN solutions for virtual networks and data centers.	Create

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
CO1	3	2	3	3	3	2	1	2	2	2	2	2	3
CO2	2	2	3	3	3	2	1	2	2	2	2	2	3
CO3	3	3	2	3	3	2	1	2	2	2	2	2	3
CO4	3	3	3	3	2	2	1	3	2	2	2	2	3
CO5	3	3	3	3	3	2	1	3	3	3	2	3	3

1 - low, 2 - medium, 3 - high

Unit I INTRODUCTION TO SDN

Traditional networks and limitations-Emergence of SDN-SDN architecture: Planes of separation-Benefits and challenges of SDN-Control plane vs data plane

Unit II SDN ARCHITECTURE AND LAYERS

Southbound APIs: OpenFlow protocol-SDN controllers: centralized vs distributed-Northbound APIs and programming network behavior-Flow tables and switches-SDN data plane functions

Unit III SDN CONTROLLERS AND TOOLS

Popular SDN controllers: Floodlight, ONOS, OpenDaylight-Interoperability and controller placement-Control layer scalability and resilience-Mininet-based SDN simulation-Open vSwitch (OVS)

Unit IV NETWORK VIRTUALIZATION AND PROGRAMMABILITY

Network function virtualization (NFV)-Virtualized network functions (VNFs)-Programming languages for SDN (e.g., P4)-Virtualization techniques: overlays, tunnels-SDN in multi-tenant data centers

Unit V SDN APPLICATIONS AND USE CASES

Traffic engineering and QoS-SDN in cloud computing and data centers-Security in SDN environments-Case studies: Google B4, Facebook SDN, AT&T WAN-Emerging trends: Intent-based networking, AI in SDN

SUGGESTED READINGS

1. Paul Goransson, Chuck Black, Timothy Culver – Software Defined Networks: A Comprehensive Approach, 2nd Edition, Morgan Kaufmann, 2016.
2. Thomas D. Nadeau & Ken Gray – SDN: Software Defined Networks, O'Reilly Media, 2013.
3. Feamster, Rexford & Zegura – The Road to SDN: An Intellectual History of Programmable Networks – ACM SIGCOMM.
4. Open Networking Foundation (ONF) White Papers and Technical Reports.
5. P4 Language Consortium – <https://p4.org> (for network programming insights).

25CSU502C

IMMERSIVE TECHNOLOGIES

Semester –V
3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives.**

- To introduce the fundamentals of immersive technologies such as Virtual Reality (VR), Augmented Reality (AR), and Mixed Reality (MR).
- To explore the hardware and software components involved in immersive systems.
- To understand design principles and interaction models used in immersive environments.
- To provide hands-on exposure to tools and platforms for developing immersive applications.
- To study the applications and future trends of immersive technologies in different domains..

Course Outcomes (COs)

At the completion of the course the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Describe the concepts, evolution, and components of immersive technologies.	Understand
CO2	Analyze the interaction and display technologies used in VR/AR/MR systems.	Analyze
CO3	Apply development tools to create simple immersive experiences.	Apply
CO4	Evaluate the design principles and user experience challenges in immersive systems..	Evaluate
CO5	Identify real-world applications and future directions of immersive technologies.	Analyze

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	2	2	2	2	2	2	1	2	2	2	2
CO2	3	3	2	2	3	2	2	2	2	2	2	2	3
CO3	3	2	2	3	3	2	2	2	2	2	2	2	2
CO4	3	3	3	2	3	2	2	3	2	2	2	2	3
CO5	3	3	3	3	3	2	2	3	3	3	2	3	3

1 - low, 2 - medium, 3 - high

Unit I INTRODUCTION TO IMMERSIVE TECHNOLOGIES

Definition and scope of immersive technologies, History and evolution of VR, AR, MR, and XR, Differences and overlaps, Key characteristics of immersive systems, Overview of immersive applications in gaming, education, healthcare, and industry

Unit II HARDWARE AND SOFTWARE COMPONENTS

Head-mounted displays (HMDs), Motion tracking systems, Haptic devices, Input controllers, 3D audio systems, Immersive display technologies (CAVE, projection), VR-ready PCs and mobile setups, SDKs and engines: Unity, Unreal Engine, ARKit, ARCore, WebXR

Unit III DESIGN AND INTERACTION IN IMMERSIVE ENVIRONMENTS

Human-computer interaction (HCI) in 3D space, 3D user interfaces, Gesture and voice recognition, Spatial mapping, Presence and immersion, UI/UX design challenges in immersive tech, Safety, ergonomics, and motion sickness issues

Unit IV CONTENT DEVELOPMENT FOR VR/AR

3D modeling basics, Scene construction and lighting, Physics and animation in immersive scenes, Scripting for interactivity, Real-time rendering techniques, Deploying VR/AR apps for mobile and desktop, Prototyping with tools like Unity and Blender

Unit V APPLICATIONS AND FUTURE TRENDS

Use cases: Gaming, training and simulation, education, tourism, medical and therapy, Industrial design, Metaverse and digital twins, Ethical and privacy concerns, Future trends: brain-computer interfaces, AI in XR, volumetric video, multisensory XR

SUGGESTED READINGS

1. Alan B. Craig, Understanding Augmented Reality: Concepts and Applications, Morgan Kaufmann, 2013.
2. Jason Jerald, The VR Book: Human-Centered Design for Virtual Reality, Morgan & Claypool, 2015.
3. Tony Parisi, Learning Virtual Reality: Developing Immersive Experiences and Applications for Desktop, Web, and Mobile, O'Reilly Media, 2015.
4. Grigore C. Burdea and Philippe Coiffet, Virtual Reality Technology, Wiley-IEEE Press, 2nd Edition, 2003.
5. Dieter Schmalstieg and Tobias Hollerer, Augmented Reality: Principles and Practice, Addison-Wesley, 2016.
6. Joseph H. Rouse, Immersive Technologies in Education: Breakthroughs in Research and Practice, IGI Global, 2019.
7. Unity Technologies, Unity Documentation and Tutorials, <https://docs.unity3d.com>
8. Khronos Group, WebXR Device API, <https://www.khronos.org/webxr/>

25CSU502D SOFTWARE ENGINEERING WITH AGILE AND SCRUM

Semester –V
3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours

Course Objectives

- To understand the fundamental principles of software engineering and development life cycles.
- To introduce Agile methodology and Scrum framework as modern approaches to software development.
- To analyze the roles, artifacts, and ceremonies involved in Scrum.
- To apply Agile practices such as iterative development, continuous integration, and user stories
- To evaluate project management and software quality practices in Agile environments, domains..

Course Outcomes (COs)

At the completion of the course the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Explain software engineering principles, process models, and Agile methodology..	Understand
CO2	Analyze and compare traditional and Agile software development life cycles..	Analyze
CO3	Apply Scrum framework to plan and manage software projects.	Apply
CO4	Develop software using Agile practices such as sprints, stand-ups, and user stories..	Create
CO5	Evaluate the effectiveness of Agile tools and practices for quality and delivery..	Evaluate

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	2	2	2	2	2	2	2	2	2	2	3
CO2	3	3	2	2	3	2	2	2	2	2	2	2	2
CO3	3	3	2	3	3	2	2	2	2	2	2	2	3
CO4	3	3	3	3	3	2	2	3	2	2	2	2	2
CO5	3	3	3	3	3	2	2	3	3	3	2	3	3

1 - low, 2 - medium, 3 - high

Unit I FUNDAMENTALS OF SOFTWARE ENGINEERING

Software engineering definition, Software process models: Waterfall, V-model, Incremental, Spiral, Prototyping, Software requirements specification, Functional vs non-functional requirements, Software project planning and cost estimation, Risk management

Unit II INTRODUCTION TO AGILE METHODOLOGY

Agile manifesto and principles, Agile vs traditional approaches, Agile process models: XP, Crystal, Kanban, Lean, Benefits and limitations of Agile, Agile metrics, Agile roles and responsibilities

Unit III SCRUM FRAMEWORK

Scrum overview, Scrum roles: Product Owner, Scrum Master, Development Team, Scrum artifacts: Product backlog, Sprint backlog, Increment, Definition of done, Scrum events: Sprint planning, Daily Scrum, Sprint review, Sprint retrospective, Velocity tracking, Burndown charts

Unit IV AGILE PRACTICES AND ENGINEERING TECHNIQUES

User stories and story points, Acceptance criteria, Estimation techniques (Planning Poker), Continuous integration and deployment (CI/CD), Test-driven development (TDD), Pair programming, Refactoring, Version control with Git, Agile testing strategies

Unit V AGILE PROJECT MANAGEMENT AND TOOLS

Agile project lifecycle, Agile documentation, Managing changing requirements, Tracking and reporting progress, Tools for Agile: Jira, Trello, GitHub Projects, Case studies of Agile adoption in industry, Challenges and best practices in scaling Agile

SUGGESTED READINGS

1. Ian Sommerville, Software Engineering, Pearson Education, 10th Edition, 2016.
2. Robert C. Martin, Agile Software Development: Principles, Patterns, and Practices, Prentice Hall, 2002.
3. Ken Schwaber and Jeff Sutherland, The Scrum Guide, Scrum.org, 2020.
4. Mike Cohn, User Stories Applied: For Agile Software Development, Addison-Wesley, 2004.
5. Craig Larman and Bas Vodde, Practices for Scaling Lean and Agile Development, Addison-Wesley, 2010.
6. Lisa Crispin and Janet Gregory, Agile Testing: A Practical Guide for Testers and Agile Teams, Addison-Wesley, 2009.
7. Henrik Kniberg, Scrum and XP from the Trenches, InfoQ, 2nd Edition, 2015.
8. Atlassian Team, Jira Software Documentation, <https://www.atlassian.com/software/jira/guides>

25CSU502E

ROBOTICS

Semester –V
3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- To understand the basic concepts and components of robotic systems.
- To study the kinematics, dynamics, and control of robotic manipulators.
- To introduce sensors, actuators, and microcontrollers used in robotics.
- To develop algorithms for robotic motion planning and navigation.
- To explore applications of robotics in various engineering domains.

Course Outcomes (COs)

At the completion of the course the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Explain the fundamentals of robotic systems and their classifications.	Understand
CO2	Apply kinematic equations for position and orientation analysis of manipulators.	Apply
CO3	Analyze robotic dynamics and implement control strategies.	Analyze
CO4	Design and simulate robotic movements and sensor-based interactions.	Create
CO5	Evaluate the applications and ethical considerations in robotics.	Evaluate

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	2	2	2	2	2	2	1	2	2	1	3
CO2	3	3	2	2	3	1	1	2	2	2	1	2	3
CO3	3	3	2	3	2	2	2	2	2	2	2	2	3
CO4	3	3	3	3	3	2	1	3	2	2	2	2	3
CO5	3	3	3	3	3	2	2	3	3	3	2	3	3

1 - low, 2 - medium, 3 - high

Unit I INTRODUCTION TO ROBOTICS

Definition and scope of robotics, Types of robots: industrial, mobile, service, medical, military, Components of robotic systems: sensors, actuators, controllers, Overview of robot anatomy, Degrees of freedom (DOF), Workspace, Robot configurations and classifications

Unit II KINEMATICS OF ROBOTS

Coordinate frames and transformations, Homogeneous transformation matrices, Forward kinematics of serial manipulators, Denavit-Hartenberg (D-H) parameters, Inverse kinematics: analytical and numerical methods, Kinematic redundancy and constraints

Unit III DYNAMICS AND CONTROL

Euler-Lagrange formulation, Newton-Euler dynamics, Joint space and operational space control, PD and PID controllers, Trajectory planning, Force control, Introduction to adaptive and nonlinear control in robotics

Unit IV SENSORS, ACTUATORS, AND EMBEDDED SYSTEMS

Types of sensors: position, velocity, proximity, vision, tactile, Actuators: DC/AC motors, servo motors, stepper motors, Pneumatic and hydraulic actuators, Interfacing sensors and actuators with microcontrollers (Arduino/Raspberry Pi), Basics of real-time control

Unit V APPLICATIONS AND FUTURE TRENDS

Robotics in manufacturing, healthcare, agriculture, exploration, Robot navigation and mapping, Simultaneous localization and mapping (SLAM), AI in robotics, Human-robot interaction, Ethical issues and safety in robotics, Trends: swarm robotics, soft robotics, humanoids

SUGGESTED READINGS

1. Mikell P. Groover, Industrial Robotics: Technology, Programming, and Applications, McGraw-Hill, 2008.
2. Saeed B. Niku, Introduction to Robotics: Analysis, Control, Applications, Wiley, 3rd Edition, 2020.
3. John J. Craig, Introduction to Robotics: Mechanics and Control, Pearson, 4th Edition, 2017. Bruno Siciliano and Lorenzo Sciavicco, Robotics: Modelling, Planning and Control, Springer, 2009.
4. Roland Siegwart, Illah R. Nourbakhsh, and Davide Scaramuzza, Introduction to Autonomous Mobile Robots, MIT Press, 2nd Edition, 2011.
5. Richard D. Klafter, Thomas A. Chmielewski, and Michael Negin, Robotic Engineering: An Integrated Approach, Prentice Hall, 2003.
6. Peter Corke, Robotics, Vision and Control: Fundamental Algorithms in MATLAB, Springer, 2011.
7. Arduino Project Hub, Arduino Robotics Projects, <https://create.arduino.cc/projecthub>

25CSU601A

DIGITAL FORENSICS

Semester –VI
3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100

End Semester Exam: 3 Hours

Course Objectives

- To understand the fundamentals of digital forensics and the investigative process.
- To learn various methods of data acquisition, preservation, and analysis from digital devices.
- To explore forensic tools and techniques used in real-world investigations.
- To understand the legal and ethical considerations in handling digital evidence.
- To gain practical skills in examining computer systems, mobile devices, and networks for forensic evidence.

