

**ACADEMIC REGULATIONS,
COURSE STRUCTURE
AND
DETAILED SYLLABUS**

(I To IV Semesters)



COMPUTER SCIENCE AND ENGINEERING

**For
B.TECH. FOUR YEAR DEGREE COURSE**

(Applicable for batches admitted from 2023-2024)



**SWARNANDHRA
COLLEGE OF ENGINEERING & TECHNOLOGY**

(AUTONOMOUS)

SEETHARAMAPURAM, NARSAPUR-534 280, W.G.DT., A.P.

INSTITUTION VISION AND MISSION

VISION

To produce global competent, ethical and dynamic professionals by creating Centre of Excellence in Technical Education for societal empowerment.

MISSION

M1: To provide quality education with knowledge and skills for rural and urban students.

M2: To collaborate the industries with academia for empowering the students to meet global standards.

M3: To induce highly ethical entrepreneurship in young minds with good leadership quality for the society.

M4: To enhance the institution in Research and Development by human intellectual capability.

DEPARTMENT VISION AND MISSION

VISION

To educate and enrich effective and responsible Mechanical Engineers to fulfil the needs of industry and society.

MISSION

M1 To lay a strong foundation of technical knowledge by concentrating on fundamental concepts of Mechanical engineering.

M2 To develop creative thinking and innovative methods for solving complex engineering problems.

M3 To develop team spirit, leadership and professional qualities.

M4 To strengthen research abilities in collaboration with industry.

PROGRAM OUTCOMES (POs)

Engineering Graduates will be able to:

- PO1. Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
- PO2. Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
- PO3. Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
- PO4. Conduct investigations of complex problems:** Use researchbased knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
- PO5. Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.
- PO6. The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
- PO7. Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
- PO8. Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
- PO9. Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
- PO10. Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
- PO11. Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
- PO12. Lifelong learning:** Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change.

Academic Regulations (R23) for B. Tech (Regular-Full time)

(Effective for the students admitted into I year from the Academic Year **2023-24** onwards)

1. Award of the Degree

(a) Award of the B.Tech. Degree if he/she fulfils the following:

- I. Pursues a course of study for not less than four academic years and not more than eight academic years. However, for the students availing Gap year facility this period shall be extended by two years at the most and these two years would in addition to the maximum period permitted for graduation (Eight years).
- II. Registers for 160 credits and secures all 160 credits.

(b) **Award of B.Tech. degree with Honors** if he/she fulfils the following:

- I. Student secures additional 15 credits fulfilling all the requisites of a B.Tech. programme i.e., 160 credits.
- II. Registering for Honors is optional.
- III. Honors is to be completed simultaneously with B.Tech. programme.

2. Students, who fail to fulfil all the academic requirements for the award of the degree within eight academic years from the year of their admission, shall forfeit their seat in B.Tech. course and their admission stands cancelled. This clause shall be read along with clause 1 a) i).

3. Admissions

Admission to the B. Tech Program shall be made subject to the eligibility, qualifications and specialization prescribed by the A.P. State Government/University from time to time. Admissions shall be made either based on the merit rank obtained by the student in the common entrance examination conducted by the A.P. Government/University or any other order of merit approved by the A.P. Government/University, subject to reservations as prescribed by the Government/University from time to time.

3.1 Regular Admission

(Join in first year B. Tech Programme)

Admissions in the Institution are classified into **CATEGORY – A**, through convener, EAPCET and **CATEGORY B** filled by the college management.

3.2 Lateral Entry Admission

(Join in the Second year/third semester of B. Tech Programme)

Eligibility: Diploma in Engineering / B.Sc Degree with Mathematics as one course .

Based on the rank secured by the candidate at Engineering Common Entrance Test (ECET) conducted by APSCE, Government of Andhra Pradesh.

3.3 Advance standing Admission

(Transfer from other Colleges/ Readmission due to discontinuation)

These may arise in the following cases:

3.3.1 When a student seeks transfer from other colleges to SCET and desirous to pursue the study at SCET in an eligible branch of study.

3.3.2 When students of SCET get transferred from one regulation to another regulation.

In all such cases, approval is mandatory from the statutory bodies

3.4 B. Tech Programmes offered

The College is offering the following programmes:

- Computer Science and Engineering (CSE)
- Electronics and communication Engineering (ECE)
- Electrical and Electronics Engineering (EEE)
- Information Technology (IT)
- Mechanical Engineering (ME)
- Civil Engineering (CE)
- Artificial Intelligence and Machine Learning (AI&ML)
- Robotics (ROBO)
- Computer Science and Engineering (Data Science)
- Computer Science and Engineering (Cyber Security)
- Computer Science and Engineering and Business Systems
- Artificial Intelligence (AI) and Data Science

4. Program related terms

Credit: A unit by which the course work is measured. It determines the number of hours of instruction required per week. One credit is equivalent to one hour of teaching (Lecture/Tutorial) or two hours of practical work/field work per week.

Credit Definition:

1 Hr. Lecture (L) per week	1 credit
1 Hr. Tutorial (T) per week	1 credit
1 Hr. Practical (P) per week	0.5 credit
2 Hrs. Practical (Lab) per week	1 credit

- a) **Academic Year:** Two consecutive (one odd + one even) semesters constitute one academic year.
- b) **Choice Based Credit System (CBCS):** The CBCS provides a choice for students to select from the prescribed courses.

5. Semester/Credits:

- i) A semester comprises 90 working days and an academic year is divided into two semesters.
- ii) The summer term is for eight weeks during summer vacation. Internship/ apprenticeship / work-based vocational education and training can be carried out during the summer term, especially by students who wish to exit after two semesters or four semesters of study.
- iii) Regular courses may also be completed well in advance through MOOCs satisfying prerequisites.

6. Structure of the Undergraduate Programme

All courses offered for the undergraduate programme (B. Tech.) are broadly classified as follows:

S.No.	Category
1.	Humanities and Social Science including Management (HM)
2.	Basic Sciences (BS)
3.	Engineering Sciences (ES)
4.	Professional Core (PC)
5.	Electives – Professional (PE) & Open (OE); Skill Enhancement Courses (SEC)
6.	Internships & Project work (PR)
7.	Mandatory Courses (MC)

7. Course Classification:

All courses offered for the undergraduate programme in Engineering & Technology (B.Tech. degree programmes) are broadly classified as follows:

S.No.	Broad Course Classification	Course Category	Description
1.	Foundation Courses	Foundation courses	Includes Mathematics, Physics and Chemistry; fundamental engineering courses; humanities, social sciences and management courses
2.	Core Courses	Professional Core Courses (PC)	Includes subjects related to the parent discipline/department/branch of Engineering
3.	Elective Courses	Professional Elective Courses (PE)	Includes elective subjects related to the parent discipline/department/ branch of Engineering
		Open Elective Courses (OE)	Elective subjects which include interdisciplinary subjects or subjects in an area outside the parent discipline/ department/ branch of Engineering
		Skill enhancement courses (SEC)	interdisciplinary/job-oriented/domain courses which are relevant to the industry
4.	Project & Internships	Project	B.Tech. Project or Major Project
		Internships	Summer Internships – Community based and Industry Internships; Industry oriented Full Semester Internship
5.	Audit Courses	Mandatory non-credit courses	Covering subjects of developing desired attitude among the learners

8. Programme Pattern

- i. Total duration of the of B. Tech (Regular) Programme is four academic years.
- ii. Each academic year of study is divided into two semesters.
- iii. Minimum number of instruction days in each semester is 90 days.
- iv. There shall be mandatory student induction program for freshers, with a three weeks duration before the commencement of first semester. Physical activity, Creative Arts, Universal Human Values, Literary, Proficiency Modules, Lectures by Eminent People, Visits to local Areas,

- Familiarization to Dept./Branch & Innovations etc., are included as per the guidelines issued by AICTE
- v. E.
 - v. Health/wellness/yoga/sports and NSS /NCC /Scouts & Guides / Community service activities are made mandatory as credit courses for all the undergraduate students.
 - vi. Courses like Environmental Sciences, Indian Constitution, Technical Paper Writing & IPR are offered as non-credit mandatory courses for all the undergraduate students.
 - vii. Design Thinking for Innovation & Tinkering Labs are made mandatory as credit courses for all the undergraduate students.
 - viii. Increased flexibility for students through an increase in the elective component of the curriculum, with 05 Professional Elective courses and 04 Open Elective courses.
 - ix. Professional Elective Courses, include the elective courses relevant to the chosen specialization/branch. Proper choice of professional elective courses can lead to students specializing in emerging areas within the chosen field of study.
 - x. While choosing the electives, students shall ensure that they do not opt for the courses with syllabus contents similar to courses already pursued.
 - xi. A pool of interdisciplinary/job-oriented/domain skill courses which are relevant to the industry are integrated into the curriculum of all disciplines. There shall be 05 skill-oriented courses offered during III to VII semesters. Among the five skill courses, four courses shall focus on the basic and advanced skills related to the domain/interdisciplinary courses and the other shall be a soft skills course.
 - xii. Students shall undergo mandatory summer internships, for a minimum of eight weeks duration at the end of second and third year of the programme. The internship at the end of second year shall be community oriented and industry internship at the end of third year.
 - xiii. There shall also be mandatory full internship in the final semester of the programme along with the project work.
 - xiv. Undergraduate degree with Honors is introduced for the students having good academic record.
 - xv. College takes measures to implement Virtual Labs (<https://www.vlab.co.in>) which provide remote access to labs in various disciplines of Engineering and will help student in learning basic and advanced concept through remote experimentation. Student shall be made to work on virtual lab experiments during the regular labs.
 - xvi. College will assign a faculty advisor/mentor after admission to a group of students from same department to provide guidance in courses registration/ career growth/placements/opportunities for higher studies/GATE/other competitive exams etc.
 - xvii. Preferably 25% of course work for the theory courses in every semester shall be conducted in the blended mode of learning.

9. Evaluation Process

The performance of a student in each semester shall be evaluated subject wise with a maximum of 100 marks for theory and 100 marks for practical courses. Summer Internships shall be evaluated for 50 marks, Full Internship & Project work in final semester shall be evaluated for 200 marks, mandatory courses with no credits shall be evaluated for 30 mid semester marks.

A student has to secure not less than 35% of marks in the end examination and a minimum of 40% of marks in the sum total of the mid semester and end examination marks taken together for the theory, practical, design, drawing subject or project etc. In case of a mandatory course, he/she should secure 40% of the total marks.

Theory Courses

Assessment Method	Marks
Continuous Internal Assessment	30
Semester End Examination	70
Total	100

- i) For theory subject, the distribution shall be 30 marks for Internal Evaluation and 70 marks for the End-Examination.
- ii) For practical subject, the distribution shall be 30 marks for Internal Evaluation and 70 marks for the End- Examination.
- iii) If any course contains two different branch subjects, the syllabus shall be written in two parts with 3 units each (Part-A and Part-B) and external examination question paper shall be set with two parts each for 35 marks.
- iv) If any subject is having both theory and practical components, they will be evaluated separately as theory subject and practical subject. However, they will be given same subject code with an extension of 'T' for theory subject and 'L' for practical subject.

(a) Continuous Internal Evaluation

- i) For theory subjects, during the semester, there shall be two midterm examinations. Each midterm examination shall be evaluated for 30 marks of which 10 marks for objective paper (20 minutes duration), 15 marks for subjective paper (90 minutes duration) and 5 marks for assignment.
- ii) Objective paper shall contain for 05 short answer questions with 2 marks each or maximum of 20 bits for 10 marks. Subjective paper shall contain 3 either or type questions (totally six questions from 1 to 6) of which student has to answer one from each either-or type of questions. Each question carries 10 marks. The marks obtained in the subjective paper are condensed to 15 marks.

Note:

- The objective paper shall be prepared in line with the quality of competitive examinations questions.
 - The subjective paper shall contain 3 either or type questions of equal weightage of 10 marks. Any fraction shall be rounded off to the next higher mark.
 - Assignments shall be in the form of problems, mini projects, design problems, slip tests, quizzes etc., depending on the course content. It should be continuous assessment throughout the semester and the average marks shall be considered.
- iii) If the student is absent for the mid semester examination, no re-exam shall be conducted and mid semester marks for that examination shall be considered as zero.
 - iv) First midterm examination shall be conducted for I, II units of syllabus with one either or type question from each unit and third either or type question from both the units. The second midterm examination shall be conducted for III, IV and V units with one either or type question from each unit.

- v) Final mid semester marks shall be arrived at by considering the marks secured by the student in both the mid examinations with 80% weightage given to the better mid exam and 20% to the other.

For Example:

Marks obtained in first mid: Absent

Marks obtained in second mid: 25

Final mid semester Marks: $(25 \times 0.8) + (0 \times 0.2) = 20$

b) End Examination Evaluation:

The **end semester** examination will be conducted for 70 marks which covers full syllabus. In end examination pattern, **Part–A** consists of five short questions from all units (Brainstorming/Thought provoking/Case study) for 10 marks. **Part – B** has **5 questions** with internal choice from each unit and valued for 60 marks.

End examination of theory courses consisting of two parts of different subjects, for Example: Basic Electrical & Electronics Engineering shall have the following pattern:

- i) Question paper shall be in two parts viz., Part A and Part B with equal weightage of 35 marks each.
- ii) In each part, question 1 shall contain 5 compulsory short answer questions for a total of 5 marks such that each question carries 1 mark.
- iii) In each part, questions from 2 to 7, there shall be either/or type questions of 10 marks each. Student shall answer any one of them.
- iv) The questions from 2 to 7 shall be set by covering one unit of the syllabus for each question.

Practical Courses

Assessment Method	Marks
Continuous Internal Assessment	30
Semester End Examination	70
Total	100

- i. For practical courses, there shall be a continuous evaluation during the semester for 30 sessional marks and end examination shall be for 70 marks.
- ii. Day-to-day work in the laboratory shall be evaluated for 15 marks by the concerned laboratory teacher based on the record/viva and 15 marks for the internal test.
- iii. The end examination shall be evaluated for 70 marks, conducted by the concerned laboratory teacher and a senior expert in the subject from the same department or expert from other institutions.
 - Procedure: 20 marks
 - Experimental work & Results: 30 marks
 - Viva voce: 20 marks.
- iv. In a practical subject consisting of two parts (Eg: Basic Electrical & Electronics Engineering Lab), the end examination shall be conducted for 70 marks as a single laboratory in 3 hours. Mid semester examination shall be evaluated as above for 30 marks in each part and final mid semester marks shall be arrived by considering the average of marks obtained in two parts.

- c) For the subject having design and/or drawing, such as Engineering Drawing, the distribution of marks shall be 30 for mid semester evaluation and 70 for end examination.

Assessment Method	Marks
Continuous Internal Assessment	30
Semester End Examination	70
Total	100

- i. Day-to-day work shall be evaluated for 15 marks by the concerned subject teacher based on the reports/submissions prepared in the class. And there shall be two midterm examinations in a semester for duration of 2 hours each for 15 marks with weightage of 80% to better mid marks and 20% for the other. The subjective paper shall contain 3 either or type questions of equal weightage of 5 marks. There shall be no objective paper in mid semester examination. The sum of day-to-day evaluation and the mid semester marks will be the final sessional marks for the subject.
- ii. The end examination pattern for Engineering Graphics, shall consists of 5 questions, either/or type, of 14 marks each. There shall be no objective type questions in the end examination. However, the end examination pattern for other subjects related to design/drawing, multiple branches, etc is mentioned along with the syllabus.
- d) There shall be no external examination for mandatory courses with zero credits. However, attendance shall be considered while calculating aggregate attendance and student shall be declared to have passed the mandatory course only when he/she secures 40% or more in the internal examinations. In case, the student fails, a reexamination shall be conducted for failed candidates for 30 marks satisfying the conditions mentioned in item 1 & 2 of the regulations.
- e) The laboratory records and mid semester test papers shall be preserved for a minimum of 3 years in the respective institutions as per the University norms and shall be produced to the Committees of the University as and when the same are asked for.

10. Skill oriented Courses

- i) There shall be five skill-oriented courses offered during III to VII semesters.
- ii) Out of the five skill courses two shall be skill-oriented courses from the same domain. Of the remaining three skill courses, one shall be a soft skill course and the remaining two shall be skill-advanced courses from the same domain/Interdisciplinary/Job oriented.
- iii) The course shall carry 100 marks and shall be evaluated through continuous assessments during the semester for 30 sessional marks and end examination shall be for 70 marks. Day-to-day work in the class / laboratory shall be evaluated for 30 marks by the concerned teacher based on the regularity/assignments/viva/mid semester test. The end examination similar to practical examination pattern shall be conducted by the senior expert in the subject from the same department or expert from other institutions.
- iv) The Head of the Department shall identify a faculty member as coordinator for the course. A committee consisting of the Head of the Department, coordinator and a senior Faculty member

nominated by the Head of the Department shall monitor the evaluation process. The marks/grades shall be assigned to the students by the above committee based on their performance.

- v) The student shall be given an option to choose either the skill courses being offered by the Department or to choose a certificate course being offered by industries/Professional bodies or any other accredited bodies. If a student chooses to take a Certificate Course offered by external agencies, the credits shall be awarded to the student upon producing the Course Completion Certificate from the agency. A committee shall be formed at the level of the Department to evaluate the grades/marks given for a course by external agencies and convert to the equivalent marks/grades.
- vi) The recommended courses offered by external agencies, conversions and appropriate grades/marks are to be approved by the BOS at the beginning of the semester. The principal of the respective college shall forward such proposals to the University for approval.
- vii) If a student prefers to take a certificate course offered by external agency, the department shall mark attendance of the student for the remaining courses in that semester excluding the skill course in all the calculations of mandatory attendance requirements upon producing a valid certificate as approved by the Department.

11. Massive Open Online Courses (MOOCs):

- a. A Student has to pursue and complete one course compulsorily through MOOCs approved by the University. A student can pursue courses other than core through MOOCs and it is mandatory to complete one course successfully through MOOCs for awarding the degree. A student is not permitted to register and pursue core courses through MOOCs.
- b. A student shall register for the course (Minimum of either 8 weeks or 12 weeks) offered through MOOCs with the approval of Head of the Department. The Head of the Department shall appoint one mentor to monitor the student's progression. The student needs to earn a certificate by passing the exam. The student shall be awarded the credits assigned in the curriculum only by submission of the certificate. Examination fee, if any, will be borne by the student.
- c. Students who have qualified in the proctored examinations conducted through MOOCs platform can apply for credit transfer as specified and are exempted from appearing internal as well as external examination (for the specified equivalent credit course only) conducted by the College.
- d. Necessary amendments in rules and regulations regarding adoption of MOOC courses would be proposed from time to time.

12. Credit Transfer Policy

Adoption of MOOCs is mandatory, to enable Blended model of teaching-learning as also envisaged in the NEP 2020. As per University Grants Commission (Credit Framework for Online Learning Courses through SWAYAM) Regulation, 2016, the College shall allow up to a maximum of 20% of the total courses being offered in a particular programme i.e., maximum of 32 credits through MOOCs platform.

- i) The Institution shall offer credit mobility for MOOCs and give the equivalent credit weightage to the students for the credits earned through online learning courses.
- ii) Student registration for the MOOCs shall be only through the respective department of the institution, it is mandatory for the student to share necessary information with the department.

- iii) Credit transfer policy will be applicable to the Professional & Open Elective courses only.
- iv) The concerned department shall identify the courses permitted for credit transfer.
- v) The Institution shall notify at the beginning of semester the list of the online learning courses eligible for credit transfer.
- vi) The institution shall designate a faculty member as a Mentor for each course to guide the students from registration till completion of the credit course.
- vii) In case of delay in results, the Institution will re-issue the marks sheet for such students.
- viii) Student pursuing courses under MOOCs shall acquire the required credits only after successful completion of the course and submitting a certificate issued by the competent authority along with the percentage of marks and grades.
- ix) The institution shall submit the following to the examination section of the university:
 - a) List of students who have passed MOOC courses in the current semester along with the certificate of completion.
 - b) Undertaking form filled by the students for credit transfer.
- x) The universities shall resolve any issues that may arise in the implementation of this policy from time to time and shall review its credit transfer policy in the light of periodic changes brought by UGC, SWAYAM, NPTEL and state government.

Note: Students shall be permitted to register for MOOCs offered through online platforms approved by the University from time to time.

13. Academic Bank of Credits (ABC)

The University has implemented Academic Bank of Credits (ABC) to promote flexibility in curriculum as per NEP 2020 to

- i. provide option of mobility for learners across the universities of their choice
- ii. provide option to gain the credits through MOOCs from approved digital platforms.
- iii. facilitate award of certificate/diploma/degree in line with the accumulated credits in ABC
- iv. execute Multiple Entry and Exit system with credit count, credit transfer and credit acceptance from students' account.

14. Mandatory Internships

Summer Internships : Two summer internships either onsite or virtual each with a minimum of 08 weeks duration, done at the end of second and third years, respectively are mandatory. It shall be completed in collaboration with local industries, Govt. Organizations, construction agencies, Power projects, software MNCs or any industries in the areas of concerned specialization of the Undergraduate program. One of the two summer internships at the end of second year (Community Service Project) shall be society oriented and shall be completed in collaboration with government organizations/NGOs & others. The other internship at the end of third year is Industry Internship and shall be completed in collaboration with Industries. The student shall register for the internship as per course structure after commencement of academic year. The guidelines issued by the APSICHE / University shall be followed for carrying out and evaluation of Community Service Project and Industry Internship.

Evaluation of the summer internships shall be through the departmental committee. A student will be required to submit a summer internship report to the concerned department and appear for an oral presentation before the departmental committee comprising of Head of the Department, supervisor of the internship and a senior faculty member of the department. A certificate of successful completion from industry shall be included in the report. The report and the oral

presentation shall carry 50% weightage each. It shall be evaluated for 50 external marks. There shall be no internal marks for Summer Internship. A student shall secure minimum 40% of marks for successful completion. In case a student fails, he/she shall reappear as and when semester supplementary examinations are conducted by the Institution.

Full Semester Internship and Project work: In the final semester, the student should mandatorily register and undergo internship (onsite/virtual) and in parallel he/she should work on a project with well-defined objectives. At the end of the semester the candidate shall submit an internship completion certificate and a project report. A student shall also be permitted to submit project report on the work carried out during the internship.

The project report shall be evaluated with an external examiner. The total marks for project work 200 marks and distribution shall be 60 marks for internal and 140 marks for external evaluation. The supervisor assesses the student for 30 marks (Report: 15 marks, Seminar: 15 marks). At the end of the semester, all projects shall be showcased at the department for the benefit of all students and staff and the same is to be evaluated by the departmental Project Review Committee consisting of supervisor, a senior faculty and HOD for 30 marks. The external evaluation of Project Work is a Viva-Voce Examination conducted in the presence of internal examiner and external examiner appointed by the College and is evaluated for 140 marks.

The college shall facilitate and monitor the student internship programs. Completion of internships is mandatory, if any student fails to complete internship, he/she will not be eligible for the award of degree. In such cases, the student shall repeat and complete the internship.

15. Guidelines for offering a Minor

Minor Degree as per the JNTUK Guidelines

16. Guidelines for offering Honors

The objective of introducing B.Tech. (Hons.) is to facilitate the students to choose additionally the specialized courses of their choice and build their competence in a specialized area in the UG level. The programme is a best choice for academically excellent students having good academic record and interest towards higher studies and research.

- i) Honors is introduced in the curriculum of all B. Tech. programs offering a major degree and is applicable to all B. Tech (Regular and Lateral Entry) students admitted in Engineering & Technology.
- ii) A student shall earn additional 15 credits for award of B.Tech.(Honors) degree from same branch/department/discipline registered for major degree. This is in addition to the credits essential for obtaining the Undergraduate degree in Major Discipline (i.e., 160 credits).
- iii) A student is permitted to register for Honors in IV semester after the results of III Semester are declared and students may be allowed to take maximum two subjects per semester pertaining to the Honors from V Semester onwards.
- iv) The Department shall arrange separate class work and timetable of the courses offered under Honors program.
- v) Courses that are used to fulfil the student's primary major may not be double counted towards the Honors. Courses with content substantially equivalent to courses in the student's primary Major may not be counted towards the Honors.
- vi) Students can complete the courses offered under Honors either in the college or in online platforms like SWAYAM with a minimum duration of 12 weeks for a 3-credit course and 8 weeks duration for a 2-credit course satisfying the criteria for credit mobility. If the courses under Honors are offered in conventional mode, then the teaching and evaluation procedure shall be similar to regular B. Tech courses.

- vii) The attendance for the registered courses under Honors and regular courses offered for Major degree in a semester are to be considered separately.
- viii) A student shall maintain an attendance of 75% in all registered courses under Honors to be eligible for attending semester end examinations.
- ix) **A student registered for Honors shall pass in all subjects that constitute the requirement for the Honors degree program.** No class/division (i.e., second class, first class and distinction, etc.) shall be awarded for Honors degree programme.
- x) If a student drops or is terminated from the Honors program, the additional credits so far earned cannot be converted into open or core electives; they will remain extra. However, such students will receive a separate grade sheet mentioning the additional courses completed by them.
- xi) The Honors will be mentioned in the degree certificate as Bachelor of Technology (Honors) in XYZ. For example, B.Tech. (Honors) in Mechanical Engineering

Enrolment into Honors:

- i) Students of a Department/Discipline are eligible to opt for Honors program offered by the same Department/Discipline
- ii) The enrolment of student into Honors is based on the CGPA obtained in the major degree program. CGPA shall be taken up to III semester in case of regular entry students and only III semester in case of lateral entry students. Students having 7 CGPA without any backlog subjects will be permitted to register for Honors.
- iii) If a student is detained due to lack of attendance either in Major or in Honors, registration shall be cancelled.
- iv) Transfer of credits from Honors to regular B. Tech degree and vice-versa shall not be permitted.
- v) Honors is to be completed simultaneously with a Major degree program.

