DEPARTMENT OF INFORMATION TECHNOLOGY

ACADEMIC REGULATIONS (R23)

FOR

B. Tech Four Year Degree Programme

(Applicable for the batches admitted from the A.Y. 2023-24)

PACE INSTITUTE OF TECHNOLOGY AND SCIENCES
(Autonomous)
Approved by AICTE and Govt. of Andhra Pradesh, Accredited by NAAC (A Grade)
Recognized under 2(f) & 12(B) of UGC, Permanently Affiliated to JNTUK, Kakinada
NH-16, Near Valluramma Temple, Ongole-523272
Andhra Pradesh, India.

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 / B.Tech. Degree with a Minor 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. program 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. **Courses of study:**
   The following courses of study are offered at Pace Institute of Technology and Sciences,Ongole

Sl No	Branch	Short name	Code
1	Civil Engineering	CE	01
2	Electrical and Electronics Engineering	EEE	02
3	Mechanical Engineering	ME	03
4	Electronics & Communication Engineering	ECE	04
5	Computer Science and Engineering	CSE	05
6	Computer Science and Information Technology	CSIT	07
7	Information Technology	IT	12
8	Computer Science and Engineering (Internet of Things and Cyber Security Including Block Chain Technology)	CSE(IoT&CSBT)	47
9	Artificial Intelligence and Data Science	AIDS	54
10	Artificial Intelligence and Machine Learning	AIML	61
11	Computer Science and Engineering (Indian Language)	CSE-R	63
12	Electronics Engineering (VLSI Design & Technology)	EE(VLSID&T)	66

4. **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.

5. **Program related terms**

   (a) **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.

   (b) **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

   (c) **Academic Year:** Two consecutive (one odd + one even) semesters constitute one academic year.

   (d) **Choice Based Credit System (CBCS):** The CBCS provides a choice for students to select from the prescribed courses.

6. **Semester/Credits:**

   (a) A semester comprises 90 working days and an academic year is divided into two semesters.

   (b) 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.

   (c) Regular courses may also be completed well in advance through MOOCs satisfying prerequisites.

7. **Structure of the Undergraduate Programme**

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

S.No.	Category	Breakup of Credits (Total 160)	Percentage of total credits	AICTE Recommendation (%)
1	Humanities and Social Science including Management (HM)	13	8%	8-9%
2	Basic Sciences (BS)	20	13%	12-16%
3	Engineering Sciences (ES)	23.5	14%	10-18%
4	Professional Core (PC)	54.5	34%	30-36%
5	Electives – Professional (PE) & Open (OE); Domain Specific Skill Enhancement Courses (SEC)	33	21%	19-23%
6	Internships & Project work (PR)	16	10%	8-11%
7	Mandatory Courses (MC)	Non-credit	Non-credit	-

8. **Course Classification:** All subjects/ 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
		Domain specific 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

9. **Programme Pattern**

(a) Total duration of the of B. Tech (Regular) Programme is four academic years.

(b) Each academic year of study is divided into two semesters.

(c) Minimum number of instruction days in each semester is 90 days.

(d) There shall be mandatory student induction program for freshers, with a three-week 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.

(e) Health/wellness/yoga/sports and NSS /NCC /Scouts & Guides / Community service activities are made mandatory as credit courses for all the undergraduate students.

(f) Courses like Environmental Sciences, Indian Constitution, Technical Paper Writing & IPR are offered as non-credit mandatory courses for all the undergraduate students.

(g) Design Thinking for Innovation & Tinkering Labs are made mandatory as credit courses for all the undergraduate students.

(h) Increased flexibility for students through an increase in the elective component of the curriculum, with 05 Professional Elective courses and 04 Open Elective courses.

(i) 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.

(j) A total of 04 Open Electives are offered in the curriculum. A student can complete the requirement for B.Tech. Degree with a Minor within the 160 credits by opting for the courses offered through various verticals/tracks under Open Electives.

(k) While choosing the electives, students shall ensure that they do not opt for the courses with syllabus contents similar to courses already pursued.

(l) 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.

(m) 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.

(n) There shall also be mandatory full internship in the final semester of the programme along with the project work.

(o) Undergraduate degree with Honors is introduced by the University for the students having good academic record.

(p) Each college shall take 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.

(q) Each college shall 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.

(r) Preferably 25% of course work for the theory courses in every semester shall be conducted in the blended mode of learning.

10. **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 subject. 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.

(A) **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 'P' 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.
- The objective paper shall be conducted by the respective institution on the day of subjective paper test.
- 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: 25
* Marks obtained in second mid: 20
* Final mid semester Marks: (25x0.8) + (20x0.2) = 24

If the student is absent for any one midterm examination, the final mid semester marks shall be arrived at by considering 80% weightage to the marks secured by the student in the appeared examination and zero to the other. For Example:

⋆ Marks obtained in first mid: Absent
⋆ Marks obtained in second mid: 25
⋆ Final mid semester Marks: (25x0.8) + (0x0.2) =20

(b) **Semester End Examination Evaluation:**

End examination of theory subjects shall have the following pattern:

i. There shall be 6 questions and all questions are compulsory.

ii. Question I shall contain 10 compulsory short answer questions for a total of 20 marks such that each question carries 2 marks.

iii. There shall be 2 short answer questions from each unit.

◇ In each of the questions from 2 to 6, there shall be either/or type questions of 10 marks each. Student shall answer any one of them.

iv. The questions from 2 to 6 shall be set by covering one unit of the syllabus for each question.

End examination of theory subjects 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 1mark.

iii. In each part, questions from 2 to 4, there shall be either/or type questions of 10 marks each. Student shall answer any one of them.

iv. The questions from 2 to 4 shall be set by covering one unit of the syllabus for each question.

(B) **Practical Courses**

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

(a) For practical courses, there shall be a continuous evaluation during the semester for 30 sessional marks and end examination shall be for 70 marks.

(b) 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.

(c) 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.

i. Procedure: 20 marks

ii. Experimental work & Results: 30 marks

iii. Viva voce: 20 marks.

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.

(d) 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. Day-to-day work shall be evaluated for 15 marks

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

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.

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.

(e) 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 re=examination shall be conducted for failed candidates for 30 marks satisfying the conditions mentioned in item 1 & 2 of the regulations.

(f) 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.

11. **Skill oriented Courses**

(a) There shall be five skill-oriented courses offered during III to VII semesters.

(b) 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.

(c) 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 concerned teacher and an expert in the subject nominated by the principal.

(d) 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.

(e) The student shall be given an option to choose either the skill courses being offered by the college 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 college to evaluate the grades/marks given for a course by external agencies and convert to the equivalent marks/grades.

(f) The recommended courses offered by external agencies, conversions and appropriate grades/marks are to be approved by the principal at the beginning of the semester. The Head of the Department shall forward such proposals to the principal for approval.

(g) 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 principal.

12. **Massive Open Online Courses (MOOCs):**

A Student has to pursue and complete one course compulsorily through MOOCs approved by the principal. 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.

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.

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.

Necessary amendments in rules and regulations regarding adoption of MOOC courses would be proposed from time to time.

13. **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 institute 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.

(a) The institute shall offer credit mobility for MOOCs and give the equivalent credit weightage to the students for the credits earned through online

learning courses.

(b) 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.

(c) Credit transfer policy will be applicable to the Professional & Open Elective courses only.

(d) The concerned department shall identify the courses permitted for credit transfer.

(e) The institution shall notify at the beginning of semester the list of the online learning courses eligible for credit transfer.

(f) The department shall designate a faculty member as a Mentor for each course to guide the students from registration till completion of the credit course.

(g) The institution shall ensure no overlap of MOOC exams with that of the semester end examination schedule. In case of delay in results, the institution will re-issue the marks sheet for such students.

(h) 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.

(i) The department shall submit the following to the examination cell & systems:

  i. List of students who have passed MOOC courses in the current semester along with the certificate of completion.

  ii. Undertaking form filled by the students for credit transfer.

(j) The institution 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.

14. **Academic Bank of Credits (ABC)**

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

(a) provide option of mobility for learners across the universities of their choice

(b) provide option to gain the credits through MOOCs from approved digital platforms.

(c) facilitate award of certificate/diploma/degree in line with the accumulated credits in ABC

(d) execute Multiple Entry and Exit system with credit count, credit transfer and credit acceptance from students' account.

15. **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 APSCHE / 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, if 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 Exami-

nation conducted in the presence of internal examiner and external examiner appointed by the principal 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.

16. **Guidelines for offering a Minor**

To promote interdisciplinary knowledge among the students, the students admitted into B.Tech. in a major stream/branch are eligible to obtain degree in Minor in another stream.

(a) The Minor program requires the completion of 12 credits in Minor stream chosen.

(b) Two courses for 06 credits related to a Minor are to be pursued compulsorily for the minor degree, but maybe waived for students who have done similar/equivalent courses. If waived for a student, then the student must take an extra elective course in its place. It is recommended that students should complete the compulsory courses (or equivalents) before registering for the electives.

(c) Electives (minimum of 2 courses) to complete a total of 12 credits.

Note: A total of 04 Open Electives are offered in the curriculum. A student can complete the requirement for Minor by opting for the courses offered through various verticals/tracks under Open Electives.

17. **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.

(a) 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.

(b) 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).

(c) 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.

(d) The concerned Head of The Department shall arrange separate class work and timetable of the courses offered under Honors program.

(e) 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.

(f) 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.

(g) The attendance for the registered courses under Honors and regular courses offered for Major degree in a semester are to be considered separately.

(h) A student shall maintain an attendance of 75% in all registered courses under Honors to be eligible for attending semester end examinations.

(i) **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.

(j) 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.

(k) 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:

(a) Students of a Department/Discipline are eligible to opt for Honors program offered by the same Department/Discipline

(b) 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.

(c) If a student is detained due to lack of attendance either in Major or in Honors, registration shall be cancelled.

(d) Transfer of credits from Honors to regular B. Tech degree and vice-versa shall not be permitted.

(e) Honors is to be completed simultaneously with a Major degree program.

Registration for Honors:

(a) 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.

(b) 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.

(c) 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.

(d) There is no fee for registration of subjects for Honors program offered in offline at the respective institutions.

18. **Attendance Requirements:**

(a) A student shall be eligible to appear for the semester end examinations if he/she acquires a minimum of 40% attendance in each subject and 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.

(b) Shortage of Attendance below 65% in aggregate shall in NO CASE be condoned.

(c) A stipulated fee shall be payable towards condonation of shortage of attendance to the University.

(d) 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.

(e) 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.

(f) If any candidate fulfils the attendance requirement in the present semester, he shall not be eligible for readmission into the same class.

(g) 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.

(h) For induction programme attendance shall be maintained as per AICTE norms.

19. **Promotion Rules:**

The following academic requirements must be satisfied in addition to the attendance requirements mentioned in section 16.

(a) A student shall be promoted from first year to second year if he/she fulfils the minimum attendance requirement as per university norms.

(b) 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 *lower* digit) up to in the subjects that have been studied up to III semester.

(c) 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 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.

(d) 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.

20. **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

(a) 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.

(b) 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.

While computing the SGPA the subjects in which the student is awarded Zero grade points will also be included.

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	$\geq$ 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) x 10

21. **With–holding of Results**

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

22. **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 6-credit 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 6-credit 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.

23. **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 Head of The Department shall forward such proposals submitted by the students to the principal. 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

24. **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.

25. **Minimum Instruction Days for a Semester:**

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

26. **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. Similarly, the medium of instruction and examinations in AICTE approved Indian language B.Tech programme are in Telugu and English.

27. **Student Transfers:**

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

28. **General Instructions:**

(a) The academic regulations should be read as a whole for purpose of any interpretation.

(b) Malpractices rules-nature and punishments are appended.

(c) Where the words “he”, “him”, “his”, occur in the regulations, they also include “she”, “her”, “hers”, respectively.

(d) In the case of any doubt or ambiguity in the interpretation of the above rules, the decision of the competent authorities of institution is final.

(e) The institution 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 institute.

(f) In the case of any doubt or ambiguity in the interpretation of the guidelines given, the decision of the 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 / B.Tech. Degree with a Minor 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

   (a) 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.

   (b) 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 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 IV year I semester class work of next year.

4. **Course Pattern**

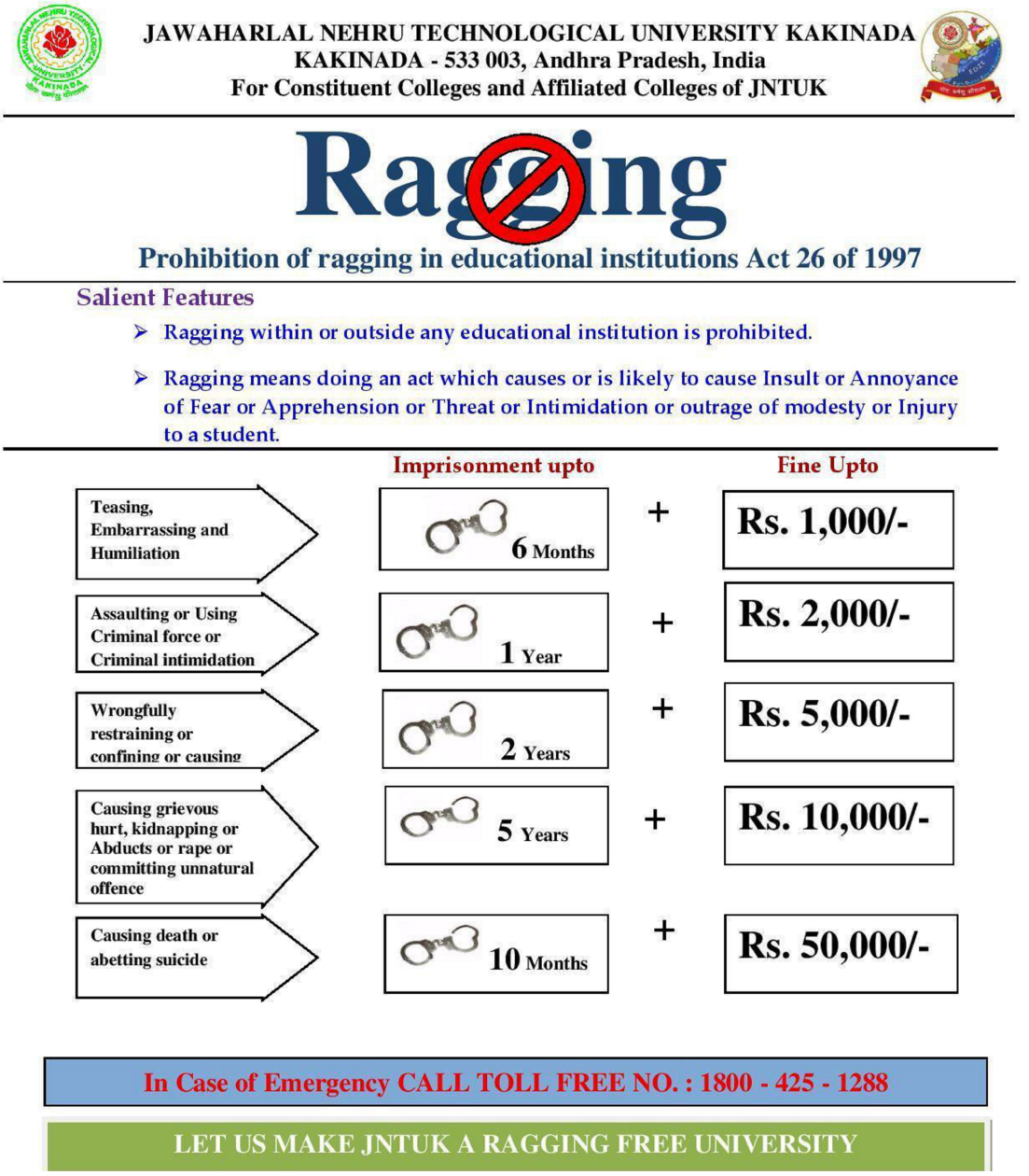
   (a) The entire course of study is three academic years on semester pattern.

(b) 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.

(c) 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.

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

Annexure-I


JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY KAKINADA
KAKINADA - 533 003, Andhra Pradesh, India
For Constituent Colleges and Affiliated Colleges of JNTUK
Ragging
Prohibition of ragging in educational institutions Act 26 of 1997
Salient Features
Ragging within or outside any educational institution is prohibited.
Ragging means doing an act which causes or is likely to cause Insult or Annoyance of Fear or Apprehension or Threat or Intimidation or outrage of modesty or Injury to a student.
Imprisonment upto
Fine Upto
Teasing, Embarrassing and Humiliation
6 Months
+
Rs. 1,000/-
Assaulting or Using Criminal force or Criminal intimidation
1 Year
+
Rs. 2,000/-
Wrongfully restraining or confining or causing
2 Years
+
Rs. 5,000/-
Causing grievous hurt, kidnapping or Abducts or rape or committing unnatural offence
5 Years
+
Rs. 10,000/-
Causing death or abetting suicide
10 Months
+
Rs. 50,000/-
In Case of Emergency CALL TOLL FREE NO. : 1800 - 425 - 1288
LET US MAKE JNTUK A RAGGING FREE UNIVERSITY

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY: KAKINADA
KAKINADA - 533 003, Andhra Pradesh, India
For Constituent Colleges and Affiliated Colleges of JNTUK

Ragging

ABSOLUTELY NO TO RAGGING

1. **Ragging is prohibited as per Act 26 of A.P. Legislative Assembly, 1997.**
2. **Ragging entails heavy fines and/or imprisonment.**
3. **Ragging invokes suspension and dismissal from the College.**
4. **Outsiders are prohibited from entering the College and Hostel without permission.**
5. **Girl students must be in their hostel rooms by 7.00 p.m.**
6. **All the students must carry their Identity Cards and show them when demanded**
7. **The Principal and the Wardens may visit the Hostels and inspect the rooms any time.**

Jawaharlal Nehru Technological University Kakinada
For Constituent Colleges and Affiliated Colleges of JNTUK

In Case of Emergency CALL TOLL FREE NO. : 1800 - 425 - 1288

LET US MAKE JNTUK A RAGGING FREE UNIVERSITY

DEPARTMENT OF INFORMATION TECHNOLOGY

R-23 UNDER GRADUATE (B.Tech) COURSE STRUCTURE

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

DEPARTMENT OF INFORMATION TECHNOLOGY

R-23 UNDER GRADUATE (B.Tech) COURSE STRUCTURE

I Year - I Semester						
S.No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	P23BST04	Engineering Physics	3	0	0	3
2	P23BST02	Linear Algebra & Calculus	3	0	0	3
3	P23EST03	Basic Electrical and Electronics Engineering	3	0	0	3
4	P23EST04	Engineering Graphics	1	0	4	3
5	P23EST02	Introduction to Programming	3	0	0	3
6	P23ESL04	IT Workshop	0	0	2	1
7	P23BSL02	Engineering Physics Lab	0	0	2	1
8	P23ESL03	Electrical and Electronics Engineering Workshop	0	0	3	1.5
9	P23ESL02	Computer Programming Lab	0	0	3	1.5
10	P23BST08	NSS/NCC/Scouts & Guides/Community Service	-	-	1	0.5
Total Credits						20.5

I Year - II Semester						
S.No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	P23BST01	Communicative English	2	0	0	2
2	P23BST06	Chemistry	3	0	0	3
3	P23BST03	Differential Equations & Vector Calculus	3	0	0	3
4	P23EST01	Basic Civil & Mechanical Engineering	3	0	0	3
5	P23CST01	Data Structures	3	0	0	3
6	P23BSL01	Communicative English Lab	0	0	2	1
7	P23BSL04	Chemistry Lab	0	0	2	1
8	P23ESL01	Engineering Workshop	0	0	3	1.5
9	P23CSL01	Data Structures Lab	0	0	3	1.5
10	P23BST07	Health and wellness, Yoga and sports	-	-	1	0.5
Total Credits						19.5

DEPARTMENT OF INFORMATION TECHNOLOGY

R-23 UNDER GRADUATE (B.Tech) COURSE STRUCTURE

II Year I Semester						
S.No.	**Course Code**	**Title**	**L**	**T**	**P**	**C**
1	P23BST10	Discrete Mathematics and Graph Theory	3	0	0	3
2	P23BST12	Universal Human Values – Understanding Harmony	2	1	0	3
3	P23EST09	Digital Logic & Computer Organization	3	0	0	3
4	P23CST03	Object Oriented Programming through Java	3	0	0	3
5	P23ITT01	Software Engeneering	3	0	0	3
6	P23ACT01	Environmental Science (Audit)	2	0	0	-
7	P23CSL03	Object Oriented Programming through Java Lab	0	0	3	1.5
8	P23ITL01	Software Engeneering Lab	0	0	3	1.5
9	P23CBS01	Python Programming (Skill Enhancement)	0	1	2	2
10						
Total Credits			16	2	8	20

DEPARTMENT OF INFORMATION TECHNOLOGY

R-23 UNDER GRADUATE (B.Tech) COURSE STRUCTURE

II Year - II Semester						
S.No.	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	P23MNT01	Managerial Economics and Financial Analysis	2	0	0	2
2	P23BST10	Probability & Statistics	3	0	0	3
3	P23CST05	Operating Systems	3	0	0	3
4	P23CST06	Database Management Systems	3	0	0	3
5	P23CST07	Design and Analysis of Algorithms	3	0	0	3
6	P23CSL04	Operating Systems Lab	0	0	2	1
7	P23CSL05	Database Management Systems Lab	0	0	3	1.5
8	P23CSL06	Django Framework Lab	0	0	3	1.5
9	P23SCT02	Skill Enhancement (Elective)	0	1	2	2
10	P23HST02	Design Thinking & Innovation	1	0	2	2
Total Credits			15	1	12	22
Mandatory Community Service Project Internship of 08 weeks duration during summer vacation						

DEPARTMENT OF INFORMATION TECHNOLOGY

R-23 UNDER GRADUATE (B.Tech) COURSE STRUCTURE

III Year – I Semester						
S.No.	**Course Code**	**Title**	**L**	**T**	**P**	**C**
1	P23ITT02	Advanced Java	3	0	0	3
2	P23CST07	Data Warehousing and Data Mining	3	0	0	3
3	P23CST08	Computer Networks	3	0	0	3
4	P23XXXXX	Professional Elective–I	3	0	0	3
5	P23XXXXX	Open Elective–I	3	0	0	3
6	P23CSL06	Data Mining Lab (Professional Core)	0	0	3	1.5
7	P23CSL07	Computer Networks Lab (Professional Core)	0	0	3	1.5
8	P23ITS03	Full Stack Development – I (Skill Enhancement Course)	0	1	2	2
9	P23XXXXX	User Interface Design using Flutter (Engineeting Science)	0	0	2	1
Evaluation of community Service Internship			0	0	0	2
Total Credits			15	1	10	23

Professional Elective - I		
S.No.	**Course Code**	**Course Title**
1	P23ITE01	Object Oriented Analysis and Design
2	P23ITE02	Artificial Intelligence
3	P23ITE03	Microprocessors & Microcontrollers
4	P23ITE04	Automata Theory & Compiler Design
5	P23ITN01	MOOCS [NPTEL]

Open Elective - I		
S.No.	**Course Code**	**Course Title**
1	P23EOE01	Electronic Devices & Circuits
2	P23EOE05	Introduction to Simulation Software
3	P23ADO04	Introduction to Data Science
4	P23MBO01	Entrepreneurship Development & Venture Creation

DEPARTMENT OF INFORMATION TECHNOLOGY

R-23 UNDER GRADUATE (B.Tech) COURSE STRUCTURE

III Year – II Semester						
S.No.	**Course Code**	**Title**	**L**	**T**	**P**	**C**
1	P23XXXXX	Machine Learning	3	0	0	3
2	P23XXXXX	Cloud Computing	3	0	0	3
3	P23XXXXX	Cryptography & Network Security	3	0	0	3
4	P23XXXXX	Professional Elective–II	3	0	0	3
5	P23XXXXX	Professional Elective–III	3	0	0	3
6	P23XXXXX	Open Elective–II	3	0	0	3
7	P23XXXXX	Machine Learning Lab	0	0	2	1
8	P23XXXXX	Cloud Computing Lab	0	0	3	1.5
9	P23XXXXX	Soft Skills (Skill Enhancement Course)	0	1	2	2
10	P23XXXXX	Technical Paper Writing & IPR (Audit Course)	2	0	0	0
Total Credits			16	1	10	27
Mandatory Community Service Project Internship of 08 weeks duration during summer vacation						

Professional Elective - II		
S.No.	**Course Code**	**Course Title**
1	P23XXXXX	Software Testing Methodologies
2	P23XXXXX	Cyber Security
3	P23XXXXX	DevOps
4	P23XXXXX	Generative AI

Professional Elective - III		
S.No.	**Course Code**	**Course Title**
1	P23XXXXX	Software Project Management
2	P23XXXXX	Mobile Adhoc Networks
3	P23XXXXX	Bigdata Technologies
4	P23XXXXX	Agile Methodologies

Open Elective - II		
S.No.	**Course Code**	**Course Title**
1	P23XXXXX	Digital Marketing
2	P23XXXXX	Principles of Communication
3	P23XXXXX	Management Science
4	P23XXXXX	Principles of Signal Processing

DEPARTMENT OF INFORMATION TECHNOLOGY

R-23 UNDER GRADUATE (B.Tech) COURSE STRUCTURE

IV Year I Semester						
S.No.	**Category**	**Title**	**L**	**T**	**P**	**C**
1	Professional Core		3	0	0	3
2	Professional Core		3	0	0	3
3	Management Course - II		2	0	0	2
4	Professional Elective - IV		3	0	0	3
5	Professional Elective - V		3	0	0	3
6	Open Elective - IV		3	0	0	3
7	Professional Core		0	0	2	1
8	Professional Core		0	0	2	1
9	Skill Enhancement Course		0	1	2	2
10	Audit Course	Constitution of India	2	0	0	-
11	Internship	Evaluation of Industry Internship	-	-	-	2
Total Credits			19	1	6	23

IV Year II Semester						
S.No.	**Category**	**Title**	**L**	**T**	**P**	**C**
1	Internship & Project Work	Full semester Internship & Project Work	0	0	24	12

Course Code	Course Name	Course Structure			
		L	T	P	C
P23BST04	Engineering Physics (Common to All Branches of Engineering)	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives:

To bridge the gap between the Physics in school at 10+2 level and UG level engineering courses by identifying the importance of the optical phenomenon like interference, diffraction etc, enlightening the periodic arrangement of atoms in crystalline solids and concepts of quantum mechanics, introduce novel concepts of dielectric and magnetic materials, physics of semiconductors.