Course Outcomes (COs)

At the completion of the course the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Describe the digital forensic process and its application in cybercrime investigation.	Understand
CO2	Apply forensic techniques to acquire and analyze digital evidence from various media.	Apply
CO3	Analyze forensic data from file systems, emails, and networks to identify traces.	Analyze
CO4	Use appropriate tools and methods for mobile, cloud, and memory forensics.	Apply
CO5	Evaluate the legal and ethical issues involved in digital forensics.	Evaluate

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	2	2	2	2	3	2	2	2	2	2	3
CO2	3	3	2	2	3	1	3	2	2	2	2	2	3
CO3	3	3	2	3	3	1	3	2	2	2	2	2	3
CO4	3	3	3	3	3	2	3	3	2	2	2	2	3
CO5	3	3	3	3	3	2	2	3	3	3	2	3	3

1 - low, 2 - medium, 3 - high

Unit I FUNDAMENTALS OF DIGITAL FORENSICS

Introduction to digital forensics, Cybercrime and cyber laws, Phases of forensic investigation: identification, acquisition, preservation, analysis, and reporting, Digital evidence: types, collection, admissibility in court, Chain of custody, Anti-forensics techniques

Unit II DATA ACQUISITION AND FORENSIC IMAGING

Storage devices and file systems (FAT, NTFS, EXT), Data acquisition methods: static and live acquisition, Bit-stream imaging, Write blockers, Data recovery techniques, Hashing and verification, Use of tools like FTK Imager, Autopsy

Unit III OPERATING SYSTEM AND NETWORK FORENSICS

Windows forensics: registry, log files, user activity, recycle bin, prefetch Linux forensics: logs, shell history, configuration files Network forensics: sniffing tools, packet analysis, logs from routers and firewalls, Email forensics: headers, attachments, spam traces, Wireshark, TCPDump, Xplico

Unit IV MOBILE AND MEMORY FORENSICS

Mobile OS architecture (Android, iOS), SIM and SD card forensics, Deleted data recovery, Mobile forensic tools (Cellebrite, MOBILedit), Memory acquisition tools, Volatile memory analysis, RAM dump, Malware forensics, Reverse engineering basics

Unit V LEGAL, ETHICAL, AND ADVANCED TOPICS

Laws and standards: IT Act 2000, HIPAA, GDPR, ISO 27037, Ethics in digital forensics, Expert witness testimony, Cloud forensics: data collection challenges, Chain of custody in distributed environments, Emerging trends: IoT forensics, Blockchain forensics, AI in forensics

SUGGESTED READINGS

1. Nelson, Phillips, and Steuart, Guide to Computer Forensics and Investigations, Cengage Learning, 6th Edition, 2018.
2. Marjie T. Britz, Computer Forensics and Cyber Crime: An Introduction, Pearson, 3rd Edition, 2013.
3. John Sammons, The Basics of Digital Forensics, Syngress (Elsevier), 2nd Edition, 2014.
4. Bill Nelson, Amelia Phillips, Christopher Steuart, Computer Forensics and Digital Investigation with EnCase Forensic v7, Cengage Learning, 2014.
5. Eoghan Casey, Digital Evidence and Computer Crime: Forensic Science, Computers, and the Internet, Academic Press, 3rd Edition, 2011.
6. Brian Carrier, File System Forensic Analysis, Addison-Wesley, 2005.
7. Harlan Carvey, Windows Forensic Analysis Toolkit, Syngress, 4th Edition, 2014.
8. NIST, Guide to Integrating Forensic Techniques into Incident Response, NIST SP 800-86

25CSU601B

STORAGE TECHNOLOGIES

Semester –VI
3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- To understand the fundamentals of data storage systems and their architectures.
- To study various storage technologies including DAS, NAS, SAN, and cloud-based storage.
- To explore data backup, recovery, and replication strategies.
- To examine storage virtualization and storage security mechanisms.
- To analyze performance, scalability, and management aspects of enterprise storage solutions.

Course Outcomes (COs)

At the completion of the course the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Explain the architecture, components, and types of storage systems.	Understand
CO2	Analyze and differentiate between DAS, NAS, SAN, and cloud storage models..	Analyze
CO3	Apply backup, recovery, and replication techniques in storage environments.	Apply
CO4	Evaluate storage virtualization and security practices	Evaluate
CO5	Design and recommend appropriate storage solutions for various business needs.	Evaluate

CO-PO Mapping

CO / PO	P01	P02	P03	P04	P05	P06	P07	P08	P09	P010	P011	PS01	PS02
CO1	3	2	2	2	2	1	2	2	2	2	1	1	3
CO2	3	3	2	2	3	1	1	2	2	2	1	2	3
CO3	3	3	2	3	3	1	1	2	2	2	2	2	3
CO4	3	3	3	3	3	2	1	3	2	2	2	2	3
CO5	3	3	3	3	3	2	2	3	3	3	2	3	3

1 - low, 2 - medium, 3 - high

Unit I INTRODUCTION TO STORAGE SYSTEMS

Overview of storage needs, Data storage hierarchy, Evolution of storage technology, Components of a storage system: controllers, disks, RAID levels and architectures, Performance metrics: throughput, IOPS, latency, reliability, availability, serviceability (RAS)

Unit II DIRECT ATTACHED STORAGE (DAS), NAS AND SAN

DAS architecture, benefits and limitations, Network Attached Storage (NAS): architecture, file sharing protocols (NFS, SMB/CIFS), Storage Area Network (SAN): Fibre Channel, iSCSI, FCoE, SAN topologies and zoning, Comparison of DAS, NAS, and SAN

Unit III BACKUP, RECOVERY, AND DATA PROTECTION

Backup types: full, incremental, differential, Backup strategies and scheduling, Backup to disk and tape, Snapshot and replication techniques, Recovery objectives (RTO, RPO), Archiving solutions, Deduplication, Business continuity and disaster recovery

Unit IV STORAGE VIRTUALIZATION AND CLOUD STORAGE

Storage virtualization concepts, Block vs file virtualization, Virtual Storage Appliances (VSAs), Thin provisioning, Storage tiering, Software-defined storage (SDS), Cloud storage models: public, private, hybrid, Cloud providers (AWS S3, Azure Blob), Object-based storage

Unit V STORAGE SECURITY AND MANAGEMENT

Storage security threats and countermeasures, Data encryption at rest and in transit, Key management, Role-based access control (RBAC), Storage management tools (e.g., EMC Unisphere, NetApp OnCommand), Storage performance tuning, SLA and capacity planning

SUGGESTED READINGS

1. EMC Corporation, Information Storage and Management: Storing, Managing, and Protecting Digital Information, Wiley India, 2nd Edition, 2012.
2. Robert Spalding, Storage Networks: The Complete Reference, Tata McGraw-Hill, 2003.
3. Ulf Troppens, Rainer Erkens, Wolfgang Müller, Storage Networks Explained: Basics and Application of Fibre Channel SAN, NAS, iSCSI, InfiniBand and FCoE, Wiley, 2nd Edition, 2009
4. Marc Farley, Building Storage Networks, McGraw-Hill, 2nd Edition, 2000.
5. Tom Clark, Designing Storage Area Networks, Pearson Education, 2nd Edition, 2003
6. John Chirillo and Scott Blaul, Storage Area Networks, Wiley, 2001.
7. Hitachi Data Systems, Data Protection and Disaster Recovery Strategies, Tech Whitepapers, 2021.

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- To understand the fundamentals of video and audio formats, standards, and technologies.
- To learn editing techniques using professional software for post-production.
- To explore video and audio enhancement, effects, transitions, and rendering techniques.
- To apply storytelling principles to audio-visual content.
- To gain hands-on experience with editing tools for real-world media production..

Course Outcomes (COs)

At the completion of the course the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Identify and explain various video and audio formats, codecs, and editing principles.	Understand
CO2	Apply video and audio editing techniques using industry-standard tools.	Apply
CO3	Create professional-quality media with effects, transitions, and synchronization.	Create
CO4	Analyze and enhance media quality for different platforms and audiences.	Analyze
CO5	Design short projects by combining editing, storytelling, and post-production skills.	Create

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	1	1	2	1	3	2	2	1	2	1	3
CO2	3	3	2	2	3	1	2	2	2	2	2	2	3
CO3	3	3	2	3	3	1	3	2	2	1	2	2	3
CO4	3	3	3	3	3	2	3	3	2	2	2	2	3
CO5	3	3	3	3	3	2	2	3	3	3	2	3	3

1 - low, 2 - medium, 3 - high

Unit I INTRODUCTION TO VIDEO AND AUDIO EDITING

Basics of multimedia systems, Video formats (MP4, MOV, AVI), Audio formats (WAV, MP3, AAC), Frame rate, resolution, bit rate, Aspect ratios, Introduction to timelines and keyframes, Storyboarding and planning a media project

Unit II TOOLS AND INTERFACES

Overview of editing software: Adobe Premiere Pro, Final Cut Pro, DaVinci Resolve, Audacity, Adobe Audition User interface: timeline, media bins, track layers, preview monitor, Toolsets for cutting, trimming, moving, zooming, Multitrack editing concepts

Unit III VIDEO EDITING TECHNIQUES

Non-linear editing, Timeline and clip management, Transitions and fades, Color correction and grading, Motion graphics basics, Titles and text effects, Speed ramping, green screen (chroma keying), Exporting and compression settings

Unit IV AUDIO EDITING TECHNIQUES

Noise reduction, Equalization (EQ), Audio transitions and crossfades, Synchronizing audio with video, Voice-over editing, Mixing background scores and effects, Normalization and mastering, Creating soundscapes

Unit V MEDIA PROJECTS AND POST-PRODUCTION

Combining audio and video tracks, Editing for social media, film, advertising, and documentaries, Rendering settings and formats, Subtitling and captioning, Ethical use of media, Copyright considerations, Final project: edit a 1–2-minute video/audio clip

SUGGESTED READINGS

1. Adobe Creative Team, Adobe Premiere Pro Classroom in a Book, Adobe Press, Latest Edition.
2. Wooster Blake, The Art of Digital Video Editing, Focal Press, 2nd Edition, 2020.
3. Ken C. Pohlmann, Principles of Digital Audio, McGraw-Hill, 6th Edition, 2010.
4. Maxim Jago, Adobe Premiere Pro CC Classroom in a Book, Adobe Press, Latest Edition.
5. Jay Rose, Audio Postproduction for Film and Video, Focal Press, 3rd Edition, 2014.
6. Ashley Kennedy, Video Editing with Adobe Premiere Pro for Beginners, LinkedIn Learning (online).
7. Mike Collins, Pro Tools for Music Production, Focal Press, 2nd Edition, 2004
8. YouTube Creator Academy, Video Editing Tutorials, <https://creatoracademy.youtube.com>

25CSU601D

AUTOMATION AND TESTING

Semester –VI
3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- Understand the need and principles of software testing and automation.
- Explore various software testing techniques and strategies.
- Learn about test automation frameworks and tools.
- Design and implement test scripts using automation tools.
- Evaluate and improve testing performance through metrics and continuous integration.