Registration for Honors:

- i) The eligible and interested students shall apply through the HOD of his/her parent department. The whole process should be completed within one week before the start of every semester. Selected students shall be permitted to register the courses under Honors.
- ii) The selected students shall submit their willingness to the principal through his/her parent department offering Honors. The parent department shall maintain the record of student pursuing the Honors.
- iii) The students enrolled in the Honors courses will be monitored continuously. An advisor/mentor from parent department shall be assigned to a group of students to monitor the progress.

17. Attendance Requirements:

- i) A student shall be eligible to appear for the external examinations if he/she acquires a minimum of 75% of attendance in aggregate of all the subjects. Condonation of shortage of attendance in aggregate up to 10% (65% and above and below 75%) in each semester may be granted by the College Academic Committee.
- ii) Shortage of Attendance below 65% in aggregate shall in NO CASE be condoned.
- iii) A stipulated fee shall be payable towards condonation of shortage of attendance to the University.
- iv) Students whose shortage of attendance is not condoned in any semester are not eligible to take their end examination of that class and their registration shall stand cancelled.
- v) A student will not be promoted to the next semester unless he satisfies the attendance requirements of the present semester. They may seek readmission for that semester from the date of commencement of class work.

- vi) If any candidate fulfils the attendance requirement in the present semester, he shall not be eligible for readmission into the same class.
- vii) If the learning is carried out in blended mode (both offline & online), then the total attendance of the student shall be calculated considering the offline and online attendance of the student.
- viii) For induction programme attendance shall be maintained as per AICTE norms.

18. Promotion Rules:

The following academic requirements must be satisfied in addition to the attendance requirements mentioned in section 17.

- i) A student shall be promoted from first year to second year if he/she fulfils the minimum attendance requirement as per university norms.
- ii) A student will be promoted from II to III year if he/she fulfils the academic requirement of securing 40% of the credits (any *decimal* fraction should be *rounded off* to *Upper* digit) up to in the subjects that have been studied up to IV semester.
- iii) A student shall be promoted from III year to IV year if he/she fulfils the academic requirements of securing 40% of the credits (any *decimal* fraction should be *rounded off* to *Upper* digit) in the subjects that have been studied up to VI semester. And in case a student is detained for want of credits for a particular academic year by ii) & iii) above, the student may make up the credits through supplementary examinations and only after securing the required credits he/she shall be permitted to join in the V semester or VII semester respectively as the case may be.
- iv) When a student is detained due to lack of credits/shortage of attendance he/she may be re-admitted when the semester is offered after fulfilment of academic regulations. In such case, he/she shall be in the academic regulations into which he/she is readmitted.

19. Grading:

As a measure of the student's performance, a 10-point Absolute Grading System using the following Letter Grades and corresponding percentage of marks shall be followed:

After each course is evaluated for 100 marks, the marks obtained in each course will be converted to a corresponding letter grade as given below, depending on the range in which the marks obtained by the student fall.

Structure of Grading of Academic Performance

Range in which the marks in the subject fall	Grade	Grade points
		Assigned
90 & above	S (Superior)	10
80 - 89	A (Excellent)	9
70 - 79	B (Very Good)	8
60 - 69	C (Good)	7
50 - 59	D (Average)	6
40 - 49	E (Pass)	5
< 40	F (Fail)	0
Absent	Ab (Absent)	0

- i) A student obtaining Grade 'F' or Grade 'Ab' in a subject shall be considered failed and will be required to reappear for that subject when it is offered the next supplementary examination.

- ii) For non-credit audit courses, “Satisfactory” or “Unsatisfactory” shall be indicated instead of the letter grade and this will not be counted for the computation of SGPA/CGPA/Percentage.

Computation of Semester Grade Point Average (SGPA) and Cumulative Grade Point Average (CGPA):

The Semester Grade Point Average (SGPA) is the ratio of sum of the product of the number of credits with the grade points scored by a student in all the courses taken by a student and the sum of the number of credits of all the courses undergone by a student, i.e.,

$$SGPA = \frac{\sum (C_i \times G_i)}{\sum C_i}$$

where, C_i is the number of credits of the i^{th} subject and G_i is the grade point scored by the student in the i^{th} course.

The Cumulative Grade Point Average (CGPA) will be computed in the same manner considering all the courses undergone by a student over all the semesters of a program, i.e.,

$$CGPA = \frac{\sum (C_i \times S_i)}{\sum C_i}$$

where “ S_i ” is the SGPA of the i^{th} semester and C_i is the total number of credits up to that semester.

Both SGPA and CGPA shall be rounded off to 2 decimal points and reported in the transcripts.

Grade Point: It is a numerical weight allotted to each letter grade on a 10-point scale. Letter Grade: It is an index of the performance of students in a said course. Grades are denoted by the letters S, A, B, C, D and F.

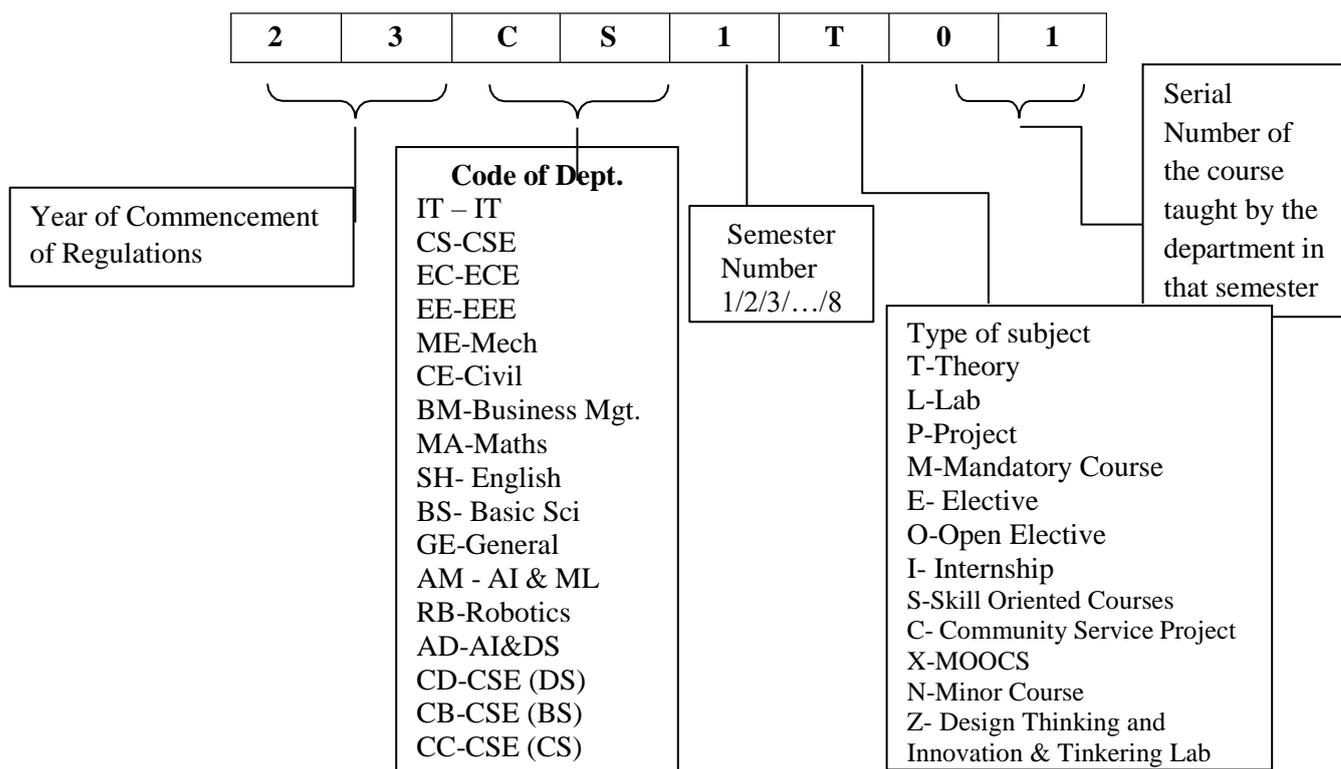
Award of Class: After a student has satisfied the requirements prescribed for the completion of the program and is eligible for the award of B. Tech. Degree, he/she shall be placed in one of the following four classes:

Class Awarded	CGPA Secured
First Class with Distinction	≥ 7.5
First Class	$\geq 6.5 < 7.5$
Second Class	$\geq 5.5 < 6.5$
Pass Class	$\geq 5.0 < 5.5$

CGPA to Percentage conversion Formula – $(CGPA - 0.5) \times 10$

20. Course Code & Course Numbering Scheme:

The subject codes will be given by the department teaching the subject. Each subject code contains 8 characters. The 8 characters for each subject will be filled as per the following guidelines.



21. Supplementary Examinations

In addition to the Regular Final Examinations held at the end of each semester, a Supplementary Examination will be conducted. A student can appear for any number courses of supplementary examinations till he/she clears the courses. However the maximum stipulated period of programme cannot be relaxed under any circumstance.

22. Advanced Supplementary Examinations:

Candidate who fails the courses in 7th and 8th Semester can appear for Advanced Supplementary Examinations.

23. With-holding of Results

If the candidate has any dues not paid to the Institution or if any case of indiscipline or malpractice is pending against him/her, the result of the candidate shall be withheld in such cases.

24. Multiple Entry / Exit Option

(a) Exit Policy:

The students can choose to exit the four-year programme at the end of first/second/third year.

i) **UG Certificate in (Field of study/discipline)** - Programme duration: First year (first two semesters) of the undergraduate programme, 40 credits followed by an additional exit 10-credit bridge course(s) lasting two months, including at least 6credit job-specific internship/apprenticeship that would help the candidates acquire job-ready competencies required to enter the workforce.

ii) **UG Diploma (in Field of study/discipline)** - Programme duration: First two years (first four semesters) of the undergraduate programme, 80 credits followed by an additional exit 10-credit bridge course(s) lasting two months, including at least 6credit job-specific internship/

apprenticeship that would help the candidates acquire job-ready competencies required to enter the workforce.

iii) **Bachelor of Science (in Field of study/discipline) i.e., B.Sc. Engineering in(Field of study/discipline)**- Programme duration: First three years (first six semesters) of the undergraduate programme, 120 credits.

(b) Entry Policy:

Modalities on multiple entry by the student into the B.Tech. programme will be provided in due course of time.

Note: The Universities shall resolve any issues that may arise in the implementation of Multiple Entry and Exit policies from time to time and shall review the policies in the light of periodic changes brought by UGC, AICTE and State government.

25. Gap Year Concept:

Gap year concept for Student Entrepreneur in Residence is introduced and outstanding students who wish to pursue entrepreneurship / become entrepreneur are allowed to take a break of one year at any time after II year to pursue full-time entrepreneurship programme/to establish startups. This period may be extended to two years at the most and these two years would not be counted for the time for the maximum time for graduation. The principal of the college shall forward such proposals submitted by the students. An evaluation committee constituted by the principal shall evaluate the proposal submitted by the student and the committee shall decide whether to permit the student(s) to avail the Gap Year or not

26. Transitory Regulations

Discontinued, detained, or failed candidates are eligible for readmission as and when the semester is offered after fulfilment of academic regulations. Candidates who have been detained for want of attendance or not fulfilled academic requirements or who have failed after having undergone the course in earlier regulations or have discontinued and wish to continue the course are eligible for admission into the unfinished semester from the date of commencement of class work with the same or equivalent subjects as and when subjects are offered, subject to Section 2 and they will follow the academic regulations into which they are readmitted.

Candidates who are permitted to avail Gap Year shall be eligible for re-joining into the succeeding year of their B. Tech from the date of commencement of class work, subject to Section 2 and they will follow the academic regulations into which they are readmitted.

27. Minimum Instruction Days for a Semester:

The minimum instruction days including exams for each semester shall be 90 days.

28. Medium of Instruction:

The medium of instruction of the entire B. Tech undergraduate programme in Engineering & Technology (including examinations and project reports) will be in English only.

29. Student Transfers:

Student transfers shall be as per the guidelines issued by the Government of Andhra Pradesh and the Universities from time to time.

30. General Instructions:

- i) The academic regulations should be read as a whole for purpose of any interpretation.
- ii) Malpractices rules-nature and punishments are appended.

- iii) Where the words “he”, “him”, “his”, occur in the regulations, they also include “she”, “her”, “hers”, respectively.
- iv. In the case of any doubt or ambiguity in the interpretation of the above rules, the decision of the Academic Council is final.
- v. The College may change or amend the academic regulations or syllabi at any time and the changes or amendments shall be made applicable to all the students on rolls with effect from the dates notified by the Institutaion.
- vi. In the case of any doubt or ambiguity in the interpretation of the guidelines given, the decision of the Academic Council / Head of the institution is final.

ACADEMIC REGULATIONS (R23) FOR B.TECH. (LATERAL ENTRY SCHEME)

(Effective for the students admitted into II year through Lateral Entry Scheme from the Academic Year 2024-25 onwards)

1. Award of the Degree

(a) Award of the B.Tech. Degree if he/she fulfils the following:

- (i) Pursues a course of study for not less than three academic years and not more than six academic years. However, for the students availing Gap year facility this period shall be extended by two years at the most and these two years would in addition to the maximum period permitted for graduation (Six years).
- (ii) Registers for 120 credits and secures all 120 credits.

(b) **Award of B.Tech. degree with Honors** if he/she fulfils the following:

- (i) Student secures additional 15 credits fulfilling all the requisites of a B.Tech. program i.e., 120 credits.
- (ii) Registering for Honors is optional.
- (iii) Honors is to be completed simultaneously with B.Tech. programme.

2. Students, who fail to fulfil the requirement for the award of the degree within six consecutive academic years from the year of admission, shall forfeit their seat.

3. Minimum Academic Requirements

The following academic requirements have to be satisfied in addition to the requirements mentioned in item no.2

- i. A student shall be deemed to have satisfied the minimum academic requirements and earned the credits allotted to each theory, practical, design, drawing subject or project if he secures not less than 35% of marks in the end examination and a minimum of 40% of marks in the sum total of the mid semester evaluation and end examination taken together.
- ii. A student shall be promoted from III year to IV year if he/she fulfils the academic requirements of securing 40% of the credits (any decimal fraction should be rounded off to lower digit) in the subjects that have been studied up to V or VI semester.

And in case if student is already detained for want of credits for particular academic year, the student may make up the credits through supplementary exams of the above exams before the commencement of VII semester class work of next year.

4. Course Pattern

- i) The entire course of study is three academic years on semester pattern.
- iii) A student eligible to appear for the end examination in a subject but absent at it or has failed in the end examination may appear for that subject at the next supplementary examination offered.
- iv) When a student is detained due to lack of credits/shortage of attendance the student may be re-admitted when the semester is offered after fulfilment of academic regulations, the student shall be in the academic regulations into which he/she is readmitted.

All other regulations as applicable for B. Tech. Four-year degree course (Regular) will hold good for B. Tech. (Lateral Entry Scheme).

B.TECH. - COURSE STRUCTURE – R23
(Applicable from the academic year 2023-24 onwards)

INDUCTION PROGRAMME

S.No.	Course Name	Category	L-T-P-C
1	Physical Activities -- Sports, Yoga and Meditation, Plantation	MC	0-0-6-0
2	Career Counselling	MC	2-0-2-0
3	Orientation to all branches -- career options, tools, etc.	MC	3-0-0-0
4	Orientation on admitted Branch -- corresponding labs, tools and platforms	EC	2-0-3-0
5	Proficiency Modules & Productivity Tools	ES	2-1-2-0
6	Assessment on basic aptitude and mathematical skills	MC	2-0-3-0
7	Remedial Training in Foundation Courses	MC	2-1-2-0
8	Human Values & Professional Ethics	MC	3-0-0-0
9	Communication Skills -- focus on Listening, Speaking, Reading, Writing skills	BS	2-1-2-0
10	Concepts of Programming	ES	2-0-2-0

SEMESTER-I

S.No.	Category	Course Code	Title	L/D	T	P	Credits	IM	EM	TM
1	BS&H	23HS1T01	Communicative English	2	0	0	2	30	70	100
2	BS&H	23BS1T01	Chemistry	3	0	0	3	30	70	100
3	BS&H	23BS1T03	Linear Algebra & Calculus	3	0	0	3	30	70	100
4	Engineering Science	23ME1T01	Basic Civil & Mechanical Engineering	3	0	0	3	30	70	100
5	Engineering Science	23CS1T01	Introduction to Programming	3	0	0	3	30	70	100
6	BS&H	23HS1L01	Communicative English Lab	0	0	2	1	30	70	100
7	BS&H	23BS1L01	Chemistry Lab	0	0	2	1	30	70	100
8	Engineering Science	23ME1L01	Engineering Workshop	0	0	3	1.5	30	70	100
9	Engineering Science	23CS1L01	Computer Programming Lab	0	0	3	1.5	30	70	100
10	BS&H	23BS1P01	Health and Wellness, Yoga and Sports	-	-	1	0.5	-	-	-
Total				14	00	11	19.5	270	630	900

SEMESTER-II

S.No.	Category	Course Code	Title	L/D	T	P	Credits	IM	EM	TM
1	BS&H	23BS2T02	Engineering Physics	3	0	0	3	30	70	100
2	BS&H	23BS2T04	Differential Equations & Vector Calculus	3	0	0	3	30	70	100
3	Engineering Science	23EE2T01	Basic Electrical & Electronics Engineering	3	0	0	3	30	70	100
4	Engineering Science	23ME2T02	Engineering Graphics	1	0	4	3	30	70	100
5	Professional Core	23CS2T01	Data Structure	3	0	0	3	30	70	100
6	Engineering Science	23CS2L02	IT Workshop	0	0	2	1	30	70	100
7	BS&H	23BS2L02	Engineering Physics Lab	0	0	2	1	30	70	100
8	Engineering Science	23EE2L01	Electrical & Electronics Engineering Workshop	0	0	3	1.5	30	70	100
9	Professional Core	23CS2L01	Data Structures Lab	0	0	3	1.5	30	70	100
10	BS&H	23BS2P02	NSS/NCC/Scouts & Guides/ Community Service	-	-	1	0.5	-	-	-
Total				13	00	15	20.5	270	630	900

SEMESTER-III

S. No.	CATEGORY	Course Code	Course Title	L	T	P	C	IM	EM	TM
1	BS & H	23BS3T03	Discrete Mathematics & Graph Theory	3	0	0	3	30	70	100
2	Management Course-I	23BM3T01	Managerial Economics and Financial Analysis	2	0	0	2	30	70	100
3	Engineering Science	23CS3T01	Computer Organization	3	0	0	3	30	70	100
4	Professional Core	23CS3T02	Advanced Data Structures & Algorithms	3	0	0	3	30	70	100
5	Professional Core	23CS3T03	Database Management Systems	3	0	0	3	30	70	100
6	Professional Core	23CS3L01	Advanced Data Structures & Algorithms Lab	0	0	3	1.5	30	70	100
7	Professional Core	23CS3L02	Database Management Systems Lab	0	0	3	1.5	30	70	100
8	Skill Enhancement course	23CS3S01	Python Programming	0	1	2	2	30	70	100
9	Audit Course	23CS3M01	Environmental Science	2	0	0	-	-	-	-
Total				16	1	8	19	240	560	800

SEMESTER-IV

S. No	CATEGORY	Course Code	Course Title	L	T	P	C	IM	EM	TM
1	BS & H	23HS4T01	Universal human values – understanding harmony and Ethical human conduct	2	1	0	3	30	70	100
2	Engineering Science/ Basic Science	23BS4T04	Probability & Statistics	3	0	0	3	30	70	100
3	Professional Core	23CS4T01	Operating Systems	3	0	0	3	30	70	100
4	Professional Core	23CS4T02	Object Oriented Programming Through Java	3	0	0	3	30	70	100
5	Professional Core	23IT4T02	Software Engineering	3	0	0	3	30	70	100
6	Professional Core	23CS4L01	Operating Systems & Software Engineering Lab	0	0	3	1.5	30	70	100
7	Professional Core	23CS4L02	Object Oriented Programming Through Java Lab	0	0	3	1.5	30	70	100
8	Skill Enhancement Course	23CS4S01	Full Stack Development - 1	0	1	2	2	30	70	100
9	BS&H	23CS4Z01	Design Thinking & Innovation	1	0	2	2	30	70	100
Total				15	2	10	22	240	560	900

L-LECTURE HOURS, T- TUTORIAL HOURS, P-PRACTICAL HOURS, C-CREDITS, IM- INTERNAL MARKS, EM- EXTERNAL MARKS, TM- TOTAL MARKS

I SEMESTER	L	T	P	C
	2	-	-	2

23HS1T01- COMMUNICATIVE ENGLISH

Course Objectives:

The main objective of introducing this course, *Communicative English*, is to facilitate effective listening, Reading, Speaking and Writing skills among the students. It enhances the same in their comprehending abilities, oral presentations, reporting useful information and providing knowledge of grammatical structures and vocabulary. This course helps the students to make them effective in speaking and writing skills and to make them industry ready.

Course Outcomes:

CO1: Understand the context, topic, and pieces of specific information from social or Transactional dialogues.

CO2: Apply grammatical structures to formulate sentences and correct word forms.

CO3: Analyze discourse markers to speak clearly on a specific topic in informal discussions.

CO4: Evaluate reading / listening texts and to write summaries based on global comprehension of these texts.

CO5: Create a coherent paragraph, essay, and resume.

UNIT-I**Lesson: HUMAN VALUES: Gift of Magi (Short Story)**

Listening: Identifying the topic, the context and specific pieces of information by listening to short audio texts and answering a series of questions.

Speaking: Asking and answering general questions on familiar topics such as home, family, work, studies and interests; introducing oneself and others.

Reading: Skimming to get the main idea of a text; scanning to look for specific pieces of information.

Writing: Mechanics of Writing-Capitalization, Spellings, Punctuation-Parts of Sentences.

Grammar: Parts of Speech, Basic Sentence Structures-forming questions

Vocabulary: Synonyms, Antonyms, Affixes (Prefixes/Suffixes), Root words.

UNIT-II**Lesson: NATURE: The Brook by Alfred Tennyson (Poem)**

Listening: Answering a series of questions about main ideas and supporting ideas after listening to audio texts.

Speaking: Discussion in pairs/small groups on specific topics followed by short structure talks.

Reading: Identifying sequence of ideas; recognizing verbal techniques that help to link the ideas in a paragraph together.

Writing: Structure of a paragraph - Paragraph writing (specific topics)

Grammar: Cohesive devices - linkers, use of articles and zero article; prepositions.

Vocabulary: Homonyms, Homophones, Homographs.

UNIT-III**Lesson: BIOGRAPHY: Elon Musk**

Listening: Listening for global comprehension and summarizing what is listened to.

Speaking: Discussing specific topics in pairs or small groups and reporting what is discussed

Reading: Reading a text in detail by making basic inferences -recognizing and interpreting specific context clues; strategies to use text clues for comprehension.

Writing: Summarizing, Note-making, paraphrasing

Grammar: Verbs - tenses; subject-verb agreement; Compound words, Collocations

Vocabulary: Compound words, Collocations

UNIT-IV

Lesson: **INSPIRATION: The Toys of Peace** by Saki

Listening: Making predictions while listening to conversations/ transactional dialogues without video; listening with video.

Speaking: Role plays for practice of conversational English in academic contexts (formal and informal) - asking for and giving information/directions.

Reading: Studying the use of graphic elements in texts to convey information, reveal trends/patterns/relationships, communicate processes or display complicated data.

Writing: Letter Writing: Official Letters, Resumes

Grammar: Reporting verbs, Direct & Indirect speech, Active & Passive Voice

Vocabulary: Words often confused, Jargons

UNIT-V

Lesson: **MOTIVATION: The Power of Intrapersonal Communication** (An Essay)

Listening: Identifying key terms, understanding concepts and answering a series of relevant questions that test comprehension.

Speaking: Formal oral presentations on topics from academic contexts

Reading: Reading comprehension.

Writing: Writing structured essays on specific topics.

Grammar: Editing short texts –identifying and correcting common errors in grammar and usage (articles, prepositions, tenses, subject verb agreement)

Vocabulary: Technical Jargons

Textbooks:

1. Pathfinder: Communicative English for Undergraduate Students, 1st Edition, Orient Black Swan, 2023

Reference Books:

1. Dubey, Sham Ji & Co. English for Engineers, Vikas Publishers, 2020
2. Bailey, Stephen. Academic writing: A Handbook for International Students. Routledge, 2014.
3. Murphy, Raymond. English Grammar in Use, Fourth Edition, Cambridge University Press, 2019.
4. Lewis, Norman. Word Power Made Easy- The Complete Handbook for Building a Superior Vocabulary. Anchor, 2014.

I SEMESTER	L	T	P	C
	3	-	-	3

23BS1T01- CHEMISTRY

Course Objectives:

- To familiarize engineering chemistry and its applications
- To train the students on the principles and applications of electrochemistry and polymers
- To introduce instrumental methods, molecular machines and switches.

Course Outcomes: At the end of the course, the students will be able to:

CO1: Compare the materials of construction for battery.

CO2: Explain the preparation, properties, and applications of thermoplastics, thermosetting & elastomers conducting polymers.

CO3: Explain the fundamentals of quantum mechanics and molecular orbital theory.

CO4: Apply the principle of Band diagrams in the application of conductors and semiconductors.

CO5: Summarize the concepts of Instrumental methods.

UNIT- I**Electrochemistry and Applications**

Electrochemical cell, Nernst equation, cell potential calculations and numerical problems, potentiometry-potentiometric titrations (redox titrations), concept of conductivity, conductivity cell, conductometric titrations (acid-base titrations).

Primary cells – Zinc-air battery, Secondary cells – lithium-ion batteries- working of the batteries including cell reactions; Fuel cells, hydrogen-oxygen fuel cell – working of the cells. Polymer Electrolyte Membrane Fuel cells (PEMFC).

UNIT- II**Modern Engineering materials**

Semiconductors – Introduction, basic concept, application

Semiconductor preparation by distillation method and purification by Zone refining. Superconductors- Introduction basic concept, applications.

Nano materials: Introduction, classification, properties and applications of Fullerenes, carbon nano tubes and Graphene nanoparticles.