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

CO1: Analyze the intensity variation of light due to interference, diffraction and polarization.

CO2: Familiarize with the basics of crystals and their structures.

CO3: Summarize various types of polarization of dielectrics and classify the magnetic materials.

CO4: Explain the basic concepts of Quantum Mechanics, free electron theory.

CO5: Apply the band theory of solids and Hall Effect to study the semiconductors.

UNIT–I: Wave Optics

Interference: Introduction - Principle of superposition –Interference of light - Interference in thin films (Reflection Geometry) & applications - Colours in thin films- Newton's Rings, Determination of wavelength and refractive index.

Diffraction: Introduction - Fresnel and Fraunhofer diffractions - Fraunhofer diffraction due to single slit, double slit & N-slits (Qualitative) – Diffraction Grating - Dispersive power and resolving power of Grating (Qualitative). Polarization: Introduction -Types of polarization - Polarization by reflection, refraction and Double refraction - Nicol's Prism -Half wave and Quarter wave plates.

UNIT–II: 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 - Miller indices – separation between successive (hkl) planes.

X- ray diffraction: Bragg's law - X-ray Diffractometer – crystal structure determination by Laue's and powder methods

UNIT–III: 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 - complex dielectric constant – Frequency dependence of polarization – dielectric

loss

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–IV: 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–V: 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.

Text Books:

1. A Text book of Engineering Physics, M. N. Avadhanulu, P.G.Kshirsagar & TVS Arun Murthy, S. Chand Publications, 11^{th} 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:

1. `https://www.loc.gov/rr/scitech/selected-internet/physics.html`

Course Code	Course Name	Course Structure			
		L	T	P	C
P23BST02	Linear Algebra & Calculus (Common to All Branches of Engineering)	3	0	0	3

Internal Marks: 30 External Marks: 70

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: Demonstrate the understanding of rank of a matrix. Analyze the solutions of the system of linear equations.

CO2: Find the Eigen values and Eigen vectors of a matrix, apply Cayley-Hamilton theorem to determine inverse and power of a matrix and identify the nature of quadratic form.

CO3: Utilize mean value theorems to real life problems.

CO4: Familiarize with functions of several variables which are useful in optimization.

CO5: Familiarize with double and triple integrals of functions of several variables in two dimensions using Cartesian and polar coordinates and in three dimensions using cylindrical and spherical coordinates.

UNIT–I: Matrices

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: 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.

Text Books:

1. Higher Engineering Mathematics, B. S. Grewal, Khanna Publishers, 2017, 44^{th} Edition
2. Advanced Engineering Mathematics, Erwin Kreyszig, John Wiley & Sons, 2018, 10^{th} Edition.

Reference Books:

1. Thomas Calculus, George B. Thomas, Maurice D. Weir and Joel Hass, Pearson Publishers, 2018, 14^{th} Edition.
2. Advanced Engineering Mathematics, R. K. Jain and S. R. K. Iyengar, Alpha ScienceInternational Ltd., 2021 5^{th} Edition (9^{th} reprint).
3. Advanced Modern Engineering Mathematics, Glyn James, Pearson publishers, 2018, 5^{th} Edition.
4. Advanced Engineering Mathematics, Micheael Greenberg, , Pearson publishers, 9^{th} edition.
5. Higher Engineering Mathematics, H. K Das, Er. Rajnish Verma, S. Chand Publications, 2014, Third Edition (Reprint 2021)

Course Code	Course Name	Course Structure			
		L	T	P	C
P23EST03	Basic Electrical & Electronics Engineering (Common to All branches of Engineering)	3	0	0	3

Internal Marks: 30 External Marks: 70

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

CO1: Describe fundamental laws, operating principles of motors/generators, MC/MI instruments (L2)

CO2: Demonstrate the working of electrical machines, measuring instruments and power generation stations. (L2)

CO3: Apply mathematical tools and fundamental concepts to derive various equations related to electrical circuits and machines. (L3)

CO4: Calculate electrical load and electricity bill of residential and commercial buildings. (L4)

PART A: BASIC ELECTRICAL ENGINEERING

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, Applications.

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

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, 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.

Text 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

Reference Books:

1. Basic Electrical Engineering, D. P. Kothari and I. J. Nagrath, Mc Graw 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. Bhatacharya, Person Publications, 2018, Second Edition.

Web Resources:

1. https://nptel.ac.in/courses/108105053
2. https://nptel.ac.in/courses/108108076

PART B: BASIC ELECTRONICS ENGINEERING

Course Objectives:

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

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

CO1: Compare the operation and characteristics of various semiconductor devices and their utilization.

CO2: Implement various power supply circuits using diodes and amplifier circuits using BJT.

CO3: Categorize various types of Logic gates and implement simple combinational logic circuits.

UNIT–I: SEMICONDUCTOR DEVICES

Introduction - Evolution of electronics – Vacuum tubes to nano electronics - Characteristics of PN Junction Diode — Zener Effect — Zener Diode and its Characteristics. Bipolar Junction

Transistor — CB, CE, CC Configurations and Characteristics.

UNIT–II: BASIC ELECTRONIC CIRCUITS AND INSTRUMENTTAION

Rectifiers and power supplies: Block diagram description of a dc power supply, working of a full wave bridge rectifier, capacitor filter (no analysis), working of simple zener voltage regulator.

Electronic Instrumentation: Block diagram of an electronic instrumentation system.

UNIT–III: DIGITAL ELECTRONICS

Overview of Number Systems, Logic gates including Universal Gates, BCD codes, Excess-3 code, Gray code, Hamming code. Boolean Algebra, Basic Theorems and properties of Boolean Algebra, Truth Tables and Functionality of Logic Gates – NOT, OR, AND, NOR, NAND, XOR and XNOR. Simple combinational circuits–Half and Full Adders.

Text Books:

1. R. L. Boylestad & Louis Nashlesky, Electronic Devices & Circuit Theory, Pearson Education, 2021.
2. R. P. Jain, Modern Digital Electronics, 4^{th} 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.

Semester end examination pattern:

1. Question paper shall be in two parts viz., Part A and Part B with equal weightage of 35 marks each.
2. 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.
3. In each part, questions from 2 to 4, there shall be either/or type questions of 10 marks each. Student shall answer any one of them.
4. The questions from 2 to 4 shall be set by covering one unit of the syllabus for each question.

Course Code	Course Name	Course Structure			
		L	T	P	C
P23EST04	Engineering Graphics (Common to All branches of Engineering)	1	0	4	3

Internal Marks: 30 External Marks: 70

Course Objectives:

1. To enable the students with various concepts like dimensioning, conventions and standards related to Engineering Drawing
2. To impart knowledge on the projection of points, lines and plane surfaces
3. To improve the visualization skills for better understanding of projection of solids
4. To develop the imaginative skills of the students required to understand Section of solids and Developments of surfaces.
5. To make the students understand the viewing perception of a solid object in Isometric and orthographic projections.

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

CO1: Understand the principles of engineering drawing, including polygons, engineering curves, scales.

CO2: Draw and interpret orthographic projections of points, lines, planes.

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

CO4: Explain principles behind sections of solids and development of surfaces.

CO5: Convert the isometric view into orthographic view and vice versa.

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, Involutes, Normal and tangent to Curves.

Scales: Plain scales, diagonal scales and vernier 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: Perpendicular and inclined section planes, Sectional views and True shape of section, 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: Semiconductors

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

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

Text Books:

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

Reference Books:

1. Engineering Drawing, K.L. Narayana and P. Kannaiah, 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.

Course Code	Course Name	Course Structure			
		L	T	P	C
P23EST02	Introduction To Programming (Common to All branches of Engineering)	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives:

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

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

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

CO2: Analyse a problem and develop programs involving decision structures and loops.

CO3: Implement different operations on arrays and solve problems using Strings.

CO4: Design the programs by applying the features of pointers, structures and unions.

CO5: Develop problem-solving skills and the ability to debug and optimize the code by using functions and files.

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, Operations, 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, do- while) Break and Continue.

UNIT–III: Arrays and Strings

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

UNIT–IV: Pointers & User Defined Data types

Pointers, dereferencing (Pointer to Pointer) and address operators, pointer and address arithmetic, 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

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

Text Books:

1. "The C Programming Language", Brian W. Kernighan and Dennis M. Ritchie, Prentice- Hall, 1988
2. 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, 2^{nd} edition
3. C Programming, A Problem Solving Approach, Forouzan, Gilberg, Prasad, CENGAGE, 3rd edition

Course Code	Course Name	Course Structure			
		L	T	P	C
P23ESL04	IT Workshop (Common to All branches of Engineering)	0	0	2	1

Internal Marks: 30 External Marks: 70

Course Objectives:

1. To introduce the internal parts of a computer, peripherals, I/O ports, connecting cables
2. To demonstrate configuring the system as Dual boot both Windows and other Operating Systems Viz. Linux, BOSS
3. To teach basic command line interface commands on Linux.
4. To teach the usage of Internet for productivity and self-paced life-long learning
5. To introduce Compression, Multimedia and Antivirus tools and Office Tools such as Word processors, Spread sheets and Presentation tools.

Course Outcomes:

CO1: Understand the fundamental principles of computer hardware components and their interconnections.

CO2: Analyze the historical development of the Internet and its impact on global communication.

CO3: Analyze the underlying principles and structure of LaTeX and Word documents.

CO4: Implement essential toolbars and ribbons for common spreadsheet tasks.

CO5: Understand the principles of effective content organization in presentations.

CO6: Implement AI tools like ChatGPT into their professional workflows for content creation and translation.

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 LaTeX and Microsoft (MS) office or equivalent (FOSS) tool word: Importance of LaTeX 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 LaTeX and word – Accessing, overview of toolbars, saving files, Using help and resources, rulers, format painter in word.

Task 2: Using LaTeX 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 LaTeX 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, 3^{rd} edition
3. Introduction to Information Technology, ITL Education Solutions limited, Pearson Education, 2012, 2^{nd} 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 Anfins on and Ken Quamme. – CISCO Press, Pearson Education, 3^{rd} edition
7. IT Essentials PC Hardware and Software Labs and Study Guide, Patrick Regan– CISCO Press, Pearson Education, 3^{rd} edition

Course Code	Course Name	Course Structure			
		L	T	P	C
P23BSL02	Engineering Physics Lab (Common to All Branches of Engineering)	0	0	2	1

Internal Marks: 30 External Marks: 70

Course Objectives:

To bridge the gap between the Physics in school at 10+2 level and UG level engineering courses by identifying the importance of the optical phenomenon like interference, diffraction etc, enlightening the periodic arrangement of atoms in crystalline solids and concepts of quantum mechanics, introduce novel concepts of dielectric and magnetic materials, physics of semiconductors.

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

CO1: Apply the concepts of wave optics to get wavelength of light.

CO2: Apply the concept of resonance in sonometer and melde's experiments for getting frequency.

CO3: Study the earth's gravity and rigidity modulus of material.

CO4: Study electrostatics and magnetism to determine its dependent properties.

CO5: Determine the properties of semiconductors and Planks constant.

List of Experiments:

1. Determination of radius of curvature of a given Plano-convex lens by Newton's rings.
2. Determination of wavelengths of different spectral lines in mercury spectrum using diffraction grating in normal incidence configuration.
3. Verification of Brewster's law
4. Determination of dielectric constant using charging and discharging method.
5. Study the variation of B versus H by magnetizing the magnetic material (B-H curve).
6. Determination of wavelength of Laser light using diffraction grating.
7. Estimation of Planck's constant using photoelectric effect.
8. Determination of the resistivity of semiconductors by four probe methods.
9. Determination of energy gap of a semiconductor using p-n junction diode.
10. Magnetic field along the axis of a current carrying circular coil by Stewart Gee's Method.
11. Determination of Hall voltage and Hall coefficient of a given semiconductor using Hall effect.

12. Determination of temperature coefficients of a thermistor.
13. Determination of acceleration due to gravity and radius of Gyration by using a compound pendulum.
14. Determination of magnetic susceptibility by Kundt's tube method.
15. Determination of rigidity modulus of the material of the given wire using Torsional pendulum.
16. Sonometer: Verification of laws of stretched string.
17. Determination of young's modulus for the given material of wooden scale by non- uniform bending (or double cantilever) method.
18. Determination of Frequency of electrically maintained tuning fork by Melde's experiment.

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

References:

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

Web Resources:

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

Course Code	Course Name	Course Structure			
		L	T	P	C
P23ESL03	Electrical & Electronics Engineering Workshop (Common to All branches of Engineering)	0	0	3	1.5

Internal Marks: 30 External Marks: 70

Course Objectives:

To impart knowledge on the fundamental laws & theorems of electrical circuits, functions of electrical machines and energy calculations.

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

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

CO2: **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 Device & Circuit Theory, Pearson Education, 2021.
2. R. P. Jain, Modern Digital Electronics, 4^{th} 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.

Course Code	Course Name	Course Structure			
		L	T	P	C
P23ESL02	Computer Programming Lab (Common to All branches of Engineering)	0	0	3	1.5

Internal Marks: 30 External Marks: 70

Course Objectives:

The course aims to give students hands – on experience and train them on the concepts of the C- programming language.

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

CO1: Understand and trace the execution of programs written in C language.

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

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

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

1. Basic Linux environment and its editors like Vi, Vim & Emacs etc.
2. Exposure to Turbo C, gcc
3. 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.

Lab2: Converting algorithms/flow charts into C Source code.

Developing the algorithms/flowcharts for the following sample programs

1. Sum and average of 3 numbers
2. Conversion of Fahrenheit to Celsius and vice versa
3. 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:
Lab3: Simple computational problems using arithmetic expressions.

1. Finding the square root of a given number
2. Finding compound interest
3. Area of a triangle using heron's formulae
4. 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

1. 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)
2. Find the maximum of three numbers using conditional operator
3. 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, null- else, 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:
Lab5: Problems involving if-then-else structures.

1. Write a C program to find the max and min of four numbers using if-else.

2. Write a C program to generate electricity bill.
3. TFind the roots of the quadratic equation.
4. Write a C program to simulate a calculator using switch case.
5. 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
Lab6: Iterative problems e.g., the sum of series

1. Find the factorial of given number using any loop.
2. Find the given number is a prime or not.
3. Compute sine and cos series
4. Checking a number palindrome
5. 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.
Lab7: 1D Array manipulation, linear search

1. Find the min and max of a 1-D integer array.
2. Perform linear search on1D array.
3. The reverse of a 1D integer array
4. Find 2's complement of the given binary number.
5. 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

1. Addition of two matrices
2. Multiplication two matrices
3. Sort array elements using bubble sort
4. Concatenate two strings without built-in functions
5. 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.

1. Write a C program to find the sum of a 1D array using malloc()
2. Write a C program to find the total, average of n students using structures
3. Enter n students data using calloc() and display failed students list
4. Read student name and marks from the command line and display the student details along with the total.
5. 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

1. Create and display a singly linked list using self-referential structure.
2. Demonstrate the differences between structures and unions using a C program.
3. Write a C program to shift/rotate using bitfields.
4. 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.

1. Write a C function to calculate NCR value.
2. Write a C function to find the length of a string.
3. Write a C function to transpose of a matrix.
4. 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

1. Write a recursive function to generate Fibonacci series.
2. Write a recursive function to find the lcm of two numbers.
3. Write a recursive function to find the factorial of a number.
4. Write a C Program to implement Ackermann function using recursion.
5. 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.

1. Write a C program to swap two numbers using call by reference.
2. Demonstrate Dangling pointer problem using a C program.
3. Write a C program to copy one string into another using pointer.
4. Write a C program to find no of lowercase, uppercase, digits and other characters using pointers.

WEEK 14

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 12: Recursion, the structure of recursive calls
Lab 14: File operations

1. Write a C program to write and read text into a file.
2. Write a C program to write and read text into a binary file using fread() and fwrite()
3. Copy the contents of one file to another file.
4. Write a C program to merge two files into the third file using command-line arguments.
5. Find no. of lines, words and characters in a file
6. Write a C program to print last n characters of a given file.

Text Books:

1. Ajay Mittal, Programming in C: A practical approach, Pearson.
2. Byron Gottfried, Schaum & 39; 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

Course Code	Course Name	Course Structure			
		L	T	P	C
P23BST08	NSS/NCC/SCOUTS & Guides/Community Service (Common to All branches of Engineering)	0	0	1	0.5

Course Objectives:

The objective of introducing this course is to impart discipline, character, fraternity, teamwork, social consciousness among the students and engaging them in selfless service.

Course Outcomes: At the end of the course, the student 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:

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

UNIT–II: Nature & Care :

Activities:

1. Best out of waste competition.
2. Poster and signs making competition to spread environmental awareness.
3. Recycling and environmental pollution article writing competition.
4. Organising Zero-waste day.
5. Digital Environmental awareness activity via various social media platforms.
6. Virtual demonstration of different eco-friendly approaches for sustainable living.

7. Write a summary on any book related to environmental issues.

UNIT–III: Community Service
Activities:

1. 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- authorities- experts-etc.
2. Conducting awareness programs on Health-related issues such as General Health, Mental health, Spiritual Health, HIV/AIDS,
3. Conducting consumer Awareness. Explaining various legal provisions etc.
4. Women Empowerment Programmes- Sexual Abuse, Adolescent Health and Population Education.
5. Any other programmes in collaboration with local charities, NGOs etc.

Reference Books:

1. 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:

1. Evaluated for a total of 100 marks.
2. 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.
3. A student shall be evaluated by the concerned teacher for 10 marks by conducting viva voce on the subject.

Course Code	Course Name	Course Structure			
		L	T	P	C
P23BST01	Communicative English (Common to All Branches of Engineering)	2	0	0	2

Internal Marks: 30 External Marks: 70

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: Readinga 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

Text Books:

1. Pathfinder: Communicative English for Undergraduate Students, 1^{st} Edition, Orient Black Swan, 2023 (Units 1,2 & 3)
2. Empowering with Language by Cengage Publications, 2023 (Units 4 & 5)

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.

Web Resources:

1. www.bbc.co.uk/learningenglish
2. https://dictionary.cambridge.org/grammar/british-grammar/
3. www.eslpod.com/index.html
4. https://www.learngrammar.net/
5. https://english4today.com/english-grammar-online-with-quizzes/
6. https://www.talkenglish.com/grammar/grammar.aspx
7. https://www.youtube.com/c/DailyVideoVocabulary/videos

VOCABULARY

1. https://www.youtube.com/c/DailyVideoVocabulary/videos
2. https://www.youtube.com/channel/UC4cmBAit8i_NJZE8qK8sfpA

Course Code	Course Name	Course Structure			
		L	T	P	C
P23BST06	Chemistry (Common to EEE, ECE, CSE, IT & allied branches)	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives: At the end of the course the student can be able

1. To predict the Fundamentals of Quantum mechanics, energy level diagrams in homo, hetero nuclear molecules.
2. To Illustrate the commonly used industrial materials.
3. To train the students on the principles and applications of electrochemistry.
4. To train the students on the principles and applications of polymers.
5. To introduce instrumental methods, molecular machines and switches.

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

CO1: Explain the Fundamentals of Quantum mechanics, energy level diagrams in homo, hetero nuclear molecules.
CO2: Explain the. commonly used industrial materials.
CO3: Explain the principles and applications of electrochemistry.
CO4: Explain the principles and applications of polymers.
CO5: Explain the instrumental methods and applications.

UNIT–I: Structure and Bonding Models

Fundamentals of Quantum mechanics, Schrodinger Wave equation, significance of ψ and $\psi 2$, particle in one dimensional box, molecular orbital theory – bonding in homo- and heteronuclear diatomic molecules – energy level diagrams of O_2, N_2 and CO, etc. π-molecular orbitals of butadiene and benzene, calculation of bond order.

UNIT–II: Modern Engineering materials

Semiconductors – Introduction, basic concept, application

Super conductors-Introduction basic concept, applications.

Supercapacitors: Introduction, Basic Concept-Classification – Applications.

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

UNIT–III: 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).

Electrochemical sensors – potentiometric sensors with examples, amperometric sensors with examples.

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

UNIT–IV: Polymer Chemistry

Introduction to polymers, functionality of monomers, chain growth and step growth polymerization, coordination 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, carbon fibres.

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

Conducting polymers – polyacetylene, polyaniline, – mechanism of conduction and applications. Bio-Degradable polymers - Poly Glycolic Acid (PGA), Polyl Lactic Acid (PLA).

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. Chromatography-Basic Principle, Classification-HPLC: Principle, Instrumentation and Applications.

Text Books:

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, 5^{th} Edition, Wiley Publications, Feb.2008
3. Textbook of Polymer Science, Fred W. Billmayer Jr, 3^{rd} Edition

Course Code	Course Name	Course Structure			
		L	T	P	C
P23BST03	Differential Equations and Vector Calculus (Common to All Branches of Engineering)	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives:

1. To enlighten the learners in the concept of differential equations and multivariable calculus.
2. 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: Find the complete solution to the higher order linear differential equations and apply these methods to find the current in complex electrical circuits.

CO3: 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, particular integral, general solution, Wronskian, Method of variation of parameters, Applications to L-C-R Circuit problems and Simple Harmonic motion.

UNIT–III: Partial Differential Equations

Introduction, Formation of Partial Differential Equations by elimination of arbitrary constants and arbitrary functions, solutions of first order linear equations using Lagrange's method and non-linear(standard types) equations, Second order PDE: solution of linear PDE with constant coefficients- RHS term of the type $e^{ax+by}, sin(ax+by), cos(ax+by), x^m y^n$

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-Divergence and Curl, Scalar potential functions, Vector identities.

UNIT–V: Vector integration

LWithoutegral-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.

Text Books:

1. Higher Engineering Mathematics, B. S. Grewal, Khanna Publishers, 2017, 44^{th} Edition
2. Advanced Engineering Mathematics, Erwin Kreyszig, John Wiley & Sons, 2018, 10^{th} Edition.

Reference Books:

1. Thomas Calculus, George B. Thomas, Maurice D. Weir and Joel Hass, Pearson Publishers, 2018, 14^{th} Edition.
2. Advanced Engineering Mathematics, R. K. Jain and S. R. K. Iyengar, Alpha ScienceInternational Ltd., 2021 5^{th} Edition(9^{th} reprint).
3. Advanced Modern Engineering Mathematics, Glyn James, Pearson publishers, 2018, 5^{th} Edition.
4. Advanced Engineering Mathematics, Micheael Greenberg, , Pearson publishers, 9^{th} edition
5. Higher Engineering Mathematics, B. V. Ramana, McGraw Hill Education, 2017

Course Code	Course Name	Course Structure			
		L	T	P	C
P23EST01	Basic Civil and Mechanical Engineering (Common to All branches of Engineering)	3	0	0	3

Internal Marks: 30 External Marks: 70

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

CO1: Summarize various sub-divisions of Civil Engineering and to appreciate their role in ensuring better society.