Course Outcomes (COs)

At the completion of the course the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Explain software testing principles, lifecycle, and strategies...	Understand
CO2	Apply functional and non-functional testing techniques.	Apply
CO3	Develop automated test scripts using appropriate tools..	Create
CO4	Analyze test cases and improve testing efficiency.	Analyze
CO5	Evaluate testing metrics and integrate testing in CI/CD pipelines.	Create

CO-PO Mapping

CO / PO	P01	P02	P03	P04	P05	P06	P07	P08	P09	P010	P011	PS01	PS02
CO1	3	2	1	1	2	1	1	2	1	2	2	1	3
CO2	3	3	2	2	3	1	1	2	2	2	2	2	3
CO3	3	3	2	3	3	1	1	2	2	2	2	2	3
CO4	3	3	3	3	3	2	1	2	2	2	2	2	3
CO5	3	3	3	3	3	2	2	2	3	3	2	3	3

1 - low, 2 - medium, 3 - high

Unit I FUNDAMENTALS OF SOFTWARE TESTING

Testing principles, testing process and lifecycle, test levels (unit, integration, system, acceptance), test types (functional, non-functional), defect lifecycle, test documentation (test plan, test case, traceability matrix), role of QA in SDLC.

Unit II SOFTWARE TESTING TECHNIQUES

Black-box testing: Equivalence partitioning, boundary value analysis, decision table, state transition.

White-box testing: Statement, branch, path, condition coverage. Static and dynamic analysis, exploratory and ad-hoc testing.

Unit III TEST AUTOMATION BASICS

Introduction to automation, benefits and limitations, test automation lifecycle, selecting test cases for automation, frameworks (data-driven, keyword-driven, hybrid), scripting basics, automation ROI.

Unit IV TOOLS FOR TEST AUTOMATION

Overview and usage of tools, Selenium (WebDriver, IDE, Grid), JUnit/TestNG for test case management, Jenkins for Continuous Integration, Postman for API testing, LoadRunner/JMeter for performance testing, Tool selection criteria and integration with test frameworks.

Unit V TEST MANAGEMENT AND QUALITY METRICS

Defect management tools (e.g., Bugzilla, Jira), test metrics: defect density, test coverage, defect leakage. Test process improvement: CMMI, TMMi, ISO standards. CI/CD in testing, automated build and test execution, DevOps and QA collaboration.

SUGGESTED READINGS

1. Srinivasan Desikan, Gopaldaswamy Ramesh, Software Testing: Principles and Practices, Pearson Education, 2006.
2. Paul C. Jorgensen, Software Testing: A Craftsman's Approach, CRC Press, 2021 (5th Edition).
3. Aditya P. Mathur, Foundations of Software Testing, Pearson, 2008.
4. Rex Black, Pragmatic Software Testing, Wiley, 2007.
5. Selenium Documentation – <https://www.selenium.dev/>
6. Jenkins User Guide – <https://www.jenkins.io/doc/>

25CSU601E

BIG DATA ANALYTICS

Semester –VI
3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- Introduce students to the concept, architecture, and applications of big data.
- Equip students with skills to work with Hadoop ecosystem and related tools.
- Provide hands-on experience with data analytics using MapReduce, Hive, and Pig.
- Explore real-time big data processing tools like Spark and NoSQL databases.
- Apply big data analytics in solving real-world problems.

Course Outcomes (COs)

At the completion of the course the student will be able to

Cos	Course Outcomes	Blooms Level
CO1	Explain the fundamentals and technologies of big data systems	Understand
CO2	Apply Hadoop ecosystem components for storing and processing big data.	Apply
CO3	Develop MapReduce programs and perform data queries using Hive and Pig.	Create
CO4	Analyze large-scale data sets using Spark and NoSQL databases.	Analyze
CO5	Design big data solutions for practical applications in industry and research..	Create

CO-PO Mapping

CO / PO	P01	P02	P03	P04	P05	P06	P07	P08	P09	P010	P011	PS01	PS02
CO1	3	2	2	2	2	1	2	2	2	2	2	1	3
CO2	3	3	2	2	3	1	1	2	2	2	1	2	3
CO3	3	3	2	3	3	1	2	2	2	2	2	2	3
CO4	3	3	3	3	3	2	1	3	2	2	2	2	3
CO5	3	3	3	3	3	2	2	3	3	3	2	3	3

1 - low, 2 - medium, 3 - high

Unit I INTRODUCTION TO BIG DATA

Big Data overview, characteristics of Big Data (Volume, Variety, Velocity, Veracity, Value), challenges, Big Data vs Traditional Data, Data life cycle, applications of Big Data in various domains, role of data scientist.

Unit II HADOOP ECOSYSTEM AND HDFS

Hadoop architecture, Hadoop Distributed File System (HDFS), data replication, HDFS commands, YARN, Hadoop configuration, introduction to Hadoop ecosystem tools: Hive, Pig, HBase, Sqoop, Flume.

Unit III MAPREDUCE AND DATA PROCESSING

MapReduce programming model, combiner, partitioner, counters, sorting and shuffling, writing simple MapReduce programs using Java or Python, performance tuning, debugging MapReduce jobs.

Unit IV BIG DATA QUERYING AND NOSQL DATABASES

Hive: architecture, data types, queries, partitions, UDFs, Pig: architecture, Pig Latin basics, data models, operators, NoSQL databases: key-value, document, column-family, graph databases Introduction to MongoDB and Cassandra.

Unit V REAL-TIME ANALYTICS WITH APACHE SPARK

Need for real-time data processing, introduction to Spark, RDDs, transformations and actions, Spark SQL, Spark Streaming, Machine Learning with MLlib, integration with Hadoop and other tools.

SUGGESTED READINGS

1. Seema Acharya & Subhasini Chellappan, Big Data and Analytics, Wiley India, 2015.
2. Tom White, Hadoop: The Definitive Guide, O'Reilly Media, 2012.
3. V. Srinivasa Rao, Big Data Analytics, Wiley India, circa 2015.
4. Bill Franks, Taming the Big Data Tidal Wave, Wiley, 2012.
5. Jure Leskovec, Anand Rajaraman, Jeff Ullman, Mining of Massive Datasets, Cambridge University Press, 2014.
6. Apache Hadoop Documentation – <https://hadoop.apache.org>.
7. Apache Spark Documentation – <https://spark.apache.org>

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- Provide fundamental knowledge about various forms of intellectual property.
- Familiarize students with the laws and legal frameworks governing IPR.
- Understand the processes involved in the registration and protection of IPR.
- Explore the global and national IPR systems and their significance.
- Discuss the ethical and commercial aspects of IPR in engineering and innovation...

Course Outcomes (COs)

At the completion of the course the student will be able to

Cos	Course Outcomes	Blooms Level
CO1	Describe different types of intellectual property and their relevance.	Understand
CO2	Explain the procedures for obtaining patents, copyrights, and trademarks.	Understand
CO3	Analyze the legal aspects of IPR and related case studies.	Analyze
CO4	Evaluate the strategic and economic value of IPR in innovation.	Evaluate
CO5	Apply IPR knowledge in protecting one's inventions and designs.	Apply

CO-PO Mapping

CO / PO	P01	P02	P03	P04	P05	P06	P07	P08	P09	P010	P011	PS01	PS02
CO1	2	2	1	1	2	1	1	2	1	2	1	1	2
CO2	2	2	2	2	2	1	1	2	2	2	1	2	2
CO3	2	1	2	2	1	1	1	2	2	2	2	2	2
CO4	2	1	1	2	2	2	1	3	2	2	2	2	2
CO5	2	2	1	1	2	2	2	3	3	3	2	3	2

1 - low, 2 - medium, 3 - high

Unit I INTRODUCTION TO IPR AND LEGAL FRAMEWORK

Definition, scope, and importance of IPR, history and development of IPR systems, types of IPR: patents, copyrights, trademarks, trade secrets, industrial designs, geographical indications, layout designs, international treaties (WIPO, WTO, TRIPS), Indian legal framework.

Unit II PATENTS

Patent definition and types, criteria for patentability, process of patent filing in India and abroad, patent specification, rights and obligations of patent holders, infringement and remedies, revocation of patents, patent agents and their role, patent search and databases.

Unit III COPYRIGHTS AND RELATED RIGHTS

Basics of copyright law, literary and artistic works, rights of authors and owners, limitations and exceptions (fair use), registration procedure, term of protection, digital copyrights, moral rights, neighboring rights, infringement and penalties.

Unit IV TRADEMARKS AND INDUSTRIAL DESIGNS

Trademark definition and types, symbols, functions and selection of marks, registration process, infringement and protection, Industrial designs: meaning, protection, registration, rights of design holders, infringement.

Unit V EMERGING ISSUES AND CASE STUDIES IN IPR

Trade secrets and confidentiality agreements, IPR in digital age (software and data protection), IPR and AI, biotechnology patents, open source and creative commons, case studies of IPR disputes and resolutions in India and globally, ethical issues in IPR.

SUGGESTED READINGS

1. Neeraj Pandey & Khushdeep Dharni, Intellectual Property Rights, PHI Learning, 2014.
2. Prabuddha Ganguli, Intellectual Property Rights: Unleashing the Knowledge Economy, Tata McGraw Hill, 2001.
3. Deborah E. Bouchoux, Intellectual Property: The Law of Trademarks, Copyrights, Patents, and Trade Secrets, Cengage, 2023 (6th Edition).
4. P. Narayanan, Intellectual Property Law, Eastern Law House, 2020 (3rd Edition, Revised and Updated).
5. WIPO Resources – <https://www.wipo.int>
6. Indian Patent Office – <https://ipindia.gov.in>

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- Understand security architecture and models in the cloud environment.
- Learn threats, vulnerabilities, and risk management in cloud systems.
- Explore data privacy, integrity, and protection mechanisms.
- Analyze identity management, access control, and compliance standards.
- Apply best practices and technologies to secure cloud applications and data.

Course Outcomes (COs)

At the completion of the course the student will be able to

Cos	Course Outcomes	Blooms Level
CO1	Explain cloud security concepts, models, and architectures.	Understand
CO2	Identify risks, vulnerabilities, and threats in cloud computing.	Apply
CO3	Apply cryptographic techniques to ensure confidentiality and integrity..	Apply
CO4	Evaluate identity and access management solutions in cloud systems.	Evaluate
CO5	Recommend compliance practices and mitigation strategies for secure cloud environments.	Analyze

CO-PO Mapping

CO / PO	P01	P02	P03	P04	P05	P06	P07	P08	P09	P010	P011	PS01	PS02
CO1	3	2	1	1	2	1	3	2	1	2	1	1	3
CO2	3	3	2	2	3	1	3	2	2	2	1	2	3
CO3	3	3	2	3	3	1	3	2	2	2	2	2	3
CO4	3	3	3	3	3	2	3	3	2	2	2	2	3
CO5	3	3	3	3	3	2	2	3	3	3	2	3	3

1 - low, 2 - medium, 3 - high

Unit I CLOUD SECURITY ARCHITECTURE AND MODELS

Overview of cloud computing, security concepts and principles, shared responsibility model, cloud security architecture, CIA triad in cloud, security-as-a-service, threat models in cloud (STRIDE, DREAD), virtualization and container security.

Unit II RISKS, THREATS, AND VULNERABILITIES IN CLOUD

Types of threats (data breaches, DoS, malicious insiders), OWASP top 10 for cloud, risk analysis and mitigation, security policies, cloud attack surfaces, securing multi-tenancy and APIs, hypervisor vulnerabilities.

Unit III DATA SECURITY AND PRIVACY

Data lifecycle in cloud, data classification and protection, encryption (at rest, in transit, in use), key management systems (KMS), homomorphic encryption, secure data deletion, data deduplication and integrity checks, GDPR and privacy laws.

Unit IV IDENTITY AND ACCESS MANAGEMENT (IAM)

Authentication and authorization mechanisms, single sign-on (SSO), OAuth, OpenID Connect, SAML, role-based and attribute-based access control (RBAC, ABAC), multi-factor authentication (MFA), IAM in AWS/Azure/GCP.

Unit V SECURITY MONITORING, COMPLIANCE AND BEST PRACTICES

Security monitoring and logging, SIEM tools, incident response, security audits, compliance standards (ISO 27001, NIST, HIPAA, SOC 2), cloud security best practices, zero trust architecture, DevSecOps.