UNIT- III**Structure and Bonding Models:**

Fundamentals of Quantum mechanics, Schrodinger Wave equation, significance of Ψ and Ψ^2 , molecular orbital theory – bonding in homo- and heteronuclear diatomic molecules – energy level diagrams of H₂, O₂, N₂ and CO, calculation of bond order.

UNIT- IV**Polymer Chemistry**

Introduction to polymers, functionality of monomers, chain growth and step growth polymerization with specific examples and mechanisms of polymer formation.

Plastics – Thermo and Thermosetting plastics, Preparation, properties and applications of – PVC, Teflon, Bakelite, Nylon-6,6.

Elastomers–Buna-S, Buna-N–preparation, properties and applications.

Conducting polymers – polyacetylene, polyaniline, – mechanism of conduction and applications. Bio-Degradable polymers - PHBV.

UNIT- V

Instrumental Methods and Applications

Electromagnetic spectrum. Absorption of radiation: Beer-Lambert's law. UV-Visible Spectroscopy, electronic transition, Instrumentation, IR spectroscopies, fundamental modes and selection rules, Instrumentation.

Textbooks:

1. Jain and Jain, Engineering Chemistry, 16/e, DhanpatRai, 2013.
2. Peter Atkins, Julio de Paula and James Keeler, Atkins' Physical Chemistry, 10/e, Oxford University Press, 2010.

Reference Books:

1. Skoog and West, Principles of Instrumental Analysis, 6/e, Thomson, 2007.
2. J.D. Lee, Concise Inorganic Chemistry, 5th Edition, Wiley Publications, Feb.2008
3. Textbook of Polymer Science, Fred W. Billmayer Jr, 3rd Edition

I SEMESTER	L	T	P	C
	3	-	-	3
23BS1T03- LINEAR ALGEBRA & CALCULUS				

Course Objectives:

To equip the students with standard concepts and tools at an intermediate to advanced level mathematics to develop the confidence and ability among the students to handle various real-world problems and their applications.

Course Outcomes:

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

CO1: Solve a given system of linear equations (K2)

CO2: Develop the matrix algebra techniques that are needed by engineers for practical applications (K4)

CO3: Utilize mean value theorems for real life problems (K2)

CO4: Apply the concept of partial differentiation in various engineering applications (K3)

CO5: Evaluate double, triple integrals and their applications (K3)

UNIT-I**Matrices**

Linear Transformation-Rank of a matrix by echelon form, normal form. Cauchy–Binet formulae (without proof). Inverse of Non- singular matrices by Gauss-Jordan method, System of linear equations: Consistency of System of Linear Equations–Solving system of Homogeneous and Non-Homogeneous equations by Gauss elimination method, Jacobi and Gauss Seidel Iteration Methods.

UNIT-II**Eigenvalues, Eigenvectors and Orthogonal Transformation**

Eigenvalues, Eigenvectors and their properties, Diagonalization of a matrix, Cayley-Hamilton Theorem (without proof), finding inverse and power of a matrix by Cayley-Hamilton Theorem, Quadratic forms and Nature of the Quadratic Forms, Reduction of Quadratic form to canonical forms by Orthogonal Transformation.

UNIT-III**Calculus**

Mean Value Theorems: Rolle’s Theorem, Lagrange’s mean value theorem with their geometrical interpretation, Cauchy’s mean value theorem, Taylor’s and Maclaurin theorems with remainders (without proof), Problems and applications on the above theorems.

UNIT-IV**Partial differentiation and Applications (Multi variable calculus)**

Functions of several variables: Continuity and Differentiability, Partial derivatives, total derivatives, chain rule, Taylor’s and Maclaurin’s series expansion of functions of two variables. Jacobians, Functional dependence, maxima and minima of functions of two variables, method of Lagrange multipliers.

UNIT-V**Multiple Integrals (Multi variable Calculus)**

Double integrals, triple integrals, change of order of integration, change of variables to polar, cylindrical and spherical coordinates. Finding areas (by double integrals) and volumes (by double integrals and triple integrals).

Textbook:

1. Higher Engineering Mathematics, B. S. Grewal, Khanna Publishers, 2017, 44th Edition

Reference Book:

1. Advanced Engineering Mathematics, Erwin Kreyszig, John Wiley & Sons, 2018, 10th Edition

I SEMESTER	L	T	P	C
	3	-	-	3

23ME1T01- BASIC CIVIL AND MECHANICAL ENGINEERING

Course Objectives:

1. Get familiarized with the scope and importance of Civil Engineering sub-divisions.
2. Introduce the preliminary concepts of surveying.
3. Acquire preliminary knowledge on Transportation and its importance in nation's economy.
4. Get familiarized with the importance of quality, conveyance and storage of water.
5. Introduction to basic civil engineering materials and construction techniques.

Course Outcomes:

On completion of the course, the student should be able to:

- CO1:** Understand various sub-divisions of Civil Engineering and to appreciate their role in ensuring better society.
- CO2:** Know the concepts of surveying and to understand the measurement of distances, angles and levels through surveying.
- CO3:** Realize the importance of Transportation in nation's economy and the engineering measures related to Transportation.
- CO4:** Understand the importance of Water Storage and Conveyance Structures so that the social responsibilities of water conservation will be appreciated.
- CO5:** Understand the basic characteristics of Civil Engineering Materials and attain knowledge on prefabricated technology.

UNIT-I

Basics of Civil Engineering: Role of Civil Engineers in Society- Various Disciplines of Civil Engineering- Structural Engineering- Geo-technical Engineering- Transportation Engineering – Hydraulics and Water Resources Engineering – Environmental Engineering – Scope of each discipline - Building Construction and Planning- Construction Materials-Cement - Aggregate - Bricks- Cement concrete- Steel. Introduction to Prefabricated construction Techniques.

UNIT-II

Surveying: Objectives of Surveying – Horizontal Measurements – Angular Measurements - Introduction to Bearings Levelling instruments used for levelling -Simple problems on levelling and bearings- Contour mapping.

UNIT-III

Transportation Engineering Importance of Transportation in Nation's economic development- Types of Highway Pavements – Flexible Pavements and Rigid Pavements –Simple Differences. Basics of Harbour, Tunnel, Airport, and Railway Engineering.

Water Resources and Environmental Engineering: Introduction, Sources of water-Quality of water-Specifications- Introduction to Hydrology–Rainwater Harvesting-Water Storage and Conveyance Structures (Simple introduction to Dams and Reservoirs).

Textbooks:

1. Basic Civil Engineering, M.S.Palanisamy,, TataMcgraw Hill publications(India) Pvt.Ltd.Fourth Edition.
2. Introduction to Civil Engineering, S.S.Bhavikatti, New Agenternational Publishers.2022.First Edition.
3. Basic Civil Engineering, Satheesh Gopi, Pearson Publications, 2009, FirstEdition.

Reference Books:

1. Surveying, Vol-IandVol-II,S.K.Duggal,Tata McGrawHillPublishers2019.FifthEdition.
2. Hydrology and Water Resources Engineering, Santosh Kumar Garg, Khanna Publishers,Delhi. 2016
3. Irrigation Engineering and Hydraulic Structures - Santosh Kumar Garg, Khanna Publishers ,Delhi 2023. 38th Edition.
4. Highway Engineering, S.K.Khanna, C.E.G.Justoand Veeraraghavan, Nemchand and Brothers Publications 2019. 10th Edition.
5. Indian Standard DRINKING WATER—SPECIFICATION IS10500-2012.

PART – B
BASIC MECHANICAL ENGINEERING

Course Objectives:

The students after completing the course are expected to

1. Get familiarized with the scope and importance of Mechanical Engineering in different sectors and industries.
2. Explain different engineering materials and different manufacturing processes.
3. Provide an overview of different thermal and mechanical transmission systems and introduce basics of robotics and its applications.

Course Outcomes:

On completion of the course, the student should be able to

CO1: Understand the different manufacturing processes.

CO2: Explain the basics of thermal engineering and its applications.

CO3: Describe the working of different mechanical power transmission systems and power plants.

CO4: Describe the basics of robotics and its applications.

UNIT-I

Introduction to Mechanical Engineering: Role of Mechanical Engineering in Industries and Society-Technologies in different sectors such as Energy, Manufacturing, Automotive, Aerospace, and Marine sectors.

Engineering Materials: Metals-Ferrous and Non-ferrous, Ceramics, Composites, Smart materials.

UNIT-II

Manufacturing Processes: Principles of Casting, Forming, joining processes, Machining, Introduction to CNC machines, 3D printing, and Smart manufacturing.

Thermal Engineering: Working principle of Boilers, Otto cycle, Diesel cycle, Refrigeration and air-conditioning cycles IC engines, 2-Stroke and 4-Stroke engines, SI/CI Engines, Components of Electric and Hybrid Vehicles.

UNIT -III

Power plants: Working principle of Steam, Diesel, Hydro, Nuclear power plants. Mechanical Power Transmission - Belt Drives, Chain, Rope drives, Gear Drives and their applications.

Introduction to Robotics - Joints & links, configurations, and applications of robotics.

Textbooks:

1. Internal Combustion Engines by V.Ganesan, By Tata McGraw Hill publications (India) Pvt. Ltd.
2. A text book of Theory of Machines by S.S. Rattan, Tata McGraw Hill Publications, (India) Pvt. Ltd.
3. An introduction to Mechanical Engg by Jonathan Wicker and Kemper Lewis, Cengage learning India Pvt. Ltd.

Reference Books:

1. G. Shanmugam and M.S.Palanisamy, Basic Civil and the Mechanical Engineering, Tata McGraw Hill publications (India) Pvt. Ltd.
2. Thermal Engineering by Mahesh M Rathore Tata McGraw Hill publications (India) Pvt.Ltd.
3. 3D printing & Additive Manufacturing Technology- L. Jyothish Kumar, Pulak M Pandey, Springer publications
4. Appuu Kuttan KK, Robotics, I.K. International Publishing House Pvt. Ltd. Volume-I

I SEMESTER	L	T	P	C
	3	-	-	3

23CS1T01- INTRODUCTION TO PROGRAMMING

Course Objectives:

- To introduce students to the fundamentals of computer programming.
- To provide hands-on experience with coding and debugging.
- To foster logical thinking and problem-solving skills using programming.
- To familiarize students with programming concepts such as data types, control structures, functions, and arrays.
- To encourage collaborative learning and teamwork in coding projects.

Course Outcomes:

A student after completion of the course will be able to

CO1: Understand basics of computers, the concept of algorithm and algorithmic thinking.

CO2: Analyze a problem and develop an algorithm to solve it.

CO3: Implement various algorithms using the C programming language.

CO4: Understand more advanced features of C language.

CO5: Develop problem-solving skills and the ability to debug and optimize the code.

UNIT- I

Introduction to Programming and Problem Solving

History of Computers, Basic organization of a computer: ALU, input-output units, memory, program counter, Introduction to Programming Languages, Basics of a Computer Program Algorithms, flowcharts (Using Dia Tool), pseudo code. Introduction to Compilation and Execution, Primitive Data Types, Variables, and Constants, Basic Input and Output, Operators, Type Conversion, and Casting.

Problem solving techniques: Algorithmic approach, characteristics of algorithm, Problem solving strategies: Top-down approach, Bottom-up approach, Time and space complexities of algorithms.

UNIT-II

Control Structures

Simple sequential programs Conditional Statements (if, if-else, switch), Loops (for, while, dowhile) Break, Continue, goto and exit.

UNIT-III

Arrays and Strings

Arrays indexing, memory model, programs with array of integers, two dimensional arrays, Introduction to Strings- String Handling Functions.

UNIT- IV

Pointers & User Defined Data types

Pointers-Declaration, Initialization and Accessing , Pointer Arithmetic, Dynamic memory Allocation, array manipulation using pointers, User-defined data types-Structures and Unions.

UNIT -V

Functions & File Handling

Introduction to Functions, Function Declaration and Definition, Function call Return Types and Arguments, modifying parameters inside functions using pointers, arrays as parameters. Scope and Lifetime of Variables, Basics of File Handling- Types of File, File Modes, File Accessing functions.

Note: The syllabus is designed with C Language as the fundamental language of implementation.

Textbooks:

1. Programming in ANSI C by E. Balaguruswamy, Tata Mc-Graw Hill
2. "The C Programming Language", Brian W. Kernighan and Dennis M. Ritchie, Prentice Hall, 1988
3. Schaum's Outline of Programming with C, Byron S Gottfried, McGraw-Hill Education, 1996

Reference Books:

1. Computing fundamentals and C Programming, Balagurusamy, E., McGraw-Hill Education, 2008.
2. Programming in C, Rema Theraja, Oxford, 2016, 2nd edition
3. C Programming, A Problem Solving Approach, Forouzan, Gilberg, Prasad, CENGAGE, 3rd edition

I SEMESTER	L	T	P	C
	-	-	2	1

23HS1L01- COMMUNICATIVE ENGLISH LAB

Course Objectives:

The main objective of introducing this course, Communicative English Laboratory, is to expose the students to a variety of self-instructional, learner friendly modes of language learning. The students will get trained in basic communication skills and also make them ready to face job interviews.

Course Outcomes:

CO1: Understand the different aspects of the English language proficiency with emphasis on LSRW skills.

CO2: Apply communication skills through various language learning activities.

CO3: Analyze the English speech sounds, stress, rhythm, intonation and syllable division for better listening and speaking comprehension.

CO4: Evaluate and exhibit professionalism in participating in debates and group discussions.

CO5: Create effective Course Objectives

List of Topics:

Vowels & Consonants

1. Neutralization/Accent Rules
2. Communication Skills & JAM
3. Role Play or Conversational Practice
4. E-mail Writing
5. Resume Writing, Cover letter, SOP
6. Group Discussions-methods & practice
7. Debates - Methods & Practice
8. PPT Presentations/ Poster Presentation
9. Interviews Skills

Suggested Software:

- Walden Infotech
- Young India Films
-

Reference Books:

1. Raman Meenakshi, Sangeeta-Sharma. *Technical Communication*. Oxford Press.2018.
2. Taylor Grant: *English Conversation Practice*, Tata McGraw-Hill Education India, 2016
3. Hewing's, Martin. *Cambridge Academic English (B2)*. CUP, 2012.
4. J. Sethi & P.V. Dhamija. *A Course in Phonetics and Spoken English*, (2nd Ed), Kindle, 2013

Web Resources:**Spoken English:**

1. www.esl-lab.com
2. www.englishmedialab.com
3. www.englishinteractive.net
4. <https://www.britishcouncil.in/english/online>
5. <http://www.letstalkpodcast.com/>

I SEMESTER	L	T	P	C
	-	-	2	1
23BS1L01- CHEMISTRY LAB				

Course Objectives

- Verify the fundamental concepts with experiments.

Course Outcomes: At the end of the course, the students will be able to

CO1: Determine the cell constant and conductance of solutions.

CO2: Prepare advanced polymer Bakelite materials.

CO3: Measure the strength of an acid present in secondary batteries.

CO4: Determine the redox potentials and emf.

CO5: Calculate strength of acid in Pb-Acid battery.

List of Experiments:

1. Conductometric titration of strong acid vs. strong base
2. Conductometric titration of weak acid vs. strong base
3. Potentiometric titration of strong acid vs. strong base
4. Potentiometric titration of weak acid vs. strong base
5. Determination of cell constant and conductance of solutions
6. Potentiometry - determination of redox potentials and emfs
7. Determination of Strength of an acid in Pb-Acid battery
8. Preparation of a Bakelite
9. Preparation of nanomaterials by precipitation method
10. Estimation of Ferrous ion by Dichrometry

Reference:

- "Vogel's Quantitative Chemical Analysis 6th Edition 6th Edition" Pearson Publications by J. Mendham, R.C.Denney, J.D.Barnes and B. Sivasanka

I SEMESTER	L	T	P	C
		0	0	3

23ME1L01-ENGINEERING WORKSHOP

Course Objectives:

To familiarize students with wood working, sheet metal operations, fitting, electrical house wiring skills, and basic repairs of two-wheeler vehicle.

Course Outcomes:

CO1: Identify workshop tools and their operational capabilities. [K2]

CO2: Practice on manufacturing of components using workshop trades including fitting, carpentry, and foundry and welding. [K3]

CO3: Apply fitting operations in various fitting exercises. [K3]

CO4: Apply basic electrical engineering knowledge for House Wiring Practice [K3]

SYLLABUS

1. **Demonstration:** Safety practices and precautions to be observed in workshop.
2. **Wood Working:** Familiarity with different types of woods and tools used in wood working and make following joints.
 - a) Half – Lap joint b) Mortise and Tenon joint c) Corner Dovetail joint or Bridle joint
3. **Sheet Metal Working:** Familiarity with different types of tools used in sheet metal working, Developments of following sheet metal job from GI sheets.
 - a) Tapered tray b) Conical funnel c) Elbow pipe d) Brazing
4. **Fitting:** Familiarity with different types of tools used in fitting and do the following fitting exercises.
 - a) V-fit b) Dovetail fit c) Semi-circular fit d) Bicycle tire puncture and change of two-wheeler tyre.
5. **Electrical Wiring:** Familiarity with different types of basic electrical circuits and make the following connections.
 - a) Parallel and series b) Two-way switch c) Godown lighting d) Tube light e) Three phase motor f) Soldering of wires
6. **Foundry Trade:** Demonstration and practice on Moulding tools and processes, Preparation of Green Sand Moulds for given Patterns.
7. **Welding Shop:** Demonstration and practice on Arc Welding and Gas welding. Preparation of Lap joint and Butt joint.
8. **Plumbing:** Demonstration and practice of Plumbing tools, Preparation of Pipe joints with coupling for same diameter and with reducer for different diameters.

9. **Basic repairs of Two-wheeler vehicle** – Demonstration of working of two-wheeler vehicle and its repairs.

Textbooks:

1. Basic Workshop Technology: Manufacturing Process, Felix W.; Independently Published, 2019. Workshop Processes, Practices and Materials; Bruce J. Black, Routledge publishers, 5th Edn. 2015.
2. A Course in Workshop Technology Vol I. & II, B.S. Raghuvanshi, Dhanpath Rai & Co., 2015 & 2017.

Reference Books:

1. Elements of Workshop Technology, Vol. I by S. K. Hajra Choudhury & Others, Media Promoters and Publishers, Mumbai. 2007, 14th edition
2. Workshop Practice by H. S. Bawa, Tata-McGraw Hill, 2004.
3. Wiring Estimating, Costing and Contracting; Soni P.M. & Upadhyay P.A.; Atul Prakashan, 2021-22.

CO PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2		3										2	
CO2	2		3		2								2	
CO3	2		2										2	
CO4	2		3										2	
Avg	2		2.75		2								2	

I SEMESTER	L	T	P	C
		0	0	3

23CS1L01- COMPUTER PROGRAMMING LAB

Course Outcomes:

CO1: Read, understand, and trace the execution of programs written in C language.

CO2: Select the right control structure for solving the problem.

CO3: Develop C programs which utilize memory efficiently using programming constructs like pointers.

CO4: Develop, Debug and Execute programs to demonstrate the applications of arrays, functions, basic concepts of pointers in C.

UNIT- I**WEEK 1**

Objective: Getting familiar with the programming environment on the computer and writing the first program.

Suggested Experiments/Activities:

Tutorial 1: Problem-solving using Computers.

Lab1: Familiarization with programming environment

- i) Basic Linux environment and its editors like Vi, Vim & Emacs etc.
- ii) Exposure to Turbo C, gcc
- iii) Writing simple programs using printf(), scanf()

WEEK 2

Objective: Getting familiar with how to formally describe a solution to a problem in a series of finite steps both using textual notation and graphic notation.

Suggested Experiments /Activities:

Tutorial 2: Problem-solving using Algorithms and Flow charts.

Lab 1: Converting algorithms/flow charts into C Source code.

Developing the algorithms/flowcharts for the following sample programs

- i) Sum and average of 3 numbers
- ii) Conversion of Fahrenheit to Celsius and vice versa
- iii) Simple interest calculation

WEEK 3

Objective: Learn how to define variables with the desired data-type, initialize them with appropriate values and how arithmetic operators can be used with variables and constants.

Suggested Experiments/Activities:

Tutorial 3: Variable types and type conversions:

Lab 3: Simple computational problems using arithmetic expressions.

- i) Finding the square root of a given number
- ii) Finding compound interest
- iii) Area of a triangle using heron's formulae
- iv) Distance travelled by an object

UNIT-II**WEEK 4**

Objective: Explore the full scope of expressions, type-compatibility of variables & constants and operators used in the expression and how operator precedence works.

Suggested Experiments/Activities:

Tutorial4: Operators and the precedence and as associativity:

Lab4: Simple computational problems using the operator' precedence and associativity

- i) Evaluate the following expressions.
 - a. $A+B*C+(D*E) + F*G$
 - b. $A/B*C-B+A*D/3$
 - c. $A+++B---A$
 - d. $J=(i++) + (++i)$
- ii) Find the maximum of three numbers using conditional operator
- iii) Take marks of 5 subjects in integers, and find the total, average in float

WEEK 5

Objective: Explore the full scope of different variants of “if construct” namely if-else, nullelse, if-else if*-else, switch and nested-if including in what scenario each one of them can be used and how to use them. Explore all relational and logical operators while writing conditionals for “if construct”.

Suggested Experiments/Activities:

Tutorial 5: Branching and logical expressions:

Lab 5: Problems involving if-then-else structures.

- i) Write a C program to find the max and min of four numbers using if-else.
- ii) Write a C program to generate electricity bill.
- iii) Find the roots of the quadratic equation.
- iv) Write a C program to simulate a calculator using switch case.
- v) Write a C program to find the given year is a leap year or not.

WEEK 6

Objective: Explore the full scope of iterative constructs namely while loop, do-while loop and for loop in addition to structured jump constructs like break and continue including when each of these statements is more appropriate to use.

Suggested Experiments/Activities:

Tutorial 6: Loops, while and for loops

Lab 6: Iterative problems e.g., the sum of series

- i) Find the factorial of given number using any loop.
- ii) Find the given number is a prime or not.
- iii) Compute sine and cos series
- iv) Checking a number palindrome
- v) Construct a pyramid of numbers.

UNIT -III

WEEK 7:

Objective: Explore the full scope of Arrays construct namely defining and initializing 1-D and 2-D and more generically n-D arrays and referencing individual array elements from the defined array. Using integer 1-D arrays, explore search solution linear search.

Suggested Experiments/Activities:

Tutorial 7: 1 D Arrays: searching.

Lab 7: 1D Array manipulation, linear search

- i) Find the min and max of a 1-D integer array.
- ii) Perform linear search on 1D array.
- iii) The reverse of a 1D integer array
- iv) Find 2's complement of the given binary number.
- v) Eliminate duplicate elements in an array.

WEEK 8:

Objective: Explore the difference between other arrays and character arrays that can be used as Strings

by using null character and get comfortable with string by doing experiments that will reverse a string and concatenate two strings. Explore sorting solution bubble sort using integer arrays.

Suggested Experiments/Activities:

Tutorial 8: 2 D arrays, sorting and Strings.

Lab 8: Matrix problems, String operations, Bubble sort

- i) Addition of two matrices
- ii) Multiplication two matrices
- iii) Sort array elements using bubble sort
- iv) Concatenate two strings without built-in functions
- v) Reverse a string using built-in and without built-in string functions

UNIT IV

WEEK 9:

Objective: Explore pointers to manage a dynamic array of integers, including memory allocation & value initialization, resizing changing and reordering the contents of an array and memory de-allocation using malloc (), calloc (), realloc () and free () functions. Gain experience processing command-line arguments received by C

Suggested Experiments/Activities:

Tutorial 9: Pointers, structures and dynamic memory allocation

Lab 9: Pointers and structures, memory dereference.

- i) Write a C program to find the sum of a 1D array using malloc()
- ii) Write a C program to find the total, average of n students using structures
- iii) Enter n students data using calloc() and display failed students list
- iv) Read student name and marks from the command line and display the student details along with the total.
- v) Write a C program to implement realloc()

WEEK 10:

Objective: Experiment with C Structures, Unions, bit fields and self-referential structures (Singly linked lists) and nested structures

Suggested Experiments/Activities:

Tutorial 10: Bitfields, Self-Referential Structures, Linked lists

Lab10: Bitfields, linked lists

Read and print a date using dd/mm/yyyy format using bit-fields and differentiate the same without using bit- fields

- i) Create and display a singly linked list using self-referential structure.
- ii) Demonstrate the differences between structures and unions using a C program.
- iii) Write a C program to shift/rotate using bitfields.
- iv) Write a C program to copy one structure variable to another structure of the same type.

UNIT V

WEEK 11:

Objective: Explore the Functions, sub-routines, scope and extent of variables, doing some experiments by parameter passing using call by value. Basic methods of numerical integration

Suggested Experiments/Activities:

Tutorial 11: Functions, call by value, scope and extent,

Lab 11: Simple functions using call by value, solving differential equations using Eulers theorem.

- i) Write a C function to calculate NCR value.
- ii) Write a C function to find the length of a string.
- iii) Write a C function to transpose of a matrix.
- iv) Write a C function to demonstrate numerical integration of differential equations using Euler's method

WEEK 12:

Objective: Explore how recursive solutions can be programmed by writing recursive functions that can be invoked from the main by programming at-least five distinct problems that have naturally recursive solutions.

Suggested Experiments/Activities:

Tutorial 12: Recursion, the structure of recursive calls

Lab 12: Recursive functions

- i) Write a recursive function to generate Fibonacci series.
- ii) Write a recursive function to find the lcm of two numbers.
- iii) Write a recursive function to find the factorial of a number.
- iv) Write a C Program to implement Ackermann function using recursion.
- v) Write a recursive function to find the sum of series.

WEEK 13:

Objective: Explore the basic difference between normal and pointer variables, Arithmetic operations using pointers and passing variables to functions using pointers

Suggested Experiments/Activities:

Tutorial 13: Call by reference, dangling pointers

Lab 13: Simple functions using Call by reference, Dangling pointers.