CO2: Identity 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.

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: Polymers and Fuel Chemistry

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).

Text Books:

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

Reference Books:

1. Surveying, Vol- I and Vol-II, S.K. Duggal, Tata McGraw Hill Publishers 2019. Fifth Edition.
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. 38^{th} Edition.
4. Highway Engineering, S.K.Khanna, C.E.G. Justo and Veeraraghavan, Nemchand and Brothers Publications 2019. 10^{th} Edition.
5. Indian Standard DRINKING WATER — SPECIFICATION IS 10500-2012.

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

CO1: Understand the role of mechnical engineering and different engineering materials.

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

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

(**Note:** The subject covers only the basic principles of Civil and Mechanical Engineering systems. The evaluation shall be intended to test only the fundamentals of the subject.)

Text Books:

1. Internal Combustion Engines by V.Ganesan, By Tata McGraw Hill publications (India) Pvt.
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.

Reference Books:

1. Shanmugam and M.S.Palanisamy, Basic Civil and the Mechanical Engineering, Tata McGraw Hill publications (India) Pvt.
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

Course Code	Cyber Security (Professional Elective-I)	Course Structure			
		L	T	P	C
P23XXXXX	Cyber Security (Professional Elective-I)	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives:

1. Identify security risks and take preventive steps.
2. Understand the forensics fundamentals.
3. Understand the evidence capturing process.
4. Understand the preservation of digital evidence.

Course Outcomes:

CO1: Understand types of cybercrimes and threats to information security. (L2)
CO2: Identify and explain common cyberattack tools and techniques. (L1)
CO3: Apply basic methods for cybercrime investigation and evidence handling. (L3)
CO4: Use forensics tools for analyzing systems and retrieving digital evidence. (L3)
CO5: Understand cyber laws, legal frameworks, and IT Act provisions in India. (L2)

UNIT–I:

Introduction to Cybercrime: Introduction, Cybercrime: Definition and Origins of the Word, Cybercrime and Information Security, Cybercriminals, Classifications of Cybercrime, Cyberstalking, Cybercafé and Cybercrimes, Botnets, Attack Vector, Proliferation of Mobile and Wireless Devices, Security Challenges Posed by Mobile Devices, Attacks on Mobile/Cell Phones, Network and Computer Attacks.

UNIT–II:

Tools and Methods: Proxy Servers and Anonymizers, Phishing, Password Cracking, Keyloggers and Spywares, Virus and Worms, Trojan Horses and Back doors, Steganography, Sniffers, Spoofing, Session Hijacking, Buffer overflow, DoS and DDoS Attacks, SQL Injection, Buffer Overflow, Attacks on Wireless Networks, Identity Theft (ID Theft), Footprinting and Social Engineering, Port Scanning, Enumeration.

UNIT–III:

Cybercrime Investigation: Introduction, Investigation Tools, eDiscovery, Digital Evidence Collection, Evidence Preservation, E-Mail Investigation, E-Mail Tracking, IP Tracking, E-Mail Recovery, Hands-on Case Studies. Encryption and Decryption Methods, Search and Seizure of Computers, Recovering Deleted Evidence, Password Cracking.

UNIT–IV:

Computer Forensics and Investigations: Understanding Computer Forensics, Preparing for Computer Investigations. Current Computer Forensics Tools: Evaluating Computer Forensics Tools, Computer Forensics Software Tools, Computer

Forensics Hardware Tools, Validating and Testing Forensics Software, Face, Iris and Fingerprint Recognition, Audio Video Analysis, Windows System Forensics, Linux System Forensics, Graphics and Network Forensics, E-mail Investigations, Cell Phone and Mobile Device Forensics.

UNIT–V:

Cyber Crime Legal Perspectives: Introduction, Cybercrime and the Legal Landscape around the World, The Indian IT Act, Challenges to Indian Law and Cybercrime Scenario in India, Consequences of Not Addressing the Weakness in Information Technology Act, Digital Signatures and the Indian IT Act, Amendments to the Indian IT Act, Cybercrime and Punishment, Cyber law, Technology and Students: Indian Scenario.

Text Books:

1. Sunit Belapure, Nina Godbole, "Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives", WILEY, 2011.
2. Nelson Phillips and Enfinger Steuart, "Computer Forensics and Investigations", Cengage Learning, New Delhi, 2009.

Reference Books:

1. Michael T. Simpson, Kent Backman, James E. Corley, "Hands on Ethical Hacking and Network Defence", Cengage, 2019.
2. John R. Vacca, "Computer Forensics, Computer Crime Investigation", Firewall Media, New Delhi.
3. Alfred Basta, Nadine Basta, Mary Brown, Ravinder Kumar, "Cyber Security and Cyber Laws", Cengage, 2018.

E-Resources:

1. `http://www.cert-in.org.in/`
2. `https://www.coursera.org/learn/introduction-cybersecurity-cyber-attacks`
3. `https://computersecurity.stanford.edu/free-online-videos`
4. `https://ocw.mit.edu/courses/6-858-computer-systems-security-fall-2014`

Course Code	Course Name	Course Structure			
		L	T	P	C
P23BSL01	Communicative English Lab (Common to All Branches of Engineering)	0	0	2	1

Internal Marks: 30 External Marks: 70

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:

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

Suggested Software:

1. Walden Infotech
2. 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/
6. https://www.youtube.com/c/mmmEnglish_Emma/featured
7. https://www.youtube.com/c/ArnelsEverydayEnglish/featured
8. https://www.youtube.com/c/engvidAdam/featured
9. https://www.youtube.com/c/EnglishClass101/featured
10. https://www.youtube.com/c/SpeakEnglishWithTiffani/playlists
11. https://www.youtube.com/channel/UCV1h_cBE0Drdx19qkTM0WNw

Voice & Accent:

1. https://www.youtube.com/user/letstalkaccent/videos
2. https://www.youtube.com/c/EngLanguageClub/featured
3. https://www.youtube.com/channel/UC_OskgZBoS4dAnVUgJVexc
4. https://www.youtube.com/channel/UCNfm92h83W2i2ijc5Xwp_IA

Course Code	Course Name	Course Structure			
		L	T	P	C
P23BSL04	Chemistry Lab (Common to EEE, ECE, CSE, IT & allied branches)	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives:

- To verify the fundamental concepts with experiments

Course Outcomes: At the end of the course, the student 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: Analyse the IR spectra of some organic compounds.
CO5: Calculate strength of acid in Pb-Acid battery

List of Experiments:

1. Measurement of 10Dq by spectrophotometric method
2. Conductometric titration of strong acid vs. strong base
3. Conductometric titration of weak acid vs. strong base
4. Determination of cell constant and conductance of solutions
5. Potentiometry - determination of redox potentials and emfs
6. Determination of Strength of an acid in Pb-Acid battery
7. Preparation of a Bakelite
8. Verify Lambert-Beer's law
9. Wavelength measurement of sample through UV-Visible Spectroscopy
10. Identification of simple organic compounds by IR
11. Preparation of nanomaterials by precipitation method
12. Estimation of Ferrous Iron by Dichrometry

References:

1. "Vogel's Quantitative Chemical Analysis 6^{th} Edition" Pearson Publications by J. Mendham, R.C. Denney, J.D. Barnes and B. Sivasankar

Course Code	Course Name	Course Structure			
		L	T	P	C
P23ESL01	Engineering Workshop (Common to All branches of Engineering)	0	0	3	1.5

Internal Marks: 30 External Marks: 70

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

CO1: Identify workshop tools and their operational capabilities.

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

CO3: Apply sheet metal working operations in various applications and basic repairs of two weeler vehicle.

CO4: Apply basic electrical engineering knowledge for House Wiring Practice.

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.

Text Books:

1. Basic Workshop Technology: Manufacturing Process, Felix W.; Independently Published, 2019.

2. Workshop Processes, Practices and Materials; Bruce J. Black, Routledge publishers, 5^{th} Edn. 2015.

3. A Course in Workshop Technology Vol I. & II, B.S. Raghuwanshi, Dhanpath Rai & Co., 2015 & 2017.

Reference Books:

1. Wiring Estimating, Costing and Contracting; Soni P.M. & Upadhyay P.A.; Atul Prakashan, 2021-22.

2. Workshop Practice by H. S. Bawa, Tata-McGraw Hill, 2004.

3. Elements of Workshop Technology, Vol. I by S. K. Hajra Choudhury & Others, Media Promoters and Publishers, Mumbai. 2007, 14^{th} edition

Course Code	Course Name	Course Structure			
		L	T	P	C
P23CSL01	Data Structures Lab (Common to CSE, IT & allied branches)	0	0	3	1.5

Internal Marks: 15 External Marks: 35

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

CO1: Understand the role of linear data structures in organizing and accessing data efficiently in algorithms.

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

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

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.

CO5: Recognize scenarios where Trees, hashing is advantageous, and design hash-based solutions for specific problems.

List of Experiments:

Exercise 1: Array Manipulation

1. Write a program to reverse an array.
2. C Programs to implement the Searching Techniques – Linear & Binary Search
3. C Programs to implement Sorting Techniques – Bubble, Selection and Insertion Sort

Exercise 2: Linked List Implementation

1. Implement a singly linked list and perform insertion and deletion operations.
2. Develop a program to reverse a linked list iteratively and recursively.
3. Solve problems involving linked list traversal and manipulation.

Exercise 3: Linked List Applications

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

Exercise 4: Double Linked List Implementation

1. Implement a doubly linked list and perform various operations to understand its properties and applications.
2. Implement a circular linked list and perform insertion, deletion, and traversal.

Exercise 5: Stack Operations

1. Implement a stack using arrays and linked lists.
2. Write a program to evaluate a postfix expression using a stack.
3. Implement a program to check for balanced parentheses using a stack.

Exercise 6: Queue Operations

1. Implement a queue using arrays and linked lists.
2. Develop a program to simulate a simple printer queue system.
3. Solve problems involving circular queues.

Exercise 7: Stack and Queue Applications

1. Use a stack to evaluate an infix expression and convert it to postfix.
2. Create a program to determine whether a given string is a palindrome or not.
3. Implement a stack or queue to perform comparison and check for symmetry.

Exercise 8: Binary Search Tree

1. Implementing a BST using Linked List.
2. Traversing of BST.

Exercise 9: Hashing

1. Implement a hash table with collision resolution techniques.
2. Write a program to implement a simple cache using hashing.

Textbooks:

1. Data Structures and algorithm analysis in C, Mark Allen Weiss, Pearson, 2^{nd} 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.

Course Code	Course Name	Course Structure			
		L	T	P	C
P23BST07	Health and Wellness, Yoga and Sports (Common to All branches of Engineering)	0	0	1	0.5

Course Objectives:

The main objective of introducing this course is to make the students maintain their mental and physical wellness by balancing emotions in their life. It mainly enhances the essential traits required for the development of the personality.

Course Outcomes: At the end 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:

1. Organizing health awareness programmes in community
2. Preparation of health profile
3. 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:

1. 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
2. Practicing cardiorespiratory fitness, treadmill, run test, 9 min walk, skipping and running.

Reference Books:

1. Gordon Edlin, Eric Golanty. Health and Wellness, 14^{th} 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. – 3^{rd} 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:

1. Evaluated for a total of 100 marks.
2. 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.
3. A student shall be evaluated by the concerned teacher for 10 marks by conducting viva voce on the subject.

Course Code	Course Name	Course Structure			
		L	T	P	C
P23EST09	DIGITAL LOGIC COMPUTER ORGANIZATION	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives:

1. provide students with a comprehensive understanding of digital logic design principles and computer organization fundamentals.
2. Describe memory hierarchy concepts.
3. Explain input/output (I/O) systems and their interaction with the CPU, memory, and peripheral devices.

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

CO1: Understand the role of linear data structures in organizing and accessing data efficiently in algorithms.

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

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

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.

CO5: Devise novel solutions to small scale programming challenges involving data structures such as stacks, queues, Trees and Recognize scenarios where hashing is advantageous, and design hash-based solutions for specific problems.

UNIT–I:

Data Representation: Binary Numbers, Fixed Point Representation. Floating Point Representation. Number base conversions, Octal and Hexadecimal Numbers, components, Signed binary numbers, Binary codes.

Digital Logic Circuits-I: Basic Logic Functions, Logic gates, universal logic gates, Minimization of Logic expressions. K-Map Simplification, Combinational Circuits, Decoders, Multiplexers

UNIT–II:

Digital Logic Circuits-II: Sequential Circuits, Flip-Flops, Binary counters, Registers, Shift Registers, Ripple counters.

Basic Structure of Computers: Computer Types, Functional units, Basic operational concepts, Bus structures, Software, Performance, multiprocessors and multi computers, Computer Generations, Von- Neumann Architecture

UNIT–III:

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

Processor Organization: Fundamental Concepts, Execution of a Complete Instruction, Multiple-Bus Organization, Hardwired Control and Multi programmed Control

UNIT–IV:

The Memory Organization: Basic Concepts, Semiconductor RAM Memories, Read-Only Memories, Speed, Size and Cost, Cache Memories, Performance Considerations, Virtual Memories, Memory Management Requirements, Secondary Storage

UNIT–V:

Input/Output Organization: Accessing I/O Devices, Interrupts, Processor Examples, Direct Memory Access, Buses, Interface Circuits, Standard I/O Interfaces

Text Books:

1. Computer Organization, Carl Hamacher, Zvonko Vranesic, SafwatZaky, 6th edition, McGraw Hill
2. Digital Design, 6th Edition, M. Morris Mano, Pearson Education.
3. Computer Organization and Architecture, William Stallings, 11th Edition,

Reference Books:

1. Computer Systems Architecture, M.Moris Mano, 3rd Edition, Pearson
2. Computer Organization and Design, David A. Paterson, John L.Hennessy, Elsevier
3. Fundamentals of Logic Design, Roth, 5th Edition, Thomson

Course Code	Course Name	Course Structure			
		L	T	P	C
P23BST10	DISCRETE MATHEMATICS AND GRAPH THEORY	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives:

1. To introduce the students to the topics and techniques of discrete methods and combinatorial reasoning.
2. 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, the student will be able to

CO1: Understand and apply mathematical logic statements and express logical sentences in terms of logical connectives.(L2)

CO2: Understand sets, relations, functions.(L2)

CO3: Able to formulate problems and solve recurrence relations.(L3)

CO4: Analyze the various types of graphs in different geometries related to edges. (L3)

CO5: Able to model and solve real-world problems using graphs and trees.(L3)

UNIT–I: Mathematical Logic:

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:

Backtracking: TBasic 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, 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. Fundamentals of Data Structures in C++, Horowitz, Ellis; Sahni, Sartaj: Mehta, Dinesh, 2nd Edition Universities Press.
2. Computer Algorithms in C++, Ellis Horowitz, Sartaj Sahni, Sangu Thevar Rajasekaran, 2nd Edition University Press

Reference Books:

1. Discrete Mathematical Structures with Applications to Computer Science, J.P. Tremblay and P. Manohar, Tata McGraw Hill.
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.

Course Code	Course Name	Course Structure			
		L	T	P	C
P23ACT01	ENVIRONMENTAL SCIENCE	2	0	0	0

Internal Marks: 30 External Marks: 70

Course Objectives:

1. Overall knowledge of natural resources
2. Basic knowledge of the ecology and its variety
3. familiarity with numerous environmental difficulties caused by unexpected human activity

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

CO1: The ecosystem idea and its role in the environment. The importance of preserving producers and consumers in diverse ecosystems, as well as their roles in the food web.

CO2: The natural resources and their relevance for life sustenance, as well as the necessity to protect natural resources

CO3: Pollution characteristics, repercussions, and strategies to prevent or regulate pollution, as well as waste management techniques.

CO4: India's biodiversity, biodiversity risks, and conservation efforts to conserve biodiversity.

CO5: Social difficulties in both rural and urban environments, as well as potential solutions to obstacles and environmental assessment phases included in EIA and environmental audit.

UNIT–I:

MULTIDISCIPLINARY NATURE OF ENVIRONMENTAL STUDIES: Definition, Scope and Importance– Need for Public Awareness. Renewable energy Resources, Solar energy-solar cells, solar batteries, wind energy, wind mills, ocean energy, tidal energy and non renewable energy resources: LPG, water gas, producer gas. World food problems, degradation and Soil erosion - overgrazing, effects of modern agriculture, fertilizer-pesticide problems, water logging and salinity.

UNIT–II:

ECOSYSTEMS: Concept of an ecosystem. – Structure, Components and function of an ecosystem – Producers, consumers and decomposers – Energy flow in the ecosystem – Ecological succession – Ecological pyramids - Food chains, food webs and Introduction, types, characteristic features, structure and function of the following ecosystem: a. Forest ecosystem. b. Grassland ecosystem c. Desert ecosystem d. Aquatic – River and Lake Ecosystems.

UNIT–III:

BIODIVERSITY AND ITS CONSERVATION: Introduction, Definition: genetic, species and ecosystem diversity – Bio-geographical classification of India –Value of biodiversity: consumptive use, Productive use, social, ethical and aesthetic 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, Endangered and endemic species of India – Conservation of biodiversity: In-situ and Ex-situ conservation of biodiversity.

UNIT–IV:

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. Nuclear hazards.

Solid Waste Management: Causes, effects and control measures of urban and industrial wastes

UNIT–V:

SOCIAL ISSUES AND THE ENVIRONMENT: From unsustainable to sustainable development – Urban problems related to energy – Water conservation, rain water harvesting and watershed management –Environmental ethics: Issues and possible solutions – Climate change, global warming, acid rain, ozone layer depletion, nuclear accidents and holocaust. Case Studies Population growth – Impacts on society, variation among nations. Environmental Impact Assessment (EIA) and Environmental Protection Acts.

Text Books:

1. Text book of Environmental Studies for Undergraduate Courses by Erach Bharucha for University Grants Commission, Universities Press, 2005.
2. Environmental Studies by Benny Joseph, Tata McGraw Hill Co, New Delhi, 2008.

Reference Books:

1. Environmental Science & Engineering by Dr. A. Ravikrishnan, Hitech Publishing Company Pvt. Ltd. 2013
2. Perspectives in Environmental Studies, Second edition, Anubha Koushik and C.P. Koushik, New Age International (P) Limited, Publishers, 2004.

Course Code	Course Name	Course Structure			
		L	T	P	C
P23CST03	OBJECT ORIENTED PROGRAMMING THROUGH JAVA	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives:

1. Identify Java language components and how they work together in applications.
2. Learn the fundamentals of object-oriented programming in Java, including defining classes, invoking methods, using class libraries.
3. Learn how to extend Java classes with inheritance and dynamic binding and how to use exception handling in Java applications.
4. Understand how to design applications with threads in Java.

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

CO1: Understand the OOPS concepts in Java programs. (L2)

CO2: Describe the concepts of classes with inheritance and dynamic binding to design applications. (L3)

CO3: Apply the concepts of inheritance and interfaces to build java programs. (L3)

CO4: Apply the concepts of exceptions, I/O streams and multithreading to develop java applications. (L3)

CO5: Develop Java applications using API. (L6)

UNIT–I:

Object Oriented Programming: Basic concepts, Principles, Program Structure in Java: Introduction, Writing Simple Java Programs, Tokens in Java Programs, Java Statements, Command Line Arguments, User Input to Programs, Escape Sequences Comments.

Data Types, Variables, and Operators: Introduction, Data Types in Java, Declaration of Variables, Type Casting, Scope of Variable Identifier, Constants, Formatted Output with printf() Method, Static Variables and Methods, Attribute Final, Introduction to Operators, Types of operators, Precedence and Associativity of Operators.

Control Statements: Introduction, Conditional Statements, Iteration Statements, Break Statement, Continue Statement.

UNIT–II:

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:

Arrays: Introduction, Assigning Array to Another Array, Dynamic Change of Array Size, Sorting of Arrays, Search for Values in Arrays, Two-dimensional Arrays, Three-dimensional Arrays.

Inheritance: Introduction, Process of Inheritance, Types of Inheritances, Object Class, Inhibiting Inheritance of Class Using Final, Super Keyword, Method Overriding, Dynamic Method Dispatch, Abstract Classes.

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

UNIT–IV:

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, class Math, Wrapper Classes, Auto-boxing and Auto-unboxing, Java util package, Formatter Class, Random Class, Time Package, Class Instant (java.time.Instant), Formatting for Date/Time in Java.

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.

Java I/O and File: Java I/O API, standard I/O streams, types, Byte streams, Character streams, Scanner class, Files in Java(Text Book 2)

UNIT–V:

String Handling in Java: Introduction, Interface Char Sequence, Class String, Methods for Strings, Class String Buffer.

Multithreaded Programming: Introduction, Need for Multiple Threads, 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 FX GUI: Java FX Scene Builder, Java FX App Window Structure, displaying text and image, event handling, laying out nodes in scene graph, mouse events

Text Books:

1. JAVA one step ahead, Anitha Seth, B.L.Juneja, Oxford.
2. Joy with JAVA, Fundamentals of Object Oriented Programming, Debasis Samanta, Monalisa Sarma, Cambridge, 2023.

Reference Books:

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

Course Code	Course Name	Course Structure			
		L	T	P	C
P23CST06	SOFTWARE ENGINEERING	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives: The objectives of this course are to introduce

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

Course Outcomes: At the time of graduation, all Software Engineering students will have demonstrated:

CO1: How to apply the software engineering lifecycle by demonstrating competence in communication planning, analysis, design, construction, and deployment.
CO2: An ability to work in one or more significant application domains.
CO3: Understand the basics of software engineering, object-oriented paradigms, object-oriented a. methodologies used, and various basic terminologies
CO4: Design software requirement specification document, software design document, and test case matrix
CO5: Demonstrate an ability to use the techniques and tools necessary for engineering practice.

UNIT–I:

Introduction: Evolution, Software development projects, Exploratory ¡tyle 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, risk management

Requirements Analysis and Specification: Requirements gathering and analysis, Software Requirements Specification (SRS), Formal system specification, Axiomatic specification, # ebraic specification

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 **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 support life cycle. other characteristics of CASE: tools, Towards second generation Architecture of a CASE Environment

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

Text Books:

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

Reference Books:

1. Software Engineering, lan Sommerville, 10aEdition, Pearson.
2. C 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_012605 89506387148827_s hared/overview
3. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013382 690411003904735_shared/overview

Course Code	Course Name	Course Structure			
		L	T	P	C
P23CSL03	OBJECT ORIENTED PROGRAMMING THROUGH JAVA LAB	0	0	3	1.5

Internal Marks: 50 External Marks: 50

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.

Sample Experiments:

Exercise – 1:

a) Write a JAVA program to display default value of all primitive data type of JAVA.

b) Write a JAVA program that display the roots of a quadratic equation $ax^2+bx=0$. Calculate the discriminant D and based on the value of D, describe the nature of root.

Exercise – 2:

a) Write a JAVA program to search for an element in a given list of elements using binary search mechanism.

b) Write a JAVA program to sort for an element in a given list of elements using bubble sort.

c) Write a JAVA program using StringBuffer to delete, remove character.

Exercise – 3:

a) Write a JAVA program to implement class mechanism. Create a class, methods and invoke them inside main method.

b) Write a JAVA program implements method overloading.

c) Write a JAVA program to implement constructor.

d) Write a JAVA program to implement constructor overloading.

Exercise – 4:

a) Write a JAVA program to implement Single Inheritance.

b) Write a JAVA program to implement multi-level Inheritance.

c) Write a JAVA program for abstract class to find areas of different shapes.

Exercise – 5:

a) Write a JAVA program give example for "super" keyword.

b) Write a JAVA program to implement Interface. What kind of Inheritance can be achieved?

c) Write a JAVA program that implements Runtime polymorphism.

Exercise – 6:

a) Write a JAVA program that describes exception handling mechanism.

b) Write a JAVA program Illustrating Multiple catch clauses.

c) Write a JAVA program for creation of Java Built-in Exceptions.

d) 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 displays "Good Morning" every 1 sec, the second thread displays "Hello" every 2 seconds and the third displays "Welcome" every 3 seconds, (Repeat the same by implementing Runnable).

b) Write a program illustrating isAlive and join().

c) Write a Program illustrating Daemon Threads.

d) Write a JAVA program Producer Consumer Problem.