SUGGESTED READINGS

1. Tim Mather, Subra Kumaraswamy & Shahed Latif, Cloud Security and Privacy, O'Reilly, 2009.
2. Ronald L. Krutz & Russell Dean Vines, Cloud Security: A Comprehensive Guide to Secure Cloud Computing, Wiley, 2010.
3. Kumar Saurabh, Cloud Security: Principles and Practice, Wiley, 2012.
4. NIST Cloud Computing Security Guidelines – <https://nvlpubs.nist.gov>
5. Cloud Security Alliance (CSA) – <https://cloudsecurityalliance.org>
6. AWS Security Documentation – <https://docs.aws.amazon.com/security>

25CSU602C

UI AND UX DESIGN

Semester –VI
3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- Understand the principles and process of User Interface (UI) and User Experience (UX) design.
- Analyze user needs and create user personas and user journeys.
- Apply design principles for creating effective and aesthetic interfaces.
- Use tools and techniques to prototype, test, and evaluate UI/UX designs
- Understand the role of accessibility, usability, and responsiveness in interface design..

Course Outcomes (COs)

At the completion of the course the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Describe UI/UX design principles, processes, and trends.	Understand
CO2	Analyze user requirements and design user flows and personas.	Analyze
CO3	Apply wireframing, prototyping, and visual design techniques using design tools.	Apply
CO4	Evaluate designs for usability, accessibility, and responsiveness.	Evaluate
CO5	Develop and test user-centric UI/UX solutions for real-world applications.	Create

CO-PO Mapping

CO / PO	P01	P02	P03	P04	P05	P06	P07	P08	P09	P010	P011	PS01	PS02
CO1	3	2	1	1	2	1	1	2	1	2	1	1	3
CO2	3	3	2	2	3	1	1	2	2	2	1	2	3
CO3	3	3	2	3	3	1	1	2	2	2	2	2	3
CO4	3	3	3	3	3	2	1	3	2	2	2	2	3
CO5	3	3	3	3	3	2	2	3	3	3	2	3	3

1 - low, 2 - medium, 3 - high

Unit I INTRODUCTION TO UI AND UX DESIGN

Definition and difference between UI and UX, importance of user-centered design, design thinking process, evolution of interfaces, design trends and challenges, roles of UX/UI designer, overview of interaction design and HCI.

Unit II UNDERSTANDING USERS AND REQUIREMENTS

User research techniques: interviews, surveys, observations; creating user personas, empathy maps, user journeys, user stories and use cases, information architecture, task analysis, content strategy.

Unit III DESIGN PRINCIPLES AND WIREFRAMING

Gestalt principles, visual hierarchy, consistency, affordance, feedback, UI patterns, color theory, typography, layout design (grids, spacing), wireframing tools (Figma, Adobe XD, Balsamiq), designing for different devices.

Unit IV PROTOTYPING, USABILITY AND ACCESSIBILITY

Low- and high-fidelity prototyping, interaction design, micro-interactions, usability heuristics (Nielsen's heuristics), usability testing methods (A/B testing, think aloud), accessibility standards (WCAG), inclusive design.

Unit V UX EVALUATION AND REAL-WORLD APPLICATIONS

UX metrics (SUS, NPS, task success rate), feedback and iteration, responsive and adaptive design, mobile-first and progressive enhancement, designing for web vs. mobile vs. desktop, UX case studies, portfolio preparation.

SUGGESTED READINGS

1. Alan Cooper, Robert Reimann & David Cronin, *About Face: The Essentials of Interaction Design*, Wiley, 2014.
2. Jesse James Garrett, *The Elements of User Experience*, Pearson Education, 2002.
3. Don Norman, *The Design of Everyday Things*, MIT Press, 1988.
4. Jeff Gothelf, *Lean UX: Applying Lean Principles to Improve User Experience*, O'Reilly, 2013.
5. Nielsen Norman Group articles – <https://www.nngroup.com>
6. Figma documentation – <https://www.figma.com/resources>

25CSU602D

APPLICATION DEVELOPMENT USING DEVOPS

Semester –VI
3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- Understand DevOps principles, practices, and its role in modern application development.
- Learn to integrate development and operations using CI/CD pipelines.
- Use version control, build automation, testing, and containerization tools.
- Automate deployment and monitor applications in cloud environments.
- Develop scalable, testable, and secure applications through agile DevOps practices.

Course Outcomes (COs)

At the completion of the course the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Describe the DevOps culture, tools, and practices.	Understand
CO2	Implement version control and automation in software development workflows..	Create
CO3	Design and deploy CI/CD pipelines using tools like Jenkins, Git, and Docker..	Create
CO4	Analyze application monitoring, testing, and feedback integration in DevOps..	Analyze
CO5	Develop cloud-native applications with scalability and security using DevOps principles.	Create

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
CO1	3	2	1	1	2	1	1	2	1	2	1	1	3
CO2	3	3	2	2	3	1	1	2	2	2	1	2	3
CO3	3	3	2	3	3	1	1	2	2	2	2	2	3
CO4	3	3	3	3	3	2	1	3	2	2	2	2	3
CO5	3	3	3	3	3	2	2	3	3	3	2	3	3

1 - low, 2 - medium, 3 - high

Unit I INTRODUCTION TO DEVOPS

DevOps overview and need, DevOps culture and principles (CALMS), Agile vs. DevOps, DevOps lifecycle, collaboration between development and operations, DevOps tools overview, DevOps in cloud and microservices.

Unit II SOURCE CONTROL AND BUILD AUTOMATION

Version control systems – Git, GitHub/GitLab, branching strategies, Git workflows (Git Flow, Forking), Git hooks. Build automation tools – Maven, Gradle, npm, automated builds and artifact repositories (Nexus, Artifactory).

Unit III CONTINUOUS INTEGRATION AND TESTING

CI concepts, Jenkins pipelines (declarative vs scripted), Travis CI, GitHub Actions. Automated testing: unit, integration, and functional testing using JUnit, Selenium, PyTest. Static code analysis (SonarQube), test coverage tools, test automation in pipelines.

Unit IV CONTINUOUS DEPLOYMENT AND CONTAINERIZATION

CD concepts, deployment automation, Infrastructure as Code (IaC) using Ansible/Terraform. Docker: architecture, Dockerfiles, Docker Compose, container registries. Kubernetes basics: pods, services, deployments, scaling, Helm charts, Blue/green and canary deployments.

Unit V MONITORING, LOGGING, AND SECURITY IN DEVOPS

Monitoring tools: Prometheus, Grafana, ELK Stack, Log management and alerting, performance metrics, SRE basics. Security in DevOps (DevSecOps): secure code, container security, secrets management, shift-left security, OWASP DevSecOps practices.

SUGGESTED READINGS

1. Len Bass, Ingo Weber & Liming Zhu, DevOps: A Software Architect's Perspective, Addison-Wesley, 2015.
2. Kanaan Gill, The Phoenix Project: A Novel About IT, DevOps, and Helping Your Business Win, IT Revolution Press, 2013.
3. Gene Kim, Jez Humble & Patrick Debois, The DevOps Handbook, IT Revolution Press, 2015.
4. Ritesh Modi, Learning DevOps, Packt Publishing, 2017.
5. Kubernetes Documentation – <https://kubernetes.io>
6. Jenkins User Documentation – <https://www.jenkins.io/doc/>
7. DevSecOps Guide by OWASP – <https://owasp.org/www-project-devsecops-guideline>

25CSU602E

LINUX TECHNOLOGIES

Semester –VI
3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- Introduce the fundamental concepts of Linux operating systems, system architecture, and command-line interface.
- Enable students to manage Linux file systems, users, groups, permissions, and storage devices.
- Develop scripting skills using Bash to automate system tasks and manage processes.
- Familiarize students with Linux networking, system services, and remote administration tools.
- Equip students with knowledge of Linux package management, logging, backup strategies, and system security practices

Course Outcomes (COs)

At the completion of the course the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Explain the architecture, boot process, and core components of a Linux system.	Understand
CO2	Perform file system operations, manage users and groups, and handle disk partitions and mounting.	Apply
CO3	Write and execute shell scripts to automate administrative tasks and manage system processes	Create
CO4	Configure network settings, manage services, and perform secure remote system administration	Analyze
CO5	Install and manage software packages, monitor system logs, implement basic security, and perform backup operations	Create

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	1	1	2	1	1	2	1	2	1	1	3
CO2	3	3	2	2	3	1	1	2	2	2	1	2	3
CO3	3	3	2	3	3	1	1	2	2	2	2	2	3
CO4	3	3	3	3	3	2	1	3	2	2	2	2	3
CO5	3	3	3	3	3	2	2	3	3	3	2	3	3

1 - low, 2 - medium, 3 - high

Unit I INTRODUCTION TO LINUX AND SYSTEM ARCHITECTURE

History and evolution of Linux, Linux distributions and installation, Linux system architecture and boot process, Kernel and init systems (SysVinit, systemd), Linux shell overview and command line basics, Manual pages and help commands (man, info, help).

Unit II FILE SYSTEMS AND USER MANAGEMENT

Linux file system hierarchy (FHS), file and directory operations, hard and soft links, file permissions and ownership (chmod, chown, umask), user and group administration (useradd, usermod, groupadd), disk partitioning and mounting (fdisk, mount, fstab), file compression and archiving (gzip, tar, zip, rsync).

Unit III SHELL SCRIPTING AND PROCESS MANAGEMENT

Bash shell scripting basics including variables, conditionals, and loops; script execution and debugging, command chaining and redirection (|, >, >>, 2>), job control for foreground and background processes, process management (ps, top, kill, nice, cron, at), scheduling with cron and systemd timers.

Unit IV NETWORKING AND SYSTEM SERVICES

Basic networking commands (ping, netstat, ss, ifconfig, ip, nmcli), configuration files and network troubleshooting, managing services using systemctl, remote access tools such as SSH, FTP, SCP, basic firewall configuration using iptables and firewalld, introduction to essential network services like DNS, DHCP, NFS, and SAMBA.

Unit V PACKAGE MANAGEMENT, LOGGING, AND SECURITY

Package management using rpm, yum, dnf, and apt, installing software and configuring repositories, system logging using journalctl, rsyslog, and files in /var/log, backup strategies and tools (rsync, tar, cron), Linux security concepts including sudo, SELinux/AppArmor, basic firewall rules, introduction to audit frameworks, authentication using /etc/passwd and PAM.

SUGGESTED READINGS

1. Evi Nemeth, Garth Snyder, Trent Hein, UNIX and Linux System Administration Handbook, Pearson Education, 5th Edition, 2017.
2. Richard Blum, Linux Command Line and Shell Scripting Bible, Wiley, 3rd Edition, 2015.
3. Christopher Negus, Linux Bible, Wiley, 10th Edition, 2020.
4. Mark G. Sobell, A Practical Guide to Linux Commands, Editors, and Shell Programming, Pearson, 4th Edition, 2017.
5. Red Hat Documentation – <https://access.redhat.com/documentation>
6. Linux man pages – <https://man7.org/linux/man-pages/>
7. The Linux Documentation Project – <https://tldp.org>

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- Understand the security challenges in cloud computing environments.
- Explore advanced cryptographic techniques for securing cloud infrastructure and data.
- Examine authentication, access control, and identity management in cloud.
- Learn about virtualization, container security, and secure cloud architecture.
- Study compliance, governance, and secure DevOps in cloud platforms..

Course Outcomes (COs)

At the completion of the course the student will be able to

Cos	Course Outcomes	Blooms Level
CO1	Explain security threats, models, and mitigation strategies in cloud environments.	Understand
CO2	Apply advanced cryptographic and encryption techniques to secure cloud data.	Apply
CO3	Analyze and configure secure access control and identity management mechanisms.	Analyze
CO4	Evaluate secure virtualization, containerization, and multi-tenant cloud models..	Evaluate
CO5	Design secure cloud solutions using governance, monitoring, and compliance tools.	Apply

CO-PO Mapping

CO / PO	P01	P02	P03	P04	P05	P06	P07	P08	P09	P010	P011	PS01	PS02
CO1	3	2	1	1	2	1	3	2	1	2	1	1	3
CO2	3	3	2	2	3	1	3	2	2	2	1	2	3
CO3	3	3	2	3	3	1	3	2	2	2	2	2	3
CO4	3	3	3	3	3	2	3	3	2	2	2	2	3
CO5	3	3	3	3	3	2	2	3	3	3	2	3	3

1 - low, 2 - medium, 3 - high

Unit I CLOUD SECURITY OVERVIEW AND THREAT LANDSCAPE

Cloud computing models (IaaS, PaaS, SaaS), shared responsibility model, threat taxonomy (data breaches, APTs, insecure APIs, insider threats), cloud security architecture, NIST cloud security standards, STRIDE/DREAD models.

Unit II CRYPTOGRAPHY AND DATA PROTECTION IN CLOUD

Data lifecycle (at rest, in transit, in use), symmetric and asymmetric encryption, homomorphic encryption, searchable encryption, data tokenization, secure key management (HSM, KMS), TLS/SSL and VPNs for secure communication.