- i) Write a C program to swap two numbers using call by reference.
- ii) Demonstrate Dangling pointer problem using a C program.
- iii) Write a C program to copy one string into another using pointer.
- iv) Write a C program to find no of lowercase, uppercase, digits and other characters using pointers.

WEEK14:

Objective: To understand data files and file handling with various file I/O functions. Explore the differences between text and binary files.

Suggested Experiments/Activities:

Tutorial 14: File handling

Lab 14: File operations

- i) Write a C program to write and read text into a file.
- ii) Write a C program to write and read text into a binary file using fread() and fwrite()
- iii) Copy the contents of one file to another file.
- iv) Write a C program to merge two files into the third file using command-line arguments.
- v) Find no. of lines, words and characters in a file
- vi) Write a C program to print last n characters of a given file.

Textbooks:

1. Ajay Mittal, Programming in C: A practical approach, Pearson.
2. Byron Gottfried, Schaum's Outline of Programming with C, McGraw Hill

Reference Books:

1. Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language, Prentice- Hall of India
2. C Programming, A Problem-Solving Approach, Forouzan, Gilberg, Prasad, CENGAGE

I SEMESTER	L	T	P	C
	0	0	1	0.5

23BS1P01- HEALTH AND WELLNESS, YOGA AND SPORTS

Course Outcomes: After completion of the course the student will be able to

CO1: Understand the importance of yoga and sports for Physical fitness and sound health.

CO2: Demonstrate an understanding of health-related fitness components.

CO3: Compare and contrast various activities that help enhance their health.

CO4: Assess current personal fitness levels.

CO5: Develop Positive Personality

UNIT -I

Concept of health and fitness, Nutrition and Balanced diet, basic concept of immunity Relationship between diet and fitness, Globalization and its impact on health, Body Mass Index (BMI) of all age groups.

Activities:

- i) Organizing health awareness programmes in community ii) Preparation of health profile s
- iii) Preparation of chart for balance diet for all age groups

UNIT- II

Concept of yoga, need for and importance of yoga, origin and history of yoga in Indian context, classification of yoga, Physiological effects of Asanas- Pranayama and meditation, stress management and yoga, Mental health and yoga practice.

Activities:

Yoga practices – Asana, Kriya, Mudra, Bandha, Dhyana, Surya Namaskar

UNIT -III

Concept of Sports and fitness, importance, fitness components, history of sports, Ancient and Modern Olympics, Asian games and Commonwealth games.

Activities:

- i) Participation in one major game and one individual sport viz., Athletics, Volleyball, Basketball, Handball, Football, Badminton, Kabaddi, Kho-kho, Table tennis, Cricket etc.
- Practicing general and specific warm up, aerobics ii) Practicing cardiorespiratory fitness, treadmill, run test, 9 min walk, skipping and running.

Reference Books:

1. Gordon Edlin, Eric Golanty. Health and Wellness, 14th Edn. Jones & Bartlett Learning, 2022
2. T.K.V.Desikachar. The Heart of Yoga: Developing a Personal Practice
3. Archie J.Bahm. Yoga Sutras of Patanjali, Jain Publishing Company, 1993
4. Wiseman, John Lofty, SAS Survival Handbook: The Ultimate Guide to Surviving Anywhere Third Edition, William Morrow Paperbacks, 2014
5. The Sports Rules Book/ Human Kinetics with Thomas Hanlon. -- 3rd ed. Human Kinetics, Inc.2014

General Guidelines:

1. Institutes must assign slots in the Timetable for the activities of Health/Sports/Yoga.
2. Institutes must provide field/facility and offer the minimum of five choices of as many as Games/Sports.
3. Institutes are required to provide sports instructor / yoga teacher to mentor the students.

Evaluation Guidelines:

- Evaluated for a total of 100 marks.
- A student can select 6 activities of his/her choice with a minimum of 01 activity per unit. Each activity shall be evaluated by the concerned teacher for 15 marks, totalling to 90 marks.
- A student shall be evaluated by the concerned teacher for 10 marks by conducting viva voce on the subject.

II SEMESTER	L	T	P	C
	3	-	-	3
23BS2T02- ENGINEERING PHYSICS				

Course Objectives:

To bridge the gap between the Physics in school at 10+2 level and UG level Engineering courses are designed by identifying the importance of the periodic arrangement of atoms in crystalline solids and concepts of quantum mechanics, introduce novel concepts of dielectric and magnetic materials, physics of semiconductors, enlightening the characteristics and applications of lasers and optical fibers.

Course Outcomes:

At the end of the course students are able to

CO1: Familiarize with the basics of crystals and their structures (K1).

CO2: Summarize various types of polarization of Dielectrics and classify the Magnetic materials (K2).

CO3: Explain fundamentals of Quantum mechanics and apply it to one dimensional motion of particles (K3).

CO4: Identify the type of Semiconductor using Hall Effect (K2).

CO5: Describe the characteristics of Lasers and properties of Optical fibers and their applications (K2).

UNIT- I

Crystallography and X-ray diffraction

Crystallography: Space lattice, Basis, Unit Cell and lattice parameters – Bravais Lattices – crystal systems (3D) – coordination number - packing fraction of SC, BCC & FCC

X - ray diffraction: Miller indices – separation between successive (h k l) planes. Bragg's law- X-ray Diffractometer – crystal structure determination by Laue and powder methods

UNIT- II

Dielectric and Magnetic Materials

Dielectric Materials: Introduction - Dielectric polarization - Dielectric polarizability, Susceptibility, Dielectric constant and Displacement Vector – Relation between the electric vectors - Types of polarizations- Electronic (Quantitative), Ionic (Quantitative) and Orientation polarizations (Qualitative) - Lorentz internal field - Clausius- Mossotti equation

Magnetic Materials: Introduction - Magnetic dipole moment - Magnetization-Magnetic susceptibility and permeability – Atomic origin of magnetism - Classification of magnetic materials: Dia, para, Ferro, anti-ferro & Ferri magnetic materials - Domain concept for Ferromagnetism & Domain walls (Qualitative) - Hysteresis - soft and hard magnetic materials.

UNIT- III

Quantum Mechanics and Free electron Theory

Quantum Mechanics: Dual nature of matter – Heisenberg's Uncertainty Principle – Significance and properties of wave function – Schrodinger's time independent and dependent wave equations– Particle in a one-dimensional infinite potential well.

Free Electron Theory: Classical free electron theory (Qualitative with discussion of merits and demerits) – Quantum free electron theory – electrical conductivity based on quantum free electron theory - Fermi-Dirac distribution-Density of states - Fermi energy

UNIT- IV

Semiconductors

Semiconductors: Formation of energy bands – classification of crystalline solids - Intrinsic semiconductors: Density of charge carriers – Electrical conductivity – Fermi level – Extrinsic

semiconductors: density of charge carriers – dependence of Fermi energy on carrier concentration and temperature - Drift and diffusion currents – Einstein’s equation – Hall effect and its applications.

UNIT-V

Lasers and Optical Fibers

Lasers: Introduction– Characteristics of lasers – Spontaneous and Stimulated emission of radiation – Einstein’s coefficients – Population inversion - Three level and four level laser pumping schemes - Ruby laser – Helium-Neon laser- Applications of Laser.

Optical Fibers: Introduction to Optical fibers- Critical angle of propagation- Total internal reflection- Acceptance angle and acceptance cone- Numerical aperture- Classification of optical fibers based on refractive index profile-Classification of optical fibers based on modes- Attenuation in optical fibers and Applications of optical fibers.

Textbooks:

1. A Text book of Engineering Physics, M. N. Avadhanulu, P.G.Kshirsagar & TVS ArunMurthy, S. Chand Publications, 11th Edition 2019.
2. Engineering Physics - D.K.Bhattacharya and Poonam Tandon, Oxford press (2015)

Reference Books:

1. Engineering Physics - B.K. Pandey and S. Chaturvedi, Cengage Learning 2021.
2. Engineering Physics - Shatendra Sharma, Jyotsna Sharma, Pearson Education, 2018.
3. Engineering Physics” - Sanjay D. Jain, D. Sahasrabudhe and Girish, University Press.2010
4. Engineering Physics - M.R. Srinivasan, New Age international publishers (2009).

Web Resources: <https://www.loc.gov/rr/scitech/selected-internet/physics.html>

II SEMESTER	L	T	P	C
	3	-	-	3

23BS2T04- DIFFERENTIAL EQUATIONS AND VECTOR CALCULUS

Course Objectives:

- To enlighten the learners in the concept of differential equations and multivariable calculus.
- To furnish the learners with basic concepts and techniques at plus two level to lead them into advanced level by handling various real-world applications.

Course Outcomes:

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

CO1: Solve the differential equations related to various engineering fields.

CO2: Identify solution methods for partial differential equations that model physical processes.

CO3: Interpret the physical meaning of different operators such as gradient, curl and divergence.

CO4: Estimate the work done against a field, circulation and flux using vector calculus.

UNIT-I**Differential equations of first order and first degree**

Linear differential equations – Bernoulli's equations- Exact equations and equations reducible to exact form. Applications: Newton's Law of cooling – Law of natural growth and decay- Electrical circuits.

UNIT-II**Linear differential equations of higher order (Constant Coefficients)**

Definitions, homogenous and non-homogenous, complimentary function, general solution, particular integral, Wronskian, Method of variation of parameters. Simultaneous linear equations, Applications to L-C-R Circuit problems.

UNIT-III**Partial Differential Equations**

Introduction and formation of Partial Differential Equations by elimination of arbitrary constants and arbitrary functions, solutions of first order linear equations using Lagrange's method. Homogeneous Linear Partial differential equations with constant coefficients.

UNIT –IV**Vector differentiation**

Scalar and vector point functions, vector operator Del, Del applies to scalar point functions Gradient, Directional derivative, del applied to vector point functions-Scalar Potential Function-Divergence-Solenoidal and Curl-Irrotational, vector identities.

UNIT-V**Vector integration**

Line intetegral-circulation-work done, Surface integral-flux, Green's theorem in the plane (without proof), Stoke's theorem (without proof), Volume integral, Divergence theorem (without proof) and related problems.

Textbooks:

1. Higher Engineering Mathematics, B. S. Grewal, Khanna Publishers, 2017, 44th Edition

Reference book:

2. Advanced Engineering Mathematics, Erwin Kreyszig, John Wiley & Sons, 2018, 10th Edition

II SEMESTER	L	T	P	C
	3	-	-	3
23EE2T01- BASIC ELECTRICAL AND ELECTRONICS ENGINEERING				

PART – A
BASIC ELECTRICAL ENGINEERING

Course Objectives:

To expose to the field of electrical & electronics engineering, laws and principles of electrical/electronic engineering and to acquire fundamental knowledge in the relevant field)

Course Outcomes:

After the completion of the course students will be able to

CO1. Describe fundamental laws, operating principles of motors/generators, MC/MI instruments

CO2. Demonstrate the working of electrical machines, measuring instruments and power generation stations.

CO3. Apply mathematical tools and fundamental concepts to derive various equations related to electrical circuits and machines.

CO4. Calculate electrical load and electricity bill of residential and commercial buildings

UNIT-I**DC & AC CIRCUITS**

DC Circuits: Electrical circuit elements (R, L and C), Ohm's Law and its limitations, KCL & KVL, series, parallel, series-parallel circuits, Super Position theorem, Simple numerical problems.

AC Circuits: A.C. Fundamentals: Equation of AC Voltage and current, waveform, time period, frequency, amplitude, phase, phase difference, average value, RMS value, form factor, peak factor, Voltage and current relationship with phasor diagrams in R, L, and C circuits, Concept of Impedance, Active power, reactive power and apparent power, Concept of power factor (Simple Numerical problems)..

UNIT-II**MACHINES AND MEASURING INSTRUMENTS**

Machines: Construction, principle and operation of (i) DC Motor, (ii) DC Generator, (iii) Single Phase Transformer, (iv) Three Phase Induction Motor and (v) Alternator, Applications of electrical machines.

Measuring Instruments: Construction and working principle of Permanent Magnet Moving Coil (PMMC), Moving Iron (MI) Instruments and Wattmeter.

UNIT-III**ENERGY RESOURCES, ELECTRICITY BILL & SAFETY MEASURES**

Energy Resources: Conventional and non-conventional energy resources; Layout and operation of various Power Generation systems: Hydel, Nuclear, Solar & Wind power generation.

Electricity bill: Power rating of household appliances including air conditioners, PCs, Laptops, Printers, etc. Definition of "unit" used for consumption of electrical energy, two-part electricity tariff, calculation of electricity bill for domestic consumers.

Equipment Safety Measures: Working principle of Fuse and Miniature circuit breaker (MCB), merits and demerits. Personal safety measures: Electric Shock, Earthing and its types, Safety Precautions to

avoid shock.

Textbooks:

1. Basic Electrical Engineering, D. C. Kulshreshtha, Tata McGraw Hill, 2019, First Edition.
2. Power System Engineering, P.V. Gupta, M.L. Soni, U.S. Bhatnagar and A. Chakrabarti, DhanpatRai& Co, 2013.
3. Fundamentals of Electrical Engineering, Rajendra Prasad, PHI publishers, 2014, Third Edition.

REFERENCE BOOKS:

1. Basic Electrical Engineering, D. P. Kothari and I. J. Nagrath, McGraw Hill, 2019, Fourth Edition.
2. Principles of Power Systems, V.K. Mehtha, S.Chand Technical Publishers, 2020.
3. Basic Electrical Engineering, T. K. Nagsarkar and M. S. Sukhija, Oxford University Press, 2017.
4. Basic Electrical and Electronics Engineering, S. K.Bhattacharya, Person Publications, 2018, Second Edition.

PART – B
BASIC ELECTRONICS ENGINEERING

Course Objectives:

To teach the fundamentals of semiconductor devices and its applications, principles of digital electronics

Course Outcomes:

After the completion of the course students will be able to

- CO1:** Describe the characteristics of various semiconductor devices (K1)
- CO2:** Construct voltage regulators and rectifiers (K3)
- CO3:** Able to explain about the amplifiers and instrumentation system (K2)
- CO4:** Able to explain the behavior of combinational and sequential circuits (K4)

UNIT-I

SEMICONDUCTOR DEVICES

Introduction - Introduction to Electronics, characteristics of PN Junction Diode, Zener Diode — Zener break down and its Characteristics, LED and its characteristics

Transistors — BJT- CB, CE, CC Configurations and Characteristics, FET-Characteristics, Transistor acts as Amplifier.

UNIT-II

BASIC ELECTRONIC CIRCUITS AND INSTRUMENTATION

Rectifiers and power supplies: Block diagram description of a dc power supply, working of rectifiers (HW, FW & Bridge), capacitor filter (no analysis), working of simple zener voltage regulator.

Amplifiers: Block diagram of Public Address system, Circuit diagram and working of RC coupled amplifier with its frequency response. (no analysis)

Electronic Instrumentation: Block diagram of an electronic instrumentation system.

UNIT-III

DIGITALELECTRONICS

Number Systems & Logic Gates: Overview of Number Systems, Logic gates NOT, OR, AND, NOR, NAND, XOR, XNOR Truth Tables and Functionality, BCD codes, Excess-3code, Gray code,

Boolean Algebra: Basic Theorems and properties of Boolean Algebra, Simple combinational circuits– Half and Full Adders & Half and Full Subtractor

Introduction to sequential circuits, Flip flops, Registers and counters (Elementary Treatment only)

Textbooks:

1. R. L. Boylestad & Louis Nashlesky, Electronic Devices & Circuit Theory, Pearson Education, 2021.
2. R. P. Jain, Modern Digital Electronics, 4th Edition, Tata Mc Graw Hill, 2009

Reference Books:

1. R. S. Sedha, A Textbook of Electronic Devices and Circuits, S. Chand & Co, 2010.
2. Santiram Kal, Basic Electronics- Devices, Circuits and IT Fundamentals, Prentice Hall, India, 2002.
3. R. T. Paynter, Introductory Electronic Devices & Circuits – Conventional Flow Version, Pearson Education,2009.

II SEMESTER	L	T	P	C
	1	-	4	3
23ME2T02- ENGINEERING GRAPHICS				

Course Objectives:

To enable the students with various concepts like dimensioning, conventions and standards related to Engineering Drawing

1. To impart knowledge on the projection of points, lines and plane surfaces
2. To improve the visualization skills for better understanding of projection of solids
3. To develop the imaginative skills of the students required to understand Section of solids and Developments of surfaces.
4. To make the students understand the viewing perception of a solid object in Isometric and Perspective projections.

Course Outcomes:

CO1: Understand the principles of engineering drawing, including engineering curves, scales, orthographic and isometric projections.

CO2: Draw and interpret orthographic projections of points, lines, planes and solids in front, top and side views.

CO3: Understand and draw projection of solids in various positions in first quadrant.

CO4: Explain principles behind development of surfaces.

CO5: Prepare isometric and perspective sections of simple solids.

UNIT-I

Introduction: Lines, Lettering and Dimensioning, Geometrical Constructions and Constructing regular polygons by general methods.

Curves: construction of ellipse, parabola and hyperbola by general, Cycloids, Involutives, Normal and tangent to Curves.

Scales: Plain scales and diagonal scales.

UNIT-II

Orthographic Projections: Reference plane, importance of reference lines or Plane, Projections of a point situated in any one of the four quadrants.

Projections of Straight Lines: Projections of straight lines parallel to both reference planes, perpendicular to one reference plane and parallel to other reference plane, inclined to one reference plane and parallel to the other reference plane. Projections of Straight Line Inclined to both the reference planes

Projections of Planes: regular planes Perpendicular to both reference planes, parallel to one reference plane and inclined to the other reference plane; plane inclined to both the reference planes.

UNIT-III

Projections of Solids: Types of solids: Polyhedra and Solids of revolution.

Projections of solids in simple positions: Axis perpendicular to horizontal plane, Axis perpendicular to vertical plane and Axis parallel to both the reference planes, Projection of Solids with axis inclined to one reference plane and parallel to another plane.

UNIT-IV

Sections of Solids: Sections of solids in simple position only.

Development of Surfaces: Methods of Development: Parallel line development and radial line development. Development of a cube, prism, cylinder, pyramid and cone.

UNIT-V

Conversion of Views: Conversion of isometric views to orthographic views; Conversion of orthographic views to isometric views.

Computer graphics: Creating 2D&3D drawings of objects including PCB and Transformations using Auto CAD (Not for end examination).

Textbook:

1. N. D. Bhatt, Engineering Drawing, Charotar Publishing House, 2016.

Reference Books:

1. Engineering Drawing, K.L. Narayana and P. Kanniah, Tata McGraw Hill, 2013.
2. Engineering Drawing, M.B.Shah and B.C. Rana, Pearson Education Inc,2009.
3. Engineering Drawing with an Introduction to AutoCAD, Dhananjay Jolhe, Tata McGraw Hill, 2017.

II SEMESTER	L	T	P	C
	3	-	-	3

23CS2T01- DATA STRUCTURES

Course Objectives:

- To provide the knowledge of basic data structures and their implementations.
- To understand importance of data structures in context of writing efficient programs.
- To develop skills to apply appropriate data structures in problem solving.

Course Outcomes: At the end of the course, Student will be able to

CO1: Acquire the basic concepts of data structures in organizing and accessing data efficiently in algorithms. (K2)

CO2: Design, implement, and apply linked lists for dynamic data storage, demonstrating understanding of memory allocation. (K3)

CO3: Develop programs using stacks to handle recursive algorithms, manage program states, and solve related problems. (K3)

CO4: Apply queue-based algorithms for efficient task scheduling and breadth-first traversal in graphs and distinguish between deques and priority queues, and apply them appropriately to solve data management challenges. (K4)

CO5: Devise novel solutions to small scale programming challenges involving data structures such as stacks, queues, Trees. (K3)

CO6: Design hash-based solutions for specific problems. (K2)

UNIT-I

Introduction to Data Structures: Definition and classification of data structures, Abstract data types (ADTs) and their implementation. Overview of time and space complexity analysis. Searching Techniques: Linear & Binary Search, Sorting Techniques: Bubble sort, Selection sort, Insertion Sort, Quick Sort and Merge Sort. Time complexity analysis for sorting techniques.

UNIT-II

Linked Lists: Singly linked lists: representation and operations, doubly linked lists and circular linked lists, Comparing arrays and linked lists, Applications of linked lists.

UNIT-III

Stacks: Introduction to stacks: properties and operations, implementing stacks using arrays and linked lists, Applications of stacks: Arithmetic expression conversions, expression evaluation, backtracking, reversing list and factorial calculation.

UNIT-IV

Queues: Introduction to queues: properties and operations, implementing queues using arrays and linked lists, circular queue and priority queue Applications of queues in breadth-first search, scheduling.

Deques: Introduction to deques (double-ended queues), Operations on deques and their applications.

UNIT -V

Trees: Introduction to Trees, properties of Trees, Binary Search Tree – Insertion, Deletion & Traversal

Graphs: Introduction to graphs, properties, representation, Traversal

Hashing: Brief introduction to hashing and hash functions, Collision resolution techniques: chaining and open addressing, Hash tables: basic implementation and operations, Applications of hashing in

unique identifier generation, caching.

Textbooks:

1. Data Structures and algorithm analysis in C, Mark Allen Weiss, Pearson, 2nd Edition.
2. Fundamentals of data structures in C, Ellis Horowitz, Sartaj Sahni, Susan Anderson-Freed, Silicon Press, 2008

Reference Books:

1. Algorithms and Data Structures: The Basic Toolbox by Kurt Mehlhorn and Peter Sanders
2. C Data Structures and Algorithms by Alfred V. Aho, Jeffrey D. Ullman, and John E. Hopcroft
3. Problem Solving with Algorithms and Data Structures" by Brad Miller and David Ranum
4. Introduction to Algorithms by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein
5. Algorithms in C, Parts 1 -5 (Bundle): Fundamentals, Data Structures, Sorting, Searching, and Graph Algorithms" by Robert Sedgewick

II SEMESTER	L	T	P	C
	-	-	2	1

23CS2L02- IT WORKSHOP

Course Outcomes:

CO1: Perform Hardware troubleshooting.

CO2: Understand Hardware components and inter dependencies.

CO3: Safeguard computer systems from viruses/worms.

CO4: Document/ Presentation preparation.

CO5: Perform calculations using spreadsheets.

PC Hardware & Software Installation

Task 1: Identify the peripherals of a computer, components in a CPU and its functions. Draw the block diagram of the CPU along with the configuration of each peripheral and submit to your instructor.

Task 2: Every student should disassemble and assemble the PC back to working condition. Lab instructors should verify the work and follow it up with a Viva. Also students need to go through the video which shows the process of assembling a PC. A video would be given as part of the course content.

Task 3: Every student should individually install MS windows on the personal computer. Lab instructor should verify the installation and follow it up with a Viva.

Task 4: Every student should install Linux on the computer. This computer should have windows installed. The system should be configured as dual boot (VMWare) with both Windows and Linux. Lab instructors should verify the installation and follow it up with a Viva

Task 5: Every student should install BOSS on the computer. The system should be configured as dual boot (VMWare) with both Windows and BOSS. Lab instructors should verify the installation and follow it up with a Viva

Internet & World Wide Web

Task1: Orientation & Connectivity Boot Camp: Students should get connected to their Local Area Network and access the Internet. In the process they configure the TCP/IP setting. Finally students should demonstrate, to the instructor, how to access the websites and email. If there is no internet connectivity preparations need to be made by the instructors to simulate the WWW on the LAN.

Task 2: Web Browsers, Surfing the Web: Students customize their web browsers with the LAN proxy settings, bookmarks, search toolbars and pop up blockers. Also, plug-ins like Macromedia Flash and JRE for applets should be configured.

Task 3: Search Engines & Netiquette: Students should know what search engines are and how to use the search engines. A few topics would be given to the students for which they need to search on Google. This should be demonstrated to the instructors by the student.

Task 4: Cyber Hygiene: Students would be exposed to the various threats on the internet and would be asked to configure their computer to be safe on the internet. They need to customize their browsers to block pop ups, block active x downloads to avoid viruses and/or worms.

LaTeX and WORD

Task 1: Word Orientation: The mentor needs to give an overview of La TeX and Microsoft (MS) office

or equivalent (FOSS) tool word: Importance of La TeX and MS office or equivalent (FOSS) tool Word as word Processors, Details of the four tasks and features that would be covered in each, Using La TeX and word – Accessing, overview of toolbars, saving files, Using help and resources, rulers, format painter in word.

Task 2: Using La TeX and Word to create a project certificate. Features to be covered:- Formatting Fonts in word, Drop Cap in word, Applying Text effects, Using Character Spacing, Borders and Colors, Inserting Header and Footer, Using Date and Time option in both La TeX and Word.

Task 3: Creating project abstract Features to be covered:-Formatting Styles, Inserting table, Bullets and Numbering, Changing Text Direction, Cell alignment, Footnote, Hyperlink, Symbols, Spell Check, Track Changes.

Task 4: Creating a Newsletter: Features to be covered:- Table of Content, Newspaper columns, Images from files and clipart, Drawing toolbar and Word Art, Formatting Images, Textboxes, Paragraphs and Mail Merge in word.

EXCEL

Excel Orientation: The mentor needs to tell the importance of MS office or equivalent (FOSS) tool Excel as a Spreadsheet tool, give the details of the four tasks and features that would be covered in each. Using Excel – Accessing, overview of toolbars, saving excel files, Using help and resources.

Task 1: Creating a Scheduler - Features to be covered: Gridlines, Format Cells, Summation, auto fill, Formatting Text

Task 2: Calculating GPA -. Features to be covered:- Cell Referencing, Formulae in excel – average, std. deviation, Charts, Renaming and Inserting worksheets, Hyper linking, Count function,

LOOKUP/VLOOKUP

Task 3: Split cells, freeze panes, group and outline, Sorting, Boolean and logical operators, Conditional formatting

POWER POINT

Task 1: Students will be working on basic power point utilities and tools which help them create basic power point presentations. PPT Orientation, Slide Layouts, Inserting Text, Word Art, Formatting Text, Bullets and Numbering, Auto Shapes, Lines and Arrows in PowerPoint.