Exercise – 8:

- Write a JAVA program that imports and uses the user defined packages.

Course Code	Course Name	Course Structure			
		L	T	P	C
P23ITS01	PYTHON PROGRAMMING (SKILL ENHANCEMENT COURSE)	3	0	0	3

Internal Marks: 50 External Marks: 50

Course Objectives: The main objectives of the course are to:

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

Syllabus:

Unit-I: Introduction

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:
    - Arithmetic Operators
    - Relational Operators
    - Assignment Operators
    - Logical Operators
    - Bitwise Operators
    - Ternary Operator
    - Membership Operators

- Identity Operators

5. Write a program to add and multiply complex numbers.
6. Write a program to print the multiplication table of a given number.

Unit-II: Functions, Strings, and Lists

Functions: Built-In Functions, Commonly Used Modules, Function Definition and Calling, `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 Strings 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 a string without using any library functions.
4. Write a program to check if a substring is present in a given string or not.
5. Write a program to perform the given operations on a list:
   - Addition
   - Insertion
   - Slicing
6. Write a program to perform any five built-in functions by taking any list.

Unit-III: Dictionaries, Tuples, and Sets

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, concatenate the tuples, and print the concatenated tuple.
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 and Object-Oriented Programming

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; any upper-case words from source must be converted to lower case.
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 items in an 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 a 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 an array.
6. Create a dictionary with at least five keys where each key maps to a list containing at least ten values. Convert this dictionary to a pandas DataFrame and explore the data as follows:

- Apply `head()` function to the DataFrame.
- Perform various data selection operations on the DataFrame.

7. Select any two columns from the above DataFrame, and observe the change in one attribute with respect to another using scatter and plot operations in matplotlib.

Reference Books:

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

Course Code	Course Name	Course Structure			
		L	T	P	C
P23BST12	UNIVERSAL HUMAN VALUES	2	1	0	3

Internal Marks: 30 External Marks: 70

Course Objectives: At the end of the course, the student can be able

1. 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.
2. 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.
3. 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: At the end of the course, the student will be able to

CO1: Define the terms like Natural Acceptance, Happiness and Prosperity (L1, L2)
CO2: Identify one's self, and one's surroundings (family, society nature) (L1, L2)
CO3: Apply what they have learnt to their own self in different day-to-day settings in real life (L3)
CO4: Relate human values with human relationship and human society. (L4)
CO5: Justify the need for universal human values and harmonious existence (L5) · Develop as socially and ecologically responsible engineers (L3, L6)

UNIT–I: Introduction to Value Education (6 Lectures, 3 Tutorials)

Lecture 1: Right Understanding, Relationship and Physical Facility (Holistic Development and the Role of Education). Lecture 2: Understanding Value Education.
Tutorial 1: Practice Session PS1 Sharing about Oneself.
Lecture 3: Self-exploration as the Process for Value Education. Lecture 4: Continuous Happiness and Prosperity – the Basic Human Aspirations.
Tutorial 2: Practice Session PS2 Exploring Human Consciousness.
Lecture 5: Happiness and Prosperity – Current Scenario. Lecture 6: Method to Fulfill the Basic Human Aspirations.
Tutorial 3: Practice Session PS3 Exploring Natural Acceptance.

UNIT–II: Harmony in the Human Being (6 Lectures, 3 Tutorials)

Lecture 7: Understanding Human being as the Co-existence of the self and the body. Lecture 8: 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, 3 Tutorials)

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, 2 Tutorials)

Lecture 19: Understanding Harmony in 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, 3 Tutorials)

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:

- **UNIT I**: PS1 Sharing about Oneself, PS2 Exploring Human Consciousness, PS3 Exploring Natural Acceptance
- **UNIT II**: 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

- **UNIT III**: PS7 Exploring the Feeling of Trust, PS8 Exploring the Feeling of Respect, PS9 Exploring Systems to fulfil Human Goal
- **UNIT IV**: PS10 Exploring the Four Orders of Nature, PS11 Exploring Co-existence in Existence
- **UNIT V**: PS12 Exploring Ethical Human Conduct, PS13 Exploring Humanistic Models in Education, PS14 Exploring Steps of Transition towards Universal Human Order

Textbook and Teachers Manual:

1. **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

2. **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. JeevanVidya: EkParichaya, A Nagaraj, JeevanVidyaPrakashan, 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 Abdul Kalam Azad
12. Vivekananda - Romain Rolland (English)
13. Gandhi - Romain Rolland (English)

Online Resources:

1. https://fdp-si.aicte-india.org/UHV-II%20Class%20Notes%20&%20Handouts/UHV%20Handout%201Introduction%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/D3S2%20Respect%20July%2023.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%202325%20Ethics%20v1.pdf
8. https://www.studocu.com/in/document/kiet-group-of-institutions/universal-humanvalues/chapter-5-holistic-understanding-of-harmony-on-professional-ethics/62490385
9. https://onlinecourses.swayam2.ac.in/aic22_ge23/preview

Course Code	Course Name	Course Structure			
		L	T	P	C
P23ITL01	SOFTWARE ENGINEERING LAB	0	0	3	1.5

Internal Marks: 50 External Marks: 50

Course Objectives:

The aim of this course is to

- Provide hands-on experience of software engineering principles and practices.
- Develop skills in requirements analysis, project estimation, modeling, testing, and documentation.
- Familiarize with industry-oriented tools and methodologies such as SRS, UML, COCOMO, and Function Point analysis.
- Analyze, design and test software systems using real-world case studies.

Experiments covering the Topics:

- Requirements analysis and Software Requirements Specification (SRS)
- Design and modeling using E-R diagrams, DFDs, CFD, and structured charts
- Effort estimation using COCOMO and Function Point (FP) models
- Test case design for web and mobile applications
- UML diagramming for application development

Sample Experiments:

Exercise – 1:

a) For the following systems, perform Requirement Analysis and prepare SRS:
   - Course Registration System
   - Students Marks Analyzing System
   - Online Ticket Reservation System
   - Stock Maintenance System

b) Draw the E-R diagrams, DFD, CFD, and Structured charts for the above projects.

Exercise – 2:

a) Consider any application and estimate the effort required using the COCOMO model.

Exercise – 3:

a) Consider any application and calculate the effort using Function Point (FP) oriented estimation model.

Exercise – 4:

a) Draw the UML diagrams (Use Case, Class, Sequence, Activity, etc.) for the following problems:

- Course Registration System
- Students Marks Analyzing System
- Online Ticket Reservation System
- Stock Maintenance System

Exercise – 5:

a) Design the test cases for an e-Commerce application (e.g., Flipkart, Amazon).

Exercise – 6:

a) Design the test cases for a Mobile Application (Consider any example from Appstore).

Exercise – 7:

a) Design and Implement ATM System through UML Diagrams.

Course Code	Course Name	Course Structure			
		L	T	P	C
P23CST02	ADVANCED DATA STRUCTURES & ALGORITHMS ANALYSIS	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives:

1. Provide knowledge on advanced data structures frequently used in computer science domain.
2. Develop skills in algorithm design techniques popularly used.
3. Understand the use of various data structures in algorithm design.

Course Outcomes:

CO1: Identify various Time and Space complexities of various algorithms. Apply the concepts of advanced Trees for solving problems effectively.
CO2: Understand Graph Traversal methods and Divide and Conquer Algorithms.
CO3: Apply Greedy and Dynamic Programming concept to solve various problems.
CO4: Apply Backtracking, Branch and Bound concept to solve various problems.
CO5: Implement different performance analysis methods for non-deterministic algorithms.

UNIT–I: Introduction (9 Lectures)

Introduction to Algorithm Analysis, Space and Time Complexity Analysis, Asymptotic Notations.

AVL Trees: Creation, Insertion, Deletion operations and Applications.

B-Trees: Creation, Insertion, Deletion operations and Applications.

UNIT–II: Graphs and Divide & Conquer (9 Lectures)

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

Divide and Conquer: The General Method, Binary Search, Quick Sort, Merge Sort, Strassen's Matrix Multiplication.

UNIT–III: Greedy and Dynamic Programming (10 Lectures)

Greedy Method: The General Method, Job Sequencing with deadlines, Knapsack Problem, Minimum Cost Spanning Trees, Single Source Shortest Paths.

Dynamic Programming: The General Method, All pairs shortest paths, 0/1 Knapsack, String Editing, Travelling Sales Person Problem.

UNIT–IV: Backtracking and Branch & Bound (9 Lectures)

Backtracking: The General Method, 8–Queens Problem, Sum of Subsets Problem, Graph Coloring, Hamiltonian Cycles.

Branch and Bound: The General Method, 15–Puzzle Problem, Travelling Salesperson Problem.

UNIT–V: NP Hard and NP Complete Problems (8 Lectures)

Basic Concepts, Cook's Theorem. **NP Hard Graph Problems**: Click Decision Problem (CDP), Chromatic Number Decision Problem (CNDP).
NP Hard Scheduling Problems: Scheduling Identical Processors, Job Shop Scheduling.

Text Books:

1. Fundamentals of Data Structures in C++, Horowitz, Ellis; Sahni, Sartaj: Mehta, Dinesh, 2nd Edition Universities Press.
2. Computer Algorithms in C++, Ellis Horowitz, Sartaj Sahni, Sangu Thevar 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, Introduction to Algorithms (youtube.com)

Course Code	Course Name	Course Structure			
		L	T	P	C
P23CST05	DATABASE MANAGEMENT SYSTEMS	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives:

1. To introduce about database management systems.
2. To give a good formal foundation on the relational model
3. To introduce the concepts of basic SQL as a universal Database language.
4. To demonstrate the principles behind systematic database design approaches by covering conceptual design, logical design and normalization
5. To provide an overview of physical design of a database system, by discussing Database indexing techniques and storage techniques.
6. To explain Transaction management techniques.

Course Outcomes:

CO1: Describe fundamental concepts of a relational database
CO2: Create, maintain and manipulate a relational database using SQL
CO3: Apply Conceptual and Logical database design
CO4: Apply normalization for database design
CO5: Illustrate Storage management and Transaction management techniques.

UNIT–I:

Introduction: Database system, Characteristics (Database Vs File System), Database Users (Actors on Scene, Workers behind the scenes), Advantages of Database systems, Database applications. 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.

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 **BASIC SQL:** Simple Database schema, data types, table definitions (create, alter). different DAL operations (insert, delete, update), basic SQL querying (select and project) using where clause. arithmetic & logical operations, SQL functions (Date and Time, Numeric, String conversion).

UNIT–III:

Entity Relationship Model: Introduction. Basic features of ER model. Representation of entities attributes, entity set, relationship, relationship set, constraints, ER diagrams Generalization specialization and Aggregation.

SQL: Creating tables with relationships, implementation of key and integrity constraints, nested queries sub queries, grouping, aggregation, ordering, implementation of different types of joins views(updatable and non-updatable), relational set operations

UNIT–IV:

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

UNIT–V:

Transaction Concept: Transaction State,Implementation of Atomicity and Durability. Schedules Serializability. Recoverability, Implementation of Isolation levels, 2Pl and Time stamp ordering protocols. Failure Classification, Recovery and Atomicity, ARIES Recovery algorithm Indexing Techniques: Indexing, Cluster Indexes, Primary and Secondary Indexes. Index data Structures. Hash Based Indexing, B+ Trees: Searching, Insertion, Deletion

Text Books:

1. Database System Concepts by Abraham Silberschatz, Henry F. Korth, S. Sudarshan, 7th Edition McGraw-Hill Education, 2019.
2. Database Management Systems by Raghu Ramakrishnan, Johannes Gehrke, 3rd Edition. McGraw-Hill Education (India), 2014.

Reference Books:

1. Database Principles: Fundamentals of Design. Implementation, and Management by Steven Morris, Keeley Crockett, Carlos Coronel, Craig Blewett, Cengage, 2020
2. Jay I. Devore, Probability and statistics for Engineering and the Sciences, 8th Edition, Cengage.
3. Fundamentals of Database Systems by Ramez Elmasri, Shamkant B. Navathe, 7th Edition Pearson Education India, 2015
4. Introduction to Database Systems by C J Date. 8th Edition, Pearson Education, 2000

Web Resources:

1. https:/nptelac.in courses/106/105/106105175
2. https:/www. geeksforgeeks.org/sql-tutorial Zref=shm
3. https:/www.tutorialspoint.com dbms index htm

Course Code	Course Name	Course Structure			
		L	T	P	C
P23BST17	DESIGN THINKING & INNOVATION	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives: The objectives of the course are to

1. Bring awareness on innovative design and new product development.
2. Explain the basics of design thinking.
3. Familiarize the role of reverse engineering in product development.
4. Train how to identify the needs of society and convert into demand.
5. Introduce product planning and product development process.

Course Outcomes:

CO1: Define the concepts related to design thinking.

CO2: Explain the fundamentals of Design Thinking and innovaApply the design thinking techniques for solving problems in various sectors.

CO4: Analyse to work in a multidisciplinary environment.

CO5: Evaluate the value of creativity.

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, costumer, 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.

Text Books:

1. Database System Concepts by Abraham Silberschatz, Henry F. Korth, S. Sudarshan, 7th Edition McGraw-Hill Education, 2019.
2. Database Management Systems by Raghu Ramakrishnan, Johannes Gehrke, 3rd Edition. McGraw-Hill Education (India), 2014.

Testbooks:

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, & ill butter, Universal principles of design, 2/e, Rockport Publishers, 2010.
4. Chesbrough.H, The era of open innovation, 2003.

Online Learning Resources:

1. https://nptel.ac.in/courses/110/106/110106124/
2. https://nptel.ac.in/courses/109/104/1091041097
3. https://swayam.gov.in/nd1_noc19_mg60/preview/
4. https://swayam.gov.in/nd1_noc19_mg60/preview

Course Code	Course Name	Course Structure			
		L	T	P	C
P23CST04	OPERATING SYSTEMS	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives:

1. Analyze the tradeoffs inherent in operating system design.
2. Summarize the various approaches to solving the problem of mutual exclusion in an operating system.
3. Understand the principles of Deadlocks.
4. Evaluate the trade-offs in terms of memory size (main memory, cache memory, auxiliary memory) and processor speed.
5. Demonstrate disk storage strategies, file strategies

Course Outcomes:

CO1: Describe the underlying principles and techniques of operating system.

CO2: lustrate scheduling and communication methods of processes handled by operating systemsDistinguish the process synchronization methods and deadlock handling approaches employed in operating systems.

CO4: Classify memory management techniques and virtual memory mechanisms.

CO5: Analyze the strategies of disk scheduling algorithms and file system architecture.

UNIT–I: Introduction to Operating System: Operating System Structures: Operating-System Services, User Operating-System Interface. System Calls, Types of System Calls, System Programs, Operating-System Design and Implementation. Boot Operating-System Structure, Virtual Machines, Operating-System Generation,System Boot.

UNIT–II: Process Management: Processes: Process concept, Inter-process Communication, Communication in Client-Server Systems. **Threads:** S Overview, Multithreading Models. **Process Scheduling:** Basic Concepts. Scheduling Criteria, Scheduling Algorithms **UNIT–III:**

Synchronization and Deadlocks: Synchronization: The Critical-Section Problem, Peterson's Solution, Synchronization Hardware. Semaphores, Classic Problems of Synchronization, Monitors. **Deadlocks:** System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention. Deadlock Avoidance, Deadlock Detection, Recovery from deadlock. **UNIT–IV: Memory Management: Memory Management Strategies:** Background, Swapping, Contiguous Memory Allocation. Segmentation, Paging, Structure of Page Table. **Virtual Memory Management:** Background, Demand Paging, Page Replacement, allocation of frames, Thrashing,

UNIT–V:

File System Management: File system Interface: The concept of a file, Access Methods, Directory structure, File system mounting. **File System implementation:** File system structure, allocation methods, free space management: Mass storage structure, overview of Mass-storage structure, Disk scheduling. **Text Books:**

1. Operating System Concepts, Abraham Silberschatz, Peter Baer Galvin and Greg Gagne 9th Edition. John Wiley and Sons Inc., 2012.
2. Operating System – Internals and Design Principles, William Starlings, 7th Edition,Prentice Hall,2011.

Reference Books:

1. Modern Operating Systems, Andrew S. Tanenbaum, Second Edition, Addison Wesley. 2001
2. Operating Systems: A Design-Oriented Approach, Charles Crowley. Tata Me Graw Hill Education" 1996.
3. Operating Systems:A Concept-Based Approach, D M Dhamdhere, Second Edition. Tata Me Graw-Hill Education,2007.

Web Resources:

1. https://nptel.ac.in/courses/106/106/106106144/
2. https:/www.tutorialspoint.com/operating system
3. http://www.youtube.com/playlist?list=PL EJXKK7ACSEGPOCFIQTINOEIU44 J Aun
4. https://www pd-archive.com 2016/12/25 operating-sysiem-concepts-Sth-edition sysiem-concepts-9th-edition.operaring- pdf B

Course Code	Course Name	Course Structure			
		L	T	P	C
P23MBT03	OPTIMIZATION TECHNIQUES	2	0	0	2

Internal Marks: 30 External Marks: 70

Course Objectives:

1. To define an objective function and constraint functions in terms of design variables, and then state the optimization problem.
2. To state single variable and multi variable optimization problems, without and with constraints.
3. To explain linear programming technique to an optimization problem, define slack and surplus variables, by using Simplex method.
4. To state transportation and assignment problem as a linear programming problem to determine Simplex method.
5. To study and explain nonlinear programming techniques, unconstrained or constrained, and define exterior and interior penalty functions for optimization problems.

Course Outcomes:

CO1: State and formulate the optimization problem, without and with constraints, by using design variables from an engineering design problem.

CO2: Apply classical optimization techniques to minimize or maximize a multivariable objective function, without or with constraints, and arrive at an optimal solution.

CO3: Apply and Solve transportation and assignment problem by using Linear programming Simplex method.

CO4: Apply gradient and non-gradient methods to nonlinear optimization problems and use interior or exterior penalty functions for the constraints to derive the optimal solutions

CO5: Formulate and apply Dynamic programming technique to inventory control, production planning, engineering design problems etc. to reach a final optimal solution from the current optimal solution.

UNIT–I: Introduction and Classical Optimization Techniques: (10 Lectures) Statement of an Optimization problem, design vector, design constraints, constraint surface, objective function, objective function surfaces, classification of Optimization problems.

Classical Optimization Techniques: Single variable Optimization, multi variable Optimization without constraints, necessary and sufficient conditions for minimum/maximum, multivariable Optimization with equality constraints. Solution by method of Lagrange multipliers, multivariable Optimization with inequality constraints, Kuhn – Tucker conditions.

UNIT–II: **Linear Programming:** (9 Lectures)

Standard form of a linear programming problem, geometry of linear programming problems, definitions and theorems, solution of a system of linear simultaneous equations, pivotal reduction of a general system of equations, motivation to the simplex method, simplex algorithm.

UNIT–III: **Transportation Problem:** (9 Lectures)

Finding initial basic feasible solution by north – west corner rule, least cost method and Vogel's approximation method, testing for optimality of balanced transportation problems, Special cases in transportation problem.

UNIT–IV: **Sequencing Problems:** (9 Lectures)

Definition, Principle assumptions, Problems with n jobs through two machines, Problems with n jobs through three machines, Problems with n jobs and k machines, Problems with two jobs through k machines case.

UNIT–V: **Network Scheduling By Pert/Cpm:** (8 Lectures)

Introduction, Basic terms, Common Errors, Rules of Network construction, Numbering the events (Fulkerson rule), Construction of Network, Earliest event time, Latest allowable time, Determination of Floats and Slack times, Critical Path Method (CPM), Programming Evaluation and Review Technique (PERT).

Text Books:

1. "Engineering optimization: Theory and practice", S. S.Rao, New Age International (P) Limited, 3rd edition, 1998.
2. "Introductory Operations Research", H.S. Kasene & K.D. Kumar, Springer (India), Pvt.LTd.

Reference Books:

1. "Optimization Methods in Operations Research and Systems Analysis", by K.V. Mital and C. Mohan, New Age International (P) Limited, Publishers, 3rd edition, 1996.
2. Operations Research, Dr.S.D.Sharma, Kedarnath, Ramnath & Co

Course Code	Course Name	Course Structure			
		L	T	P	C
P23BST14	PROBABILITY AND STATISTICS	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives:

1. To familiarize the students with the foundations of probability and statistical methods.
2. 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

CO1: Classify the concepts of data science and its importance.

CO2: Interpret the association of characteristics and through correlation and regression tools.

CO3: Apply discrete and continuous probability distributions.

CO4: Apply the various sampling distributions and estimation using small sample distributions.

CO5: Analyse the statistical inferential methods based on small and large sampling tests.

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: 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 an Samples, Sampling distribution of Means and Variance (definition only), Point and Interval estimations, Maximum error of estimate, Central limit theorem (without proof).

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. X 2-test.

Text Books:

1. Miller and Freund's, Probability and Statistics for Engineers, 7/e, Pearson,2008.
2. S.C. Gupta and V.K. Kapoor, Fundamentals of Mathematical Statistics, 11/e; Sultan chand & Sons Publications, 2012.

Reference Books:

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

Web Resources:

1. http://tutorial.math.lamar.edu/Classes/DE/DE.aspx
2. http://mathworld.wolfram.com/topics
3. http://www.nptel.ac.in/course.php

Course Code	Course Name	Course Structure			
		L	T	P	C
P23CSL05	DATABASE MANAGEMENT SYSTEM LAB	1	0	4	3

Internal Marks: 50 External Marks: 50

Course Objectives: This Course will enable students to

1. Populate and query a database using SQL DDL/DML Commands
2. Declare and enforce integrity constraints on a database
3. Writing Queries using advanced concepts of SQL

Course Outcomes:

1. Know about SQL DDL, DML, DCL, TCL commands
2. Know how to write SQL Quires using set operators
3. Know about how to implement PL/SQL programs using conditional loops statements
4. Know about implementing of riggers, cursors, exceptions. procedures, functions and packages
5. Know about implementing connects to a database using JDBC

Experiments covering the Topics:

1. DDL, DML, DCL commandS
2. Queries, nested queries, built-in functions,
3. PL/SQL programming- control structures
4. Procedures, Functions, Cursors, Triggers, 5. Database connectivity- ODBC/JDBC

List of Experiments

1. Creation, altering and dropping of tables and inserting rows into a table (use constraints while creating tables) examples using SELECT command
2. Queries (along with sub-Queries) using ANY, ALL. IN, EXISTS, NOTEXISTS,UNION,INTERSET Example: - Select the roll number and name of the student who secured rank in the class.
3. Queries Creation using Aggregate functions (COUNT, SUM, AVG, MAX and MIN), GROUPBY, HAVING and dropping of Views.
4. Queries using Conversion functions (to_char, to_number and to date), sing functions (Concatenation. Ipad, rpad, Itrim, rtrim, lower, upper, initcap. length, substr and instr), date functions (Sysdate. next day. add_months, last flay. months_between, least, greatest, trun, round, to_char, to_date)
5. 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 tabs 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 SAVE-POINT in PL/SQL block

6. Develop a program that includes the features NESTED IF, CASE and CASE expression. The program can be extended using the NULLIF and COALESCE functions.
7. Program development using WHILE LOOPS, numeric FOR LOOPS, nested loops using ERROR Handling. BUILT -IN Exceptions, USE defined Exceptions, RAISE-APPLICATION ERROR.
8. Programs development using creation of procedures, passing parameters " nd OUT of PROCEDURES
9. Program development functions using creation of stored functions, invoke functions in SQL Statements and write complex functions.
10. Develop of programs clause and using CURSOR features parameters in a CURSOR, FOR UPDATE CURSOR. WHERE variables.
11. Develop OF Programs Triggers using BEFORE and AFTER Triggers, Row and Statement Triggers and INSTEAD of Triggers.
12. Create a table and perform the search operation on table using indexing and non-indexing techniques.
13. Write a Java program that connects to a database using JDBC
14. Write a Java program to connect to a database using JDBC and insert values into it.
15. Write a Java program to connect to a database using JDBC and delete values from it.