Unit III IDENTITY, AUTHENTICATION, AND ACCESS CONTROL

IAM concepts: RBAC, ABAC, policy-based access, federated identity, single sign-on (SSO), OAuth 2.0, OpenID Connect, SAML, multifactor authentication (MFA), zero-trust architecture, identity federation across cloud services.

Unit IV VIRTUALIZATION, CONTAINER, AND APPLICATION SECURITY

Virtual machine and hypervisor security, VM isolation and image hardening, Docker/container security best practices, container scanning tools, Kubernetes security (RBAC, secrets, network policies), secure APIs, DevSecOps pipeline integration.

Unit V GOVERNANCE, COMPLIANCE, AND ADVANCED CLOUD SECURITY

Cloud governance frameworks, regulatory compliance (GDPR, HIPAA, ISO/IEC 27001), logging and monitoring (SIEM tools), incident response, security SLAs, auditability, cloud security reference architectures (AWS/Azure/GCP), threat modeling and risk assessment.

SUGGESTED READINGS

1. Ronald L. Krutz, Russell Dean Vines, Cloud Security: A Comprehensive Guide to Secure Cloud Computing, Wiley, 2010.
2. Tim Mather, Subra Kumaraswamy, Shahed Latif, Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance, O'Reilly, 2009.
3. Kumar Saurabh, Cloud Security: Principles and Practice, Wiley, 2012.
4. Cloud Security Alliance (CSA) Best Practices, CSA
5. NIST Special Publication 800-144: Guidelines on Security and Privacy in Public Cloud Computing, NIST, 2011.
6. OWASP Cloud Security Guidelines, OWASP

25CSU701B

EDGE AND FOG COMPUTING

Semester –VII

3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100

End Semester Exam: 3 Hours

Course Objectives

- Introduce the concepts and architecture of Edge and Fog computing.
- Understand the role of Edge/Fog computing in supporting latency-sensitive and IoT applications.
- Explore data processing, communication, and service orchestration in distributed environments.
- Evaluate platforms, tools, and frameworks for developing Edge/Fog applications.
- Address security, privacy, and resource management challenges in Edge and Fog systems.

Course Outcomes (COs)

At the completion of the course the student will be able to

Cos	Course Outcomes	Blooms Level
CO1	Explain the principles and architectures of Edge and Fog computing.	Understand
CO2	Analyze the differences and trade-offs between cloud, edge, and fog paradigms..	Analyze
CO3	Apply Edge/Fog computing models for real-time and IoT applications.	Apply
CO4	Evaluate middleware platforms, orchestration techniques, and data analytics in Edge/Fog.	Evaluate
CO5	Design secure and scalable Edge/Fog-based solutions for smart environments..	Apply

CO-PO Mapping

CO / PO	P01	P02	P03	P04	P05	P06	P07	P08	P09	P010	P011	PS01	PS02
CO1	3	2	1	1	2	1	1	2	1	2	1	1	3
CO2	3	3	2	2	3	1	1	2	2	2	1	2	3
CO3	3	3	2	3	3	1	1	2	2	2	2	2	3
CO4	3	3	3	3	3	2	1	3	2	2	2	2	3
CO5	3	3	3	3	3	2	2	3	3	3	2	3	3

1 - low, 2 - medium, 3 - high

Unit I INTRODUCTION TO EDGE AND FOG COMPUTING

Definition and evolution, comparison with cloud computing, drivers and motivation, Edge vs. Fog vs. Cloud, key characteristics and components, typical use cases (smart cities, autonomous vehicles, healthcare, industrial IoT), architecture models (Cisco Fog Computing, ETSI MEC).

Unit II EDGE AND FOG COMPUTING ARCHITECTURE AND TECHNOLOGIES

Hardware and software components, edge gateways, sensors, routers, virtualization at the edge, containers vs. VMs, microservices in Edge/Fog, OpenFog reference architecture, real-time requirements, latency constraints, network topology and QoS.

Unit III DATA MANAGEMENT AND ORCHESTRATION

Data acquisition, filtering, stream processing at the edge, distributed analytics, event processing, orchestration layers, task offloading, workload distribution, resource allocation, mobility support, orchestration tools (KubeEdge, EdgeX Foundry).

Unit IV PLATFORMS AND TOOLS

Fog computing platforms: Cisco IOx, OpenFog, Eclipse Fog05 Edge computing platforms: AWS Greengrass, Azure IoT Edge, Google Edge TPU Edge AI and ML models at the edge, TensorFlow Lite, edge model deployment, container orchestration (Docker, Kubernetes, K3s), benchmarking tools.

Unit V SECURITY, PRIVACY, AND APPLICATIONS

Security challenges: data integrity, authentication, attack surfaces, threat models. Privacy preservation: anonymization, encryption, access control. Edge/Fog in smart homes, healthcare, intelligent transportation, agriculture. Case studies and current research directions.

SUGGESTED READINGS

1. Bahar Memarian & Rajkumar Buyya, Fog and Edge Computing: Principles and Paradigms, Wiley, 2019.
2. F. Bonomi et al., Fog Computing: Concepts, Applications and Research Opportunities, Springer, 2012.
3. Puneet Sharma & Dijiang Huang, Edge Computing: Models, Technologies and Applications, CRC Press, 2019.
4. OpenFog Consortium, OpenFog Reference Architecture, 2016.

25CSU701C

GAME DEVELOPMENT

Semester –VII
3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- Introduce the fundamentals of game design and development.
- Familiarize students with game engines and their components.
- Enable students to develop 2D and 3D games using scripting and tools.
- Explore physics, animation, AI, and audio integration in games.
- Provide knowledge of testing, deployment, and monetization of games.

Course Outcomes (COs)

At the completion of the course the student will be able to

Cos	Course Outcomes	Blooms Level
CO1	Explain the process and principles of game design and development.	Understand
CO2	Design and implement interactive 2D and 3D game environments using game engines.	Apply
CO3	Integrate animation, sound, and physics to enhance game interactivity.	Analyze
CO4	Develop simple game AI for non-player character behavior..	Create
CO5	Evaluate and deploy games with performance optimization and user feedback.	Evaluate

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	2	2	1	1	2	1	1	2	3	3	1	1	3
CO2	2	3	2	2	3	1	1	2	3	3	1	2	3
CO3	2	3	2	3	3	1	1	2	3	3	2	2	3
CO4	2	3	3	3	3	2	1	3	3	3	2	2	3
CO5	2	3	3	3	3	2	2	3	3	3	2	3	3

1 - low, 2 - medium, 3 - high

Unit I INTRODUCTION TO GAME DEVELOPMENT

History and evolution of games, game genres, game development life cycle, game design elements, game architecture, storytelling in games, role of the game engine, overview of Unity and Unreal Engine, scripting basics in C# (Unity) or Blueprint (Unreal).

Unit II GAME GRAPHICS AND ASSET INTEGRATION

2D vs 3D games, sprites, tilesets, game scenes, importing assets, cameras and lighting, textures and materials, scene hierarchy, animation basics, user interface (UI) elements, player controls, collision detection.

Unit III GAME PHYSICS AND ANIMATION

Rigid body dynamics, forces, gravity, velocity, friction, collision resolution, character movement, animation blending, keyframe animation, inverse kinematics, ragdoll physics, particle systems.

Unit IV GAME ARTIFICIAL INTELLIGENCE

Pathfinding (A*, NavMesh), finite state machines, decision trees, AI behaviors: patrol, chase, attack, simple strategy AI, group AI, AI in multiplayer games, event-driven programming, input handling.

Unit V GAME TESTING, DEPLOYMENT AND MONETIZATION

Bug tracking, playtesting, performance optimization, platform-specific deployment (PC, Android, iOS), building APKs or executables, publishing to stores (Google Play, Steam), game analytics, monetization models (ads, freemium, in-app purchases).

SUGGESTED READINGS

1. Rabin, Steve, Introduction to Game Development, Cengage Learning.
2. Jeremy Gibson Bond, Introduction to Game Design, Prototyping, and Development, Addison-Wesley.
3. Mike McShaffry, Game Coding Complete, Charles River Media.
4. Unity Learn – <https://learn.unity.com>
5. Unreal Engine Documentation – <https://docs.unrealengine.com>
6. GameDev.net – <https://www.gamedev.net>

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100

End Semester Exam: 3 Hours

Course Objectives

- Equip students with the skills to develop modern full stack web applications.
- Introduce backend and frontend frameworks with hands-on practice.
- Integrate Generative AI tools and APIs to enhance web development productivity.
- Use LLMs for code generation, content creation, and intelligent interfaces.
- Deploy and manage web apps with performance and scalability in mind.

Course Outcomes (COs)

At the completion of the course the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Design and develop responsive frontend applications using modern frameworks...	Understand
CO2	Build secure and scalable backend services using Node.js, Express, and databases..	Apply
CO3	Integrate GenAI APIs (e.g., OpenAI, Hugging Face) into full stack applications.	Analyze
CO4	Automate UI/UX and content generation using LLMs and prompt engineering.	Analyze
CO5	Deploy full stack GenAI-powered apps on cloud platforms with CI/CD.	Create

CO-PO Mapping

CO / PO	P01	P02	P03	P04	P05	P06	P07	P08	P09	P010	P011	PS01	PS02
CO1	3	2	2	3	2	1	1	2	1	2	1	2	3
CO2	3	3	2	2	3	1	1	2	2	2	1	2	3
CO3	3	3	2	3	3	1	1	2	2	2	2	2	3
CO4	3	3	3	3	3	2	1	3	2	2	2	2	3
CO5	3	3	3	3	3	2	2	3	3	3	2	3	3

1 - low, 2 - medium, 3 - high

Unit I FUNDAMENTALS OF FULL STACK DEVELOPMENT

Web architecture overview (client-server, MVC), introduction to HTML5, CSS3, JavaScript (ES6+), DOM manipulation, responsive design with Bootstrap/Tailwind CSS, Git and GitHub for version control, introduction to REST APIs.

Unit II FRONTEND FRAMEWORKS AND COMPONENT DESIGN

React.js basics: components, props, state, hooks, routing, lifecycle methods. Single-page applications (SPA), data fetching with Axios or Fetch API, using local Storage and session Storage, basic frontend security practices (CSP, input validation).

Unit III BACKEND DEVELOPMENT AND DATABASES

Node.js and Express.js: routing, middleware, RESTful services. User authentication with JWT and OAuth, role-based access, using MongoDB or PostgreSQL (CRUD operations, schemas, Mongoose/Sequelize), error handling, environment configuration with dotenv.

Unit IV INTRODUCTION TO GENERATIVE AI AND INTEGRATION

Overview of GenAI (LLMs, diffusion models, prompt engineering). Using OpenAI/ Anthropic / HuggingFace APIs for: Code generation (e.g., autocompleting backend logic) UI text/image generation (hero banners, product descriptions) Chatbot interfaces using LLMs (React + Node-based chatbot) Fine-tuning and safety considerations, using GenAI responsibly.

Unit V DEPLOYMENT, CI/CD, AND CLOUD INTEGRATION

Hosting frontend on Netlify/Vercel and backend on Render/Heroku/AWS. CI/CD pipelines using GitHub Actions. Monitoring and logging (PostHog, LogRocket), securing API keys and environment variables. Case studies: GenAI in e-commerce, blogs, and portfolio sites.

SUGGESTED READINGS

1. Steve Rabin, Introduction to Game Development, Cengage Learning, 2009.
2. Jeremy Gibson Bond, Introduction to Game Design, Prototyping, and Development, Addison-Wesley, 2014.
3. Mike McShaffry, Game Coding Complete, Charles River Media, 2012.
4. Unity Learn Resources, Unity Technologies, (continuously updated).
5. Unreal Engine Documentation, Epic Games, (continuously updated).

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- Provide an understanding of digital technologies transforming modern marketing practices.
- Explore tools and platforms used in digital marketing such as SEO, SEM, and social media.
- Analyze how data, automation, and AI drive customer engagement and personalization.
- Apply digital analytics and campaign strategies in real-world marketing contexts.
- Understand ethical, legal, and technological challenges in digital marketing.