Task 2: Interactive presentations - Hyperlinks, Inserting –Images, Clip Art, Audio, Video, Objects, Tables and Charts.

Task 3: Master Layouts (slide, template, and notes), Types of views (basic, presentation, slide slotter, notes etc), and Inserting – Background, textures, Design Templates, Hidden slides.

AI TOOLS – ChatGPT

Task 1: Prompt Engineering: Experiment with different types of prompts to see how the model responds. Try asking questions, starting conversations, or even providing incomplete sentences to see how the model completes them.

- Ex: Prompt: "You are a knowledgeable AI. Please answer the following question: What is the capital of France?"

Task 2: Creative Writing: Use the model as a writing assistant. Provide the beginning of a story or a description of a scene, and let the model generate the rest of the content. This can be a fun way to brainstorm creative ideas

- Ex: Prompt: "In a world where gravity suddenly stopped working, people started floating upwards. Write a story about how society adapted to this new reality."

Task 3: Language Translation: Experiment with translation tasks by providing a sentence in one language and asking the model to translate it into another language. Compare the output to see how accurate and fluent the translations are.

- Ex: Prompt: "Translate the following English sentence to French: 'Hello, how are you doing today?'"

Reference Books:

1. Comdex Information Technology course tool kit, Vikas Gupta, WILEY Dream tech, 2003
2. The Complete Computer upgrade and repair book, Cheryl A Schmidt, WILEY Dream tech, 2013, 3rd edition
3. Introduction to Information Technology, ITL Education Solutions limited, Pearson Education, 2012, 2nd edition
4. PC Hardware - A Handbook, Kate J. Chase, PHI (Microsoft)
5. LaTeX Companion, Leslie Lamport, PHI/Pearson.
6. IT Essentials PC Hardware and Software Companion Guide, David Anfinson and Ken Quamme. – CISCO Press, Pearson Education, 3rd edition
7. IT Essentials PC Hardware and Software Labs and Study Guide, Patrick Regan– CISCO Press, Pearson Education, 3rd edition

II SEMESTER	L	T	P	C
	-	-	2	1

23BS2L02- ENGINEERING PHYSICS LAB

Course Outcomes:

The students will be able to

CO1: Estimate the wavelengths of Laser source using diffraction grating (K2).

CO2: Plot the intensity of the magnetic field of circular coil carrying current with distance (K2).

CO3: Evaluate dielectric constant and magnetic susceptibility for dielectric and magnetic materials respectively (K2).

CO4: Calculate the band gap of a given semiconductor (K3).

CO5: Evaluate general and mechanical properties of materials (K2)

List of Experiments:

1. Determination of wavelength of Laser light using diffraction grating.
2. Estimation of Planck's constant using photoelectric effect.
3. Determination of energy gap of a semiconductor using p-n junction diode.
4. Magnetic field along the axis of a current carrying circular coil by Stewart Gee's Method.
5. Determination of temperature coefficients of a thermistor.
6. Determination of Frequency of electrically maintained tuning fork by Melde's experiment.
7. Determination of Numerical aperture and Acceptance angle of a given optical fiber.
8. Determination of the resistivity of semiconductor by four probe method.
9. Study the variation of B versus H by magnetizing the magnetic material (B-H curve).
10. Determination of Frequency of AC supply using sonometer
11. Determination of time constant of CR circuit
12. Study of Characteristic curves (I/V) of a Zener diode to determine its Breakdown voltage.

Note: Any TEN of the listed experiments are to be conducted. Out of which any TWO experiments may be conducted in virtual mode.

References:

- A Textbook of Practical Physics - S. Balasubramanian, M.N. Srinivasan, S. Chand Publishers, 2017.

Web Resources

- www.vlab.co.in
- <https://phet.colorado.edu/en/simulations/filter?subjects=physics&type=html,prototype>

II SEMESTER	L	T	P	C
	-	-	2	1

23BS2L02- ELECTRICAL & ELECTRONICS ENGINEERING WORKSHOP

Course Outcomes:

After completion of this course, the student will be able to

CO1. Measure voltage, current and power in an electrical circuit. (L3)

CO2. Measure of Resistance using Wheat stone bridge (L4)

CO3. Discover critical field resistance and critical speed of DC shunt generators. (L4)

CO4. Investigate the effect of reactive power and power factor in electrical loads. (L5)

Activities:

1. Familiarization of commonly used Electrical & Electronic Workshop Tools: Bread board, Solder, cables, relays, switches, connectors, fuses, Cutter, plier, screwdriver set, wire stripper, flux, knife/blade, soldering iron, de-soldering pump etc.
 - Provide some exercises so that hardware tools and instruments are learned to be used by the students.
2. Familiarization of Measuring Instruments like Voltmeters, Ammeters, multimeter, LCR-Q meter, Power Supplies, CRO, DSO, Function Generator, Frequency counter.
 - Provide some exercises so that measuring instruments are learned to be used by the students.
3. Components:
 - Familiarization/Identification of components (Resistors, Capacitors, Inductors, Diodes, transistors, IC's etc.) – Functionality, type, size, colour coding package, symbol, cost etc.
 - Testing of components like Resistor, Capacitor, Diode, Transistor, ICs etc. - Compare values of components like resistors, inductors, capacitors etc with the measured values by using instruments

PART A: ELECTRICAL ENGINEERING LAB**List of experiments:**

1. Verification of KCL and KVL
2. Verification of Superposition theorem
3. Measurement of Resistance using Wheat stone bridge
4. Magnetization Characteristics of DC shunt Generator
5. Measurement of Power and Power factor using Single-phase wattmeter
6. Measurement of Earth Resistance using Megger
7. Calculation of Electrical Energy for Domestic Premises

Reference Books:

1. Basic Electrical Engineering, D. C. Kulshreshtha, Tata McGraw Hill, 2019, First Edition
2. Power System Engineering, P.V. Gupta, M.L. Soni, U.S. Bhatnagar and A. Chakrabarti, Dhanpat Rai & Co, 2013
3. Fundamentals of Electrical Engineering, Rajendra Prasad, PHI publishers, 2014, Third Edition

Note: Minimum Six Experiments to be performed.

PART B: ELECTRONICS ENGINEERING LAB

Course Objectives:

- To impart knowledge on the principles of digital electronics and fundamentals of electron devices & its applications.

Course Outcomes: At the end of the course, the student will be able to

CO1: Identify & testing of various electronic components.

CO2: Understand the usage of electronic measuring instruments.

CO3: Plot and discuss the characteristics of various electron devices.

CO4: Explain the operation of a digital circuit.

List of Experiments:

1. Plot V-I characteristics of PN Junction diode A) Forward bias B) Reverse bias.
2. Plot V – I characteristics of Zener Diode and its application as voltage Regulator.
3. Implementation of half wave and full wave rectifiers
4. Plot Input & Output characteristics of BJT in CE and CB configurations
5. Frequency response of CE amplifier.
6. Simulation of RC coupled amplifier with the design supplied
7. Verification of Truth Table of AND, OR, NOT, NAND, NOR, Ex-OR, Ex-NOR gates using ICs.
8. Verification of Truth Tables of S -R, J -K& D flip flops using respective ICs.

Tools / Equipment Required: DC Power supplies, Multi meters, DC Ammeters, DC Voltmeters, AC Voltmeters, CROs, all the required active devices.

References:

1. R. L. Boylestad & Louis Nashlesky, Electronic Devices & Circuit Theory, Pearson Education, 2021.
2. R. P. Jain, Modern Digital Electronics, 4th Edition, Tata Mc Graw Hill, 2009
3. R. T. Paynter, Introductory Electronic Devices & Circuits – Conventional Flow Version, Pearson Education, 2009.

Note: Minimum Six Experiments to be performed. All the experiments shall be implemented using both Hardware and Software.

II SEMESTER	L	T	P	C
	-	-	3	1.5

23CS2L01- DATA STRUCTURES LAB

Course Objectives:

The course aims to strengthen the ability of the students to identify and apply the suitable data structure for the given real-world problem. It enables them to gain knowledge in practical applications of data structures.

Course Outcomes: At the end of the course, Student will be able to

CO1: Acquire the basic concepts of data structures in organizing and accessing data efficiently in algorithms. (K2)

CO2: Design, implement, and apply linked lists for dynamic data storage, demonstrating understanding of memory allocation. (K4)

CO3: Develop programs using stacks to handle recursive algorithms, manage program states, and solve related problems. (K3)

CO4: Apply queue-based algorithms for efficient task scheduling and breadth-first traversal in graphs and distinguish between deques and priority queues and apply them appropriately to solve data management challenges. (K3)

CO5: Design hash-based solutions for specific problems. (K4)

List of Experiments:**Exercise 1: Array Manipulation**

- i) Write a program to reverse an array.
- ii) C Programs to implement the Searching Techniques – Linear & Binary Search
- iii) C Programs to implement Sorting Techniques – Bubble, Selection, Insertion Sort, Quick sort and Merge sort.

Exercise 2: Linked List Implementation

- i) Implement a singly linked list and perform insertion and deletion operations.
- ii) Develop a program to reverse a linked list iteratively and recursively.
- iii) Solve problems involving linked list traversal and manipulation.

Exercise 3: Linked List Applications

- i) Create a program to detect and remove duplicates from a linked list.
- ii) Implement a linked list to represent polynomials and perform addition.
- iii) Implement a double-ended queue (deque) with essential operations.

Exercise 4: Double Linked List Implementation

- i) Implement a doubly linked list and perform various operations to understand its properties and applications.
- ii) Implement a circular linked list and perform insertion, deletion, and traversal.

Exercise 5: Stack Operations

- i) Implement a stack using arrays and linked lists.

- ii) Write a program to evaluate a postfix expression using a stack.
- iii) Implement a program to check for balanced parentheses using a stack.

Exercise 6: Queue Operations

- i) Implement a queue using arrays and linked lists.
- ii) Develop a program to simulate a simple printer queue system.
- iii) Solve problems involving circular queues.

Exercise 7: Stack and Queue Applications

- i) Use a stack to evaluate an infix expression and convert it to postfix.
- ii) Create a program to determine whether a given string is a palindrome or not.
- iii) Implement a stack or queue to perform comparison and check for symmetry.

Exercise 8: Binary Search Tree

- i) Implementing a BST using Linked List.
- ii) Traversing of BST.

Exercise 9: Hashing

- i) Implement a hash table with collision resolution techniques.
- ii) Write a program to implement a simple cache using hashing.

Textbooks:

1. Data Structures and algorithm analysis in C, Mark Allen Weiss, Pearson, 2nd Edition.
2. Fundamentals of data structures in C, Ellis Horowitz, Sartaj Sahni, Susan AndersonFreed, Silicon Press, 2008

Reference Books:

1. Algorithms and Data Structures: The Basic Toolbox by Kurt Mehlhorn and Peter Sanders
2. C Data Structures and Algorithms by Alfred V. Aho, Jeffrey D. Ullman, and John E. Hopcroft
3. Problem Solving with Algorithms and Data Structures" by Brad Miller and David Ranum
4. Introduction to Algorithms by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein
5. Algorithms in C, Parts 1-5 (Bundle): Fundamentals, Data Structures, Sorting, Searching, and Graph Algorithms by Robert Sedgewick.

II SEMESTER	L	T	P	C
	-	-	1	0.5

23BS2P02- NSS/NCC/SCOUTS & GUIDES/COMMUNITY SERVICE

Course Outcomes: After completion of the course the students will be able to

CO1: Understand the importance of discipline, character and service motto.

CO2: Solve some societal issues by applying acquired knowledge, facts, and techniques.

CO3: Explore human relationships by analyzing social problems.

CO4: Determine to extend their help for the fellow beings and downtrodden people.

CO5: Develop leadership skills and civic responsibilities.

UNIT I

Orientation

General Orientation on NSS/NCC/ Scouts & Guides/Community Service activities, career guidance.

Activities:

- Conducting –ice breaking sessions-expectations from the course-knowing personal talents and skills
- Conducting orientations programs for the students –future plans-activities-releasing road map etc.
- Displaying success stories-motivational biopics- award winning movies on societal issues etc.
- Conducting talent show in singing patriotic songs-paintings- any other contribution.

UNIT II

Nature & Care

Activities:

- Best out of waste competition.
- Poster and signs making competition to spread environmental awareness.
- Recycling and environmental pollution article writing competition.
- Organizing Zero-waste day.
- Digital Environmental awareness activity via various social media platforms.
- Virtual demonstration of different eco-friendly approaches for sustainable living.
- Write a summary on any book related to environmental issues.

UNIT III

Community Service Activities:

- Conducting One Day Special Camp in a village contacting village-area leaders- Survey in the village, identification of problems- helping them to solve via media- authoritiesexperts-etc.
- Conducting awareness programs on Health-related issues such as General Health, Mental health, Spiritual Health, HIV/AIDS, iii) Conducting consumer Awareness. Explaining various legal provisions etc.
- Women Empowerment Programmes- Sexual Abuse, Adolescent Health and Population Education.
- Any other programmes in collaboration with local charities, NGOs etc.

Reference Books:

- Nirmalya Kumar Sinha & Surajit Majumder, *A Text Book of National Service Scheme Vol;I*, Vidya Kutir Publication, 2021 (ISBN 978-81-952368-8-6)

2. *Red Book - National Cadet Corps – Standing Instructions Vol I & II*, Directorate General of NCC, Ministry of Defence, New Delhi
3. Davis M. L. and Cornwell D. A., “Introduction to Environmental Engineering”, McGraw Hill, New York 4/e 2008
4. Masters G. M., Joseph K. and Nagendran R. “Introduction to Environmental Engineering and Science”, Pearson Education, New Delhi. 2/e 2007
5. Ram Ahuja. *Social Problems in India*, Rawat Publications, New Delhi.

General Guidelines:

1. Institutes must assign slots in the Timetable for the activities.
2. Institutes are required to provide instructor to mentor the students.

Evaluation Guidelines:

- Evaluated for a total of 100 marks.
- A student can select 6 activities of his/her choice with a minimum of 01 activity per unit. Each activity shall be evaluated by the concerned teacher for 15 marks, totalling to 90 marks.
- A student shall be evaluated by the concerned teacher for 10 marks by conducting viva voce on the subject.

III SEMESTER	L	T	P	C
	3	-	-	3
23BS3T03-DISCRETE MATHEMATICS AND GRAPH THEORY				

Course Objectives:

- To introduce the students to the topics and techniques of discrete methods and combinatorial reasoning.
- To introduce a wide variety of applications. The algorithmic approach to the solution of problems is fundamental in discrete mathematics, and this approach reinforces the close ties between this discipline and the area of computer science.

Course Outcomes: At the end of the course students will be able to

1. Build skills in solving mathematical problems (L3)
2. Comprehend mathematical principles and logic (L4)
3. Demonstrate knowledge of mathematical modeling and proficiency in using mathematical software (L6)
4. Manipulate and analyze data numerically and/or graphically using appropriate Software (L3)
5. How to communicate effectively mathematical ideas/results verbally or in writing (L1)

UNIT-I: Mathematical Logic:

Propositional Calculus: Statements and Notations, Connectives, Well Formed Formulas, Truth Tables, Tautologies, Equivalence of Formulas, Duality Law, Tautological Implications, Normal Forms, Theory of Inference for Statement Calculus, Consistency of Premises, Indirect Method of Proof, Predicate Calculus: Predicates, Predicative Logic, Statement Functions, Variables and Quantifiers, Free and Bound Variables, Inference Theory for Predicate Calculus.

UNIT-II: Set Theory:

Sets: Operations on Sets, Principle of Inclusion-Exclusion, Relations: Properties, Operations, Partition and Covering, Transitive Closure, Equivalence, Compatibility and Partial Ordering, Hasse Diagrams, Functions: Bijective, Composition, Inverse, Permutation, and Recursive Functions, Lattice and its Properties.

UNIT-III: Combinatorics and Recurrence Relations:

Basis of Counting, Permutations, Permutations with Repetitions, Circular and Restricted Permutations, Combinations, Restricted Combinations, Binomial and Multinomial Coefficients and Theorems.

Recurrence Relations:

Generating Functions, Function of Sequences, Partial Fractions, Calculating Coefficient of Generating Functions, Recurrence Relations, Formulation as Recurrence Relations, Solving Recurrence Relations by Substitution and Generating Functions, Method of Characteristic Roots, Solving Inhomogeneous Recurrence Relations

UNIT-IV: Graph Theory:

Basic Concepts, Graph Theory and its Applications, Subgraphs, Graph Representations: Adjacency and Incidence Matrices, Isomorphic Graphs, Paths and Circuits, Eulerian and Hamiltonian Graphs,

Unit-V: Multi Graphs

Multigraphs, Bipartite and Planar Graphs, Euler's Theorem, Graph Colouring and Covering, Chromatic Number, Spanning Trees, Prim's and Kruskal's Algorithms, BFS and DFS Spanning Trees.

TEXT BOOKS:

1. Discrete Mathematical Structures with Applications to Computer Science, J. P. Tremblay and P. Manohar, Tata McGraw Hill.
2. Elements of Discrete Mathematics-A Computer Oriented Approach, C. L.Liu and D.P. Mohapatra, 3rd Edition, Tata McGraw Hill.
3. Theory and Problems of Discrete Mathematics, Schaum's Outline Series, Seymour Lipschutz and Marc Lars Lipson, 3rd Edition, McGraw Hill.

REFERENCE BOOKS:

1. Discrete Mathematics for Computer Scientists and Mathematicians, J. L.Mott, A. Kandel and T. P. Baker, 2nd Edition, Prentice Hall of India.
2. Discrete Mathematical Structures, Bernand Kolman, Robert C. Busby and Sharon Cutler Ross, PHI.
3. Discrete Mathematics, S. K. Chakraborty and B.K. Sarkar, Oxford, 2011.
4. Discrete Mathematics and its Applications with Combinatorics and Graph Theory, K.H. Rosen, 7th Edition, Tata McGraw Hill.

III SEMESTER	L	T	P	C
	2	-	-	2
23BM3T01-MANAGERIAL ECONOMICS AND FINANCIAL ANALYSIS				

Course Objectives:

- To inculcate the basic knowledge of micro economics and financial accounting
- To make the students learn how demand is estimated for different products, input- out put relationship for optimizing production and cost
- To Know the Various types of market structure and pricing methods and strategy
- To give an overview on investment appraisal methods to promote the students to learn how to plan long-term investment decisions.
- To provide fundamental skills on accounting and to explain the process of preparing financial statements.

Course Outcomes: At the end of the course, student will be able to

- Define the concepts related to Managerial Economics, financial accounting and management.
- Understand the fundamentals of Economics viz., Demand, Production, cost, revenue and markets
- Apply the Concept of Production cost and revenues for effective Business decision
- Analyze how to invest their capital and maximize returns
- Evaluate the capital budgeting techniques
- Develop the accounting statements and evaluate the financial performance of business entity.

UNIT-I

Managerial Economics: Introduction – Nature, meaning, significance, functions, and advantages. Demand-Concept, Function, Law of Demand - Demand Elasticity- Types – Measurement. Demand Forecasting- Factors governing Forecasting, Methods. Managerial Economics and Financial Accounting and Management.

UNIT-II

Production and Cost Analysis: Introduction – Nature, meaning, significance, functions and advantages. Production Function– Least-cost combination– Short run and long run Production Function- Isoquants and Isocosts, MRTS -Cobb-Douglas Production Function - Laws of Returns - Internal and External Economies of scale. Cost & Break-Even Analysis - Cost concepts and Cost behaviour- Break-Even Analysis (BEA) -Determination of Break- Even Point (Simple Problems)-Managerial significance and limitations of Break- Even Analysis.

UNIT-III

Business Organizations and Markets: Introduction–Nature, meaning, significance, functions and advantages. Forms of Business Organizations- Sole Proprietary - Partnership - Joint Stock Companies - Public Sector Enterprises. Types of Markets - Perfect and Imperfect Competition - Features of Perfect Competition Monopoly-Monopolistic Competition– Oligopoly-Price-Output Determination-Pricing Methods and Strategies

UNIT-IV

Capital Budgeting: Introduction – Nature, meaning, significance, functions and advantages. Types of Working Capital, Components, Sources of Short-term and Long-term Capital, Estimating Working capital requirements. Capital Budgeting–Features, Proposals, Methods and Evaluation. Projects– Pay Back Method, Accounting Rate of Return(ARR) Net Present Value(NPV)Internal Rate Return(IRR)

Method (sample problems)

UNIT-V

Financial Accounting and Analysis: Introduction – Nature, meaning, significance, functions and advantages. Concepts and Conventions-Double-Entry Book Keeping, Journal, Ledger, Trial Balance-Final Accounts (Trading Account, Profit and Loss Account and Balance Sheet with simple adjustments). Financial Analysis-Analysis and Interpretation of Liquidity Ratios, Activity Ratios, and Capital structure Ratios and Profitability.

Textbooks:

1. Varshney &Maheswari: Managerial Economics,Sultan Chand,2013.

Reference Books:

1. Managerial Economics: Principles And Worldwide Applications, 9E (Adaptation) by Dominick Salvatore and Siddhartha Rastogi
2. Managerial Economics: Principles and Worldwide Applications by Dominick Salvatore

III SEMESTER	L	T	P	C
	3	-	-	3
23CS3T01-COMPUTER ORGANIZATION				

Course Code	Category	L	T	P	C	I.M	E.M.	Exam
	ES	3	--	--	3	30	70	3 Hrs.
COMPUTER ORGANIZATION								
(For CSE)								
Course Objectives: Students are expected to learn								
1.	Principles and the Implementation of Computer Arithmetic							
2.	Operation of CPUs including RTL, ALU, Instruction Cycle, and Busses							
3.	Functionality of central processing unit and control units							
4.	Memory System and I/O Organization							
Course Outcomes: After completion of the course, the student will be able to								
S.No	Outcome							K L
CO1	Identify set of digital components.							K3
CO2	Demonstrate functional components and micro operations in a basic computer system.							K2
CO3	Demonstrate various instructions and arithmetic operations							K3
CO4	Illustrate knowledge of functional components on central processing unit and various control units.							K2
CO5	Determine different memory components in a computer for better memory organization							K3
CO6	Explain different ways of communication with I/O devices and standard I/O interface							K2
SYLLABUS								
UNIT-I (10 Hrs)	Basic Structure of Computers: Computer Types, Functional units, Basic operational concepts, Bus structures. Register Transfer and Micro operations: Register Transfer Language, Register Transfer, Bus and Memory Transfers, Arithmetic Micro operations, Logic Micro operations, Shift Micro operations, Arithmetic Logic Shift Unit.							
UNIT-II (08 Hrs)	Basic Computer Organization and Design: Instruction Codes, Computer Register, Computer Instructions, Instruction Cycle, Memory – Reference Instructions. Input – Output and Interrupt, Complete Computer Description Computer Arithmetic: Addition and Subtraction of Signed Numbers, Design of Fast Adders, Multiplication of Positive Numbers, Signed-operand Multiplication, Fast Multiplication, Integer Division, Floating-Point Numbers and Operations							
UNIT-III (10 Hrs)	Central Processing Unit: General Register Organization, STACK Organization. Instruction Formats, Addressing Modes, Data Transfer and Manipulation, Execution of a Complete Instruction, Multiple-Bus Organization, Microprogrammed Control: Control Memory, Address Sequencing, Micro Program							

	example, Hardwired Control and Micro programmed Control.
UNIT-IV (10 Hrs)	The Memory Organization: Memory Hierarchy, Main memory, Auxiliary memory, Associate Memory, Cache Memory, and Virtual memory, Memory Management Requirements, Secondary Storage.
UNIT-V (12 Hrs.)	Input / Output Organization: Accessing I/O Devices, Interrupts, Processor Examples, modes of transfers, Direct Memory Access, Buses, Interface Circuits, Standard I/O Interfaces
Text Books:	
1.	Computer System Architecture M. M. Mano., 3rd ed., Prentice Hall of India, New Delhi, 1993
2.	Digital Design, 6th Edition, M. Morris Mano, Pearson Education.
3.	Computer Organization, Carl Hamacher, Zvonko Vranesic, Safwat Zaky, 5/e, McGraw Hill, 2002.
Reference Books:	
1.	Computer Organization and Architecture, William Stallings, 6/e, Pearson, 2006.
2.	Structured Computer Organization, Andrew S. Tanenbaum, 4/e, Pearson, 2005.
3.	Fundamentals of Computer Organization and Design, Sivarama P. Dandamudi, Springer, 2006.
e-Resources	
1.	https://nptel.ac.in/courses/106/105/106105163/
2.	http://www.cuc.ucc.ie/CS1101/David%20Tarnoff.pdf

III SEMESTER	L	T	P	C
	3	-	-	3
23CS3T02-ADVANCED DATA STRUCTURES & ALGORITHMS				

Course Objectives:

The main objectives of the course is to

- provide knowledge on advance data structures frequently used in Computer Sciencedomain
- Develop skills in algorithm design techniques popularly used
- Understand the use of various data structures in the algorithm design

Course Outcomes: At the end of the course students will be able to

CO1: Discover the performance of an algorithm using asymptotic notation. (K2)

CO2: Use divide and conquer technique to solve problems. (K3)

CO3: Understand greedy and dynamic programming techniques to solve efficient solutions for optimization problem. (K3)

CO4: Recognize problems suitable for back tracking, branch and bound solutions.(K1)

CO5: Understand the complexity classes NP-Hard and NP-Complete and solve related decision problems. (K2)

UNIT-I: [15 Hrs.]

Introduction to Algorithm Analysis, Space and Time Complexity analysis, Asymptotic Notations, Recursive functions,

AVL Trees – Creation, Insertion, Deletion operations and Applications

B-trees – Creation, Insertion, Deletion operations and Applications

B+trees - Creation, Insertion, Deletion operations and Applications

UNIT-II: [12 Hrs.]

Heap Trees (Priority Queues)–Min and Max Heaps, Operations and Applications

Graphs–Terminology, Representations, Basic Search and Traversals, Connected Components and Biconnected Components, applications

UNIT-III: [12 Hrs.]