Textbooks/Suggested Reading:

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

Course Code	Course Name	Course Structure			
		L	T	P	C
P23CSL04	OPERATING SYSTEM LAB	1	0	4	3

Internal Marks: 50 External Marks: 50

Course Objectives: The main objectives of the course are to

1. Provide insights into system calls, file systems, semaphores,
2. Develop and debug CPU Scheduling algorithms, page replacement algorithms, thread Implementation
3. Implement Bankers Algorithms to Avoid the Dead Lock

Course Outcomes:

1. Illustrate various UNIX fundamentals, commands & system calls
2. Illustrate various process scheduling algorithms.
3. Write different Page replacement algorithms
4. Write different Paging Technique of memory management.
5. Implement the file allocation strategies.

Experiments covering the Topics:

1. UNIX fundamentals, commands & system calls
2. CPU Scheduling algorithms, thread processing
3. IPC. semaphores, monitors, deadlocks
4. Page replacement algorithms, file allocation strategies
5. Memory allocation strategies.

List of Experiments

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 readdic
3. Simulate UNIX commands like cp. Is, grep, etc..
4. Simulate the following CPU scheduling algorithms a) FCFS b) SIE 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

Reference Books:

1. Operating System Concepts, Silberschatz A, Galvin PB, Gagne G, 10th Edition. Wiley 2018.
2. Modern Operating Systems, Tanenbaum A S, 4th Edition, Pearson, 2016
3. Operating Systems -Internals and Design Principles, Stallings W. 9th edition Pearson, 2018
4. Operating Systems: A Concept Based Approach, D.M Dhamdhere, 3rd Edition, McGraw- Hill, 2013.

Online Learning Resources:

1. https://wwiw.cse.ith.ac.in/~mythili/os/
2. https://peterindia.nevOperatingSystems.hum!
3. https://www.cs.washington.edu/~tom/nachosgo.

Course Code	Course Name	Course Structure			
		L	T	P	C
P23ITS02	PYTHON WITH DJANGO	0	1	2	2

Internal Marks: 30 External Marks: 70

Course Objectives:

1. Introduce core programming concepts of Python programming language.
2. Demonstrate about Python data structures like Lists, Tuples, Sets and dictionaries.
3. 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.

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

Course Code	Course Name	Course Structure			
		L	T	P	C
P23ITT01	ADVANCED JAVA	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives:

1. Introduce JDBC for database connectivity and CRUD operations in Java.
2. Explain the architecture and components of J2EE web applications.
3. Teach the use of Servlets for building dynamic web content and managing sessions.
4. Introduce JSP for server-side scripting and web interface development.
5. Explain the use of Spring MVC framework for developing scalable Java web applications.

Course Outcomes:

1. Build JDBC-based database apps. (L6)
2. Design J2EE web applications. (L6)
3. Create dynamic web apps using Servlets. (L6)
4. Develop web pages using JSP. (L6)
5. Build web apps with Spring MVC. (L6)

UNIT–I:

JDBC Programming: JDBC Architecture, Types of JDBC Drivers, Introduction to major JDBC Classes and Interface, Creating simple JDBC Application, Types of Statement (Statement Interface, Prepared Statement, Callable Statement), Exploring Result Set Operations, Batch Updates in JDBC, Creating CRUD Application, Using Row sets Objects, Managing Database Transaction.

UNIT–II:

J2EE and Web Development: J2EE Architecture Types, J2EE Containers, Types of Servers in J2EE Application, HTTP Protocols and API, Request Processing in Web Application, Web Application Structure, Web Containers and Web Architecture Models.

UNIT–III:

Servlet API and Overview: Servlet Introduction, Servlet Life Cycle(SLC), Types of Servlet, Servlet Configuration with Deployment Descriptor, Working with Servlet Context and Servlet Config Object, Attributes in Servlet, Response and Redirection using RequestDispatcher and using sendRedirect Method, Filter API, Manipulating Responses using Filter API, Session Tracking: using Cookies, HTTP Session, Hidden Form Fields and URL Rewriting, Types of Servlet Event: Context Level and Session Level.

UNIT–IV:

Java Server Pages (JSP): Introduction to JSP, Comparison with Servlet, JSP Architecture, JSP: Life Cycle, Scripting Elements, Directives, Action Tags, Implicit

Objects, Expression Language(EL), JSP Standard Tag Libraries(JSTL), Custom Tag, Session Management, Exception Handling, CRUD Application.

UNIT–V:

Java Web Frameworks: Spring MVC Spring: Introduction, Architecture, Spring MVC Module, Life Cycle of Bean Factory, Explore: Constructor Injection, Dependency Injection, Inner Beans, Aliases in Bean, Bean Scopes, Spring Annotations, Spring AOP Module, Spring DAO, Database Transaction Management, CRUD Operation using DAO and Spring API.

Text Books:

1. Dreamtech Software Team, "Black Book Java server programming" J2EE, Dream Tech Publishers, 4th Edition 2008.
2. James Keogh, "Complete Reference J2EE", McGraw Hill publication, 1st Edition, 2017.
3. Subrahmanyam Allamaraju, Cedric Buest, "Professional Java Server Programming", Wiley Publication, 3rd Edition, 2015.
4. Craig walls, "Spring in Action", Manning Publication, 6th Edition, 2022.

Reference Books:

1. Cay Horstmann, Gary Cornell, "Core Java, Volume II: Advanced Features", Pearson publication, 8th Edition, 2008.
2. Maydene Fisher, Jon Ellis, Jonathan Bruce, Addison Wesley, "JDBC™ API Tutorial and Reference", Addison Wesley, 3rd Edition, 2003.
3. Giulio Zambon, "Beginning JSP, JSF and Tomcat", Apress, 1st Edition, 2007.

Course Code	Course Name	Course Structure			
		L	T	P	C
P23CST08	COMPUTER NETWORKS	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives: The main objective of the course is to

1. Understand network types, topologies, reference models, and physical media.
2. Explain data link layer functions and protocols.
3. Describe media access control methods and Ethernet standards.
4. Understand network layer routing, congestion control, and IP addressing.
5. Explain transport and application layer protocols and their services.

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

1. Identify and compare network types, models, and physical media. (L2)
2. Apply data link layer protocols and error control techniques. (L3)
3. Analyze media access control methods and Ethernet technologies. (L4)
4. Design network layer routing and congestion control strategies and understand IP addressing. (L6)
5. Utilize transport and application layer protocols for reliable communication. (L3)

UNIT–I:

Introduction: Network Types, LAN, MAN, WAN, Network Topologies Reference models- The OSI Reference Model- the TCP/IP Reference Model - A Comparison of the OSI and TCP/IP Reference Models, OSI Vs TCP/IP.

Physical Layer –Introduction to Guided Media- Twisted-pair cable, Coaxial cable and Fiber optic cable and introduction about unguided media.

UNIT–II:

Data link layer: Design issues, **Framing:** fixed size framing, variable size framing, flow control, error control, error detection and correction codes, CRC, Checksum: idea, one's complement internet checksum, services provided to Network Layer, **Elementary Data Link Layer protocols:** simplex protocol, Simplex stop and wait, Simplex protocol for Noisy Channel. **Sliding window protocol:** One bit, Go back N, Selective repeat-Stop and wait protocol, Data link layer in HDLC, Point to point protocol (PPP)

UNIT–III:

Media Access Control: Random Access: ALOHA, Carrier sense multiple access (CSMA), CSMA with Collision Detection, CSMA with Collision Avoidance, **Controlled Access:** Reservation, Polling, Token Passing, Channelization: frequency division multiple Access(FDMA), time division multiple access(TDMA), code division multiple access(CDMA). **Wired LANs:** Ethernet, Ethernet Protocol, Standard Ethernet, Fast Ethernet(100 Mbps), Gigabit Ethernet, 10 Gigabit Ethernet.

UNIT–IV:

The Network Layer Design Issues: Store and Forward Packet Switching-Services Provided to the Transport layer- Implementation of Connectionless Service-Implementation of Connection Oriented Service- Comparison of Virtual Circuit and Datagram Networks, Routing Algorithms-The Optimality principle-Shortest path, Flooding, Distance vector, Link state, Hierarchical, Congestion Control algorithms-General principles of congestion control, Congestion prevention polices, Approaches to Congestion Control-Traffic Aware Routing- Admission Control-Traffic Throttling-Load Shedding. Traffic Control Algorithm-Leaky bucket & Token bucket.

Internet Working: How networks differ- How networks can be connected- Tunneling, internetwork routing-, Fragmentation, network layer in the internet – IP protocols-IP Version 4 protocol-IPV4 Header Format, IP addresses, Class full Addressing, CIDR, Subnets-IP Version 6-The main IPV6 header, Transition from IPV4 to IPV6, Comparison of IPV4 & IPV6.

UNIT–V:

The Transport Layer: Transport layer protocols: Introduction-services- port number-User data gram protocol-User datagram-UDP services-UDP applications-Transmission control protocol: TCP services- TCP features- Segment- A TCP connection- windows in TCP- flow control-Error control, Congestion control in TCP. **Application Layer** — World Wide Web: HTTP, Electronic mail-Architecture- web based mail-email security- TELENET-local versus remote Logging-Domain Name System.

Text Books:

1. Andrew S Tanenbaum, “Computer Networks”, Pearson Education, 6th Edition, 2022.
2. Behrouz A. Forouzan, “Data Communications and Networks”, McGraw Hill Education, 5th Edition, 2017.

References Books:

1. Achut S Godbole, Atul Kahate, “Data Communications and Networks”, McGraw Hill Education, 2nd Edition, 2017.
2. Mayank Dave, “Computer Networks”, CENGAGE, 1st Edition, 2012.

Course Code	Course Name	Course Structure			
		L	T	P	C
P23CST07	DATA WAREHOUSING & DATA MINING	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives:

1. Understand data warehousing, OLAP, and basic data mining concepts.
2. Explain key data preprocessing techniques.
3. Apply classification methods and evaluate models.
4. Perform association rule mining using Apriori and FP-Growth.
5. Describe clustering techniques and algorithms.

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

1. Explain the architecture and components of data warehousing and OLAP systems. (L2)
2. Apply data preprocessing techniques to clean and transform data effectively. (L3)
3. Implement and evaluate classification algorithms for predictive modeling. (L3)
4. Apply association rule mining to discover meaningful patterns in data. (L3)
5. Analyze and apply clustering methods for data segmentation and pattern recognition. (L4)

UNIT–I

Data Warehousing and Online Analytical Processing: Basic concepts, Data Warehouse Modeling: Data Cube and OLAP, Data Warehouse Design and Usage, Data Warehouse Implementation, Cloud Data Warehouse, Data Mining and Pattern Mining, Technologies, Applications, Major issues, Data Objects & Attribute Types, Basic Statistical Descriptions of Data, Data Visualization, Measuring Data Similarity and Dissimilarity.

UNIT-II

Data Preprocessing: An Overview, Data Cleaning, Data Integration, Data Reduction, Data Transformation and Data Discretization.

UNIT–III

Classification: Basic Concepts, General Approach to solving a classification problem, Decision Tree Induction: Attribute Selection Measures, Tree Pruning, Scalability and Decision Tree Induction, Visual Mining for Decision Tree Induction, Bayesian Classification Methods: Bayes Theorem, Naïve Bayes Classification, Rule-Based Classification, Model Evaluation and Selection.

UNIT–IV

Association Analysis: Problem Definition, Frequent Itemset Generation, Rule Generation: Confident Based Pruning, Rule Generation in Apriori Algorithm, Compact Representation of frequent item sets, FP-Growth Algorithm.

UNIT–V

Cluster Analysis: Overview, Basics and Importance of Cluster Analysis, Clustering techniques, Different Types of Clusters; K-means: The Basic K-means Algorithm, K-means Additional Issues, Bi-secting K Means, Agglomerative Hierarchical Clustering: Basic Agglomerative Hierarchical Clustering Algorithm DBSCAN: Traditional Density Center-Based Approach, DBSCAN Algorithm, Strengths and Weaknesses.

Text Books:

1. Jiawei Han, Michel Kamber, “Data Mining concepts and Techniques”, Elsevier, 3rd Edition, 2011.
2. Pang-Ning Tan & Michael Steinbach, Vipin Kumar, “Introduction to Data Mining”, Pearson, 2nd Edition, 2021.

Reference Books:

1. Vikram Pudi and P. Radha Krishna, “Data Mining”, Oxford University Press, 2009.
2. Arun K Pujari, “Data Mining Techniques”, 3rd Edition, Universities Press, 2013.

Web References:

1. https://nptel.ac.in/courses/106105174
2. https://onlinecourses.swayam2.ac.in/imb25_mg200/preview

Course Code	Course Name	Course Structure			
		L	T	P	C
P23XXXXX	OBJECT ORIENTED ANALYSIS AND DESIGN (PROFESSIONAL ELECTIVE-I)	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives:

1. Understand and design complex systems using structured methods; analyze satellite-based navigation architecture.
2. Apply UML basics to model the structure of object-oriented systems like traffic management.
3. Create detailed class and object diagrams for advanced systems such as AI-based cryptanalysis.
4. Model user behavior with UML diagrams for a vacation tracking web application.
5. Design behavioral and architectural models for a weather forecasting system.

Course Outcomes:

1. Understand complex systems and apply design principles. (L2)
2. Use UML to create basic structural models. (L3)
3. Design advanced structural models with UML diagrams. (L6)
4. Develop behavioral models to represent system behavior. (L6)
5. Combine modeling techniques for full system architecture. (L2)

UNIT-I

Introduction: The Structure of Complex systems, The Inherent Complexity of Software, Attributes of Complex System, Organized and Disorganized Complexity, Bringing Order to Chaos, Designing Complex Systems. Case Study: System Architecture: Satellite-Based Navigation

UNIT-II

Introduction to UML: Importance of modeling, principles of modeling, object-oriented modeling, conceptual model of the UML, Architecture, and Software Development Life Cycle. Basic Structural Modeling: Classes, Relationships, common Mechanisms, and diagrams. Case Study: Control System: Traffic Management.

UNIT-III

Class & Object Diagrams: Terms, concepts, modeling techniques for Class & Object Diagrams. Advanced Structural Modeling: Advanced classes, advanced relationships, Interfaces, Types and Roles, Packages. Case Study: AI: Cryptanalysis.

UNIT-IV

Basic Behavioral Modeling-I: Interactions, Interaction diagrams Use cases, Use case Diagrams, Activity Diagrams. Case Study: Web Application: Vacation Tracking System

UNIT-V

Advanced Behavioral Modeling: Events and signals, state machines, processes and Threads, time and space, state chart diagrams. Architectural Modeling: Component, Deployment, Component diagrams and Deployment diagrams. Case Study: Weather Forecasting

Text Books:

1. Grady BOOCH, Robert A. Maksimchuk, Michael W. ENGLE, Bobbi J. Young, Jim Conallen, Kellia Houston, "Object-Oriented Analysis and Design with Applications", Pearson, 3rd edition, 2013.
2. Grady Booch, James Rumbaugh, Ivar Jacobson, "The Unified Modeling Language User Guide", Pearson Education, 1st Edition, 2002.

Reference Books:

1. Meilir Page-Jones, "Fundamentals of Object-Oriented Design in UML", Pearson Education, 1st Edition, 2002.
2. Pascal Roques, "Modeling Software Systems Using UML2", Wiley India Private Limited, 2004.
3. Atul Kahate, "Object Oriented Analysis & Design", The McGraw-Hill Companies, 2004.
4. Craig Larman, "Appling UML and Patterns: An introduction to Object – Oriented Analysis and Design and Iterative Development", PHI, 3rd Edition, 2004.

Course Code	Course Name	Course Structure			
		L	T	P	C
P23XXXXX	ARTIFICIAL INTELLIGENCE (PROFESSIONAL ELECTIVE-I)	3	0	0	3

Internal Marks: 30 External Marks: 70

Pre-requisite:

1. Knowledge in Computer Programming.
2. A course on "Mathematical Foundations of Computer Science".
3. Background in linear algebra, data structures and algorithms, and probability.

Course Objectives:

1. Understand basic ML concepts, learning types, and the machine learning workflow.
2. Apply distance-based models like KNN for classification and regression.
3. Learn decision tree and Bayesian methods for classification and regression.
4. Understand linear models, SVMs, and neural networks for supervised learning.
5. Explore clustering techniques for unsupervised pattern discovery.

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

1. Enumerate the Fundamentals of Machine Learning. (L1)
2. Describe Nearest neighbour-based models. (L2)
3. Apply Models based on decision trees and Bayes rule. (L3)
4. Make use of Linear discriminators for machine Learning. (L3)
5. Choose appropriate clustering technique. (L2)

UNIT-I

Introduction: AI problems, foundation of AI and history of AI intelligent agents: Agents and Environments, the concept of rationality, the nature of environments, structure of agents, problem solving agents, problem formulation.

UNIT-II

Searching: Searching for solutions, uniformed search strategies – Breadth first search, depth first Search. Search with partial information (Heuristic search) Hill climbing, A*, AO* Algorithms, Problem reduction, Game Playing-Adversial search, Games, mini-max algorithm, optimal decisions in multiplayer games, Problem in Game playing, Alpha-Beta pruning, Evaluation functions.

UNIT-III

Representation of Knowledge: Knowledge representation issues, predicate logic-logic programming, semantic nets- frames and inheritance, constraint propagation, representing knowledge using rules, rules-based deduction systems. Reasoning under uncertainty, review of probability, Bayes' probabilistic interferences and dempster Shafer theory.

UNIT-IV

Logic concepts: First order logic. Inference in first order logic, propositional vs. first order inference, unification & lifts forward chaining, Backward chaining, Resolution, Learning from observation Inductive learning, Decision trees, Explanation based learning, Statistical Learning methods, Reinforcement Learning.

UNIT-V

Expert Systems: Architecture of expert systems, Roles of expert systems – Knowledge Acquisition Meta knowledge Heuristics. Typical expert systems – MYCIN, DART, XCON: Expert systems shells.

Text Books:

1. S. Russel and P. Norvig, "Artificial Intelligence – A Modern Approach", 3rd Edition, Pearson Education, 2009.
2. Kevin Night and Elaine Rich, Nair B., "Artificial Intelligence (SIE)", Mc Graw Hill, 3rd Edition, 2008.

Reference Books:

1. David Poole, Alan Mackworth, Randy Goebel, "Computational Intelligence: a logical approach", Oxford University Press, 1st Edition, 2004.
2. G. Luger, "Artificial Intelligence: Structures and Strategies for complex problem-solving", Fourth Edition, Pearson Education, 2002.
3. J. Nilsson, "Artificial Intelligence: A new Synthesis", Elsevier Publishers, Standard Edition, 1998.
4. Saroj Kaushik, "Artificial Intelligence", CENGAGE Learning, 1st Edition, 2011.

Online Learning Resources:

1. `https://ai.google/`
2. `https://swayam.gov.in/nd1_noc19_me71/preview`

Course Code	Course Name	Course Structure			
		L	T	P	C
P23XXXXX	MICROPROCESSORS & MICROCONTROLLERS (PROFESSIONAL ELECTIVE-I)	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives:

1. To introduce fundamental architectural concepts of microprocessors and microcontrollers.
2. To impart knowledge on addressing modes and instruction set of 8086 and 8051.
3. To introduce assembly language programming concepts.
4. To explain memory and I/O interfacing with 8086 and 8051.
5. To introduce 16 bit and 32 bit microcontrollers.

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

1. Explain the architecture, features, and modes of the 8086 microprocessor. (L2)
2. Write and debug basic assembly programs using 8086 instructions and directives. (L2)
3. Interface 8086 with memory and I/O devices using appropriate controllers. (L2)
4. Describe the architecture and programming of the 8051 microcontroller. (L2)
5. Develop interfacing applications using 8051 and compare it with other controllers. (L6)

UNIT-I

8086 Architecture: Main features, pin diagram/description, 8086 microprocessor family, internal architecture, bus interfacing unit, execution unit, interrupts and interrupt response, 8086 system timing, minimum mode and maximum mode configuration.

UNIT-II

8086 Programming: Program development steps, instructions, addressing modes, assembler directives, writing simple programs with an assembler, assembly language program development tools.

UNIT-III

8086 Interfacing: Semiconductor memories interfacing (RAM, ROM), Intel 8255 programmable peripheral interface, Interfacing switches and LEDs, Interfacing seven segment displays, software and hardware interrupt applications, Intel 8251 USART architecture and interfacing, Intel 8237a DMA controller, stepper motor, A/D and D/A converters, Need for 8259 programmable interrupt controllers.

UNIT-IV

Microcontroller, Architecture of 8051: Special Function Registers (SFRs), I/O Pins Ports and Circuits, Instruction set, Addressing modes, Assembly language programming.

UNIT-V

Interfacing Microcontroller: Programming 8051 Timers, Serial Port Programming, Interrupts Programming, LCD & Keyboard Interfacing, ADC, DAC & Sensor Interfacing, External Memory Interface, Stepper Motor and Waveform generation, Comparison of Microprocessor, Microcontroller, PIC and ARM processors.

Textbooks:

1. Douglas V Hall, SSSP Rao, "Microprocessors and Interfacing – Programming and Hardware", Tata McGraw Hill Education Private Limited, 3rd Edition, 1994.
2. K M Bhurchandi, A K Ray, "Advanced Microprocessors and Peripherals", McGraw Hill Education, 3rd Edition, 2017.
3. Raj Kamal, "Microcontrollers: Architecture, Programming, Interfacing and System Design", 2nd Edition, Pearson, 2012.

Reference Books:

1. Ramesh S Gaonkar, "Microprocessor Architecture Programming and Applications with the 8085", Penram International Publishing, 6th Edition, 2013.
2. Kenneth J. Ayala, "The 8051 Microcontroller", Cengage Learning, 3rd Edition, 2004.

Course Code	Course Name	Course Structure			
		L	T	P	C
P23XXXXX	AUTOMATA THEORY & COMPILER DESIGN	3	0	0	3

Internal Marks: 30 External Marks:70

Course Objectives: The main objective of the course is to

1. Construct and convert finite automata from regular expressions and apply to lexical analysis.
2. Implement bottom-up parsers and generate intermediate code using syntax-directed translation.
3. Classify languages and perform type checking and conversions.
4. Apply storage allocation strategies and basic code optimization techniques.
5. Generate optimized machine code using DAGs and register allocation.

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

1. Understand and apply finite automata and regular expressions for lexical analysis. (L2)
2. Develop bottom-up parsers and generate intermediate code for program translation. (L6)
3. Analyze language types and perform type checking and type conversions. (L4)
4. Explain runtime storage management and apply basic code optimization techniques. (L2)
5. Generate efficient machine code using code generation algorithms and register allocation. (L5)

UNIT-I

Formal Language and Regular Expressions: Languages, Definition Languages regular expressions, Finite Automata-DFA, NFA. Conversion of regular expressions to NFA, NFA to DFA. Applications of Finite Automata to lexical analysis, lex tools.

Context Free grammar and parsing: Context free grammar, derivation, parse trees, ambiguity LL(K) grammar and LL. (1) parsing.

UNIT-II

Bottom-up parsing handle pruning: LR Grammar Parsing, LALR parsing, parsing ambiguous grammar, YACC programming specification.

Semantics: Syntax directed translation, S-attributed and L-attributed grammar, Intermediate code - abstract syntax tree, translation of simple statements and control flow statements.

UNIT-III

Context Sensitive features: Chomsky hierarchy of languages and recognizers. Type checking, type conversions, equivalence of type expressions, overloading of functions and operations.

UNIT-IV

Run time storage: Storage organization, storage allocation strategies scope access to now local names. parameters, language facilities for dynamic storage allocation.
Code optimization: Principal sources of optimization, optimization of basic blocks, peephole optimization, flow graphs, Data flow analysis of flow graphs.

UNIT-V

Code generation: Machine dependent code generation, object code forms, generic code generation algorithm, Register allocation and assignment. Using DAG representation of Block.

Text Books:

1. Sipser Michael, "Introduction to the Theory of Computation", Thomson Press (India) Ltd, 3rd Edition, 2014.
2. Aho, Ullman, Ravisethi, "Compilers Principles, Techniques and Tools", Pearson Education. 2nd Edition, 2023.

Reference Books:

1. Andrew W. Appel, "Modern Compiler Implementation In C", Cambridge University Press, Revised ed., 2004.
2. Kenneth C. Louden, "Compiler Construction: Principles and Practice", Course Technology INC, 2006.

Course Code	Course Name	Course Structure			
		L	T	P	C
P23EOE01	ELECTRONIC DEVICES AND CIRCUITS (OPEN ELECTIVE–1)	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Outcomes:

CO1: Apply basic concepts of semiconductor physics to explain device behaviour.

CO2: Understand the formation of p-n junctions and their operation as diodes in various modes.

CO3: Describe the construction and operation of rectifiers with and without filters, including relevant expressions and comparisons.