Course Outcomes (COs)

At the completion of the course the student will be able to

Cos	Course Outcomes	Blooms Level
CO1	Explain the role of digital technologies in modern marketing ecosystems.	Understand
CO2	Apply SEO, SEM, email, and content marketing tools in real-world scenarios.	Apply
CO3	Analyze social media metrics, web traffic, and customer engagement data.	Analyze
CO4	Evaluate digital marketing campaigns using analytics and performance metrics.	Evaluate
CO5	Design customer-centric, data-driven marketing strategies using digital tools	Apply

CO-PO Mapping

CO / PO	P01	P02	P03	P04	P05	P06	P07	P08	P09	P010	P011	PS01	PS02
CO1	2	2	1	1	2	1	1	2	1	2	1	1	1
CO2	2	3	2	2	2	1	1	2	2	2	1	2	2
CO3	2	3	2	3	2	1	1	2	2	2	2	2	2
CO4	2	2	3	3	2	2	1	3	2	2	2	2	2
CO5	2	3	3	3	2	2	2	3	3	3	2	2	1

1 - low, 2 - medium, 3 - high

Unit I INTRODUCTION TO DIGITAL MARKETING AND TECHNOLOGY TRENDS

Overview of traditional vs. digital marketing, digital transformation in marketing, marketing funnel in the digital age, customer journey mapping, omnichannel marketing, key technologies: mobile, cloud, IoT, AI, VR/AR in marketing, trends in Martech.

Unit II SEARCH ENGINE OPTIMIZATION (SEO) AND SEARCH ENGINE MARKETING (SEM)

Basics of search engines, on-page and off-page SEO, keyword research tools (Google Keyword Planner, Ubersuggest), link building strategies, Google Ads, PPC campaigns, display advertising, remarketing, ad metrics (CTR, CPC, quality score).

Unit III CONTENT, EMAIL, AND INFLUENCER MARKETING

Content marketing strategy, blog and video marketing, personalization, email marketing platforms (Mailchimp, HubSpot), A/B testing in email campaigns, influencer marketing platforms and KPIs, UGC (user-generated content), content calendars and automation.

Unit IV SOCIAL MEDIA MARKETING AND ANALYTICS

Social platforms (Facebook, Instagram, LinkedIn, X, YouTube), targeting strategies, campaign creation, social media metrics (engagement rate, reach, conversion), scheduling tools (Hootsuite, Buffer), sentiment analysis, Google Analytics basics, dashboards.

Unit V AI, AUTOMATION, ETHICS AND FUTURE TRENDS

Marketing automation tools (CRM, chatbots, predictive analytics), personalization using AI (recommendation engines), programmatic advertising, voice search, digital twins, data privacy laws (GDPR, CCPA), ethical considerations in digital targeting and tracking.

SUGGESTED READINGS

1. Philip Kotler, Hermawan Kartajaya, Iwan Setiawan, Marketing 5.0: Technology for Humanity, Wiley, 2021.
2. Damian Ryan, Understanding Digital Marketing, Kogan Page, 2020.
3. Google Digital Garage Resources, Google.
4. HubSpot Academy Resources, HubSpot.
5. Neil Patel, Digital Marketing Insights, Neil Patel.
6. Hootsuite Social Trends Reports, Hootsuite

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- Introduce the fundamental concepts of blockchain and decentralized systems.
- Explain the structure, function, and mechanisms behind cryptocurrencies like Bitcoin and Ethereum.
- Explore smart contracts, consensus algorithms, and distributed ledger technologies.
- Examine blockchain applications beyond cryptocurrency, including DeFi and NFTs.
- Analyze security, scalability, and regulatory issues related to blockchain systems.

Course Outcomes (COs)

At the completion of the course the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Explain blockchain architecture, consensus mechanisms, and cryptographic foundations.	Understand
CO2	Apply SEO, SEM, email, and content marketing tools in real-world scenarios.	Apply
CO3	Analyze social media metrics, web traffic, and customer engagement data.	Analyze
CO4	Evaluate digital marketing campaigns using analytics and performance metrics.	Evaluate
CO5	Design customer-centric, data-driven marketing strategies using digital tools	Apply

CO-PO Mapping

CO / PO	P01	P02	P03	P04	P05	P06	P07	P08	P09	P010	P011	PS01	PS02
CO1	3	2	1	1	2	2	3	2	2	2	1	1	3
CO2	3	3	2	2	3	2	3	2	2	2	1	2	3
CO3	3	3	2	3	3	2	3	2	2	2	2	2	3
CO4	3	3	3	3	3	2	3	3	2	2	2	2	3
CO5	3	3	3	3	3	2	2	3	3	3	2	3	3

1 - low, 2 - medium, 3 - high

Unit I INTRODUCTION TO BLOCKCHAIN AND CRYPTOGRAPHY

Blockchain fundamentals, distributed systems, structure of a block, blockchain ledger, hash functions, Merkle trees, digital signatures, public and private keys, cryptographic hash (SHA-256), basics of consensus and decentralization.

Unit II CRYPTOCURRENCY AND BITCOIN PROTOCOL

Overview of cryptocurrency, Bitcoin architecture, transactions, proof-of-work (PoW), mining, forks, wallets, Bitcoin scripting language, UTXO model, double-spending problem, key management, real-world challenges.

Unit III ETHEREUM AND SMART CONTRACTS

Ethereum architecture and EVM, account-based model, smart contract development with Solidity, gas and transaction fees, ERC-20 and ERC-721 tokens, Remix IDE, DApps, introduction to web3.js and MetaMask.

Unit IV BLOCKCHAIN PLATFORMS AND USE CASES

Permissioned vs. permissionless blockchains, Hyperledger Fabric, Corda, enterprise applications, supply chain, identity management, healthcare, DeFi, NFTs and digital assets, tokenization, DAOs (Decentralized Autonomous Organizations).

Unit V SECURITY, SCALABILITY, AND LEGAL ASPECTS

Security risks: 51% attack, Sybil attack, front-running, flash loans; scalability challenges: sharding, Layer 2 solutions (Lightning Network, Optimistic Rollups); privacy: zk-SNARKs, mixers, regulations, compliance (FATF, AML/KYC), legal status of cryptocurrencies.

SUGGESTED READINGS

1. Arvind Narayanan et al., Bitcoin and Cryptocurrency Technologies, Princeton University Press, 2016.
2. Imran Bashir, Mastering Blockchain, Packt Publishing, 2020.
3. Ethereum Documentation, Ethereum Foundation.
4. Bitcoin Whitepaper by Satoshi Nakamoto, 2008.
5. Solidity Documentation, Ethereum Foundation.
6. Hyperledger Fabric Documentation, Linux Foundation.

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- Introduce the fundamentals and need for multicloud strategies.
- Explain architecture, deployment models, and design considerations in multicloud environments.
- Provide knowledge of inter-cloud networking, identity federation, and workload portability.
- Explore tools and platforms for multicloud management, monitoring, and automation.
- Address security, governance, compliance, and cost optimization in multicloud environments.

Course Outcomes (COs)

At the completion of the course the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Explain the concepts, drivers, and benefits of multicloud computing.	Understand
CO2	Analyze architecture and deployment models for multicloud strategies.	Analyze
CO3	Apply tools and techniques for managing and orchestrating resources across multiple clouds.	Apply
CO4	Evaluate networking, security, and workload migration issues in multicloud environments..	Evaluate
CO5	Design a cost-effective, secure, and scalable multicloud architecture.	Apply

CO-PO Mapping

CO / PO	P01	P02	P03	P04	P05	P06	P07	P08	P09	P010	P011	PS01	PS02
CO1	3	2	1	1	2	1	1	2	2	2	1	1	3
CO2	3	3	2	2	3	1	1	2	2	2	1	2	3
CO3	3	3	2	3	3	1	1	2	2	2	2	2	3
CO4	3	3	3	3	3	2	1	3	2	2	2	2	3
CO5	3	3	3	3	3	2	2	3	3	3	2	3	3

1 - low, 2 - medium, 3 - high

Unit I INTRODUCTION TO MULTICLOUD COMPUTING

Definition and need for multicloud strategies, single cloud vs. multicloud vs. hybrid cloud, benefits and challenges, vendor lock-in issues, common multicloud scenarios (disaster recovery, geo-redundancy, cloud bursting), multicloud adoption trends.

Unit II MULTICLOUD ARCHITECTURE AND DESIGN PRINCIPLES

Architectural patterns (distributed, federated, brokered), cloud service models (IaaS, PaaS, SaaS), cloud providers overview (AWS, Azure, GCP, IBM Cloud, Oracle Cloud), workload placement, interoperability, identity federation (SSO, OAuth, SAML), SLA management.

Unit III NETWORKING AND SECURITY IN MULTICLOUD

Inter-cloud connectivity (VPNs, Direct Connect, Interconnect), DNS, latency and data routing, encryption in transit and at rest, zero-trust security, cloud firewalls, shared responsibility model, secrets management, multicloud IAM, SIEM tools.

Unit IV MULTICLOUD MANAGEMENT AND AUTOMATION TOOLS

Cloud management platforms (CMPs), provisioning and orchestration (Terraform, Ansible, CloudFormation), containerization (Docker, Kubernetes, Anthos, Azure Arc), monitoring (Prometheus, Grafana), logging (ELK Stack), cost management tools (CloudHealth, Spot.io).

Unit V GOVERNANCE, COMPLIANCE, AND USE CASES

Policy enforcement, cloud governance frameworks, FinOps, cost optimization strategies, regulatory compliance (GDPR, HIPAA, ISO), incident response in multicloud, case studies in finance, healthcare, and enterprise IT, future trends (serverless in multicloud, AOps).

SUGGESTED READINGS

1. Ernesto Garbarino, *Multicloud Strategy for Cloud Architects*, Packt Publishing, 2022.
2. Jim Freeman, *Cloud Strategy: A Decision-based Approach to a Multicloud World*, Apress, 2022.
3. Google Anthos Documentation, Google Cloud.
4. Azure Arc Documentation, Microsoft.
5. AWS Multicloud Solutions Overview, Amazon Web Services.
6. Cloud Native Computing Foundation (CNCF) Tools Landscape, CNCF.

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- Introduce core AI concepts used in game programming.
- Teach the design of intelligent game behaviors for non-player characters (NPCs).
- Explore pathfinding, decision-making, and procedural generation in games.
- Implement and evaluate various AI algorithms for interactive environments.
- Develop AI-powered features that enhance player engagement and gameplay.

Course Outcomes (COs)

At the completion of the course the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Explain the role of artificial intelligence in modern game design.	Understand
CO2	Implement pathfinding and decision-making algorithms for NPC behavior..	Create
CO3	Design AI strategies using finite state machines, behavior trees, and utility systems.	Apply
CO4	Analyze and evaluate AI models for procedural content generation and adaptive gameplay.	Analyze
CO5	Develop and integrate AI-driven components into real-time game environments.	Create

CO-PO Mapping

CO / PO	P01	P02	P03	P04	P05	P06	P07	P08	P09	P010	P011	PS01	PS02
CO1	2	2	1	1	2	1	1	2	1	2	1	1	2
CO2	2	3	2	2	3	1	1	2	2	2	1	2	2
CO3	2	3	2	3	3	1	1	2	2	2	2	2	1
CO4	3	3	3	3	3	2	1	3	2	2	2	2	2
CO5	3	3	3	3	3	2	2	3	3	3	2	3	2

1 - low, 2 - medium, 3 - high

Unit I INTRODUCTION TO GAME AI

Overview of AI in games vs. traditional AI, history and evolution, types of game AI (rule-based, behavior-based), AI architecture for games, components of game loops, NPC AI design considerations, scripting vs. autonomous AI.

Unit II PATHFINDING AND MOVEMENT ALGORITHMS

Grid-based maps and navigation meshes, Dijkstra's algorithm, A* pathfinding, hierarchical pathfinding, dynamic pathfinding, steering behaviors (seek, flee, arrival, wander, flocking), obstacle avoidance, smoothing and optimization.

Unit III DECISION MAKING IN GAMES

Finite State Machines (FSM), Behavior Trees (BT), Hierarchical FSMs, Utility-based AI, Rule-based systems, GOAP (Goal-Oriented Action Planning), decision tables, influence maps, scripting AI in Unity/Unreal.

Unit IV PROCEDURAL CONTENT GENERATION AND LEARNING AI

PCG methods: random, rule-based, constraint-based, grammar-based generation of maps, levels, puzzles, and items. Adaptive AI, difficulty scaling, player modeling, intro to ML in games, Reinforcement Learning basics, Monte Carlo Tree Search, genetic algorithms.

Unit V GAME AI INTEGRATION AND CASE STUDIES

Integrating AI into game engines (Unity with C#, Unreal with Blueprints), AI performance considerations, debugging and tuning AI, multiplayer AI bots, ethical implications of AI in games, case studies of AI in commercial games (e.g., F.E.A.R., Left 4 Dead, The Sims).