Divide and Conquer: The General Method, Quick Sort, Merge Sort, Heap Sort, Strassen’s matrix multiplication.

Greedy Method: General Method, Job Sequencing with deadlines, Knapsack Problem, Minimum cost spanning trees, Single Source Shortest Paths

UNIT-IV: [15 Hrs.]

Dynamic Programming: General Method, All pairs shortest paths, Single Source Shortest Paths– General Weights (Bellman Ford Algorithm), Optimal Binary Search Trees, 0/1 Knapsack, String Editing.

Backtracking: General Method, 8-Queens Problem, Sum of Subsets problem, Graph Coloring, 0/1 Knapsack Problem

UNIT-V: [14 Hrs.]

Branch and Bound: The General Method, 0/1 Knapsack Problem, Travelling Salesperson problem.

NP Hard and NP Complete Problems: Basic Concepts, Cook's theorem NP Hard Graph Problems: Clique Decision Problem (CDP), Chromatic Number Decision Problem (CNDP), Traveling Salesperson

Textbooks:

1. Fundamentals of Data Structures in C++, Horowitz, Ellis; Sahni, Sartaj; Mehta, Dinesh 2nd Edition Universities Press
2. Computer Algorithms/C++ Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran 2nd Edition University Press

Reference Books:

1. Data Structures and program design in C, Robert Kruse, Pearson Education Asia
2. An introduction to Data Structures with applications, Trembley & Sorenson, McGraw Hill
3. The Art of Computer Programming, Vol.1: Fundamental Algorithms, Donald E Knuth, Addison-Wesley, 1997.
4. Data Structures using C & C++: Langsam, Augenstein & Tanenbaum, Pearson, 1995
5. Algorithms + Data Structures & Programs:, N.Wirth, PHI
6. Fundamentals of Data Structures in C++: Horowitz Sahni & Mehta, Galgottia Pub.
7. Data structures in Java:, Thomas Standish, Pearson Education Asia

Online Learning Resources:

1. https://www.tutorialspoint.com/advanced_data_structures/index.asp
2. <http://peterindia.net/Algorithms.html>
3. Abdul Bari, [1. Introduction to Algorithms \(youtube.com\)](https://www.youtube.com/watch?v=...)
4. <https://www.swarnandhra.ac.in/dsv>
5. bit.ly/BRK_DSV

III SEMESTER	L	T	P	C
	3	-	-	3
23CS3T03-DATABASE MANAGEMENT SYSTEMS				

Course Objectives:

The main objectives of the course is to

- Introduce database management systems and to give a good formal foundation on the relational model of data and usage of Relational Algebra
- Introduce the concepts of basic SQL as a universal Database language
- Demonstrate the principles behind systematic database design approaches by covering conceptual design, logical design through normalization
- Provide an overview of physical design of a database system, by discussing Database indexing techniques and storage techniques

COURSE OUTCOMES: Upon successful completion of this course, students should be able to:

1. Explain the basic concepts of database management system and design an Entity-Relationship (E-R) model and convert E-R model to relational model.
2. Construct database using Relational algebra and SQL.
3. Apply Normalization techniques to normalize the database.
4. Discuss transaction management using different concurrency control protocols and recovery algorithms.
5. Illustrate different file organization and indexing methods.

UNIT I:

Introduction: Database system, Characteristics (Database Vs File System), Database Users, Advantages of Database systems, Database applications. Database languages, Brief introduction of different Data Models; Concepts of Schema, Instance and data independence; Three tier schema architecture for data independence; Database system structure, environment, Centralized and Client Server architecture for the database.

Entity Relationship Model: Introduction, Representation of entities, attributes, entity set, relationship, relationship set, constraints, sub classes, super class, inheritance, specialization, generalization using ER Diagrams.

UNIT II:

Relational Model: Introduction to relational model, concepts of domain, attribute, tuple, relation, importance of null values, constraints (Domain, Key constraints, integrity constraints) and their importance, Relational Algebra, Relational Calculus. **BASIC SQL:** Simple Database schema, data types, table definitions (create, alter), different DML operations (insert, delete, update).

UNIT III:

SQL: Basic SQL querying (select and project) using where clause, arithmetic & logical operations, SQL functions (Date and Time, Numeric, String conversion). Creating tables with relationship, implementation of key and integrity constraints, nested queries, sub queries, grouping, aggregation, ordering, implementation of different types of joins, view (updatable and non-updatable), relational set operations.

UNIT IV:

Schema Refinement (Normalization): Purpose of Normalization or schema refinement, concept of functional dependency, normal forms based on functional dependency Lossless join and dependency preserving decomposition, (1NF, 2NF and 3 NF), concept of surrogate key, Boyce-Codd normal form (BCNF), MVD, Fourth normal form (4NF), Fifth Normal Form (5NF).

UNIT V:

Transaction Concept: Transaction State, ACID properties, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, Testing for Serializability, lock based, time stamp based, optimistic, concurrency protocols, Deadlocks, Failure Classification, Storage, Recovery and Atomicity, Recovery algorithm.

Introduction to Indexing Techniques: B+ Trees, operations on B+ Trees, Hash Based Indexing

Text Books:

- 1) Database Management Systems, 3rd edition, Raghurama Krishnan, Johannes Gehrke, TMH (For Chapters 2, 3, 4)
- 2) Database System Concepts, 5th edition, Silberschatz, Korth, Sudarsan, TMH (For Chapter 1 and Chapter 5)

Reference Books:

- 1) Introduction to Database Systems, 8th edition, C J Date, Pearson.
- 2) Database Management System, 6th edition, Ramez Elmasri, Shamkant B. Navathe, Pearson
- 3) Database Principles Fundamentals of Design Implementation and Management, Corlos Coronel, Steven Morris, Peter Robb, Cengage Learning.

Web-Resources:

- 1) <https://nptel.ac.in/courses/106/105/106105175/>
- 2) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01275806667282022456_shared/overview

III SEMESTER	L	T	P	C
	-	-	3	1.5
23CS3L01-ADVANCED DATA STRUCTURES & ALGORITHMS LAB				

Course Objectives:

The objectives of the course is to

- Acquire practical skills in constructing and managing Data structures
- Apply the popular algorithm design methods in problem-solving scenarios

Experiments covering the Topics:

- Operations on AVL trees, B-Trees, Heap Trees
- Graph Traversals
- Sorting techniques
- Minimum cost spanning trees
- Shortest path algorithms
- 0/1 Knapsack Problem
- Travelling Salesperson problem
- Optimal Binary Search Trees
- N-Queens Problem
- Job Sequencing

Sample Programs:

1. Construct an AVL tree for a given set of elements which are stored in a file. And implement insert and delete operation on the constructed tree. Write contents of tree into a new file using in-order.
2. Construct B-Tree an order of 5 with a set of 100 random elements stored in array. Implement searching, insertion and deletion operations.
3. Construct Min and Max Heap using arrays, delete any element and display the content of the Heap.
4. Implement BFT and DFT for given graph, when graph is represented by
 - a) Adjacency Matrix
 - b) Adjacency Lists
5. Write a program for finding the biconnected components in a given graph.
6. Implement Quick sort and Merge sort and observe the execution time for various input sizes (Average, Worst and Best cases).
7. Compare the performance of Single Source Shortest Paths using Greedy method when the graph is represented by adjacency matrix and adjacency lists.
8. Implement Job Sequencing with deadlines using Greedy strategy.
9. Write a program to solve 0/1 Knapsack problem Using Dynamic Programming.
10. Implement N-Queens Problem Using Backtracking.
11. Use Backtracking strategy to solve 0/1 Knapsack problem.
12. Implement Travelling Sales Person problem using Branch and Bound approach.

Reference Books:

1. Fundamentals of Data Structures in C++, Horowitz Ellis, SahniSartaj, Mehta, Dinesh, 2ndEdition, Universities Press
2. Computer Algorithms/C++ Ellis Horowitz, SartajSahni, SanguthevarRajasekaran, 2ndEdition, University Press
3. Data Structures and program design in C, Robert Kruse, Pearson Education Asia
4. An introduction to Data Structures with applications, Trembley& Sorenson, McGraw Hill

Online Learning Resources:

1. <http://cse01-iiith.vlabs.ac.in/>
2. <http://peterindia.net/Algorithms.html>

II SEMESTER	L	T	P	C
	-	-	3	1.5

23CS3L02-DATABASE MANAGEMENT SYSTEMS LAB

Course Objectives:

This Course will enable students to

- Populate and query a database using SQL DDL/DML Commands
- Declare and enforce integrity constraints on a database
- Writing Queries using advanced concepts of SQL
- Programming PL/SQL including procedures, functions, cursors and triggers

COURSE OUTCOMES:

At the end of the course student able to

1. Understand the different issues involved in the design and implementation of a database system
2. Create and maintain tables using SQL
3. Populate and query a database using SQL DML/DDl commands.
4. Understand the use of structured query language and its syntax, transactions, database recovery and techniques for query optimization.
5. Acquire a good understanding of database systems concepts and to be in a position to use and design databases for different applications.

List of Programs:

1. Create a table STUDENT with appropriate data types and perform the following queries. Reg no, student name, date of birth, branch and year of study.
 - a. Insert 5 to 10 rows in a table
 - b. List all the students of all branches
 - c. List student names whose name starts with 's'
 - d. List student names whose name contains 's' as third literal
 - e. List student names whose contains two 's' anywhere in the name
 - f. List students whose branch is NULL
 - g. List students of CSE who born after 1985
 - h. List all students in reverse order of their names
 - i. Delete students of any branch whose name starts with 's'
 - j. Display student name padded with '*' after the name of all the students

2. Create the following tables with appropriate data types and constraints.

EMPLOYEE (Fname, Mname, Lname, SSN, Bdate, Address, Gender, Salary, SuperSSN, Dno)

DEPARTMENT (Dnumber, Dname, MgrSSN, Mgrstartdate)

DEPENDENT (ESSN, Dependent_Name, Gender, Bdate, Relationship)

 - a. Insert 5 to 10 rows into all the tables.
 - b. Display all employee's names along with their department names.
 - c. Display all employee's names along with their dependent details.
 - d. Display name and address of all employees who work for 'CSE' department.
 - e. List the names of all employees with two or more dependents.

- f. List the names of employee who have no dependents.
 - g. List the names of employees who have at least one dependent.
 - h. List the names of the employees along with names of their supervisors using aliases.
 - i. Display name of the department and name of manager for all the departments.
 - j. Display the name of each employee who has a dependent with the same first name and gender as the employee.
 - k. List the names of managers who have at least one dependent.
 - l. Display the sum of all employees' salaries as well as maximum, minimum and average salary in the entire departments department wise if the department has more than two employees.
 - m. List the departments of each female employee along with her name.
 - n. List all employee names and also the name of the department they manage if they happen to manage a dept
3. Create the following tables for university database
- Department (deptname, building, budget, primary key (dept name))
- Course (courseid, title, deptname, credits, primary key (course id))
- Instructor (ID, name, deptname, salary, primary key (ID), foreign key (deptname) references department);
- Section (courseid, sec id, semester, year, building, room number, time slot id, primary key (courseid, sec id, semester, year), foreign key (course id) references course);
- Teaches (ID, course id, sec id, semester, year numeric, primary key (ID, course id, sec id, semester, year), foreign key (course id, sec id, semester, year) references section, foreign key (ID) references instructor);
- a. List the names of instructors along with the titles of courses that they teach
 - b. Find the names of all instructors whose salary is greater than at least one instructor in the CSE department
 - b. To find the set of all courses taught in the Even as well as in odd semester
 - c. Find the average salary of instructors in the Computer Science department
 - d. Find the average salary in each department
 - e. To find all the courses taught in the Even semester in current year but not in previous year in the same semester
 - f. Delete all instructors with a salary between 15,000 and 35,000.
4. Create the following tables based on the above Schema Diagram with appropriate data types and constraints and perform the following queries.

SAILORS (Sailid, Salname, Rating, Age)

RESERVES (Sailid, boatid, Day)

BOATS (Boatid, Boat-name, Color)

- a. Insert 5 to 10 rows in all tables?
- b. Find the name of sailors who reserved boat number 3.
- c. Find the name of sailors who reserved green boat.
- d. Find the colors of boats reserved by Particular Sailor
- e. Find the names of sailors who have reserved at least one boat.
- f. Find the all sailid of sailors who have a rating of 10 or have reserved boated 104.

- g. Find the Sailid's of sailors with age over 20 who have not registered a red boat.
 - h. Find the names of sailors who have reserved a red or green boat.
 - i. Find sailors whose rating is better than some sailor called '_Dustin '.
 - j. Find the names of sailors who are older than the oldest sailor with a rating of 10.
5. Queries (along with sub Queries and correlated sub Queries) using ANY, ALL, IN, EXISTS, NOTEXISTS, UNION, INTERSET, Constraints. Example:- Select the roll number and name of the student who secured fourth rank in the class.
 6. Queries using Aggregate functions (COUNT, SUM, AVG, MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views.
 7. Queries using Conversion functions (to_char, to_number and to_date), string functions (Concatenation, lpad, rpad, ltrim, rtrim, lower, upper, initcap, length, substr and instr), date functions (Sysdate, next_day, add_months, last_day, months_between, least, greatest, trunc, round, to_char, to_date)
 8. Queries using joins (inner, left, right, full outer join)
 9.
 - i. Create a simple PL/SQL program which includes declaration section, executable section and exception –Handling section (Ex. Student marks can be selected from the table and printed for those who secured first class and an exception can be raised if no records were found)
 - ii. Insert data into student table and use COMMIT, ROLLBACK and SAVEPOINT in PL/SQL block. Develop a program that includes the features NESTED IF, CASE and CASE expression. The program can be extended using the NULLIF and COALESCE functions.
 10. Program development using WHILE LOOPS, numeric FOR LOOPS, nested loops using ERROR Handling, BUILT –IN Exceptions, USE defined Exceptions, RAISE-APPLICATION ERROR.
 11. Programs development using creation of procedures, passing parameters IN and OUT of PROCEDURES.
 12. Program development using creation of stored functions, invoke functions in SQL Statements and write complex functions.
 13. Develop programs using features parameters in a CURSOR, FOR UPDATE CURSOR, WHERE CURRENT of clause and CURSOR variables.
 14. Develop Programs using BEFORE and AFTER Triggers, Row and Statement Triggers and INSTEAD OF Triggers.
 15. Create a table and perform the search operation on table using indexing and non-indexing techniques.

Text Books/Suggested Reading:

1. Oracle: The Complete Reference by Oracle Press
2. Nilesh Shah, "Database Systems Using Oracle", PHI, 2007
3. Rick F Vander Lns, "Introduction to SQL", Fourth Edition, Pearson Education, 2007.

III SEMESTER	L	T	P	C
---------------------	----------	----------	----------	----------

	-	1	2	2
23CS3S01-PYTHON PROGRAMMING (SKILL ENHANCEMENT COURSE)				

Course Objectives:

The main objectives of the course are to

- Introduce core programming concepts of Python programming language.
- Demonstrate about Python data structures like Lists, Tuples, Sets and dictionaries
- Implement Functions, Modules and Regular Expressions in Python Programming and to create practical and contemporary applications using these

UNIT-I:

History of Python Programming Language, Thrust Areas of Python, Installing Anaconda Python Distribution, Installing and Using Jupyter Notebook.

Parts of Python Programming Language: Identifiers, Keywords, Statements and Expressions, Variables, Operators, Precedence and Associativity, Data Types, Indentation, Comments, Reading Input, Print Output, Type Conversions, the type () Function and Is Operator, Dynamic and Strongly Typed Language.

Control Flow Statements: if statement, if-else statement, if...elif...else, Nested if statement, while Loop, for Loop, continue and break Statements, Catching Exceptions Using try and except Statement.

Sample Experiments:

1. Write a program to find the largest element among three Numbers.
2. Write a Program to display all prime numbers within an interval
3. Write a program to swap two numbers without using a temporary variable.
4. Demonstrate the following Operators in Python with suitable examples.
 - i) Arithmetic Operators ii) Relational Operators iii) Assignment Operators iv) Logical Operators
 - v) Bit wise Operators vi) Ternary Operator vii) Membership Operators viii) Identity Operators
5. Write a program to add and multiply complex numbers
6. Write a program to print multiplication table of a given number.

UNIT-II:

Functions: Built-In Functions, Commonly Used Modules, Function Definition and Calling the function, return Statement and void Function, Scope and Lifetime of Variables, Default Parameters, Keyword Arguments, *args and **kwargs, Command Line Arguments.

Strings: Creating and Storing Strings, Basic String Operations, Accessing Characters in String by Index Number, String Slicing and Joining, String Methods, Formatting Strings.

Lists: Creating Lists, Basic List Operations, Indexing and Slicing in Lists, Built-In Functions Used on Lists, List Methods, del Statement.

Sample Experiments:

1. Write a program to define a function with multiple return values.
2. Write a program to define a function using default arguments.
3. Write a program to find the length of the string without using any library functions.
4. Write a program to check if the substring is present in a given string or not.
5. Write a program to perform the given operations on a list:
 - i. addition ii. Insertion iii. slicing
6. Write a program to perform any 5 built-in functions by taking any list.

UNIT-III:

Dictionaries: Creating Dictionary, Accessing and Modifying key:value Pairs in Dictionaries, Built-In Functions Used on Dictionaries, Dictionary Methods, del Statement.

Tuples and Sets: Creating Tuples, Basic Tuple Operations, tuple() Function, Indexing and Slicing in Tuples, Built-In Functions Used on Tuples, Relation between Tuples and Lists, Relation between Tuples and Dictionaries, Using zip() Function, Sets, Set Methods, Frozenset.

Sample Experiments:

1. Write a program to create tuples (name, age, address, college) for at least two members and concatenate the tuples and print the concatenated tuples.
2. Write a program to count the number of vowels in a string (No control flow allowed).
3. Write a program to check if a given key exists in a dictionary or not.
4. Write a program to add a new key-value pair to an existing dictionary.
5. Write a program to sum all the items in a given dictionary.

UNIT-IV:

Files: Types of Files, Creating and Reading Text Data, File Methods to Read and Write Data, Reading and Writing Binary Files, Pickle Module, Reading and Writing CSV Files, Python os and os.path Modules.

Object-Oriented Programming: Classes and Objects, Creating Classes in Python, Creating Objects in Python, Constructor Method, Classes with Multiple Objects, Class Attributes Vs Data Attributes, Encapsulation, Inheritance, Polymorphism.

Sample Experiments:

1. Write a program to sort words in a file and put them in another file. The output file should have only lower-case words, so any upper-case words from source must be lowered.
2. Python program to print each line of a file in reverse order.
3. Python program to compute the number of characters, words and lines in a file.
4. Write a program to create, display, append, insert and reverse the order of the items in the array.
5. Write a program to add, transpose and multiply two matrices.
6. Write a Python program to create a class that represents a shape. Include methods to calculate its area and perimeter. Implement subclasses for different shapes like circle, triangle, and square.

UNIT-V:

Introduction to Data Science: Functional Programming, JSON and XML in Python, NumPy with Python, Pandas.

Sample Experiments:

1. Python program to check whether a JSON string contains complex object or not.
2. Python Program to demonstrate NumPy arrays creation using array () function.
3. Python program to demonstrate use of ndim, shape, size, dtype.
4. Python program to demonstrate basic slicing, integer and Boolean indexing.
5. Python program to find min, max, sum, cumulative sum of array

6. Create a dictionary with at least five keys and each key represent value as a list where this list contains at least ten values and convert this dictionary as a pandas data frame and explore the data through the data frame as follows:
 - a. Apply head () function to the pandas data frame
 - b. Perform various data selection operations on Data Frame
7. Select any two columns from the above data frame, and observe the change in one attribute with respect to other attribute with scatter and plot operations in matplotlib
8. Case studies using python
9. Home assignments
10. Mini project

Reference Books:

1. Gowrishankar S, Veena A., Introduction to Python Programming, CRC Press.
2. Python Programming, S Sridhar, J Indumathi, V M Hariharan, 2nd Edition, Pearson, 2024
3. Introduction to Programming Using Python, Y. Daniel Liang, Pearson.

Online Learning Resources/Virtual Labs:

1. <https://www.coursera.org/learn/python-for-applied-data-science-ai>
2. <https://www.coursera.org/learn/python?specialization=python#syllabus>

III SEMESTER	L	T	P	C
	2	-	-	-

23CS3M01-ENVIRONMENTAL SCIENCE

Course Objectives:

- To make the students to get awareness on environment
- To understand the importance of protecting natural resources, ecosystems for future generations and pollution causes due to the day-to-day activities of human life
- To save earth from the inventions by the engineers.

Course Outcomes:

- Grasp multidisciplinary nature of environmental studies and various renewable and non-renewable resources.
- Understand flow and bio-geo-chemical cycles and ecological pyramids.
- Understand various causes of pollution and solid waste management and related preventive measures.
- About the rainwater harvesting, watershed management, ozone layer depletion and waste land reclamation.
- Casus of population explosion, value education and welfare programmes.

UNIT – I

Multidisciplinary Nature of Environmental Studies: – Definition, Scope and Importance – Need for Public Awareness.

Natural Resources : Renewable and non-renewable resources – Natural resources and associated problems – Forest resources – Use and over – exploitation, deforestation, case studies – Timber extraction – Mining, dams and other effects on forest and tribal people – Water resources – Use and over utilization of surface and ground water – Floods, drought, conflicts over water, dams – benefits and problems – Mineral resources: Use and exploitation, environmental effects of extracting and using mineral resources, case studies – Food resources: World food problems, changes caused by agriculture and overgrazing, effects of modern agriculture, fertilizer-pesticide problems, water logging, salinity, case studies. – Energy resources:

UNIT – II

Ecosystems: Concept of an ecosystem. – Structure and function of an ecosystem – Producers, consumers and decomposers – Energy flow in the ecosystem – Ecological succession – Food chains, food webs and ecological pyramids – Introduction, types, characteristic features, structure and function of the following ecosystem:

- Forest ecosystem.
- Grassland ecosystem
- Desert ecosystem
- Aquatic ecosystems (ponds, streams, lakes, rivers, oceans, estuaries)

Biodiversity and Its Conservation : Introduction and Definition: genetic, species and ecosystem diversity – Bio-geographical classification of India – Value of biodiversity: consumptive use, Productive use, social, ethical, aesthetic and option values – Biodiversity at global, National and local levels – India as a mega-diversity nation – Hot-spots of biodiversity – Threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts – Endangered and endemic species of India – Conservation of biodiversity: In-situ and Ex-situ conservation of biodiversity.

UNIT – III

Environmental Pollution: Definition, Cause, effects and control measures of:

- a. Air Pollution.
- b. Water pollution
- c. Soil pollution
- d. Marine pollution
- e. Noise pollution
- f. Thermal pollution
- g. 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

Social Issues and the Environment: From Unsustainable to Sustainable development – Urban problems related to energy – Water conservation, rain water harvesting, watershed management – Resettlement and rehabilitation of people; its problems and concerns. Case studies – Environmental ethics: Issues and possible solutions – Climate change, global warming, acid rain, ozone layer depletion, nuclear accidents and holocaust. Case Studies – Wasteland reclamation. – Consumerism and waste products. – Environment Protection Act. – Air (Prevention and Control of Pollution) Act. – Water (Prevention and control of Pollution) Act – Wildlife Protection Act – Forest Conservation Act – Issues involved in enforcement of environmental legislation – Public awareness.

UNIT – V

Human Population And The Environment: Population growth, variation among nations. Population explosion – Family Welfare Programmes. – Environment and human health – Human Rights – Value Education – HIV/AIDS – Women and Child Welfare – Role of information Technology in Environment and human health – Case studies.

Field Work: Visit to a local area to document environmental assets River/forest grassland/hill/mountain – Visit to a local polluted site-Urban/Rural/Industrial/Agricultural Study of common plants, insects, and birds – river, hill slopes, etc.

Textbooks:

1. Erach Bharucha, Textbook of Environmental Studies for Undergraduate Courses, Universities Press (India) Private Limited, 2019.
2. Palaniswamy, Environmental Studies, 2/e, Pearson education, 2014.
3. S.Azeem Unnisa, Environmental Studies, Academic Publishing Company, 2021.
4. K.Raghavan Nambiar, “Text book of Environmental Studies for Undergraduate Courses as per UGC model syllabus”, SciTech Publications (India), Pvt. Ltd, 2010.

Reference Books:

1. Deeksha Dave and E.Sai Baba Reddy, Textbook of Environmental Science, 2/e, Cengage Publications, 2012.
2. M.Anji Reddy, “Textbook of Environmental Sciences and Technology”, BS Publication, 2014.
3. J.P. Sharma, Comprehensive Environmental studies, Laxmi publications, 2006.
4. J. Glynn Henry and Gary W. Heinke, Environmental Sciences and Engineering, Prentice Hall of India Private limited, 1988.

5. G.R. Chatwal, A Text Book of Environmental Studies, Himalaya Publishing House, 2018.
6. Gilbert M. Masters and Wendell P. Ela, Introduction to Environmental Engineering and Science, 1/e, Prentice Hall of India Private limited, 1991.

Online Learning Resources:

- https://onlinecourses.nptel.ac.in/noc23_hs155/preview
- https://www.edx.org/learn/environmental-science/rice-university-ap-r-environmental-science-part-3-pollution-and-resources?index=product&objectID=course-3a6da9f2-d84c-4773-8388-1b2f8f6a75f2&webview=false&campaign=AP%C2%AE+Environmental+Science++Part+3%3A+Pollution+and+Resources&source=edX&product_category=course&placement_url=https%3A%2F%2Fwww.edx.org%2Flearn%2Fenvironmental-science
- <http://ecoursesonline.iasri.res.in/Courses/Environmental%20Science-I/Data%20Files/pdf/lec07.pdf>
- <https://www.youtube.com/watch?v=5QxxaVfgQ3k>

IV SEMESTER	L	T	P	C
	2	1	-	3

23HS4T01-UNIVERSAL HUMAN VALUES – UNDERSTANDING HARMONY AND ETHICAL HUMAN CONDUCT

Course Objectives:

- To help the students appreciate the essential complementary between 'VALUES' and 'SKILLS' to ensure sustained happiness and prosperity which are the core aspirations of all human beings.
- To facilitate the development of a Holistic perspective among students towards life and profession as well as towards happiness and prosperity based on a correct understanding of the Human reality and the rest of existence. Such holistic perspective forms the basis of Universal Human Values and movement towards value-based living in a natural way.
- To highlight plausible implications of such a Holistic understanding in terms of ethical human conduct, trustful and mutually fulfilling human behaviour and mutually enriching interaction with Nature.