CO4: Understand the construction, working principles, and V-I characteristics of BJTs and FETs in different configurations.

CO5: Analyze the necessity of transistor biasing, different biasing techniques for BJTs and FETs, and stabilization concepts with relevant expressions.

UNIT–I:

Review of Semiconductor Physics: Hall effect, continuity equation, law of junction, Fermi Dirac function, Fermi level in intrinsic and extrinsic Semiconductors.

Junction Diode Characteristics: Energy band diagram of PN junction Diode, Open circuited p-n junction, Biased p-n junction, p-n junction diode, current components in PN junction Diode, diode equation, V-I Characteristics, temperature dependence on V-I characteristics, Diode resistance, Diode capacitance.

UNIT–II:

Special Semiconductor Devices: Zener Diode, Breakdown mechanisms, Zener diode applications, LED, Varactor Diode, Photodiode, Tunnel Diode, UJT, PNPN Diode, SCR. Construction, operation and V-I characteristics.

Rectifiers and Filters: Basic Rectifier setup, half wave rectifier, full wave rectifier, bridge rectifier, derivations of characteristics of rectifiers, rectifier circuits-operation, input and output waveforms; Filters, Inductor filter (Series inductor), Capacitor filter (Shunt inductor), -Filter; comparison of various filter circuits in terms of ripple factors.

UNIT–III:

Transistor Characteristics:

BJT: Junction transistor, transistor current components, transistor equation, transistor configurations, transistor as an amplifier, characteristics of transistor in Common Base, Common Emitter and Common Collector configurations, Ebers-Moll model of a transistor, punch through/reach through, Photo transistor, typical transistor junction voltage values.

FET: FET types, construction, operation, characteristics μ, g_m, r_d parameters, MOSFET-types, construction, operation, characteristics, comparison between JFET and MOSFET.

UNIT–IV:

Transistor Biasing and Thermal Stabilization: Need for biasing, operating point, load line analysis; BJT biasing methods; basic stability concepts: fixed bias, collector to base bias, self-bias; Stabilization against variations in V_{BE}, I_C, and β; Stability factors (S, S', S''); Bias compensation; Thermal runaway; Thermal stability biasing methods and stabilization.

UNIT–V:

Small Signal Low Frequency Transistor Amplifier Models:

BJT: Two port network, Transistor hybrid model, determination of h-parameters, conversion of h-parameters, generalized analysis of transistor amplifier model using h-parameters, Analysis of CB, CE and CC amplifiers using exact and approximate analysis, Comparison of transistor amplifiers.

FET: Generalized analysis of small signal model, Analysis of CG, CS and CD amplifiers, comparison of FET amplifiers.

Text Books:

1. J. Millman, C. Halkias, "Electronic Devices and Circuits", Tata Mc-Graw Hill, 2nd Edition, 2007.
2. David A. Bell, "Electronic Devices and Circuits", Oxford University Press.
3. Robert L. Boylestad and Louis Nashelsky, "Electronics Devices and Circuit Theory", Pearson/Prentice Hall, 10th edition, 2009.

Reference Books:

1. J. Millman, C. Halkias, "Integrated Electronics", Tata Mc-Graw Hill, 2nd Edition, 2009.
2. K. Lal Kishore, "Electronic Devices and Circuits", BS Publications, 4th Edition, 2016.

Web References:

- Electronic Devices and Circuits - Amrita Vishwa Vidyapeetham
- `https://www.udemy.com/course/electronics-devices-and-circuits-foundation-of-electronics/`

Course Code	Course Name	Course Structure			
		L	T	P	C
P23EOE05	INTRODUCTION TO SIMULATION SOFTWARE (OPEN ELECTIVE-I)	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Outcomes:

CO1: Understand the MATLAB environment, interface, and basic syntax for effective usage.

CO2: Apply MATLAB functions to perform numerical computations and data analysis.

CO3: Implement basic programming constructs such as loops and conditional statements in MATLAB.

CO4: Solve mathematical and engineering problems using built-in and user-defined MATLAB functions.

CO5: Demonstrate professionalism in presenting MATLAB-based solutions and industrial applications.

Unit-I:

MATLAB Basics: Historical Background, Scope of MATLAB, Importance of MATLAB for Engineers, Features, Applications, MATLAB Windows (Editor, Work Space, Command History, Command Window). Arithmetic Operations with Scalars, Display Formats, Clearing Operations, Commands, Data types, Operators.

Unit-II:

Data And Data Flow In MATLAB: Vectors, Matrix Operations & Operators, Reshaping Matrices, Arrays, Colon Notations, Numbers, Strings, Functions, File Input-Output, Importing and Exporting of data, Worked out Examples.

Unit-III:

MATLAB Programming: Relational and Logical Operators, Conditional Statements, Switch-Case Statements, Loops, Nested Statements, Script Files - Creating, Saving and Running script file, Current folder, Examples of MATLAB Applications.

Unit-IV:

MATLAB Plotting and Visualization: Plotting, Creating Plot & Editing Plot - 2D plots (plot, xlabel, ylabel, title, legend), Multiple plots in one figure, Subplots, 3D plots (mesh, surf, plot3), Customizing plots, MATLAB - Algebra, Calculus, Differential, Integration, Polynomials, solving a system of linear equations.

Unit-V:

GUI and Simulink: Creating and Displaying a Graphical User Interface, Graphical User Interface Components, Additional Containers: Panels and Button Groups, Dialog Boxes, Menus, Tips for Creating Efficient GUIs. Introduction to Simulink, Importance, Model Based Design, Tools, Mathematical Modeling, Converting Mathematical Model into Simulink Model, Importing Exporting Data, Solver Configuration, Masking Block/Model.

Text Books:

1. Rudra Pratap, *Getting Started With MATLAB: A Quick Introduction For Scientists And Engineers*, Oxford University Press.
2. Y. Kirani Singh, B.B. Chaudhuri, *MATLAB Programming*, PHI Publication.
3. Amos Gilat, *MATLAB: An Introduction with Applications*.

Reference Books:

1. Stephen J. Chapman, *MATLAB® Programming For Engineers*, Fourth edition.
2. Won Y. Yang, Wenwu Cao, Tae-Sang Chung, John Morris, *Applied Numerical Methods Using MATLAB*, 1st Edition.
3. David Houcque, *Introduction To MATLAB For Engineering Students*, Northwestern University.

Web References:

- `https://www.mathworks.com/help/matlab/`
- `https://matlabacademy.mathworks.com/details/matlab-onramp/getting-started`
- `https://www.coursera.org/learn/matlab`

Course Code	Course Name	Course Structure			
		L	T	P	C
P23ADO04	INTRODUCTION TO DATA SCIENCE (OPEN ELECTIVE - I)	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Prerequisite: Basic knowledge of programming (Python), mathematics, and statistics.

Course Objectives:

1. To introduce the fundamental concepts, workflow, and scope of Data Science.
2. To understand data collection, preprocessing, and visualization techniques.
3. To apply statistical methods for data analysis.
4. To use Python libraries for data science applications.
5. To solve basic real-world problems using data science methods.

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

1. Understand the data science life cycle and key roles.
2. Explore and preprocess real-world datasets.
3. Apply descriptive statistics and inferential techniques.
4. Visualize data using standard Python tools.
5. Implement data science solutions using Python.

UNIT–I:

Introduction to Data Science: What is Data Science? History and Evolution, Data Science vs Data Analytics vs Business Intelligence, Data Science Life Cycle: Business Understanding, Data Collection, Data Cleaning, Modeling, Evaluation, Deployment, Roles in Data Science: Data Scientist, Data Analyst, Data Engineer.

UNIT–II:

Data Collection and Preprocessing: Types of Data: Structured, Unstructured, Semi-Structured; Data Acquisition Techniques – APIs, Web Scraping, Databases; Data Cleaning: Handling Missing Data, Duplicates, Outliers, Normalization and Standardization, Encoding Categorical Variables.

UNIT–III:

Exploratory Data Analysis (EDA): Introduction to EDA, Summary Statistics, Distributions, Correlation; Data Visualization: Histograms, Boxplots, Scatterplots, Heatmaps, Pair Plots; EDA Tools: Pandas Profiling, Seaborn, Matplotlib.

UNIT–IV:

Statistical Methods for Data Science: Descriptive Statistics: Mean, Median, Mode, Variance, Standard Deviation; Probability Distributions: Binomial, Normal, Poisson; Inferential Statistics: Sampling, Central Limit Theorem, Confidence Intervals, Hypothesis Testing, p-values, t-test, Chi-square test.

UNIT–V:

Introduction to Data Science with Python: Working with Pandas: DataFrames,

Series, Indexing, GroupBy; NumPy for Numerical Computation; Introduction to Scikit-learn: Data Splitting, Simple Linear Regression, Classification Overview; Case Studies: Real-world mini projects in Healthcare, Retail, or Social Media.

Text Books:

1. Joel Grus, *Data Science from Scratch: First Principles with Python*, 2nd Edition, O'Reilly Media, 2019.
2. Vasant Dhar, *Data Science and Prediction: Data, Algorithms and Models*, Oxford University Press, 2020.

Reference Books:

1. Cathy O'Neil and Rachel Schutt, *Doing Data Science*, O'Reilly Media, 2013.
2. Jake VanderPlas, *Python Data Science Handbook*, O'Reilly Media, 2016.
3. Wes McKinney, *Python for Data Analysis*, O'Reilly Media, 2nd Edition, 2017.
4. Andreas C. Müller and Sarah Guido, *Introduction to Machine Learning with Python*, O'Reilly, 2016.

Web References:

- `https://www.kaggle.com/learn/data-science`
- `https://www.datacamp.com/tracks/data-scientist-with-python`
- `https://towardsdatascience.com/`
- `https://pandas.pydata.org/docs/`
- `https://numpy.org/doc/`
- `https://seaborn.pydata.org/`

Course Code	Course Name	Course Structure			
		L	T	P	C
P23MBO01	ENTREPRENEURSHIP DEVELOPMENT & VENTURE CREATION	2	0	2	2

Internal Marks: 30 External Marks: 70

Course Objectives:

1. To introduce the concept, process and importance of entrepreneurship development.
2. To acquaint students with the knowledge and skills for identifying business opportunities, project appraisal and management.
3. To familiarize learners with institutional support and legal requirements for entrepreneurship.
4. To develop an understanding of the issues and challenges of small-scale industries and entrepreneurial environment in India.

UNIT–I:

Entrepreneur and Entrepreneurship: Definition, evolution and concept of entrepreneur, functions and types of entrepreneurs, characteristics and traits of successful entrepreneurs, entrepreneurship and entrepreneurial culture, women entrepreneur: meaning, growth, problems, remedies. Theories of entrepreneurship, motivation theories (Maslow, McClelland, Alderfer).

UNIT–II:

Entrepreneurial Development: Environmental factors, socio-economic environment in the emergence of entrepreneurship. Entrepreneurial Development Programmes (EDPs): meaning, objectives, need, organizing EDPs, evaluation and problems, methods, achievements, and role of EDPs in India. Institutional support for entrepreneurship (NSIC, SIDBI, SISI, DIC, etc.).

UNIT–III:

Project Appraisal and Management: Identification and search for business ideas, project classification and formulation, project appraisal and report, feasibility and profitability analysis, sources of finance, profitability and risk analysis, risk assessment and techniques.

UNIT–IV:

Legal and Statutory Environment for Small Industry: Legal formalities for setting up small scale industrial undertakings (SSI), government setup in promoting small industries, various incentives and subsidies, financial institutions, registration procedures, social responsibilities of business, steps in starting a small industry, problems and remedies for industrial sickness.

UNIT–V:

Small Scale Industrial Undertakings: Definition and characteristics of SSI, role in economic development, government policy towards SSI, incentives and subsidies, challenges and problems for SSI, steps to overcome sickness, case studies in entrepreneurship.

Text Books:

1. Vasant Desai, "Dynamics of Entrepreneurial Development and Management", Himalaya Publishing House.
2. S.S. Khanka, "Entrepreneurial Development", S. Chand & Company Ltd.

Reference Books:

1. C.B. Gupta and N.P. Srinivasan, "Entrepreneurship Development in India", Sultan Chand & Sons.
2. P. Saravanavel, "Entrepreneurship Development", Ess Pee Kay Publishing House.
3. E. Gordon, K. Natarajan, "Entrepreneurship Development", Himalaya Publishing House.

Web Resources:

1. https://www.niesbud.nic.in/
2. https://msme.gov.in/
3. https://en.wikipedia.org/wiki/Entrepreneurship

Course Code	Course Name	Course Structure			
		L	T	P	C
P23ITL02	ADVANCED JAVA LAB (PROFESSIONAL COURSE)	0	0	3	1.5

Internal Marks: 30 External Marks: 70

Course Objectives: The main objectives of the course are

- To make use of Servlet and JSP API in the process of enterprise application deployment.
- Implement components such as JSTL.
- Distinguish Application Server, Web Container, JDBC.
- Design and Development of web applications having collaboration of Servlets, JSPs, Spring.

Course Outcomes:

- Build JDBC-based database apps. (L6)
- Design J2EE web applications. (L6)
- Create dynamic web apps using Servlets. (L6)
- Develop web pages using JSP. (L6)
- Build web apps with Spring MVC. (L6)

List of Experiments:

1. Write a JDBC application which will interact with Database and perform the following task.
   (a) Create Student Table with RollNo, Name, and Address field and insert few records.
   (b) Using Statement Object display the content of Record.
   (c) Using Statement Object Insert Two Record.
   (d) Using Statement Object Update One Record.
   (e) Using Statement Object Delete One Record.
   (f) Using Statement Object display the content of Record.
2. Write a JDBC application which will interact with Database and perform the following task.
   (a) Create Student Table with RollNo, Name, and Address field and insert few records.
   (b) Using Prepared Statement Object display the content of Record.

(c) Using Prepared Statement Object Insert Two Record.

(d) Using Prepared Statement Object Update One Record.

(e) Using Prepared Statement Object Delete One Record.

(f) Using Prepared Statement Object display the content of Record

3. Write a JDBC application which will interact with Database and perform the following task.

   (a) Create a store procedure which will insert one record into employee table.

   (b) Create a store procedure which will retrieve salary for given employee id.

   (c) Write a java application which will call the above procedure and display appropriate information on screen

4. Design a JDBC application which will demonstrate Scrollable Result Set functionality.
5. Design a JDBC application which will demonstrate Updatable Result Set functionality.
6. Write down the Program for testing the Servlet and study deployment descriptor.
7. Write down the program for testing the include action for servlet collaboration.
8. Create login form and perform state management using Cookies, Http Session and URL Rewriting.
9. Write down the Program which displays the simple JSP file.
10. Write down the program in which input the two numbers in an html file and then display the addition in JSP file.
11. Perform Database Access through JSP.
12. Write down a program which demonstrates the core tag of JSTL.
13. Write down a program which demonstrates the Format tag of JSTL.
14. Write down a program which demonstrates the Function tag of JSTL.
15. Write down a program which demonstrates the SQL tag of JSTL.
16. Study and Implement MVC using Spring Framework
17. Using Spring Template manage Database and Transaction.

Text Books:

1. Dreamtech Software Team, “Black Book Java server programming J2EE”, Dream Tech Publishers, 4th Edition, 2008.
2. James Keogh, “Complete Reference J2EE”, McGraw Hill publication, 1st Edition, 2017.
3. Subrahmanyam Allamaraju, Cedric Buest, “Professional Java Server Programming”, Wiley Publication, 3rd Edition, 2015.
4. Craig Walls, “Spring in Action”, Manning Publication, 6th Edition, 2022.

Reference Books:

1. Cay Horstmann, Gary Cornell, “Core Java, Volume II: Advanced Features”, Pearson publication, 8th Edition, 2008.
2. Maydene Fisher, Jon Ellis, Jonathan Bruce, “JDBC™ API Tutorial and Reference”, Addison Wesley, 3rd Edition, 2003.
3. Giulio Zambon, “Beginning JSP, JSF and Tomcat”, Apress, 1st Edition, 2007.

Course Code	Course Name	Course Structure			
		L	T	P	C
P23CSL07	COMPUTER NETWORKS LAB(PROFESSIONAL CORE)	0	0	3	1.5

Internal Marks: 30 External Marks: 70

Course Objectives: Learn basic concepts of computer networking and acquire practical notions of protocols with the emphasis on TCP/IP. The lab provides a practical approach to Ethernet/Internet networking: networks are assembled, and experiments are made to understand the layered architecture and how some important protocols work.

List of Experiments:

1. Study of Network devices in detail and connect the computers in Local Area Network.
2. Write a Program to implement the data link layer framing methods such as i) Character stuffing ii) Bit stuffing.
3. Write a Program to implement data link layer framing method checksum.
4. Write a program for Hamming Code generation for error detection and correction.
5. Write a Program to implement on a data set of characters the three CRC polynomials – CRC-12, CRC-16 and CRC-CCIP.
6. Write a Program to implement Sliding window protocol for Go back N.
7. Write a Program to implement Sliding window protocol for Selective repeat.
8. Write a Program to implement Stop and Wait Protocol.
9. Write a program for congestion control using leaky bucket algorithm.
10. Write a Program to implement Dijkstra's algorithm to compute the Shortest path through a graph.
11. Write a Program to implement Distance vector routing algorithm by obtaining routing table at each node (Take an example subnet graph with weights indicating delay between nodes).
12. Write a Program to implement Broadcast tree by taking subnet of hosts.
13. Wireshark
    i. Packet Capture Using Wireshark
    ii. Starting Wireshark
    iii. Viewing Captured Traffic
    iv. Analysis and Statistics & Filters.
14. How to run Nmap scan
15. Operating System Detection using Nmap
16. Do the following using NS2 Simulator
    i. NS2 Simulator-Introduction
    ii. Simulate to Find the Number of Packets Dropped

iii. Simulate to Find the Number of Packets Dropped by TCP/UDP
iv. Simulate to Find the Number of Packets Dropped due to Congestion
v. Simulate to Compare Data Rate & Throughput.

Course Code	Course Name	Course Structure			
		L	T	P	C
P23ITS03	FULL STACK DEVELOPMENT - 1(SKILL ENHANCEMENT COURSE)	0	1	2	2

Internal Marks: 30 External Marks: 70

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.

List of Experiments:

1. **Lists, Links and Images**
   (a) Write a HTML program, to explain the working of lists.
   Note: It should have an ordered list, unordered list, nested lists and ordered list in an unordered list and definition lists.
   (b) Write a HTML program, to explain the working of hyperlinks using `<a>` tag and href, target Attributes.
   (c) Create a HTML document that has your image and your friend's image with a specific height and width. Also when clicked on the images it should navigate to their respective profiles.
   (d) Write a HTML program, in such a way that, rather than placing large images on a page, the preferred technique is to use thumbnails by setting the height and width parameters to something like to 100*100 pixels. Each thumbnail image is also a link to a full sized version of the image. Create an image gallery using this technique.
2. **HTML Tables, Forms and Frames**
   (a) Write a HTML program, to explain the working of tables. (use tags: `<table>`, `<tr>`, `<th>`, `<td>` and attributes: border, rowspan, colspan)
   (b) Write a HTML program, to explain the working of tables by preparing a timetable. (*Note: Use `<caption>` tag to set the caption to the table & also use cell spacing, cell padding, border, rowspan, colspan etc.*)
   (c) Write a HTML program, to explain the working of forms by designing Registration form. (*Note: Include text field, password field, number field, date of birth field, checkboxes, radio buttons, list boxes using `<select>`&`<option>` tags, `<text area>` and two buttons ie: submit and reset. Use tables to provide a better view*).

(d) Write a HTML program, to explain the working of frames, such that page is to be divided into 3 parts on either direction. (*Note: first frame → image, second frame → paragraph, third frame → hyperlink. And also make sure of using "no frame" attribute such that frames to be fixed.*)

3. **HTML 5 and Cascading Style Sheets, Types of CSS**

(a) Write a HTML program, that makes use of `<article>`, `<aside>`, `<figure>`, `<figcaption>`, `<footer>`, `<header>`, `<main>`, `<nav>`, `<section>`, `<div>`, `<span>` tags.

(b) Write a HTML program, to embed audio and video into HTML web page.

(c) Write a program to apply different types (or levels of styles or style specification formats) - inline, internal, external styles to HTML elements. (identify selector, property and value).

4. **Selector forms**

(a) Write a program to apply different types of selector forms
i. Simple selector (element, id, class, group, universal)
ii. Combinator selector (descendant, child, adjacent sibling, general sibling)
iii. Pseudo-class selector
iv. Pseudo-element selector
v. Attribute selector

5. **CSS with Color, Background, Font, Text and CSS Box Model**

(a) Write a program to demonstrate the various ways you can reference a color in CSS.

(b) Write a CSS rule that places a background image halfway down the page, tilting it horizontally. The image should remain in place when the user scrolls up or down.

(c) Write a program using the following terms related to CSS font and text:
i. font-size ii. font-weight iii. font-style
iv. text-decoration v. text-transformation vi. text-alignment

(d) Write a program, to explain the importance of CSS Box model using
i. Content ii. Border iii. Margin iv. padding

6. **Applying JavaScript - internal and external, I/O, Type Conversion**

(a) Write a program to embed internal and external JavaScript in a web page.

(b) Write a program to explain the different ways for displaying output.

(c) Write a program to explain the different ways for taking input.

(d) Create a webpage which uses prompt dialogue box to ask a voter for his name and age. Display the information in table format along with either the voter can vote or not.

7. **JavaScript Pre-defined and User-defined Objects**

   (a) Write a program using document object properties and methods.

   (b) Write a program using window object properties and methods.

   (c) Write a program using array object properties and methods.

   (d) Write a program using math object properties and methods.

   (e) Write a program using string object properties and methods.

   (f) Write a program using regex object properties and methods.

   (g) Write a program using date object properties and methods.

   (h) Write a program to explain user-defined object by using properties, methods, accessors, constructors and display.

8. **JavaScript Conditional Statements and Loops**

   (a) Write a program which asks the user to enter three integers, obtains the numbers from the user and outputs HTML text that displays the larger number followed by the words "LARGER NUMBER" in an information message dialog. If the numbers are equal, output HTML text as "EQUAL NUMBERS".

   (b) Write a program to display week days using switch case.

   (c) Write a program to print 1 to 10 numbers using for, while and do-while loops.

   (d) Write a program to print data in object using for-in, for-each and for-of loops.

   (e) Develop a program to determine whether a given number is an 'ARMSTRONG NUMBER' or not. [Eg: 153 is an Armstrong number, since sum of the cube of the digits is equal to the number i.e., $1^3 + 5^3 + 3^3 = 153$]

   (f) Write a program to display the denomination of the amount deposited in the bank in terms of 100's, 50's, 20's, 10's, 5's, 2's & 1's. (Eg: If deposited amount is Rs.163, the output should be 1-100's, 1-50's, 1-10's, 1-2's & 1-1's)

9. **Javascript Functions and Events**

   (a) Design a appropriate function should be called to display
   i. Factorial of that number
   ii. Fibonacci series up to that number
   iii. Prime numbers up to that number
   iv. Is it palindrome or not

   (b) Design a HTML having a text box and four buttons named Factorial, Fibonacci, Prime, and Palindrome. When a button is pressed an appropriate function should be called to display:
   i. Factorial of that number
   ii. Fibonacci series up to that number

iii. Prime numbers up to that number
iv. Is it palindrome or not

(c) Write a program to validate the following fields in a registration page
i. Name (start with alphabet and followed by alphanumeric and the length should not be less than 6 characters)
ii. Mobile (only numbers and length 10 digits)
iii. E-mail (should contain format like xxxxxxx@xxxxxx.xxx)

Text Books:

1. Programming the World Wide Web, 7th Edition, Robet W Sebesta, Pearson, 2013.
2. Web Programming with HTML5, CSS and JavaScript, John Dean, Jones & Bartlett Learning, 2019 (Chapters 1-11).
3. Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node, Vasan Subramanian, 2nd edition, APress, O'Reilly.

Web Links:

1. `https://www.w3schools.com/html`
2. `https://www.w3schools.com/css`
3. `https://www.w3schools.com/js/`

Course Code	Course Name	Course Structure			
		L	T	P	C
P23XXXXX	USER INTERFACE DESIGN USING FLUTTER	0	0	2	1

Internal Marks: – External Marks: –

Course Objectives:

- Learns to implement Flutter Widgets and Layouts.
- Understands Responsive UI Design with Navigation in Flutter.
- Knowledge on Widgets and customizing widgets for specific UI elements, themes.
- Understand to include animation apart from fetching data.