SUGGESTED READINGS

1. Ian Millington, *Artificial Intelligence for Games*, CRC Press, 2018 (2nd Edition).
2. Steven Rabin, *Game AI Pro: Collected Wisdom of Game AI Professionals*, A K Peters/CRC Press, 2013.
3. Unity AI Documentation, Unity Technologies
4. Unreal Engine AI Documentation, Epic Games
5. GDC Talks on Game AI, Game Developers Conference
6. AI and Games Blog, Independent Publication

25CSU702D

WEB FRAMEWORKS

Semester –VII

3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100

End Semester Exam: 3 Hours

Course Objectives

- Introduce students to modern web frameworks for frontend and backend development.
- Enable development of dynamic, scalable, and maintainable web applications.
- Familiarize with MVC/MVVM architecture patterns used in frameworks.
- Provide hands-on experience with popular frameworks like Django, Flask, React, or Angular.
- Teach deployment, testing, and security practices in full-stack development..

Course Outcomes (COs)

At the completion of the course the student will be able to

Cos	Course Outcomes	Blooms Level
CO1	Describe the features and architecture of popular web frameworks..	Understand
CO2	Build dynamic web applications using frontend and backend frameworks.	Create
CO3	Implement routing, data binding, and form handling in web frameworks.	Create
CO4	Integrate databases, REST APIs, and third-party services in web apps.	Evaluate
CO5	Deploy, secure, and test full-stack applications in production environments.	Create

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	3	3	3	2	2	2	2	1	2	1	1	2
CO2	3	3	2	2	3	1	1	2	2	2	1	2	3
CO3	3	3	2	3	3	1	1	2	2	2	2	2	2
CO4	3	3	3	3	3	2	1	3	2	2	2	2	3
CO5	3	3	3	3	3	2	2	3	3	3	2	3	2

1 - low, 2 - medium, 3 - high

Unit I INTRODUCTION TO WEB FRAMEWORKS AND ARCHITECTURE

Web application architecture (client-server, MVC, MVVM), introduction to web frameworks, advantages, lifecycle of web applications, routing, templating, session management, RESTful design principles, introduction to full-stack development.

Unit II BACKEND WEB FRAMEWORKS

Python frameworks: Django and Flask – request handling, routing, models and ORM, templates, middleware, form handling, user authentication. Node.js/Express.js: middleware, routing, database integration (MongoDB/PostgreSQL), APIs. Environment configuration, debugging, error handling.

Unit III FRONTEND WEB FRAMEWORKS

React.js: components, props, state, hooks, conditional rendering, forms, routing, API consumption using Axios. Angular: modules, components, data binding, directives, services, dependency injection. Frontend validations, SPA (Single Page Applications).

Unit IV INTEGRATION AND FULL STACK DEVELOPMENT

Connecting frontend and backend using APIs, CORS, session management, JWT authentication, RESTful API design, database operations, CRUD functionalities, file uploads, using Postman for API testing, WebSockets (intro), version control with Git.

Unit V DEPLOYMENT, TESTING, AND SECURITY

Testing frameworks (Jest, Pytest, Postman), unit and integration testing, deployment on cloud platforms (Heroku, Vercel, Netlify, Render), containerization with Docker (intro), HTTPS, securing APIs, input validation, CSRF, XSS protection.

SUGGESTED READINGS

1. Ethan Brown, Web Development with Node and Express, O'Reilly, 2020 (2nd Edition).
2. William S. Vincent, Django for Beginners, WelcomeToCode, 2023 (5th Edition).
3. React Documentation, Meta Platforms Inc..
4. Django Documentation, Django Software Foundation.
5. Flask Documentation, Pallets Projects.
6. Angular Documentation, Google.
7. Mozilla MDN Web Documentation, Mozilla Foundation

25CSU702E

QUANTUM COMPUTING

Semester –VII

3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100

End Semester Exam: 3 Hours

Course Objectives

- Introduce fundamental concepts of quantum computation and quantum information.
- Develop understanding of qubits, quantum logic gates, and quantum circuits.
- Analyze quantum algorithms such as Grover's and Shor's.
- Explore quantum error correction and decoherence.
- Provide exposure to quantum programming platforms and applications...

Course Outcomes (COs)

At the completion of the course the student will be able to

Cos	Course Outcomes	Blooms Level
CO1	Explain the principles of quantum mechanics relevant to quantum computing.	Understand
CO2	Illustrate quantum computation using qubits, gates, and circuits..	Apply
CO3	Analyze the operation and complexity of basic quantum algorithms..	Analyze
CO4	Evaluate the challenges in quantum error correction and physical implementation.	Evaluate
CO5	Develop and simulate quantum programs using modern quantum programming tools.	Create

CO-PO Mapping

CO / PO	P01	P02	P03	P04	P05	P06	P07	P08	P09	P010	P011	PS01	PS02
CO1	3	2	1	1	2	1	1	2	1	2	1	1	3
CO2	3	3	2	2	3	1	1	2	2	2	1	2	3
CO3	3	3	2	3	3	1	1	2	2	2	2	2	3
CO4	3	3	3	3	3	2	1	3	2	2	2	2	3
CO5	3	3	3	3	3	2	2	3	3	3	2	3	3

1 - low, 2 - medium, 3 - high

Unit I FOUNDATIONS OF QUANTUM COMPUTING

Classical vs. quantum computing, postulates of quantum mechanics, Dirac notation, quantum bits (qubits), quantum superposition and entanglement, quantum measurements, Bloch sphere representation.

Unit II QUANTUM GATES AND CIRCUITS

Quantum logic gates: X, Z, H, T, CNOT, Toffoli, SWAP; universal quantum gates, quantum circuit diagrams, quantum parallelism, tensor product and matrix representation of multi-qubit systems, simulation of basic quantum circuits.

Unit III QUANTUM ALGORITHMS

Deutsch-Jozsa algorithm, Grover's search algorithm, Simon's algorithm, Shor's factoring algorithm, quantum Fourier transform (QFT), amplitude amplification and estimation, complexity classes: BPP, BQP.

Unit IV QUANTUM ERROR CORRECTION AND DECOHERENCE

Sources of noise and decoherence, quantum error correction basics (bit-flip, phase-flip, Shor code), fault tolerance, no-cloning theorem, quantum repeaters, physical implementation challenges (superconducting qubits, trapped ions).

Unit V QUANTUM PROGRAMMING AND APPLICATIONS

Quantum programming environments: IBM Qiskit, Microsoft Q#, Google Cirq, writing and executing quantum programs, simulation vs. real quantum hardware, quantum cryptography, quantum key distribution (QKD), current applications and future directions.

SUGGESTED READINGS

1. Michael A. Nielsen, Isaac L. Chuang, Quantum Computation and Quantum Information, Cambridge University Press, 10th Anniversary Edition, 2010 (original edition published in 2000).
2. Nikita Gourianov, Quantum Computing for the Very Curious, Quantum Country, 2020
3. Qiskit Textbook – <https://qiskit.org/learn> – IBM Quantum, ongoing (since ~2019)
4. Microsoft Q# Documentation – <https://learn.microsoft.com/en-us/azure/quantum> – Microsoft, ongoing (since ~2018)
5. Google Cirq – <https://quantumai.google/cirq> – Google Quantum AI, ongoing (since ~2018)
6. MIT OpenCourseWare – Quantum Computation – <https://ocw.mit.edu> – Latest version available from ~2018 onward
7. IBM Quantum Experience – <https://quantum-computing.ibm.com> – IBM, ongoing (launched in 2016)

OPEN ELECTIVES

B.Tech. Computer Science and Engineering

2025-2026

25CSU041

PROGRAMMING IN CORE JAVA

Semester –
3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100

End Semester Exam: 3 Hours

Course Objectives

1. Introduce the core concepts of Java programming language.
2. Develop object-oriented programming skills using Java.
3. Implement Java programs using classes, interfaces, and exception handling.
4. Understand file handling, collections, and GUI development.
5. Familiarize with multi-threading and basic Java networking.

Course Outcomes (COs)

At the completion of the course the student will be able to

Cos	Course Outcomes	Blooms Level
CO1	Describe Java syntax, data types, and control structures.	Understand
CO2	Apply object-oriented concepts like classes, inheritance, and polymorphism.	Apply
CO3	Implement exception handling, interfaces, and file I/O in Java applications.	Create
CO4	Develop Java programs using collections, GUI, and multithreading.	Create
CO5	Demonstrate basic network programming using Java sockets.	Create

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	1	1	2	1	1	2	1	2	1	1	2
CO2	3	3	2	2	3	1	1	2	2	2	1	2	2
CO3	3	3	2	3	3	1	1	2	2	2	2	2	2
CO4	3	3	3	3	3	2	1	3	2	2	2	2	2
CO5	3	3	3	3	3	2	2	3	3	3	2	3	2

1 - low, 2 - medium, 3 - high

Unit I INTRODUCTION TO JAVA PROGRAMMING

Java features, JDK and JRE, structure of a Java program, variables, data types, operators, control flow (if-else, loops, switch), type casting, input/output using Scanner and System.out.

Unit II OBJECT-ORIENTED PROGRAMMING IN JAVA

Classes and objects, constructors, method overloading, static members, inheritance, method overriding, super and this keywords, abstract classes, interfaces, polymorphism, final keyword.

Unit III EXCEPTION HANDLING AND FILE I/O

Types of exceptions, try-catch-finally, multiple catch blocks, throws and throw, custom exceptions, file handling using FileReader, FileWriter, BufferedReader, BufferedWriter, serialization basics.

Unit IV JAVA COLLECTIONS AND GUI PROGRAMMING

Collections framework: ArrayList, LinkedList, HashMap, HashSet, iterators, generics. GUI basics using Swing: JFrame, JPanel, JButton, JLabel, layout managers, event handling with ActionListener.

Unit V MULTITHREADING AND NETWORKING

Creating threads using Thread class and Runnable interface, thread lifecycle, synchronization, inter-thread communication, networking basics, socket programming with ServerSocket and Socket, simple client-server applications.

SUGGESTED READINGS

1. Herbert Schildt, Java: The Complete Reference, McGraw-Hill Education, 12th Edition, 2024.
2. E. Balagurusamy, Programming with Java, McGraw-Hill Education, 6th Edition, 2019.
3. Oracle Java Docs – <https://docs.oracle.com/javase/>
4. Java Tutorials by Oracle – <https://docs.oracle.com/javase/tutorial/>
5. GeeksforGeeks Java Programming – <https://www.geeksforgeeks.org/java/>

25CSU042

WEB DESIGNING TECHNOLOGIES

Semester –
3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- Introduce the fundamentals of web design and development.
- Enable students to create responsive and accessible web pages using HTML, CSS, and JavaScript.
- Familiarize students with modern frontend design practices and frameworks.
- Develop client-side interactivity and validation techniques.
- Expose students to web hosting, SEO, and performance optimization.

Course Outcomes (COs)

At the completion of the course the student will be able to

Cos	Course Outcomes	Blooms Level
CO1	Understand the structure and components of web pages.	Understand
CO2	Design static and responsive websites using HTML and CSS..	Apply
CO3	Develop interactive web pages using JavaScript and DOM manipulation.	Create
CO4	Use web design tools and frameworks for modern frontend development.	Create
CO5	Demonstrate website deployment, performance, and accessibility features.	Create

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	1	1	2	1	1	2	1	2	1	1	2
CO2	3	3	2	2	3	1	1	2	2	2	1	2	3
CO3	3	3	2	3	3	1	1	2	2	2	2	2	2
CO4	3	3	3	3	3	2	1	3	2	2	2	2	3
CO5	3	3	3	3	3	2	2	3	3	3	2	3	2

1 - low, 2 - medium, 3 - high

Unit I INTRODUCTION TO WEB DESIGN AND HTML

Basics of the internet and web, WWW, URL, HTTP/HTTPS, browsers, structure of HTML document, HTML5 semantic tags, forms, multimedia (audio/video), tables, hyperlinks, iframes.

Unit II STYLING WITH CSS

Types of CSS (inline, internal, external), selectors, box model, positioning, flexbox, grid layout, transitions and animations, responsive design using media queries, CSS3 features, variables and pseudo-classes.

Unit III JAVASCRIPT FOR INTERACTIVITY

JavaScript syntax and data types, functions, events, DOM manipulation, form validation, timers, arrays and objects, JSON handling, basic ES6 concepts (let, const, arrow functions, template literals).

Unit IV WEB DESIGN TOOLS AND FRAMEWORKS

Bootstrap basics (components, grid system, utilities), font icons, using Google Fonts, introduction to UI kits, using templates, introduction to version control with GitHub, intro to web design platforms like Figma and Canva (for mockups).