Course Outcomes:

- Define the terms like Natural Acceptance, Happiness and Prosperity (L1, L2)
- Identify one's self, and one's surroundings (family, society nature) (L1, L2)
- Apply what they have learnt to their own self in different day-to-day settings in real life (L3)
- Relate human values with human relationship and human society. (L4)
- Justify the need for universal human values and harmonious existence (L5)
- Develop as socially and ecologically responsible engineers (L3, L6)

Course Topics

The course has 28 lectures and 14 tutorials in 5 modules. The lectures and tutorials are of 1- hour duration. Tutorial sessions are to be used to explore and practice what has been proposed during the lecture sessions.

The Teacher's Manual provides the outline for lectures as well as practice sessions. The teacher is expected to present the issues to be discussed as propositions and encourage the students to have a dialogue.

UNIT I

Introduction to Value Education (6lecturesand3tutorialsforpractice session)

Lecture1: Right Understanding, Relationship and Physical Facility(Holistic Development and the Role of Education)

Lecture2: Understanding Value Education

Tutorial 1: Practice Session PS1 Sharing about Oneself

Lecture 3: self-exploration as the Process for Value Education

Lecture4: Continuous Happiness and Prosperity–the Basic Human Aspirations

Tutorial 2: Practice Session PS2 Exploring Human Consciousness

Lecture 5: Happiness and Prosperity – Current Scenario

Lecture6: Method to Fulfill the Basic Human Aspirations

Tutorial3:PracticeSessionPS3ExploringNatural Acceptance

UNIT II

Harmony in the Human Being (6 lectures and 3 tutorials for practice session)

Lecture 7: Understanding Human being as the Co-existence of the self and the body.

Lecture8: Distinguishing between the Needs of the self and the body

Tutorial 4: Practice Session PS4 Exploring the difference of Needs of self and

body.

Lecture 9: The body as an Instrument of the self.

Lecture 10: Understanding Harmony in the self.

Tutorial 5: Practice Session PS5 Exploring Sources of Imagination in the self.

Lecture 11: Harmony of the self with the body

Lecture 12: Programme to ensure self-regulation and Health

Tutorial 6: Practice Session PS6 Exploring Harmony of self with the body

UNIT III

Harmony in the Family and Society (6 lectures and 3 tutorials for practice session)

Lecture 13: Harmony in the Family – the Basic Unit of Human Interaction

Lecture 14: 'Trust' – the Foundational Value in Relationship

Tutorial 7: Practice Session PS7 Exploring the Feeling of Trust

Lecture 15: 'Respect' – as the Right Evaluation

Tutorial 8: Practice Session PS8 Exploring the Feeling of Respect

Lecture 16: Other Feelings, Justice in Human-to-Human Relationship

Lecture 17: Understanding Harmony in the Society

Lecture 18: Vision for the Universal Human Order

Tutorial 9: Practice Session PS9 Exploring Systems to fulfil Human Goal

UNIT IV

Harmony in the Nature/ Existence (4 lectures and 2 tutorials for practice session)

Lecture 19: Understanding Harmony in the Nature

Lecture 20: Interconnectedness, self-regulation and Mutual Fulfilment among the Four Orders of Nature

Tutorial 10: Practice Session PS10 Exploring the Four Orders of Nature

Lecture 21: Realizing Existence as Co-existence at All Levels

Lecture 22: The Holistic Perception of Harmony in existence

Tutorial 11: Practice Session PS11 Exploring Co-existence in Existence.

UNIT V

Implications of the Holistic Understanding – a Look at Professional Ethics (6 lectures and 3 tutorials for practice session)

Lecture 23: Natural Acceptance of Human Values

Lecture 24: Definitiveness of (Ethical) Human Conduct

Tutorial 12: Practice Session PS12 Exploring Ethical Human Conduct

Lecture 25: A Basis for Humanistic Education, Humanistic Constitution and Universal Human Order

Lecture 26: Competence in Professional Ethics

Tutorial 13: Practice Session PS13 Exploring Humanistic Models in Education

Lecture 27: Holistic Technologies, Production Systems and Management Models- Typical Case Studies

Lecture 28: Strategies for Transition towards Value-based Life and Profession

Tutorial 14: Practice Session PS14 Exploring Steps of Transition towards Universal Human Order

Practice Sessions for UNIT I – Introduction to Value Education

PS1 Sharing about Oneself

PS2 Exploring Human Consciousness

PS3 Exploring Natural Acceptance

Practice Sessions for UNIT II – Harmony in the Human Being

PS4 Exploring the difference of Needs of self and body

PS5 Exploring Sources of Imagination in the self

PS6 Exploring Harmony of self with the body

Practice Sessions for UNIT III – Harmony in the Family and Society

PS7 Exploring the Feeling of Trust

PS8 Exploring the Feeling of Respect

PS9 Exploring Systems to fulfil Human Goal

Practice Sessions for UNIT IV – Harmony in the Nature (Existence)

PS10 Exploring the Four Orders of Nature

PS11 Exploring Co-existence in Existence

Practice Sessions for UNIT V – Implications of the Holistic Understanding – a Look at Professional Ethics

PS12 Exploring Ethical Human Conduct

PS13 Exploring Humanistic Models in Education

PS14 Exploring Steps of Transition towards Universal Human Order

READINGS:

Textbook and Teachers Manual

a. The Textbook

R R Gaur, R Asthana, G P Bagaria, A Foundation Course in Human Values and Professional Ethics, 2nd Revised Edition, Excel Books, New Delhi, 2019. ISBN 978-93-87034-47-1

b. The Teacher's Manual R R Gaur, R Asthana, G P Bagaria, Teachers' Manual for A Foundation Course in Human Values and Professional Ethics, 2nd Revised Edition, Excel Books, New Delhi, 2019. ISBN 978-93-87034-53-2

Reference Books

1. Jeevan Vidya: Ek Parichaya, A Nagaraj, Jeevan Vidya Prakashan, Amarkantak, 1999.
2. Human Values, A.N. Tripathi, New Age Intl. Publishers, New Delhi, 2004.
3. The Story of Stuff (Book).
4. The Story of My Experiments with Truth - by Mohandas Karamchand Gandhi
5. Small is Beautiful - E. F Schumacher.
6. Slow is Beautiful - Cecile Andrews
7. Economy of Permanence - J C Kumarappa
8. Bharat Mein Angreji Raj – Pandit Sunderlal
9. Rediscovering India - by Dharampal
10. Hind Swaraj or Indian Home Rule - by Mohandas K. Gandhi
11. India Wins Freedom -Maulana AbdulKalam Azad
12. Vivekananda -Romain Rolland (English)
13. Gandhi -Romain Rolland (English)

Mode of Conduct:

Lecture hours are to be used for interactive discussion, placing the proposals about the topics at hand and motivating students to reflect, explore and verify them.

Tutorial hours are to be used for practice sessions.

While analyzing and discussing the topic, the faculty mentor's role is in pointing to essential elements to help in sorting them out from the surface elements. In other words, help the students explore the important or critical elements.

In the discussions, particularly during practice sessions (tutorials), the mentor encourages the student to connect with one's own self and do self-observation, self-reflection and self-exploration.

Scenarios may be used to initiate discussion. The student is encouraged to take up "ordinary" situations rather than "extra-ordinary" situations. Such observations and their analyses are shared and discussed with other students and faculty mentor, in a group sitting.

Tutorials (experiments or practical) are important for the course. The difference is that the laboratory is everyday life, and practical are how you behave and work in real life. Depending on the nature of topics, worksheets, home assignment and/or activity are included. The practice sessions (tutorials) would also provide support to a student in performing actions commensurate to his/her beliefs. It is intended that this would lead to development of commitment, namely behaving and working based on basic human values.

It is recommended that this content be placed before the student as it is, in the form of a basic foundation course, without including anything else or excluding any part of this content. Additional content may be offered in separate, higher courses. This course is to be taught by faculty from every teaching department, not exclusively by any one department.

Teacher preparation with a minimum exposure to at least one 8-day Faculty Development Program on Universal Human Values is deemed essential.

Online Resources:

1. <https://fdp-si.aicte-india.org/UHV-II%20Class%20Notes%20&%20Handouts/UHV%20Handout%201-Introduction%20to%20Value%20Education.pdf>
2. <https://fdp-si.aicte-india.org/UHV-II%20Class%20Notes%20&%20Handouts/UHV%20Handout%202-Harmony%20in%20the%20Human%20Being.pdf>
3. <https://fdp-si.aicte-india.org/UHV-II%20Class%20Notes%20&%20Handouts/UHV%20Handout%203-Harmony%20in%20the%20Family.pdf>
4. <https://fdp-si.aicte-india.org/UHV%201%20Teaching%20Material/D3-S2%20Respect%20July%202023.pdf>
5. <https://fdp-si.aicte-india.org/UHV-II%20Class%20Notes%20&%20Handouts/UHV%20Handout%205-Harmony%20in%20the%20Nature%20and%20Existence.pdf>
6. <https://fdp-si.aicte-india.org/download/FDPTeachingMaterial/3-days%20FDP-SI%20UHV%20Teaching%20Material/Day%203%20Handouts/UHV%203D%20D3-S2A%20Und%20Nature-Existence.pdf>
7. <https://fdp-si.aicte-india.org/UHV%20II%20Teaching%20Material/UHV%20II%20Lecture%2023-25%20Ethics%20v1.pdf>
8. <https://www.studocu.com/in/document/kiet-group-of-institutions/universal-human-values/chapter-5-holistic-understanding-of-harmony-on-professional-ethics/62490385>
9. https://onlinecourses.swayam2.ac.in/aic22_ge23/preview

IV SEMESTER	L	T	P	C
	3	-	-	3

23BS4T04-PROBABILITY AND STATISTICS

Course Objectives:

- To familiarize the students with the foundations of probability and statistical methods
- To impart probability concepts and statistical methods in various applications Engineering

Course Outcomes: Upon successful completion of this course, the student should be able to

1. Classify the concepts of data science and its importance (L2)
2. Interpret the association of characteristics and through correlation and regression tools (L4)
3. Apply discrete and continuous probability distributions (L3)
4. Design the components of a classical hypothesis test (L6)
5. Infer the statistical inferential methods based on small and large sampling tests (L4)

Unit – I: Descriptive statistics and methods for data science:

Data science – Statistics Introduction – Population vs Sample – Collection of data – primary and secondary data – Type of variable: dependent and independent Categorical and Continuous variables – Data visualization – Measures of Central tendency – Measures of Variability – Skewness – Kurtosis.

UNIT – II: Correlation and Regression:

Correlation – Correlation coefficient – Rank correlation.

Linear Regression: Straight line – Multiple Linear Regression - Regression coefficients and properties – Curvilinear Regression: Parabola – Exponential – Power curves.

UNIT – III: Probability and Distributions:

Probability– Conditional probability and Baye’s theorem – Random variables – Discrete and Continuous random variables – Distribution functions – Probability mass function, Probability density function and Cumulative distribution functions – Mathematical Expectation and Variance – Binomial, Poisson, Uniform and Normal distributions.

UNIT – IV: Sampling Theory:

Introduction – Population and Samples – Sampling distribution of Means and Variance (definition only) – Point and Interval estimations – Maximum error of estimate – Centrallimit theorem (without proof) – Estimation using t_2 and χ^2 F-distributions.

UNIT – V: Tests of Hypothesis:

Introduction – Hypothesis – Null and Alternative Hypothesis – Type I and Type II errors – Level of significance – One tail and two-tail tests – Test of significance for large samples and Small Samples: Single and difference means – Single and two proportions – Student’s t- test, F-test, χ^2 -test.

Text Books:

- **Miller and Freund’s**, Probability and Statistics for Engineers, 7/e, Pearson, 2008.
- **S. C. Gupta and V.K. Kapoor**, Fundamentals of Mathematical Statistics, 11/e, Sultan Chand & Sons Publications, 2012.

Reference Books:

- **Shron L. Myers, Keying Ye, Ronald E Walpole**, Probability and Statistics Engineers and the Scientists, 8th Edition, Pearson 2007.
- **Jay I. Devore**, Probability and Statistics for Engineering and the Sciences, 8th Edition, Cengage.
- **Sheldon M. Ross**, Introduction to probability and statistics Engineers and the Scientists, 4th Edition, Academic Foundation, 2011.
- **Johannes Ledolter and Robert V. Hogg**, Applied statistics for Engineers and Physical Scientists, 3rd Edition, Pearson, 2010.

IV SEMESTER	L	T	P	C
	3	-	-	3
23CS4T01-OPERATING SYSTEMS				

Course Objectives:

The main objectives of the course is to make student

- Understand the basic concepts and principles of operating systems, including process management, memory management, file systems, and Protection
- Make use of process scheduling algorithms and synchronization techniques to achieve better performance of a computer system.
- Illustrate different conditions for deadlock and their possible solutions.

UNIT - I

Operating Systems Overview: Introduction, Operating system functions, Operating systems operations, Computing environments, Free and Open-Source Operating Systems

System Structures: Operating System Services, User and Operating-System Interface, system calls, Types of System Calls, system programs, Operating system Design and Implementation, Operating system structure, Building and Booting an Operating System, Operating system debugging

UNIT - II

Process Management System: Process Concept, Process scheduling, Operations on processes, Inter-process communication.

Threads and Concurrency: Multithreading models, Thread libraries, Threading issues.

CPU Scheduling: Basic concepts, Scheduling criteria, Scheduling algorithms, Multiple processor scheduling.

UNIT – III

Synchronization Tools: The Critical Section Problem, Peterson’s Solution, Mutex Locks, Semaphores, Monitors, Classic problems of Synchronization.

Deadlocks: system Model, Deadlock characterization, Methods for handling Deadlocks, Deadlock prevention, Deadlock avoidance, Deadlock detection, Recovery from Deadlock.

UNIT - IV

Memory-Management Strategies: Introduction, Contiguous memory allocation, Paging, Structure of the Page Table, Swapping.

Virtual Memory Management: Introduction, Demand paging, Copy-on-write, Page replacement algorithm: First In First Out (FIFO), Least Recently Used (LRU) and Optimal algorithms, Allocation of frames, Thrashing.

Storage Management: Overview of Mass Storage Structure, HDD Scheduling.

UNIT - V

File System: File System Interface: File concept, Access methods, Directory Structure; File system Implementation: File-system structure, File-system Operations, Directory implementation, Allocation method, Free space management; File-System Internals: File- System Mounting, Partitions and Mounting, File Sharing.

Protection: Goals of protection, Principles of protection, Protection Rings, Domain of protection, Access matrix.

Textbooks:

1. Operating System Concepts, Silberschatz A, Galvin P B, Gagne G, 10th Edition, Wiley, 2018.

2. Modern Operating Systems, TanenbaumAS,4th Edition, Pearson,2016

Reference Books:

1. Operating Systems -Internals and Design Principles, Stallings W, 9th edition, Pearson, 2018
2. Operating Systems: A Concept Based Approach, D.M Dhamdhere, 3rd Edition, McGraw- Hill, 2013

Online Learning Resources:

1. <https://nptel.ac.in/courses/106/106/106106144/>
2. <http://peterindia.net/OperatingSystems.html>

IV SEMESTER	L	T	P	C
	3	-	-	3
23CS4T02-OBJECT ORIENTED PROGRAMMING THROUGH JAVA				

Course Objectives:

The learning objectives of this course are to:

- identify Java language components and how they work together in applications
- learn the fundamentals of object-oriented programming in Java, including defining classes, invoking methods, using class libraries.
- learn how to extend Java classes with inheritance and dynamic binding and how to use exception handling in Java applications
- understand how to design applications with threads in Java
- understand how to use Java APIs for program development

Course Outcomes:

CO1: Identify and utilize Java language components effectively in applications.

CO2: Implement object-oriented programming principles using Java.

CO3: Extend Java classes with inheritance and handle exceptions in applications

CO4: Design multi-threaded applications in Java.

CO5: Use Java APIs for program development and database connectivity.

UNIT I:**[15 Hrs.]**

Object Oriented Programming: Basic concepts: Object, Class, Abstraction, Encapsulation, Inheritance, Polymorphism, Dynamic binding and Message passing. Program Structure in Java: Introduction, Elements or Tokens in Java, Command Line Arguments, User Input to Programs.

Data Types, Variables, and Operators: Introduction, Data Types in Java, Declaration of Variables, Type Casting, Scope of Variable, Static Variables and Methods,

Introduction to Operators, Types of operators, Precedence and Associativity of Operators,

Control Statements: Introduction, if Expression, Nested if Expressions, if-else Expressions, Switch Statement, Iteration Statements, while Expression, do-while Loop, for Loop, Nested for Loop, For-Each Loop, Break Statement, Continue Statement.

UNIT II:**[16 Hrs.]**

Classes and Objects: Introduction, Class Declaration and Modifiers, Class Members, Declaration of Class Objects, Assigning One Object to Another, Access Control for Class Members, Constructor Methods for Class, Overloaded Constructor Methods, Nested Classes, Final Class and Methods, Passing Arguments by Value and by Reference, Keyword this.

Methods: Introduction, Defining Methods, Overloaded Methods, Overloaded Constructor Methods, Class Objects as Parameters in Methods, Access Control, Recursive Methods, Nesting of Methods, Overriding Methods, Attributes Final and Static.

UNIT III:**[15 Hrs.]**

Arrays: Introduction, Declaration and Initialization of Arrays, Storage of Array in Computer Memory, Accessing Elements of Arrays, Operations on Array Elements, Assigning Array to

Another Array, Dynamic Change of Array Size, Sorting of Arrays, Multidimensional arrays.

Inheritance: Introduction, Process of Inheritance, Types of Inheritances, Universal Super Class-Object Class, Inhibiting Inheritance of Class Using Final, Access Control and Inheritance, Multilevel Inheritance, Application of Keyword Super, Constructor Method and Inheritance, Method Overriding, Dynamic Method Dispatch, Abstract Classes, Interfaces and Inheritance.

Interfaces: Introduction, Declaration of Interface, Implementation of Interface, Multiple Interfaces, Nested Interfaces, Inheritance of Interfaces, Default Methods in Interfaces, Static Methods in Interface, Functional Interfaces, Annotations.

UNIT IV:

[13 Hrs.]

Packages and Java Library: Introduction, Defining Package, Importing Packages and Classes into Programs, Path and Class Path, Access Control, Packages in Java SE, Java.lang Package and its Classes, Class Object, Enumeration, class Math, Wrapper Classes, Auto-boxing and Auto-unboxing, Java util Classes and Interfaces, Formatter Class, Random Class, Time Package, Class Instant (java.time.Instant), Formatting for Date/Time in Java, Temporal Adjusters Class.

Exception Handling: Introduction, Hierarchy of Standard Exception Classes, Keywords throws and throw, try, catch, and finally Blocks, Multiple Catch Clauses, Class Throwable, Unchecked Exceptions, Checked Exceptions.

UNIT V:

[15 Hrs.]

String Handling in Java: Introduction, Interface Char Sequence, Class String, Methods for Extracting Characters from Strings, Comparison, Modifying, Searching; Class String Buffer.

Multithreaded Programming: Introduction, Need for Multiple Threads Multithreaded Programming for Multi-core Processor, Thread Class, Main Thread-Creation of New Threads, Thread States, Thread Priority-Synchronization, Deadlock and Race Situations, Inter-thread Communication - Suspending, Resuming, and Stopping of Threads.

Java Database Connectivity: Introduction, JDBC Architecture, Installing MySQL and MySQL Connector/J, JDBC Environment Setup, Establishing JDBC Database Connections, ResultSet Interface

Text Books:

1. JAVA one step ahead, Anitha Seth, B.L.Juneja, Oxford.
2. Joy with JAVA, Fundamentals of Object Oriented Programming, DebasisSamanta, MonalisaSarma, Cambridge, 2023.
3. JAVA 9 for Programmers, Paul Deitel, Harvey Deitel, 4th Edition, Pearson.

References Books:

1. The complete Reference Java, 11th edition, Herbert Schildt, TMH
2. Introduction to Java programming, 7th Edition, Y Daniel Liang, Pearson

Online Resources:

1. <https://nptel.ac.in/courses/106/105/106105191/>
2. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_012880464547618816347_shared/overview

IV SEMESTER	L	T	P	C
	3	-	-	3
23IT4T02-SOFTWARE ENGINEERING				

Course Objectives:

The objectives of this course are to introduce

- Software life cycle models, Software requirements and SRS document.
- Project Planning, quality control and ensuring good quality software.
- Software Testing strategies, use of CASE tools, Implementation issues, validation & verification procedures.

Course Outcomes:

At the end of the course students will be able to

1. Illustrate various life cycle activities like Analysis, Design, Implementation, Testing and Maintenance (K2)
2. Analyze various software engineering models and apply methods for design and development of software projects. (K4)
3. Develop system designs using appropriate techniques. (K3)
4. Outline various testing techniques for a software project. (K2)
5. Apply standards, CASE tools and techniques for engineering software projects. (K3)

UNIT I:

Introduction: Evolution, Software development projects, Exploratory style of software developments, Emergence of software engineering, Notable changes in software development practices, Computer system engineering.

Software Life Cycle Models: Basic concepts, Waterfall model and its extensions, Rapid application development, Agile development model, Spiral model.

UNIT II:

Software Project Management: Software project management complexities, Responsibilities of a software project manager, Metrics for project size estimation, Project estimation techniques, Empirical Estimation techniques, COCOMO, Halstead's software science, risk management.

Requirements Analysis And Specification: Requirements gathering and analysis, Software Requirements Specification (SRS), Formal system specification, Axiomatic specification, Algebraic specification, Executable specification and 4GL.

UNIT III:

Software Design: Overview of the design process, How to characterize a good software design? Layered arrangement of modules, Cohesion and Coupling, approaches to software design.

Agility: Agility and the Cost of Change, Agile Process, Extreme Programming (XP), Other Agile Process Models, Tool Set for the Agile Process (Text Book 2)

Function-Oriented Software Design: Overview of SA/SD methodology, Structured analysis, Developing the DFD model of a system, Structured design, Detailed design, and Design Review.

User Interface Design: Characteristics of a good user interface, Basic concepts, Types of user interfaces, Fundamentals of component-based GUI development, and user interface design methodology.

UNIT IV:

Coding And Testing: Coding, Code review, Software documentation, Testing, Black-box testing, White-Box testing, Debugging, Program analysis tools, Integration testing, Testing object-oriented programs, Smoke testing, and Some general issues associated with testing.

Software Reliability And Quality Management: Software reliability. Statistical testing, Software

quality, Software quality management system, ISO 9000. SEI Capability maturity model. Few other important quality standards and Six Sigma.

UNIT V:

Computer-Aided Software Engineering (Case): CASE and its scope, CASE environment, CASE support in the software life cycle, other characteristics of CASE tools, Towards second generation CASE Tool, and Architecture of a CASE Environment.

Software Maintenance: Characteristics of software maintenance, Software reverse engineering, Software maintenance process models and Estimation of maintenance cost.

Software Reuse: reuse- definition, introduction, reason behind no reuse so far, Basic issues in any reuse program, A reuse approach, and Reuse at organization level.

Text Books:

1. Fundamentals of Software Engineering, Rajib Mall, 5th Edition, PHI.
2. Software Engineering A Practitioner's Approach, Roger S. Pressman, 9th Edition, Mc- Graw Hill International Edition.

Reference Books:

1. Software Engineering, Ian Sommerville, 10th Edition, Pearson.
2. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press.

e-Resources:

- 1) <https://nptel.ac.in/courses/106/105/106105182/>
- 2) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01260589506387148827_shared/overview
- 3) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013382690411003904735_shared/overview

IV SEMESTER	L	T	P	C
	-	-	3	1.5
23CS4L01-OPERATING SYSTEMS & SOFTWARE ENGINEERING LAB				

Course Objectives:

The main objectives of the course are to

- Provide insights into system calls, file systems, semaphores,
- Develop and debug CPU Scheduling algorithms, page replacement algorithms, thread implementation
- Implement Bankers Algorithms to Avoid the Dead Lock
- acquire the generic software development skill through various stages of software life cycle
- generate test cases for software testing

Experiments covering the Topics:

- UNIX fundamentals, commands & system calls
- CPU Scheduling algorithms, thread processing
- IPC, semaphores, monitors, deadlocks
- Page replacement algorithms, file allocation strategies
- Memory allocation strategies
- Software Requirement Specification, DFD, CFD
- Software estimation, UML diagrams, test case design

Sample Experiments in Operating Systems:

1. Practicing of Basic UNIX Commands.
2. Write programs using the following UNIX operating system calls fork, exec, getpid, exit, wait, close, stat, opendir and readdir
3. Simulate UNIX commands like cp, ls, grep, etc.,
4. Simulate the following CPU scheduling algorithms
 - a) FCFS b) SJF c) Priority d) Round Robin
5. Control the number of ports opened by the operating system with
 - a) Semaphore b) Monitors.
6. Write a program to illustrate concurrent execution of threads using pthreads library.
7. Write a program to solve producer-consumer problem using Semaphores.
8. Implement the following memory allocation methods for fixed partition
 - a) First fit b) Worst fit c) Best fit
9. Simulate the following page replacement algorithms
 - a) FIFO b) LRU c) LFU
10. Simulate Paging Technique of memory management.
11. Implement Bankers Algorithm for Dead Lock avoidance and prevention
12. Simulate the following file allocation strategies
 - a) Sequential b) Indexed c) Linked

13. Download and install nachos operating system and experiment with it

Sample Experiments in Software Engineering:

- 1) Perform the following, for the following experiments:
 - i. Do the Requirement Analysis and Prepare SRS
 - ii. Draw E-R diagrams, DFD, CFD and structured charts for the project.
 - a. Course Registration System
 - b. Students Marks Analyzing System
 - c. Online Ticket Reservation System
 - d. Stock Maintenance
- 2) Consider any application, using COCOMO model, estimate the effort.
- 3) Consider any application, Calculate effort using FP oriented estimation model.
- 4) Draw the UML Diagrams for the problem a, b, c, d.
- 5) Design the test cases for e-Commerce application (Flipcart, Amazon)
- 6) Design the test cases for a Mobile Application (Consider any example from Appstore)
- 7) Design and Implement ATM system through UML Diagrams.