List of Experiments:

1. a) Install Flutter and Dart SDK.
   b) Write a simple Dart program to understand the language basics.
2. a) Explore various Flutter widgets (Text, Image, Container, etc.).
   b) Implement different layout structures using Row, Column, and Stack widgets.
3. a) Design a responsive UI that adapts to different screen sizes.
   b) Implement media queries and breakpoints for responsiveness.
4. a) Set up navigation between different screens using Navigator.
   b) Implement navigation with named routes.
5. a) Learn about stateful and stateless widgets.
   b) Implement state management using setState and Provider.
6. a) Create custom widgets for specific UI elements.
   b) Apply styling using themes and custom styles.
7. a) Design a form with various input fields.
   b) Implement form validation and error handling.
8. a) Add animations to UI elements using Flutter's animation framework.
   b) Experiment with different types of animations (fade, slide, etc.).
9. a) Fetch data from a REST API.
   b) Display the fetched data in a meaningful way in the UI.
10. a) Write unit tests for UI components.
    b) Use Flutter's debugging tools to identify and fix issues.

Text Books:

1. Marco L. Napoli, "Beginning Flutter: A Hands-on Guide to App Development", Wrox, 1st Edition, 2019.

2. Rap Payne, “Beginning App Development with Flutter: Create Cross-Platform Mobile Apps”, Apress, 1st Edition, 2019.
3. Richard Rose, “Flutter & Dart Cookbook, Developing Full stack Applications for the Cloud”, O’Reilly, 2023.

Course Code	Course Name	Course Structure			
		L	T	P	C
P21CSTXX	AGILE METHODOLOGIES (PROFESSIONAL ELECTIVE - III)	3	0	0	3

Internal Marks: 30 **External Marks:** 70

Course Objectives:

1. To provide students with a theoretical as well as practical understanding of agile software development practices and how small teams can apply them to create high-quality software.
2. To provide a good understanding of software design and a set of software technologies and APIs.
3. To do a detailed examination and demonstration of Agile development and testing techniques.
4. To understand the benefits and pitfalls of working in an Agile team.
5. To understand Agile development and testing.

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

1. Realize the importance of interacting with business stakeholders in determining the requirements for a software system.
2. Perform iterative software development processes: how to plan them, how to execute them.
3. Point out the impact of social aspects on software development success.
4. Develop techniques and tools for improving team collaboration and software quality.
5. Perform software process improvement as an ongoing task for development teams.
6. Show how agile approaches can be scaled up to the enterprise level.

UNIT I: Agile Methodology

Theories for Agile Management – Agile Software Development – Traditional Model vs. Agile Model – Classification of Agile Methods – Agile Manifesto and Principles – Agile Project Management – Agile Team Interactions – Ethics in Agile Teams – Agility in Design, Testing – Agile Documentation – Agile Drivers, Capabilities and Values.

UNIT II: Agile Processes

Lean Production – SCRUM, Crystal, Feature Driven Development – Adaptive Software Development – Extreme Programming: Method Overview – Lifecycle – Work Products, Roles and Practices.

UNIT III: Agility and Knowledge Management

Agile Information Systems – Agile Decision Making – Earl's Schools of KM – Institutional Knowledge Evolution Cycle – Development, Acquisition, Refinement, Distribution, Deployment, Leveraging – KM in Software Engineering – Managing Software Knowledge – Challenges of Migrating to Agile Methodologies – Agile Knowledge Sharing – Role of Story-Cards – Story-Card Maturity Model (SMM).

UNIT IV: Agility and Requirements Engineering

Impact of Agile Processes in RE – Current Agile Practices – Variance – Overview of RE Using Agile – Managing Unstable Requirements – Requirements Elicitation – Agile Requirements Abstraction Model – Requirements Management in Agile Environment, Agile Requirements Prioritization – Agile Requirements Modeling and Generation – Concurrency in Agile Requirements Generation.

UNIT V: Agility and Quality Assurance

Agile Product Development – Agile Metrics – Feature Driven Development (FDD) – Financial and Production Metrics in FDD – Agile Approach to Quality Assurance – Test Driven Development – Agile Approach in Global Software Development.

Text Books:

1. David J. Anderson and Eli Schragenheim, *Agile Management for Software Engineering: Applying the Theory of Constraints for Business Results*, Prentice Hall, 2003.
2. Hazza and Dubinsky, *Agile Software Engineering*, Series: Undergraduate Topics in Computer Science, Springer, 2009.

Reference Books:

1. Craig Larman, *Agile and Iterative Development: A Manager's Guide*, Addison-Wesley, 2004.
2. Kevin C. Desouza, *Agile Information Systems: Conceptualization, Construction, and Management*, Butterworth-Heinemann, 2007.

Web References:

1. `https://www.scrumalliance.org`
2. `https://www.agilealliance.org`
3. `https://www.tutorialspoint.com/agile/index.htm`

Course Code	Course Name	Course Structure			
		L	T	P	C
P23XXXXX	CLOUD COMPUTING (PROFESSIONAL CORE)	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives:

1. To explain the evolving utility computing model called cloud computing.
2. To introduce the various levels of services offered by cloud.
3. To discuss the fundamentals of cloud enabling technologies such as distributed computing, service-oriented architecture and virtualization.
4. To emphasize the security and other challenges in cloud computing.
5. To introduce the advanced concepts such as containers, serverless computing and cloud-centric Internet of Things.

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

1. Describe cloud computing models, services, and providers. (L2)
2. Explain enabling technologies like distributed computing and virtualization. (L2)
3. Understand virtualization, containers, and orchestration tools. (L2)
4. Analyze key challenges and security issues in cloud computing. (L4)
5. Demonstrate the advanced topics like serverless computing, IoT, and DevOps. (L3)

UNIT-I: (9 Lectures)
Introduction to Cloud Computing Fundamentals: Cloud computing at a glance, defining a cloud, cloud computing reference model, types of services (IaaS, PaaS, SaaS), cloud deployment models (public, private, hybrid), utility computing, cloud computing characteristics and benefits, cloud service providers (Amazon Web Services, Microsoft Azure, Google AppEngine).
UNIT-II: (9 Lectures)
Cloud Enabling Technologies: Ubiquitous Internet, parallel and distributed computing, elements of parallel computing, hardware architectures for parallel computing (SISD, SIMD, MISD, MIMD), elements of distributed computing, Inter-process communication, technologies for distributed computing, remote procedure calls (RPC), service-oriented architecture (SOA), Web services, and virtualization.
UNIT-III: (9 Lectures)
Virtualization and Containers: Characteristics of virtualized environments, taxonomy of virtualization techniques, virtualization and cloud Computing, pros and cons of virtualization, technology examples (XEN, VMware), building blocks of containers, container platforms (LXC, Docker), container orchestration, Docker Swarm and Kubernetes, public cloud VM (e.g. Amazon EC2) and container (e.g. Amazon Elastic Container Service) offerings.

UNIT-IV: (9 Lectures)

Cloud computing challenges: Economics of the cloud, cloud interoperability and standards, scalability and fault tolerance, energy efficiency in clouds, federated clouds, cloud computing security, fundamentals of computer security, cloud security architecture, cloud shared responsibility model, security in cloud deployment models.

UNIT-V: (9 Lectures)

Advanced concepts in cloud computing: Serverless computing, Function-as-a-Service, serverless computing architecture, public cloud (e.g. AWS Lambda) and open-source (e.g. OpenFaaS) serverless platforms, Internet of Things (IoT), applications, cloud-centric IoT and layers, edge and fog computing, DevOps, infrastructure-as-code, quantum cloud computing.

Text Books:

1. Rajkumar Buyya, Christian Vecchiola, ThamaraiSelvi, Shivananda Poojara, Satish N. Srirama, "Mastering Cloud Computing", McGraw Hill, 2nd edition, 2024.
2. Kai Hwang, Geoffery C. Fox, Jack J. Dongarra, "Distributed and Cloud Computing", Elsevier, 2012.

Reference Books:

1. Dan C Marinescu, "Cloud Computing, Theory and Practice", MK Elsevier, 2nd edition, 2018.
2. K. Chandrasekhran, "Essentials of cloud Computing", CRC press, Taylor & Francis Group, 2014.
3. Online documentation and tutorials from cloud service providers (e.g., AWS, Azure, GCP)

Course Code	Course Name	Course Structure			
		L	T	P	C
P23XXXXX	CRYPTOGRAPHY & NETWORK SECURITY(PROFESSIONAL CORE)	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives: The main objective of the course is to

1. Understand basic cryptography principles and math foundations.
2. Explain and analyze symmetric key encryption methods.
3. Describe asymmetric encryption and its mathematical basis.
4. Apply data integrity, digital signatures, and key management concepts.
5. Summarize network and system security protocols and tools.

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

1. Describe the fundamental cryptographic concepts and mathematical tools to analyze security goals and attacks. (L2)
2. Demonstrate knowledge of symmetric encryption algorithms and evaluate their security. (L3)
3. Explain the principles and implementations of asymmetric cryptographic systems. (L2)
4. Utilize techniques for ensuring data integrity, authentication, and secure key management. (L3)
5. Analyze network and system security protocols and their roles in protecting communications and systems. (L4)

UNIT-I

Basic Principles: Security Goals, Cryptographic Attacks, Services and Mechanisms, Mathematics of Cryptography- integer arithmetic, modular arithmetic, matrices, linear congruence.

UNIT-II

Symmetric Encryption: Mathematics of Symmetric Key Cryptography-algebraic structures, GF(2n) Fields, Introduction to Modern Symmetric Key Ciphers-modern block ciphers, modern stream ciphers, Data Encryption Standard- DES structure, DES analysis, Security of DES, Multiple DES, Advanced Encryption Standard-transformations, key expansions, AES ciphers, Analysis of AES.

UNIT-III

Asymmetric Encryption: Mathematics of Asymmetric Key Cryptography-primes, primality testing, factorization, CRT, Asymmetric Key Cryptography- RSA crypto system, Rabin cryptosystem, Elgamal Crypto system, ECC.

UNIT-IV

Data Integrity, Digital Signature Schemes & Key Management: Message Integrity and Message Authentication-message integrity, Random Oracle model, Message authentication, Cryptographic Hash Functions-whirlpool, SHA-512, Digital

Signature- process, services, attacks, schemes, applications, Key Management-symmetric key distribution, Kerberos.

UNIT-V

Network Security-I: Security at application layer: PGP and S/MIME, Security at the Transport Layer: SSL and TLS, Network Security-II : Security at the Network Layer: IPSec-two modes, two security protocols, security association, IKE, ISAKMP, System Security-users, trust, trusted systems, buffer overflow, malicious software, worms, viruses, IDS, Firewalls.

Text Books:

1. Behrouz A. Forouzan, Debdeep Mukhopadhyay, “Cryptography and Network Security”, McGraw Hill, 3rd Edition, 2015.
2. William Stallings, “Cryptography and Network Security”, Pearson, 4th Edition, 2006.
3. Keith M. Martin, “Everyday Cryptography”, Oxford, 1st Edition, 2016.

Reference Books:

1. Bernard Menezes, “Network Security and Cryptography”, Cengage Learning, 1st Edition, 2018.
2. William Stallings, “Cryptography and Network Security Principles and Practice”, Pearson, 7th Edition, 2020.

Course Code	Course Name	Course Structure			
		L	T	P	C
P23XXXXX	MACHINE LEARNING(PROFESSIONAL ELECTIVE-II)	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives:

1. Understand basic ML concepts, learning types, and the machine learning workflow.
2. Apply distance-based models like KNN for classification and regression.
3. Learn decision tree and Bayesian methods for classification and regression.
4. Understand linear models, SVMs, and neural networks for supervised learning.
5. Explore clustering techniques for unsupervised pattern discovery.

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

CO1: Enumerate the Fundamentals of Machine Learning. (L1)
CO2: Build Nearest neighbor-based models. (L6)
CO3: Apply Models based on decision trees and Bayes rule. (L3)
CO4: Make use of Linear discriminants for machine Learning. (L3)
CO5: Choose appropriate clustering technique. (L2)

UNIT–I:

Introduction to Machine Learning: Evolution of Machine Learning, Paradigms for ML, Learning by Rote, Learning by Induction, Reinforcement Learning, Types of Data, Matching, Stages in Machine Learning, Data Acquisition, Feature Engineering, Data Representation, Model Selection, Model Learning, Model Evaluation, Model Prediction, Search and Learning, Data Sets.

UNIT–II:

Nearest Neighbor-Based Models: Introduction to Proximity Measures, Distance Measures, Non-Metric Similarity Functions, Proximity Between Binary Patterns, Different Classification Algorithms Based on the Distance Measures, K-Nearest Neighbor Classifier, Radius Distance Nearest Neighbor Algorithm, KNN Regression, Performance of Classifiers, Performance of Regression Algorithms.

UNIT–III:

Models Based on Decision Trees: Decision Trees for Classification, Impurity Measures, Properties, Regression Based on Decision Trees, Bias–Variance Trade-off, Random Forests for Classification and Regression,The Bayes Classifier: Introduction to the Bayes Classifier, Bayes' Rule and Inference, The Bayes Classifier and its Optimality, Multi-Class Classification, Class Conditional Independence and Naive Bayes Classifier (NBC).

UNIT–IV:

Linear Discriminants for Machine Learning: Introduction to Linear Discriminants, Linear Discriminants for Classification, Perceptron Classifier, Perceptron Learning Algorithm, Support Vector Machines, Linearly Non-Separable Case, Non-linear SVM, Kernel Trick, Logistic Regression, Linear Regression, Multi-Layer Perceptrons (MLPs), Backpropagation for Training an MLP.

UNIT–V:

Clustering: Introduction to Clustering, Partitioning of Data, Matrix Factorization, Clustering of Patterns, Divisive Clustering, Agglomerative Clustering, Partitional Clustering, K-Means Clustering, Soft Partitioning, Soft Clustering, Fuzzy C-Means Clustering, Rough Clustering, Rough K-Means Clustering Algorithm, Expectation Maximization-Based Clustering, Spectral Clustering.

Text Books:

1. M N Murthy, VS Ananthanarayana, “Machine Learning Theory and Practice”, Universities Press (India), 2024.

Reference Books:

1. Tom M. Mitchell, “Machine Learning”, McGraw-Hill Publication, 2017.
2. Peter Harrington, “Machine Learning in Action”, DreamTech, 2012.
3. Pang-Ning Tan, Michel Stenbach, Vipin Kumar, “Introduction to Data Mining”, 7th Edition, 2019.

Web Resources:

- `https://scikit-learn.org/`
- `https://www.kaggle.com/`
- `https://www.tensorflow.org/tutorials`

Course Code	Course Name	Course Structure			
		L	T	P	C
P23XXXXX	SOFTWARE TESTING METHODOLOGIES (PROFESSIONAL ELECTIVE - II)	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives: The main objective of the course is to

1. Apply path testing techniques to identify bugs using flow graphs.
2. Use transaction, data flow, and domain testing to detect software issues.
3. Use path expressions and logic-based testing for test case generation.
4. Perform state and transition testing using state graphs.
5. Analyze graph matrices and use tools like JMeter or Selenium for testing.

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

1. Explain the purpose of software testing and apply path testing using flow graphs. (L2)
2. Perform transaction, data flow, and domain testing to evaluate program behavior. (L2)
3. Generate test cases using path expressions and logic-based testing techniques. (L3)
4. Apply state graph and transition testing to validate system states. (L3)
5. Use graph matrices and testing tools like JMeter or Selenium for automated testing. (L3)

UNIT-I

Introduction: Purpose of testing, Dichotomies, model for testing, consequences of bugs, taxonomy of bugs

Flow graphs and Path testing: Basics concepts of path testing, predicates, path predicates and achievable paths, path sensitizing, path instrumentation, application of path testing.

UNIT-II

Transaction Flow Testing: Transaction flows, transaction flow testing techniques.

Data Flow testing: Basics of data flow testing, strategies in data flow testing, application of data flow testing.

Domain Testing: Domains and paths, Nice & ugly domains, domain testing, domains and interfaces testing, domain and interface testing, domains and testability.

UNIT-III

Paths, Path products and Regular expressions: Path products & path expression, reduction procedure, applications, regular expressions & flow anomaly detection.

Logic Based Testing: Overview, decision tables, path expressions, kv charts, specifications.

UNIT-IV

State, State Graphs and Transition testing: State graphs, good & bad state graphs, state testing, Testability tips.

UNIT-V

Graph Matrices and Application: Motivational overview, matrix of graph, relations, power of a matrix, node reduction algorithm, building tools. (Student should be given an exposure to a tool like Jmeter/selenium/soapUI/Catalon).

Text Books:

1. Baris Beizer, "Software Testing techniques", Dreamtech, 2nd Edition, 2002.
2. Dr. K. V. K. K. Prasad, "Software Testing Tools", Dreamtech, 2009.

Reference Books:

1. Brian Marick, "The craft of software testing", Pearson Education, 2007.
2. Software Testing Techniques – SPD(Oreille)
3. Edward Kit, "Software Testing in the Real World", Pearson, 1997.
4. Perry, "Effective methods of Software Testing", John Wiley, 2002.
5. Meyers, "Art of Software Testing", John Wiley, 3rd Edition, 2015.

Course Code	Course Name	Course Structure			
		L	T	P	C
P23XXXXX	CYBER SECURITY (PROFESSIONAL ELECTIVE-II)	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives:

1. Identify security risks and take preventive steps.
2. Understand the forensics fundamentals.
3. Understand the evidence capturing process.
4. Understand the preservation of digital evidence.

Course Outcomes:

1. Understand types of cybercrimes and threats to information security. (L2)
2. Identify and explain common cyberattack tools and techniques. (L1)
3. Apply basic methods for cybercrime investigation and evidence handling. (L3)
4. Use forensics tools for analyzing systems and retrieving digital evidence. (L3)
5. Understand cyber laws, legal frameworks, and IT Act provisions in India. (L2)

UNIT-I

Introduction to Cybercrime: Introduction, Cybercrime: Definition and Origins of the Word, Cybercrime and Information Security, Cybercriminals, Classifications of Cybercrime, Cyberstalking, Cybercafe and Cybercrimes, Botnets. Attack Vector, Proliferation of Mobile and Wireless Devices, Security Challenges Posed by Mobile Devices, Attacks on Mobile / Cell Phones, Network and Computer Attacks.

UNIT-II

Tools and Methods: Proxy Servers and Anonymizers, Phishing, Password Cracking, Keyloggers and Spywares, Virus and Worms, Trojan Horses and Back doors, Steganography, Sniffers, Spoofing, Session Hijacking Buffer overflow, DoS and DDoS Attacks, SQL Injection, Buffer Overflow, Attacks on Wireless Networks, Identity Theft (ID Theft), Foot Printing and Social Engineering, Port Scanning, Enumeration.

UNIT-III

Cybercrime Investigation: Introduction, Investigation Tools, eDiscovery, Digital Evidence Collection, Evidence Preservation, E-Mail Investigation, E-Mail Tracking, IP Tracking, E-Mail Recovery, Hands on Case Studies. Encryption and Decryption Methods, Search and Seizure of Computers, Recovering Deleted Evidences, Password Cracking.

UNIT-IV

Computer Forensics and Investigations: Understanding Computer Forensics, Preparing for Computer Investigations. Current Computer Forensics Tools: Evaluating Computer Forensics Tools, Computer Forensics Software Tools, Computer

Forensics Hardware Tools, Validating and Testing Forensics Software, Face, Iris and Fingerprint Recognition, Audio Video Analysis, Windows System Forensics, Linux System Forensics, Graphics and Network Forensics, E-mail Investigations, Cell Phone and Mobile Device Forensics.

UNIT-V

Cyber Crime Legal Perspectives: Introduction, Cybercrime and the Legal Landscape around the World, The Indian IT Act, Challenges to Indian Law and Cybercrime Scenario in India, Consequences of Not Addressing the Weakness in Information Technology Act, Digital Signatures and the Indian IT Act, Amendments to the Indian IT Act, Cybercrime and Punishment, Cyber law, Technology and Students: Indian Scenario.

Text Books:

1. Sunit Belapure, Nina Godbole, "Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives", WILEY, 2011.
2. Nelson Phillips and Enfinger Steuart, "Computer Forensics and Investigations", Cengage Learning, New Delhi, 2009.

Reference Books:

1. Michael T. Simpson, Kent Backman and James E. Corley, "Hands on Ethical Hacking and Network Defence", Cengage, 2019.
2. John R. Vacca, "Computer Forensics, Computer Crime Investigation", Firewall Media, New Delhi.
3. Alfred Basta, Nadine Basta, Mary Brown and Ravinder Kumar, "Cyber Security and Cyber Laws", Cengage, 2018.

E-Resources:

1. CERT-In Guidelines: `http://www.cert-in.org.in/`
2. `https://www.coursera.org/learn/introduction-cybersecurity-cyber-attacks`
3. `https://computersecurity.stanford.edu/free-online-videos`
4. Nickolai Zeldovich. 6.858 Computer Systems Security. MIT OpenCourseWare: `https://ocw.mit.edu`

Course Code	Course Name	Course Structure			
		L	T	P	C
P23XXXXX	DEVOPS (PROFESSIONAL ELECTIVE - II)	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives:

1. Understand DevOps principles and lifecycle for agile software delivery.
2. Manage source code and ensure code quality using Git and testing tools.
3. Automate builds and continuous integration using Jenkins.
4. Implement continuous delivery and containerization with Docker and Selenium.
5. Automate configuration management and orchestration using Ansible and Kubernetes.

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

1. Explain DevOps concepts and workflows. (L2)
2. Demonstrate proficiency in Git and perform Unit testing with code quality tools. (L3)
3. Configure Jenkins pipelines for continuous integration. (L2)
4. Deploy applications using Docker and apply automated testing. (L3)
5. Use Ansible and Kubernetes for configuration management and container orchestration. (L3)

UNIT-I

Introduction to DevOps: Introduction to SDLC, Agile Model. Introduction to Devops. DevOps Features, DevOps Architecture, DevOps Lifecycle, Understanding Workflow and principles, Introduction to DevOps tools, Build Automation, Delivery Automation, Understanding Code Quality, Automation of CI/ CD. Release management, Scrum, Kanban, delivery pipeline, bottlenecks, examples.

UNIT-II

Source Code Management (GIT): The need for source code control, The history of source code management, Roles and code, source code management system and migrations. What is Version Control and GIT, GIT Installation, GIT features, GIT workflow, working with remote repository, GIT commands, GIT branching, GIT staging and collaboration.

UNIT TESTING - CODE COVERAGE: Junit, nUnit & Code Coverage with Sonar Qube, SonarQube - Code Quality Analysis.

UNIT-III

Build Automation - Continuous Integration (CI): Build Automation, What is CI Why CI is Required, CI tools, Introduction to Jenkins (With Architecture), Jenkins workflow, Jenkins master slave architecture, Jenkins Pipelines, PIPELINE BASICS

- Jenkins Master, Node, Agent, and Executor Freestyle Projects & Pipelines, Jenkins for Continuous Integration, Create and Manage Builds, User Management in Jenkins Schedule Builds, Launch Builds on Slave Nodes.

UNIT-IV

Continuous Delivery (CD): Importance of Continuous Delivery, CONTINUOUS DEPLOYMENT CD Flow.

Containerization with Docker: Introduction to Docker, Docker installation, Docker commands, Images & Containers, DockerFile, Running containers, Working with containers and publish to Docker Hub.

Testing Tools: Introduction to Selenium and its features, JavaScript testing.

UNIT-V

Configuration Management - ANSIBLE: Introduction to Ansible, Ansible tasks, Roles, Jinja templating, Vaults, Deployments using Ansible.

CONTAINERIZATION USING KUBERNETES (OPENSHIFT): Introduction to Kubernetes Namespace & Resources, CI/CD - On OCP, BC, DC &ConfigMaps, Deploying Apps on Openshift Container Pods. Introduction to Puppet master and Chef.

Text Books:

1. Joyner, Joseph., "DevOps for Beginners: DevOps Software Development Method Guide for Software Developers and IT Professionals", Mihails Konoplows, 1st Edition, 2015.
2. Alisson Machado de Menezes, "Hands-on DevOps with Linux", BPB Publications, 1st Edition, India, 2021.