Unit V DEPLOYMENT, OPTIMIZATION, AND BEST PRACTICES

Web hosting (free and paid options), domain registration, FTP, introduction to SEO, image and script optimization, accessibility (WCAG guidelines), cross-browser compatibility, progressive enhancement.

SUGGESTED READINGS

1. Jon Duckett, HTML and CSS: Design and Build Websites, Wiley, 1st Edition, 2011.
2. Ethan Brown, Web Development with Node and Express: Leveraging the JavaScript Stack, O'Reilly Media, 2nd Edition, 2019.
3. Mozilla Developer Network (MDN) – <https://developer.mozilla.org>
4. W3Schools – <https://www.w3schools.com>
5. FreeCodeCamp Web Design Certification – <https://www.freecodecamp.org>

25CSU043

MOBILE COMPUTING TECHNOLOGIES

Semester –
3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours

Course Objectives

- Introduce the principles of mobile computing and wireless communication.
- Explore the architecture, protocols, and standards for mobile networks.
- Explain mobile application platforms and development environments.
- Understand mobile data management, synchronization, and cloud integration.
- Address security, privacy, and emerging trends in mobile computing..

Course Outcomes (COs)

At the completion of the course the student will be able to

Cos	Course Outcomes	Blooms Level
CO1	Understand the structure and components of web pages.	Understand
CO2	Design static and responsive websites using HTML and CSS..	Apply
CO3	Develop interactive web pages using JavaScript and DOM manipulation.	Create
CO4	Use web design tools and frameworks for modern frontend development.	Create
CO5	Demonstrate website deployment, performance, and accessibility features.	Create

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	1	1	2	1	1	2	1	2	1	1	2
CO2	3	3	2	2	3	1	1	2	2	2	1	2	3
CO3	3	3	2	3	3	1	1	2	2	2	2	2	2
CO4	3	3	3	3	3	2	1	3	2	2	2	2	2
CO5	3	3	3	3	3	2	2	3	3	3	2	3	3

1 - low, 2 - medium, 3 - high

Unit I INTRODUCTION TO MOBILE COMPUTING

Definition and scope, mobile and wireless devices, mobile computing architecture, mobile communication models, types of mobility, characteristics and limitations, cellular concepts, 1G to 5G evolution.

Unit II MOBILE COMMUNICATION TECHNOLOGIES

GSM, GPRS, CDMA, UMTS, LTE/4G/5G overview, mobile IP and mobility management, handoff and roaming, wireless LAN (802.11), Bluetooth, ZigBee, mobile TCP/IP, routing in mobile ad hoc networks (MANETs).

Unit III MOBILE APPLICATION DEVELOPMENT

Mobile OS overview (Android, iOS, HarmonyOS), Android architecture, activities, intents, UI design, storage options (SQLite, shared preferences), permissions, services, sensors and location APIs, introduction to React Native and Flutter.

Unit IV MOBILE DATA MANAGEMENT AND CLOUD INTEGRATION

Data synchronization types (push/pull), mobile databases, offline storage, syncing with remote servers, RESTful APIs and JSON, integration with cloud services (Firebase, AWS, Azure), content providers and broadcast receivers.

Unit V MOBILE SECURITY AND EMERGING TRENDS

Threats and attacks on mobile systems, authentication, encryption, app security best practices, securing communications, MDM (Mobile Device Management), BYOD policies, IoT integration, edge computing, mobile AI applications, AR/VR in mobile devices.

SUGGESTED READINGS

1. Asoke K. Talukder, Roopa R. Yavagal, Hasan Ahmed, Mobile Computing: Technology, Applications, and Service Creation, McGraw-Hill Education, 2nd Edition, 2010.
2. Jochen Schiller, Mobile Communications, Pearson Education, 2nd Edition, 2003.
3. Reto Meier, Ian Lake, Professional Android, Wrox/Wiley, 4th Edition, 2018.
4. Bill Phillips, Chris Stewart, Kristin Marsicano, Android Programming: The Big Nerd Ranch Guide, Big Nerd Ranch, 4th Edition, 2019.
5. David Flanagan, JavaScript: The Definitive Guide, O'Reilly Media, 7th Edition, 2020 (for understanding cross-platform frameworks like React Native).
6. Firebase Documentation – <https://firebase.google.com/docs>
7. Android Developers Guide – <https://developer.android.com/guide>

25CSU044

E-COMMERCE

Semester –
3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours**Course Objectives**

- Provide a comprehensive understanding of the concepts, models, and strategies involved in electronic commerce.
- Introduce students to the technologies and infrastructure that enable e-commerce operations.
- Equip learners with the skills to design and manage e-commerce websites and platforms.
- Familiarize students with digital marketing techniques and customer relationship strategies.
- Highlight legal, ethical, and security issues related to conducting business online, and explore emerging e-commerce trends.

Course Outcomes (COs)

At the completion of the course the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Explain the foundational concepts, types, and business models of e-commerce.	Understand
CO2	Analyze the technological components that support e-commerce systems, including payment and network infrastructure.	Analyze
CO3	Design and develop user-friendly and secure e-commerce websites using suitable platforms and tools..	Create
CO4	Apply digital marketing techniques and tools to improve online business performance and manage customer relations.	Apply
CO5	Evaluate legal, ethical, and security aspects of e-commerce and identify the impact of emerging technologies on the digital marketplace.	Evaluate

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PS01	PS02
CO1	2	2	1	1	2	1	2	2	1	2	1	1	2
CO2	2	3	2	2	3	1	2	2	2	2	1	2	2
CO3	2	2	2	3	3	1	2	2	2	2	2	2	2
CO4	2	3	3	3	3	2	2	3	2	2	2	2	2
CO5	2	2	3	3	3	2	2	3	3	3	2	3	2

1 - low, 2 - medium, 3 - high

Unit I INTRODUCTION TO E-COMMERCE

Definition, scope, and evolution of e-commerce, traditional commerce vs. e-commerce, types of e-commerce (B2B, B2C, C2C, C2B, G2C), benefits and limitations, e-commerce business models, value chains in e-commerce, emerging trends in e-commerce.

Unit II E-COMMERCE INFRASTRUCTURE AND TECHNOLOGIES

Internet and WWW as enablers of e-commerce, network infrastructure, web technologies (HTML, CSS, JavaScript), web servers and hosting, electronic payment systems (credit/debit cards, digital wallets, UPI, cryptocurrencies), payment gateways, and e-commerce software platforms.

Unit III WEBSITE DESIGN AND DEVELOPMENT FOR E-COMMERCE

E-commerce website planning and structure, design principles and UI/UX considerations, responsive design, content management systems (CMS), e-commerce platforms (Shopify, Magento, WooCommerce), shopping cart functionality, product catalog and inventory management, secure transactions using SSL/TLS.

Unit IV DIGITAL MARKETING AND CUSTOMER RELATIONSHIP

Introduction to digital marketing, SEO, SEM, email marketing, social media marketing, affiliate marketing, content marketing, analytics and performance measurement using Google Analytics, CRM systems and strategies, personalization and customer engagement.

Unit V LEGAL, SECURITY, AND FUTURE TRENDS IN E-COMMERCE

E-commerce laws and regulations (IT Act, GDPR, cyber laws), intellectual property rights, online contracts, taxation and compliance, cybersecurity issues in e-commerce, authentication and encryption, fraud detection, ethical issues, emerging trends – mobile commerce, voice commerce, augmented reality (AR) in e-commerce, and AI-driven e-commerce.

SUGGESTED READINGS

1. Kenneth C. Laudon, Carol Guercio Traver, E-Commerce: Business, Technology, Society, Pearson Education, 16th Edition, 2020.
2. David Whiteley, E-Commerce: Strategy, Technologies and Applications, McGraw-Hill Education, 1st Edition, 2000.
3. P. T. Joseph, E-Commerce: An Indian Perspective, PHI Learning, 5th Edition, 2015.
4. Gary Schneider, Electronic Commerce, Cengage Learning, 12th Edition, 2016.
5. Chaffey, Dave, Digital Business and E-Commerce Management, Pearson Education, 6th Edition, 2015.
6. Digital Marketing Guide – <https://support.google.com/google-ads/>
7. Shopify Help Center – <https://help.shopify.com>
8. Magento Documentation – <https://developer.adobe.com/commerce>

25CSU045

APPLICATIONS OF IMMERSIVE TECHNOLOGIES

Semester –
3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours

Course Objectives

- Introduce students to the foundations and evolution of immersive technologies, including VR, AR, MR, and XR.
- Familiarize learners with the hardware and software ecosystems essential for developing immersive applications.
- Explore real-world applications and transformative potential of immersive technologies across multiple industries.
- Develop knowledge of design principles and content creation strategies for user-centered immersive experiences.
- Address challenges, ethical issues, and emerging trends shaping the future of immersive technologies.

Course Outcomes (COs)

At the completion of the course the student will be able to

COs	Course Outcomes	Blooms Level
CO1	Understand and distinguish between VR, AR, MR, and XR concepts, tools, and technologies.	Understand
CO2	Demonstrate familiarity with major immersive hardware devices and software platforms such as Unity, Unreal Engine, and WebXR.	Understand
CO3	Analyze and evaluate immersive technology applications in domains like education, healthcare, retail, industry, and entertainment.	Analyze
CO4	Apply user-centered design principles to create effective and engaging immersive content and experiences.	Apply
CO5	Assess current limitations, ethical implications, and future directions in the immersive technology landscape, including the Metaverse and AI integration.	Analyze

CO-PO Mapping

CO / PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	1	1	2	1	1	2	1	2	1	1	2
CO2	2	3	2	2	3	1	1	2	2	2	1	2	2
CO3	2	2	2	3	3	1	1	2	2	2	2	2	2
CO4	2	3	3	3	3	2	1	3	2	2	2	2	2
CO5	3	3	3	3	3	2	2	3	3	3	2	3	2

1 - low, 2 - medium, 3 - high

Unit I INTRODUCTION TO IMMERSIVE TECHNOLOGIES

Definition and evolution of immersive technologies, components and taxonomy: Virtual Reality (VR), Augmented Reality (AR), Mixed Reality (MR), and Extended Reality (XR). Key concepts: immersion, presence, interactivity, and user experience. Historical development and milestones in immersive tech.

Unit II HARDWARE AND SOFTWARE ECOSYSTEM

Immersive hardware: head-mounted displays (HMDs), smart glasses, motion controllers, haptic devices, omnidirectional treadmills. Software frameworks and platforms: Unity, Unreal Engine, ARCore, ARKit, WebXR, OpenXR. Devices and operating systems for VR/AR: Oculus, HTC Vive, HoloLens, Magic Leap, and mobile-based systems.

Unit III APPLICATION DOMAINS AND USE CASES

Immersive technology applications in education and training (virtual classrooms, simulation-based learning), healthcare (VR therapy, surgical simulation), retail and e-commerce (virtual try-ons, AR shopping), manufacturing and industry (digital twins, remote assistance), real estate and architecture (virtual walkthroughs), tourism, and entertainment (gaming, interactive storytelling).

Unit IV DESIGN PRINCIPLES AND CONTENT DEVELOPMENT

Human-centered design for immersive environments, 3D modeling and asset creation, interaction models and user navigation, storytelling in XR, spatial audio design, performance optimization for immersive experiences. User interface (UI) and user experience (UX) challenges and best practices.

Unit V CHALLENGES, ETHICS, AND FUTURE TRENDS

Technical challenges: latency, motion sickness, resolution, device limitations. Ethical and social implications: privacy, data collection, psychological effects. Accessibility and inclusivity in immersive systems. Future trends: AI integration, Metaverse, brain-computer interfaces (BCI), digital humans, and the convergence of immersive technologies with IoT and 5G.

SUGGESTED READINGS

1. Alan B. Craig, *Understanding Augmented Reality: Concepts and Applications*, Morgan Kaufmann, 2nd Edition, 2018.
2. Jason Jerald, *The VR Book: Human-Centered Design for Virtual Reality*, Morgan & Claypool, 1st Edition, 2015.
3. Chetankumar G. Shetty, *Augmented Reality and Virtual Reality: The Power of AR and VR for Business*, BPB Publications, 1st Edition, 2020.
4. Tony Parisi, *Learning Virtual Reality: Developing Immersive Experiences and Applications for Desktop, Web, and Mobile*, O'Reilly Media, 1st Edition, 2015.
5. Unity Documentation – <https://docs.unity3d.com>
6. Unreal Engine Docs – <https://docs.unrealengine.com>
7. WebXR and OpenXR Resources – <https://immersiveweb.d>