IV SEMESTER	L	T	P	C
	-	-	3	1.5

23CS4L02-OBJECT ORIENTED PROGRAMMING THROUGH JAVA LAB
Course Objectives:

The aim of this course is to

- Practice object-oriented programming in the Java programming language
- implement Classes, Objects, Methods, Inheritance, Exception, Runtime Polymorphism, User defined Exception handling mechanism
- Illustrate inheritance, Exception handling mechanism, JDBC connectivity
- Construct Threads, Event Handling, implement packages, Java FX GUI

Experiments covering the Topics:

- Object Oriented Programming fundamentals- data types, control structures
- Classes, methods, objects, Inheritance, polymorphism,
- Exception handling, Threads, Packages, Interfaces
- Files, I/O streams, JDBC

Sample Experiments:
Exercise – 1:

- Write a JAVA program to display default value of all primitive data type of JAVA
- Write a java program that display the roots of a quadratic equation $ax^2+bx=0$. Calculate the discriminate D and basing on value of D, describe the nature of root.

Exercise - 2

- Write a JAVA program using StringBuffer to delete, remove character.
- Write a JAVA program to implement class mechanism. Create a class, methods and invoke them inside main method.

Exercise - 3

- Write a JAVA program implements method overloading.
- Write a JAVA program to implement constructor.
- Write a JAVA program to implement constructor overloading.

Exercise - 4

- Write a JAVA program to implement Single Inheritance
- Write a JAVA program to implement multi-level Inheritance
- Write a JAVA program for abstract class to find areas of different shapes

Exercise - 5

- Write a JAVA program give example for “super” keyword.
- Write a JAVA program to implement Interface. What kind of Inheritance can be achieved?
- Write a JAVA program that implements Runtime polymorphism

Exercise - 6

- Write a JAVA program that describes exception handling mechanism
- Write a JAVA program Illustrating Multiple catch clauses

- Write a JAVA program for creation of Java Built-in Exceptions
- Write a JAVA program for creation of User Defined Exception

Exercise - 7

- a) Write a JAVA program that creates threads by extending Thread class. First thread display “Good Morning “every 1 sec, the second thread displays “Hello “every 2 seconds and the third display “Welcome” every 3 seconds, (Repeat the same by implementing Runnable)
- b) Write a program illustrating **is Alive** and **join ()**
- c) Write a Program illustrating Daemon Threads.

Exercise – 8

- a) Write a java program that connects to a database using JDBC
- b) Write a java program to connect to a database using JDBC and insert values into it.
- c) Write a java program to connect to a database using JDBC and delete values from it

IV SEMESTER	L	T	P	C
	-	1	2	2

23CS4S01-FULL STACK DEVELOPMENT-1 (SKILL ENHANCEMENT COURSE)

Course Objectives:

The main objectives of the course are to

- Design and build static as well as dynamic web pages and interactive web-based applications
- Web development using Django framework.
- Analyze and create functional website in Django and deploy Django Web Application on Cloud

Course Objectives:

The main objectives of the course are to

1. Make use of HTML elements and their attributes for designing static web pages
2. Build a web page by applying appropriate CSS styles to HTML elements
3. Experiment with JavaScript to develop dynamic web pages and validate forms

FULL STACK WEB APPLICATION DEVELOPMENT**UNIT I FUNDAMENTALS & TYPESCRIPT LANGUAGE**

Server-Side Web Applications. Client-Side Web Applications. Single Page Application. About TypeScript. Creating TypeScript Projects. TypeScript Data Types. Variables. Expression and Operators. Functions. OOP in Typescript. Interfaces. Generics. Modules. Enums. Decorators. Enums. Iterators. Generators.

Experiments:**Server-Side Web Applications:**

1. **Setting Up a Node.js Server:**
 - Install Node.js and create a basic HTTP server using the built-in http module.
 - Experiment with routing and handling different HTTP methods (GET, POST, etc.).
2. **Using Express.js:**
 - Create a server-side web application using Express.js.
 - Implement routes for handling CRUD operations (Create, Read, Update, Delete).
3. **Database Integration:**
 - Integrate a database (e.g., MongoDB, PostgreSQL) with your Node.js application using Mongoose or Sequelize.
 - Implement data models and perform basic database operations (insert, retrieve, update, delete).
4. **Authentication and Authorization:**
 - Implement user authentication using packages like Passport.js.
 - Experiment with different authentication strategies (e.g., local strategy, OAuth).
5. **RESTful API Development:**
 - Design and implement RESTful APIs using Express.js.
 - Test APIs using tools like Postman or curl.
6. **Error Handling and Middleware:**
 - Implement error handling middleware to manage errors gracefully in your Node.js application.
 - Create custom middleware for logging, authentication checks, etc.

Client-Side Web Applications:

7. **HTML and CSS Basics:**
 - Create a simple HTML page with CSS styling.
 - Experiment with different layouts, styles, and responsive design principles.
8. **JavaScript DOM Manipulation:**
 - Use JavaScript to manipulate the DOM (Document Object Model) dynamically.
 - Implement interactive features like form validation, image sliders, etc.

9. AJAX and Fetch API:

- Make asynchronous requests to fetch data from a server using AJAX or the Fetch API.
- Update DOM elements dynamically based on the fetched data.

10. Single Page Application (SPA):

- Implement a basic SPA using a front-end framework like React.js, Angular, or Vue.js.
- Manage routing, state management, and data binding within the SPA.

11. Client-Side Frameworks:

- Choose a client-side framework (e.g., React, Angular, Vue) and build a simple application using its components and features.
- Explore component-based architecture, state management solutions (like Redux or Vuex), and reactive programming.

TypeScript:

12. Setting Up a TypeScript Project:

- Initialize a new TypeScript project using tsc (TypeScript compiler) or a modern JavaScript framework like Angular.

13. TypeScript Data Types:

- Explore and experiment with TypeScript data types: primitives (number, string, boolean), arrays, tuples, enums, etc.
- Create variables and functions with specified types and observe type inference.

14. Variables, Expressions, and Operators:

- Define variables with different scopes (local, global).
- Experiment with TypeScript expressions (arithmetic, logical) and operators.

15. Functions in TypeScript:

- Create and call functions with typed parameters and return types.
- Explore optional parameters, default parameters, and rest parameters in TypeScript.

16. Object-Oriented Programming (OOP) in TypeScript:

- Implement classes, constructors, methods, and properties in TypeScript.
- Explore inheritance, encapsulation, and polymorphism concepts.

17. Interfaces and Generics:

- Define interfaces to describe the structure of objects in TypeScript.
- Experiment with optional properties, readonly properties, and function types within interfaces.
- Use generics to create reusable components that work with a variety of types.

18. Modules in TypeScript:

- Organize TypeScript code into modules using import and export statements.
- Create modules that encapsulate related functionality and demonstrate module resolution.

19. Enums and Decorators:

- Define and use enums to declare a set of named constants in TypeScript.
- Implement decorators to modify the behavior of classes, methods, or properties.

20. Iterators and Generators:

- Implement custom iterators and generators in TypeScript.
- Use iterators to iterate over collections and generators to generate sequences lazily.

UNIT II ANGULAR

About Angular. Angular CLI. Creating an Angular Project. Components. Components Interaction. Dynamic Components. Angular Elements. Angular Forms. Template Driven Forms. Property, Style, Class and Event Binding. Two way Bindings. Reactive Forms. Form Group. Form Controls. About Angular Router. Router Configuration. Router State. Navigation Pages. Router Link. Query Parameters. URL matching. Matching Strategies. Services. Dependency Injection. HttpClient. Read Data from the Server. CRUD Operations. Http Header Operations. Intercepting requests and responses.

Experiments

Angular CLI and Project Setup

1. **Install Angular CLI:** Install Angular CLI globally on your machine.

2. **Create a New Angular Project:** Use Angular CLI to generate a new Angular project.
3. **Serve the Application:** Use ng serve to serve the Angular application locally.

Components

4. **Create Components:** Generate new components using Angular CLI (ng generate component) and explore their structure.
5. **Component Interaction:** Implement communication between components using @Input and @Output.

Dynamic Components and Angular Elements

6. **Dynamic Components:** Create components dynamically at runtime based on user interactions or data.
7. **Angular Elements:** Build an Angular component as a custom element (Web Component) for use in non-Angular applications.

Forms in Angular

8. **Template Driven Forms:** Create forms using template-driven approach (ngModel, validation).
9. **Reactive Forms:** Implement reactive forms with FormGroup, FormControl, and validation.
10. **Form Group and Form Controls:** Explore complex form structures using nested form groups and controls.

Binding and Directives

11. **Property Binding:** Use [property]="value" to bind properties in templates.
12. **Style Binding:** Dynamically apply styles using [style.property]="value".
13. **Class Binding:** Conditionally apply CSS classes using [class.class-name]="condition".
14. **Event Binding:** Handle DOM events using (event)="expression".

Angular Router

15. **Configure Routes:** Set up routing in your Angular application using RouterModule.forRoot() and define routes (Routes array).
16. **Router State:** Access route parameters and query parameters using ActivatedRoute.
17. **Navigation and RouterLink:** Navigate between routes using [routerLink] directive and programmatic navigation with Router service.
18. **Query Parameters and URL Matching:** Pass and retrieve query parameters in route navigation and use different matching strategies (prefix, full, etc.).

Services and Dependency Injection

19. **Create Services:** Generate Angular services (ng generate service) to encapsulate business logic or data fetching.
20. **Dependency Injection:** Inject services into components, modules, and other services using Angular's built-in dependency injection.

HttpClient and CRUD Operations

21. **HttpClient Module:** Import HttpClientModule and use HttpClient to make HTTP requests to a server.
22. **Read Data from the Server:** Fetch data from a RESTful API and display it in your Angular application.
23. **CRUD Operations:** Implement Create, Read, Update, and Delete operations with HttpClient (get(), post(), put(), delete()).
24. **Http Headers Operations:** Set headers (e.g., Content-Type) for HTTP requests and handle response headers.

Interceptors and Error Handling

25. **Intercepting Requests and Responses:** Create HTTP interceptors to modify requests or responses globally in your application.
26. **Global Error Handling:** Implement a global error handler using ErrorHandler to manage and log errors.

Miscellaneous

27. **Testing:** Write unit tests for components, services, and HTTP requests using Jasmine and Karma.

28. **Deployment:** Deploy your Angular application to platforms like Firebase, AWS, or GitHub Pages.

Advanced Concepts (Optional)

29. **Lazy Loading Modules:** Optimize performance by lazy loading feature modules using `loadChildren` in routes.
30. **Animation:** Implement animations in Angular using Angular's animation library (`@angular/animations`).

Project-Based Experiments

31. **Task Management Application:** Create a task management application with CRUD operations, authentication, and routing.
32. **E-commerce Platform:** Build an e-commerce platform with product listings, shopping cart functionality, and checkout process.
33. **Real-Time Chat Application:** Develop a real-time chat application using Angular with WebSocket integration.
34. **Dashboard with Data Visualization:** Build a dashboard that fetches and visualizes data using charts (e.g., Chart.js).

1. Develop Angular JS program that allows user to input their first name and last name and display their full name. Note: The default values for first name and last name may be included in the program.
2. Develop an Angular JS application that displays a list of shopping items. Allow users to add and remove items from the list using directives and controllers. Note: The default values of items may be included in the program
3. Develop a simple Angular JS calculator application that can perform basic mathematical operations (addition, subtraction, multiplication, division) based on user input.
4. Write an Angular JS application that can calculate factorial and compute square based on given user input.
5. Develop Angular JS application that displays a detail of students and their CGPA. Allow users to read the number of students and display the count. Note: Student details may be included in the program
6. Develop an Angular JS program to create a simple to-do list application. Allow users to add, edit, and delete tasks. Note: The default values for tasks may be included in the program.
7. Write an Angular JS program to create a simple CRUD application (Create, Read, Update, and Delete) for managing users.
8. Develop Angular JS program to create a login form, with validation for the username and password fields.
9. Create an Angular JS application that displays a list of employees and their salaries. Allow users to search for employees by name and salary. Note: Employee details may be included in the program.
10. Create Angular JS application that allows users to maintain a collection of items. The application should display the current total number of items, and this count should automatically update as items are added or removed. Users should be able to add items to the collection and remove them as needed. Note: The default values for items may be included in the program.
11. Create Angular JS application to convert student details to Uppercase using angular filters. Note: The default details of students may be included in the program.
12. Create an Angular JS application that displays the date by using date filter parameters

UNIT III NODE.js

About Node.js. Configuring Node.js environment. Node Package Manager NPM. Modules. Asynchronous Programming. Call Stack and Event Loop. Callback functions. Callback errors. Abstracting callbacks. Chaining callbacks. File System. Synchronous vs. asynchronous I/O. Path and directory operations. File Handle. File Synchronous API. File Asynchronous API. File Callback API. Timers. Scheduling Timers. Timers Promises API. Node.js Events. Event Emitter. Event Target and Event API. Buffers. Buffers and Typed Arrays. Buffers and iteration. Using buffers for binary data. Flowing vs. non-flowing streams. JSON.

Experiments:

Setting Up Node.js Environment:

- Install Node.js and set up a development environment on your machine. Experiment with different versions using nvm.

Using NPM:

- Create a new Node.js project, install dependencies using NPM, and explore package.json configurations.

Modules and Module System:

- Create and use custom modules in Node.js. Explore how modules can encapsulate functionality and promote code reuse.

Asynchronous Programming with Callbacks:

- Write functions that use callbacks to handle asynchronous tasks such as reading files or making HTTP requests.

Handling Callback Errors:

- Practice error handling in callback-based APIs. Implement strategies to propagate and handle errors effectively.

Abstracting Callbacks with Promises:

- Convert callback-based functions to use Promises. Implement promise chains for sequential asynchronous operations.

Chaining Callbacks with Async/Await:

- Rewrite asynchronous code using async functions and await expressions. Compare readability and error handling with callback-based and Promise-based approaches.

File System Operations:

- Perform file read/write operations synchronously and asynchronously using the fs module. Handle errors and explore different file handling techniques.

Path and Directory Operations:

- Use the path module to manipulate file paths. Experiment with path joining, resolution, and directory operations.

Timers and Event Loop:

- Explore Node.js timers with setTimeout, setInterval, and setImmediate. Understand how these interact with the event loop.

Event Emitter and Custom Events:

- Create custom events using the events module in Node.js. Implement event emitters and listeners for different scenarios.

Buffers and Typed Arrays:

- Experiment with creating and manipulating buffers and TypedArrays in Node.js. Handle binary data and perform operations like slicing and copying.

□ Streams:

- Implement flowing and non-flowing streams using built-in Node.js modules. Experiment with piping data between streams and handling events.

JSON Handling:

- Read and write JSON data using built-in functions in Node.js. Explore parsing and stringifying JSON objects.

UNIT IV EXPRESS.JS

Express.js. How Express.js Works. Configuring Express.js App Settings. Defining Routes. Starting the App. Express.js Application Structure. Configuration, Settings. Middleware. body-parser. cookie-parser. express-session. response-time. Template Engine. Jade. EJS. Parameters. Routing. router.route(path). Router Class. Request Object. Response Object. Error Handling. RESTful.

Experiments

Basic Experiments

1. **Hello World:** Create a simple Express.js server that responds with "Hello, World!" when accessed.
2. **Static Files:** Serve static files (HTML, CSS, JavaScript) using `express.static` middleware.
3. **Routing:** Define multiple routes (GET, POST, PUT, DELETE) and handle different HTTP methods.
4. **Middleware:** Create custom middleware functions to log requests, manipulate request/response objects, etc.
5. **Error Handling:** Implement error handling middleware to catch and respond to errors.

Intermediate Experiments

6. **Template Engines:** Use a template engine like EJS or Pug to render dynamic HTML pages.
7. **Form Handling:** Create a form that sends data to the server using POST method, and handle the form data on the server.
8. **Sessions and Cookies:** Implement session management using `express-session` and handle cookies with `cookie-parser`.
9. **RESTful APIs:** Build a RESTful API with endpoints for CRUD operations on a resource (e.g., users, products).
10. **Database Integration:** Connect Express.js with a database (e.g., MongoDB, MySQL) using appropriate libraries (`mongoose`, `mysql2`, etc.) and perform CRUD operations.

Advanced Experiments

11. **Authentication:** Implement user authentication and authorization using libraries like `Passport.js`.
12. **File Uploads:** Allow users to upload files to the server and handle file uploads using `multer` middleware.
13. **Real-Time Communication:** Integrate WebSocket functionality using libraries like `Socket.io` for real-time communication between client and server.
14. **Security:** Implement security measures such as CSRF protection, input validation, and secure HTTP headers.
15. **Testing:** Write unit tests for your Express.js routes and middleware using testing frameworks like `Mocha` and `Chai`.

Project-Based Experiments

16. **Blog Application:** Build a simple blog application with features like user registration, post creation, and commenting.
17. **E-commerce Store:** Develop an e-commerce platform with product listings, shopping cart functionality, and checkout process.
18. **Social Media Platform:** Create a social media platform with user profiles, post feeds, likes, and comments.
19. **Task Management:** Build a task management application with features for creating tasks, assigning tasks to users, and tracking task status.
20. **Location-Based Service:** Develop a location-based service that retrieves and displays nearby places using geospatial queries.

UNIT V MONGODB

Introduction to MongoDB. Documents. Collections. Subcollections. Database. Data Types. Dates. Arrays. Embedded Documents. CRUD Operations. Batch Insert. Insert Validation. Querying The Documents. Cursors. Indexing. Unique Indexes. Sparse Indexes. Special Index and Collection Types. Full-Text Indexes. Geospatial Indexing. Aggregation framework.

Experiments

1. Documents:

- In MongoDB, data is stored as **documents**, which are BSON (Binary JSON) objects.
- Documents are analogous to rows in relational databases but are schema-less, meaning fields can vary from document to document in the same collection.

2. Collections:

- A **collection** is a grouping of MongoDB documents.
- Collections are analogous to tables in relational databases but do not enforce a schema.

3. Subcollections:

- MongoDB does not have a built-in concept of subcollections like some relational databases.
- Instead, subcollections can be emulated using embedded documents within a parent document.

4. Database:

- A **database** in MongoDB is a container for collections.
- It holds multiple collections which in turn hold documents.

5. Data Types:

- MongoDB supports various data types including strings, numbers, booleans, arrays, objects, dates, and more.

6. Dates:

- Dates in MongoDB are stored as BSON Date objects.
- They can be manipulated and queried using various date operators.

7. Arrays:

- MongoDB supports arrays as a data type within documents.
- Arrays can contain multiple values of any type, including other arrays or embedded documents.

8. Embedded Documents:

- Embedded documents are documents nested within another document.
- They allow for hierarchical relationships to be stored within a single document.

CRUD Operations

CRUD stands for Create, Read, Update, Delete, which are fundamental operations in MongoDB:

- **Create:** Inserting new documents into a collection.
- **Read:** Querying and retrieving documents from a collection.
- **Update:** Modifying existing documents in a collection.
- **Delete:** Removing documents from a collection.

Additional Operations

- **Batch Insert:** Inserting multiple documents into a collection in a single operation for efficiency.
- **Insert Validation:** Validating documents before they are inserted into a collection using validators.
- **Querying:** MongoDB supports powerful querying capabilities using a JSON-like query language.
- **Cursors:** When querying, MongoDB returns a cursor which allows iteration over the query results.

Indexing

- **Indexing:** MongoDB supports indexes to improve query performance.
- **Unique Indexes:** Ensures that the indexed fields do not store duplicate values.
- **Sparse Indexes:** Indexes where documents that do not contain the indexed field are not included in the index.
- **Special Index Types:** Includes text indexes for full-text search and geospatial indexes for location-based queries.

Aggregation Framework

- **Aggregation Framework:** MongoDB provides an aggregation framework for performing aggregation operations such as grouping, sorting, and transforming data.

- It uses pipelines to process documents and output aggregated results.

Lab Exercise: Basic MongoDB Operations

Requirements:

- MongoDB installed on your local machine or accessible MongoDB server.
- MongoDB shell (mongo) or a MongoDB client (like MongoDB Compass).

Exercise Steps:

1. Connect to MongoDB:

- Open your terminal or command prompt.
- Connect to your MongoDB instance by running:
Copy code
mongo
- If MongoDB is running on a different host or port, connect using:
Copy code
mongo --host <hostname>:<port>

2. Create a Database and Collection:

- In MongoDB, databases and collections are created implicitly when data is inserted into them. You can explicitly create them if needed:
Copy code
use mydatabase
db.createCollection("mycollection")

3. Insert Documents:

- Insert some sample documents into your collection:
Copy code
db.mycollection.insertMany([
 { name: "John Doe", age: 30, city: "New York" },
 { name: "Jane Smith", age: 25, city: "San Francisco" },
 { name: "Mike Johnson", age: 35, city: "Seattle" }
])

4. Query Documents:

- Retrieve documents from the collection:
Copy code
db.mycollection.find()

5. Update Documents:

- Update a document in the collection:
Copy code
db.mycollection.updateOne(
 { name: "John Doe" },
 { \$set: { age: 31, city: "Boston" } }
)

6. Delete Documents:

- Delete a document from the collection:
Copy code
db.mycollection.deleteOne({ name: "Mike Johnson" })

7. Indexing:

- Create an index on a field (e.g., name):
Copy code

```
db.mycollection.createIndex({ name: 1 })
```

8. **Aggregation:**

- Perform an aggregation operation (e.g., calculate average age):

php

Copy code

```
db.mycollection.aggregate([  
  { $group: { _id: null, avgAge: { $avg: "$age" } } }  
])
```

Textbook: 1. Full Stack Web Application Development, ISBN:9789357574068,SIPH

2. Full-Stack Web Development with MERN: Design, Build, Test and Deploy Production-Grade Web Applications with MongoDB, Express, React and NodeJS (Paperback, Nabendu Biswas)

References 1. Full Stack Web Development: The Comprehensive Guide (Grayscale Indian Edition) (Paperback, Philip Ackermann)

Weblinks: 1. [Full Stack Developer Roadmap](https://roadmap.sh/full-stack) (https://roadmap.sh/full-stack)

3. <https://www.tutorialspoint.com/the-full-stack-web-development/index.asp>

IV SEMESTER	L	T	P	C
	1	0	2	2
23CS4Z01-DESIGN THINKING & INNOVATION				

Course Objectives: The objectives of the course are to

- Bring awareness on innovative design and new product development.
- Explain the basics of design thinking.
- Familiarize the role of reverse engineering in product development.
- Train how to identify the needs of society and convert into demand.
- Introduce product planning and product development process.

Course Outcomes:

COs	Statements	Blooms Level
CO1	Define the concepts related to design thinking.	L1
CO2	Explain the fundamentals of Design Thinking and innovation.	L2
CO3	Apply the design thinking techniques for solving problems in various sectors.	L3
CO4	Analyze to work in a multidisciplinary environment.	L4
CO5	Evaluate the value of creativity.	L5

UNIT – I

Introduction to Design Thinking

Introduction to elements and principles of Design, basics of design-dot, line, shape, form as fundamental design components. Principles of design. Introduction to design thinking, history of Design Thinking, New materials in Industry.

UNIT - II

Design Thinking Process

Design thinking process (empathize, analyze, idea & prototype), implementing the process in driving inventions, design thinking in social innovations. Tools of design thinking - person, customer, journey map, brainstorming, product development

Activity: Every student presents their idea in three minutes, Every student can present design process in the form of flow diagram or flow chart etc. Every student should explain about product development.

UNIT - III

Innovation

Art of innovation, Difference between innovation and creativity, role of creativity and innovation in organizations. Creativity to Innovation. Teams for innovation, Measuring the impact and value of creativity.

Activity: Debate on innovation and creativity, Flow and planning from idea to innovation, Debate on value-based innovation.

UNIT - IV

Product Design

Problem formation, introduction to product design, Product strategies, Product value, Product planning, product specifications. Innovation towards product design Case studies.

Activity: Importance of modeling, how to set specifications, Explaining their own product design.

UNIT – V

Design Thinking in Business Processes

Design Thinking applied in Business & Strategic Innovation, Design Thinking principles that redefine business – Business challenges: Growth, Predictability, Change, Maintaining Relevance, Extreme competition, Standardization. Design thinking to meet corporate needs. Design thinking for Startups. Defining and testing Business Models and Business Cases. Developing & testing prototypes.

Activity: How to market our own product, about maintenance, Reliability and plan for startup.

Textbooks:

1. Tim Brown, Change by design, 1/e, Harper Bollins, 2009.
2. Idris Mootee, Design Thinking for Strategic Innovation, 1/e, Adams Media, 2014.

Reference Books:

1. David Lee, Design Thinking in the Classroom, Ulysses press, 2018.
2. Shrrutin N Shetty, Design the Future, 1/e, Norton Press, 2018.
3. William lidwell, Kritinaholden, &Jill butter, Universal principles of design, 2/e, Rockport Publishers, 2010.
4. Chesbrough.H, The era of open innovation, 2003.

Online Learning Resources:

- <https://nptel.ac.in/courses/110/106/110106124/>
- <https://nptel.ac.in/courses/109/104/109104109/>
- https://swayam.gov.in/nd1_noc19_mg60/preview
- https://onlinecourses.nptel.ac.in/noc22_de16/preview