Reference Books:

1. Len Bass, Ingo Weber, Liming Zhu, "DevOps: A Software Architect's Perspective", Addison Wesley.
2. Gene Kim, Jez Humble, Patrick Debois, John Willis, "The DevOps Handbook", IT Revolution Press, 1st Edition, 2016.
3. Verona, Joakim, "Practical DevOps", Packt Publishing, 1st Edition, 2016.
4. Joakim Verona, "Practical DevOps, Ingram short title", 2nd edition (2018). ISBN10: 1788392574.
5. Deepak Gaikwad, Viral Thakkar, "DevOps Tools from Practitioner's Viewpoint", Wiley publications. ISBN: 9788126579952.

Course Code	Course Name	Course Structure			
		L	T	P	C
P23XXXXX	GENERATIVE AI (PROFESSIONAL ELECTIVE-II)	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives:

1. Understand and differentiate key generative models and their ethical implications.
2. Design effective prompts and explain core text generation architectures.
3. Compare image generation models and implement a basic example.
4. Apply GAN variants and RNNs for creative content generation.
5. Fine-tune and deploy generative models using open-source tools.

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

1. Describe key generative models and their ethical considerations. (L2)
2. Apply prompt engineering and text generation techniques. (L3)
3. Implement and analyze image generation models. (L3)
4. Create generative art, music, or game strategies using AI. (L5)
5. Deploy and fine-tune generative models using open-source tools. (L3)

UNIT-I

Introduction To Gen AI: Historical Overview of Generative modelling, Difference between Gen AI and Discriminative Modeling, Importance of generative models in AI and Machine Learning, Types of Generative models, GANs, VAEs, autoregressive models and Vector quantized Diffusion models, understanding of probabilistic modeling and generative process, Challenges of Generative Modeling, Future of Gen AI, Ethical Aspects of AI, Responsible AI, Use Cases.

UNIT-II

Generative Models For Text: Language Models Basics, Building blocks of Language models, Transformer Architecture, Encoder and Decoder, Attention mechanisms, Generation of Text, Models like BERT and GPT models, Generation of Text, Autoencoding, Regression Models, Exploring ChatGPT, Prompt Engineering: Designing Prompts, Revising Prompts using Reinforcement Learning from Human Feedback (RLHF), Retrieval Augmented Generation, Multimodal LLM, Issues of LLM like hallucination.

UNIT-III

Generation of Images: Introduction to Generative Adversarial Networks, Adversarial Training Process, Nash Equilibrium, Variational Autoencoders, Encoder-Decoder Architectures, Stable Diffusion Models, Introduction to Transformer-based Image Generation, CLIP, Visual Transformers ViT- Dall-E2 and Dall-E3, GPT-4V, Issues of Image Generation models like Mode Collapse and Stability.

UNIT-IV

Generation of Painting, Music, and Play: Variants of GAN, Types of GAN, Cyclic GAN, Using Cyclic GAN to Generate Paintings, Neural Style Transfer, Style Transfer, Music Generating RNN, MuseGAN, Autonomous agents, Deep Q Algorithm, Actor-critic Network.

UNIT-V

Open Source Models And Programming Frameworks: Training and Fine tuning of Generative models, GPT-4All, Transfer learning and Pretrained models, Training vision models, Google Copilot, Programming LLM, LangChain, Open-Source Models, Llama, Programming for TimeSformer, Deployment, Hugging Face.

Text Books:

1. Denis Rothman, "Transformers for Natural Language Processing and Computer Vision", Packt Books, 3rd Edition, 2024.

Reference Books:

1. David Foster, "Generative Deep Learning", O'Reilly Books, 2024.
2. Altaf Rehmani, "Generative AI for Everyone", BlueRose One, 2024.

Course Code	Course Name	Course Structure			
		L	T	P	C
P23XXXXX	SOFTWARE PROJECT MANAGEMENT (PROFESSIONAL ELECTIVE-III)	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives:

1. Learn traditional vs. modern software management and software economics.
2. Understand software lifecycle phases and key project artifacts.
3. Explore software architectures, workflows, and iterative planning.
4. Study project roles, automation, and software metrics.
5. Learn Agile, Scrum, and DevOps practices in software development.

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

1. Distinguish between conventional and modern software management approaches.
2. Identify software lifecycle phases and describe key development artifacts.
3. Apply iterative planning and understand process workflows in software projects.
4. Analyze project organization, automation tools, and software metrics.
5. Implement Agile and DevOps practices in real-world software development.

UNIT-I

Conventional Software Management: The waterfall model, conventional software Management performance.

Evolution of Software Economics: Software Economics, pragmatic software cost estimation.

Improving Software Economics: Reducing Software product size, improving software processes, improving team effectiveness, improving automation, Achieving required quality, peer inspections.

The old way and the new: The principles of conventional software Engineering, principles of modern software management, transitioning to an iterative process.

UNIT-II

Life cycle phases: Engineering and production stages, inception, Elaboration, construction, transition phases.

Artifacts of the process: The artifact sets, Management artifacts, Engineering artifacts, programmatic artifacts.

UNIT-III

Model-based software architectures: A Management perspective and technical perspective.

Workflows of the process: Software process workflows, Iteration workflows.

Checkpoints of the process: Major milestones, Minor Milestones, Periodic status assessments.

Iterative Process Planning: Work breakdown structures, planning guidelines, cost and schedule estimating, Iteration planning process, Pragmatic planning.

UNIT-IV

Project Organizations and Responsibilities: Line-of-Business Organizations, Project Organizations, evolution of Organizations.

Process Automation: Building blocks, The Project Environment.

Project Control and Process instrumentation: The seven core Metrics, Management indicators, quality indicators, life cycle expectations, pragmatic Software Metrics, Metrics automation.

UNIT-V

Agile Methodology: ADAPTing to Scrum, Patterns for Adopting Scrum, Iterating towards Agility.

Fundamentals of DevOps: Architecture, Deployments, Orchestration, Need, Instance of applications, DevOps delivery pipeline, DevOps eco system.

DevOps adoption in projects: Technology aspects, Agiling capabilities, Tool stack implementation, People aspect, processes

Text Books:

1. Walker Royce, “Software Project Management”, PEA, 2005.
2. Mike Cohn, “Succeeding with Agile: Software Development Using Scrum”, Addison Wesley.
3. Gene Kim, John Willis, Patrick Debois, Jez Humble, “The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations”, 1st Edition, O’Reilly publications, 2016.

Reference Books:

1. Bob Hughes, Mike Cotterell, “Software Project Management”, TMH, 3rd edition.
2. Joel Henry, “Software Project Management”, PEA, 2003.
3. Pankaj Jalote, “Software Project Management in practice”, PEA, 2005.
4. Robert K.Wysocki, “Effective Software Project Management”, Wiley, 2006.
5. Kathy Schwalbe, “Project Management in IT”, Cengage.

Course Code	Course Name	Course Structure			
		L	T	P	C
P23XXXXX	MOBILE ADHOC NETWORKS (PROFESSIONAL ELECTIVE-III)	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives:

1. Understand MANET characteristics, applications, and MAC protocol design.
2. Explain routing and transport protocols for Ad Hoc networks.
3. Identify security challenges and solutions in MANETs.
4. Describe design and communication in Wireless Sensor Networks.
5. Summarize security and operating systems for Wireless Sensor Networks.

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

1. Explain the fundamentals and challenges of MANETs and MAC protocols. (L2)
2. Analyze routing and transport layer protocols for Ad Hoc networks. (L4)
3. Assess security issues and solutions in Mobile Ad Hoc Networks. (L2)
4. Understand design principles and communication in Wireless Sensor Networks. (L2)
5. Apply knowledge of security mechanisms and operating systems in Wireless Sensor Networks. (L3)

UNIT-I

Introduction to Ad Hoc Wireless Networks: Cellular and Ad Hoc Wireless Networks, Characteristics of MANETs, Applications of MANETs, Issues and Challenges of MANETs, Ad Hoc Wireless Internet, MAC protocols for Ad hoc Wireless Networks-Issues, Design Goals and Classifications of the MAC Protocols.

UNIT-II

Routing Protocols for Ad Hoc Wireless Networks: Issues in Designing a Routing Protocol, Classifications of Routing Protocols, Topology-based versus Position-based Approaches, Issues and design goals of a Transport layer protocol, Classification of Transport layer solutions, TCP over Ad hoc Wireless Networks, Solutions for TCP over Ad Hoc Wireless Networks, Other Transport layer protocols.

UNIT-III

Security protocols for Ad hoc Wireless Networks: Security in Ad hoc Wireless Networks, Network Security Requirements, Issues and Challenges in Security Provisioning, Network Security Attacks, Key Management, Secure Routing in Ad hoc Wireless Networks, Cooperation in MANETs, Intrusion Detection Systems.

UNIT-IV

Basics of Wireless Sensors and Applications: The Mica Mote, Sensing and Communication Range, Design Issues, Energy Consumption, Clustering of Sensors, Applications, Data Retrieval in Sensor Networks-Classification of WSNs, MAC

layer, Routing layer, Transport layer, High-level application layer support, Adapting to the inherent dynamic nature of WSNs.

UNIT-V

Security in WSNs: Security in WSNs, Key Management in WSNs, Secure Data Aggregation in WSNs, Sensor Network Hardware-Components of Sensor Mote, Sensor Network Operating Systems–TinyOS, LA-TinyOS, SOS, RETOS, Imperative Language-nesC, Dataflow Style Language-TinyGALS, Node-Level Simulators, NS-2 and its sensor network extension, TOSSIM.

Text Books:

1. C. Siva Ram Murthy, B. S. Murthy, "Ad Hoc Wireless Networks – Architectures and Protocols", Pearson Education, 1st edition, 2004.
2. Carlos Corderio Dharma P.Aggarwal, "Ad Hoc and Sensor Networks – Theory and Applications", World Scientific Publications / Cambridge University Press, 2nd edition March 2006.

Reference Books:

1. Feng Zhao, Leonidas Guibas, "Wireless Sensor Networks: An Information Processing Approach", Elsevier Science imprint, Morgan Kauffman Publishers, 1st edition, 2005, rp2009.
2. Subir Kumar Sarkar, et al., "Wireless Ad hoc Mobile Wireless Networks – Principles, Protocols and Applications", Auerbach Publications, Taylor & Francis Group, 1st edition, 2008.
3. Charles E.Perkins, "Ad hoc Networking", Pearson Education, 1st edition, 2001.
4. Shih-Lin Wu, Yu-Chee Tseng, "Wireless Ad hoc Networking", Auerbach Publications, Taylor & Francis Group, 1st edition, 2007.
5. Fei Hu, Xiaojun Cao, "Wireless Sensor Networks – Principles and Practice", An Auerbach book, CRC Press, Taylor & Francis Group, 1st edition, 2010.

Course Code	Course Name	Course Structure			
		L	T	P	C
P23XXXXX	BIG DATA ANALYTICS (PROFESSIONAL ELECTIVE-III)	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives:

1. Understand what big data is and how Big Data Technologies can help organizations achieve a competitive advantage.
2. Provide an overview of Apache Hadoop and its ecosystem components.
3. Understand Map Reduce Jobs.
4. Process Big Data with advanced architectures like Spark.
5. Understand practical machine learning scalable and easy.

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

1. Understand fundamentals of Big Data Technologies.
2. Investigate Hadoop framework and Hadoop Distributed File System.
3. Demonstrate the MapReduce programming model to process big data along with Hadoop tools.
4. Implement Big Data code in Apache Spark (in PySpark).
5. Run Supervised and Unsupervised machine learning on Large-Scale Data.

UNIT-I

Introduction to Big Data Analytics: Big Data, Scalability and Parallel Processing, Designing Data Architecture, Data Sources, Quality, Pre-Processing and Storing, Data Storage and Analysis, Big Data Analytics Applications and Case Studies.

UNIT-II

Introduction to Hadoop: Introduction, Hadoop and its Ecosystem, Hadoop Distributed File System, MapReduce Framework and Programming Model, Hadoop Yarn, Hadoop Ecosystem Tools.

Hadoop Distributed File System Basics: HDFS Design Features, Components, HDFS User Commands.

Hadoop Ecosystem Components: Using Apache Pig, Hive, Sqoop, Flume, Oozie, HBase.

UNIT-III

MapReduce, Hive and Pig: Introduction, MapReduce Map Tasks, Reduce Tasks and MapReduce Execution, Composing MapReduce for Calculations and Algorithms, Hive, HiveQL, Pig.

UNIT-IV

Large-Scale Data Processing with PySpark: Apache Spark, Spark programming (Python and PySpark), RDDs, Data Frames, Spark SQL, PySpark, NumPy, SciPy, Code Optimization, Cluster Configurations, Linear Algebra Computation in Large Scale, Distributed File Storage Systems.

UNIT-V

Large Scale Machine Learning with Spark: Basic statistics, Data sources, Pipelines, Extracting, transforming and selecting features, Classification and Regression, Clustering, Collaborative filtering, Frequent Pattern Mining, Model selection and tuning.

Text Books:

1. Raj Kamal and Preeti Saxena, "Big Data Analytics Introduction to Hadoop, Spark, and Machine-Learning", McGraw Hill Education, 2018 ISBN: 9789353164966, 9353164966.
2. Douglas Eadline, "Hadoop 2 Quick-Start Guide: Learn the Essentials of Big Data Computing in the Apache Hadoop 2 Ecosystem", 1st Edition, Pearson Education, 2016. ISBN13: 978-9332570351.

Reference Books:

1. Tom White, "Hadoop: The Definitive Guide", 4th Edition, O'Reilly Media, 2015. ISBN-13: 978-9352130672.
2. Perrin, J. (2020). *Spark in Action* (2nd ed.). (Covers Apache Spark 3 with examples in Java, Python, and Scala) O'Reilly Media Inc.
3. Arshdeep Bahga, Vijay Madisetti, "Big Data Analytics: A Hands-On Approach", 1st Edition, VPT Publications, 2018. ISBN-13: 978-0996025577.
4. Damji, J., Wenig, B., Das, T., Lee, D. (2020). *Learning Spark* (2nd ed.) O'Reilly Media Inc.
5. Nudurupati, S. (2021). *Essential PySpark for scalable data analytics: A beginner's guide to harnessing the power and ease of PySpark 3.* Packt Publishing.

Course Code	Course Name	Course Structure			
		L	T	P	C
P23EOE07	PRINCIPLES OF COMMUNICATION (OPEN ELECTIVE-II)	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Outcomes:

1. Analyze analog modulation schemes and angle modulated signals in time and frequency domains.
2. Understand the behaviour of analog signals as random processes and noise.
3. Apply knowledge of channel characteristics to assess effects on modulated signals.
4. Evaluate the performance of analog communication systems using SNR metrics.
5. Analyze pulse modulation techniques such as PAM, PPM, PCM, and TDM.

UNIT–I:

Basic tools for communication: Fourier Series/Transform, Properties, Autocorrelation, Energy Spectral Density, Parseval's Relation, Amplitude Modulation (AM), Spectrum of AM, Envelope Detection, Power Efficiency, Modulation Index.

UNIT–II:

Double Sideband Suppressed Carrier (DSB-SC) Modulation: Demodulation, Costas Receiver, Single Sideband Modulation (SSB), Hilbert Transform, Complex Preenvelope/Envelope, Demodulation of SSB, Vestigial Sideband Modulation (VSB)

UNIT–III:

Angle Modulation: Frequency Modulation (FM), Phase Modulation (PM), Modulation Index, Instantaneous Frequency, Spectrum of FM Signals, Carsons Rule for FM Bandwidth, Narrowband FM Generation, Wideband FM Generation via Indirect Method, FM Demodulation

UNIT–IV:

Introduction to Sampling: Spectrum of Sampled Signal, Aliasing, Nyquist Criterion, Signal Reconstruction from Sampled Signal, Pulse Amplitude Modulation, Quantization, Uniform Quantizers – Midrise and Midtread, Quantization noise, Non-uniform Quantizers, Delta Modulation, Differential Pulse Code Modulation (DPCM)

UNIT–V:

Basics of Probability: Conditional Probability, MAP Principle, Random Variables, Probability Density Functions, Applications in Wireless Channels, Basics of Random Processes, Gaussian Random Process, Noise.

Text Books:

1. Simon Haykin “Communications Systems”, John Wiley and Sons, Inc, 4th Edition, 2006.
2. David Tse “Fundamentals of Wireless Communication”, Cambridge University Press, 2005.

Web References:

1. Signals and Systems from Massachusetts Institute of Technology — Class Central
2. Coursera Plus — Unlimited Access to 10,000+ Online Courses

Course Code	Course Name	Course Structure			
		L	T	P	C
P21MBO05	MANAGEMENT SCIENCE(OPEN ELECTIVE -II)	2	0	2	2

Internal Marks: 30 External Marks: 70

Course Objectives:

1. To understand the application of management science in decision making process and its importance, evaluation of management thought, how organisation structure is designed along with its principles and types.
2. To understand types of management related to work study, quality control, control charts and inventory control and their types.
3. To learn the main functional areas of organisation such as Financial Management, Production Management, Marketing Management, Human Resource Management, Product life cycles and Channels of Distribution.
4. To understand the development of network and identifying critical path.
5. To understand the concept of strategic management and basic concepts of MIS, MRP, JIT, TQM, Six Sigma, CMM, Supply Chain Management, ERP, Business Process Outsourcing, Benchmarking and Business Process Re-engineering.

Course Outcomes:

1. Able to apply the concepts and principles of management in real life and design and develop organizational structures.
2. Able to apply PPC techniques, Quality Control, and Work-study principles in industry.
3. Able to identify and apply Marketing, HRM, and Production strategies effectively.
4. Able to develop PERT/CPM charts for projects and estimate time and cost.
5. Able to develop Mission, Objectives, Goals and strategies using modern management techniques like MIS, ERP, TQM, SCM, BPR, and Benchmarking.

UNIT–I:

Introduction to management: Concept, nature and importance of Management – Generic Functions of Management – Evaluation of Management thought – Theories of Motivation – Decision making process – Designing organization structure – Principles of organization – Organizational typology – International Management: Global Leadership and Organizational behavior Effectiveness (GLOBE) structure.

UNIT–II:

Operations Management: Principles and types of management – Work study – Statistical Quality Control – Control charts (P-chart, R-chart, and C-chart) with simple problems – Material Management: Need for Inventory control – EOQ, ABC analysis (simple problems) and Types of ABC analysis (HML, SDE, VED, and FSN analysis).

UNIT–III:

Functional Management: Concept of HRM, HRD and PMIR – Functions of HR Manager – Wage payment plans (Simple Problems) – Job Evaluation and Merit Rating – Marketing Management – Functions of Marketing – Marketing strategies based on product Life Cycle, Channels of distributions – Operationalising change through performance management.

UNIT–IV:

Project Management (PERT/CPM): Development of Network – Difference between PERT and CPM – Identifying Critical Path – Probability – Project Crashing (Simple Problems).

UNIT–V:

Strategic Management: Vision, Mission, Goals, Strategy – Elements of Corporate Planning Process – SWOT analysis – Steps in Strategy Formulation and Implementation – Generic Strategy Alternatives – Global strategies.

Contemporary Management Practices: Basic concepts of MIS, Total Quality Management (TQM), Six Sigma, Supply Chain Management, Enterprise Resource Planning (ERP), Business Process Re-engineering and Benchmarking.

Text Books:

1. Dr. P. Vijaya Kumar & Dr. N. Appa Rao, 'Management Science', Cengage, Delhi, 2012.
2. Dr. A. R. Aryasri, 'Management Science', TMH, 2011.

Reference Books:

1. Philip Kotler & Armstrong: Principles of Marketing, Pearson Publications.
2. Biswajit Patnaik: Human Resource Management, PHI, 2011.
3. Hitt and Vijaya Kumar: Strategic Management, Cengage Learning.
4. Seth & Rastogi: Global Management Systems, Cengage Learning, Delhi, 2011.

Web References:

1. https://mrcet.com/downloads/digital_notes/ECE/II%20Year/Management%20Science.pdf
2. https://books.askvenkat.org/management-science-textbook-aryasri-pdf/
3. https://nptel.ac.in/courses/122/102/122102007/
4. https://nptel.ac.in/courses/122/108/122108038/
5. http://www.universityofcalicut.info/SDE/Management_science_corrected_on13April2016.pdf

Course Code	Course Name	Course Structure			
		L	T	P	C
P23EOE08	PRINCIPLES OF SIGNAL PROCESSING (OPEN ELECTIVE-II)	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Outcomes:

1. Acquire fundamental knowledge of signals and systems.
2. Interpret and apply various transforms for analyzing continuous-time signals.
3. Explain sampling theorem and apply z-transform to discrete systems.
4. Analyze discrete-time signals using appropriate transforms.
5. Design and analyze digital filters for signal processing applications.

UNIT–I:

Introduction: Definition of Signals and Systems, Classification of Signals, Classification of Systems, Operations on signals: time-shifting, time-scaling, amplitude-shifting, Amplitude-scaling. Problems on classification and characteristics of Signals and Systems. Complex exponential and sinusoidal signals, impulse function, step function, signum function and ramp function. Introduction, Linear system, impulse response, Linear time invariant (LTI) system, Linear time invariant (LTV) system, Concept of convolution in time domain and frequency domain, Graphical representation of convolution, Transfer function of a LTI system, Related problems.

UNIT–II:

Analysis of Continuous Time Signals: Fourier Series and Fourier Transform: Fourier series representation of continuous time periodic signals, Dirichlet's conditions, Trigonometric Fourier series and Exponential Fourier series, Deriving Fourier transform from Fourier series, Fourier transform of standard signals, properties of Fourier transforms, Related problems.
Laplace Transforms: Introduction, Concept of region of convergence (ROC) for Laplace transforms, Properties of L.T's, Inverse Laplace transform, Relation between Laplace Transform and Fourier Transform of a signal.

UNIT–III:

Sampling and Z–Transforms: Sampling Theorem: Graphical and analytical proof for Band Limited Signals, impulse sampling, Reconstruction of signal from its samples, Aliasing.
Z–Transforms: Concept of Z-Transform of a discrete sequence. Region of convergence in Z-Transform, Inverse Z-transform, properties of Z-transforms.

UNIT–IV:

Fourier Analysis of Discrete Signals: Fourier Transform of Discrete Signal, Properties, and Inverse Fourier Transforms, related problems. Discrete Fourier Trans-

forms: Definition, Properties, Inverse DFT, related problems. Fast Fourier Transform: Decimation in Time domain and Decimation in Frequency Algorithms.

UNIT–V:

Digital Filters: Structures of IIR filters and FIR filters: Direct form–1 and Direct form 2; cascade form; parallel form. Analog filter design: LPF, BPF, HPF and BEF filter design using Butterworth. Frequency Transformations: Analog to Analog; Digital and Digital. IIR Filter Design: IIR filter from analog filter – IIR filter design by Impulse Invariance, Bilinear transformation. FIR Filter Design: Filter design using windowing techniques. Rectangular Window, Hamming Window, Hanning Window.

Text Books:

1. B. P. Lathi, “Signals, Systems and Communications”, BS Publications, 2003.
2. P. Ramesh Babu, “Digital Signal Processing” , SCITECH Publishers, 5th Edition, 2003.

Reference Books:

1. Simon Haykin and Van Veen, “Signals and Systems”, Wiley, 2nd Edition, 2007.
2. A.V. Oppenheim, A.S. Willsky and S.H. Nawab, “Signals and Systems”, PHI, 2nd Edition, 1997.
3. A.V. Oppenheim and R.W. Schaffer, “Discrete Time Signal Processing”, 3rd Edition, Pearson, 2014.

Web References:

1. Digital Signal Processing 1: Basic Concepts and Algorithms — Coursera
2. Digital Signal Processing (online) — Oxford University Department for Continuing Education

Course Code	Course Name	Course Structure			
		L	T	P	C
P23XXXX	DIGITAL MARKETING (OPEN ELECTIVE - II)	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives:

1. To impart knowledge of digital marketing concepts, channels, and analytics.
2. To distinguish traditional marketing from digital, and teach integrated strategies.
3. To familiarize students with SEO, SEM, Email, Content and Social Media Marketing.
4. To develop the ability to create, monitor and analyze digital marketing campaigns.
5. To provide practical knowledge in key digital tools and platforms for business growth.

Course Outcomes:

1. Understand the concepts, evolution, and significance of digital marketing.
2. Apply digital marketing strategies including SEO, SEM, Content, Email, and Social Media.
3. Design and optimize digital campaigns; use analytics for performance tracking.
4. Analyze and compare different digital media platforms and their suitability for various business objectives.
5. Demonstrate proficiency in digital tools, campaign planning, and measuring digital ROI.

UNIT–I:

Introduction to Digital Marketing: Fundamentals of Digital Marketing, Significance and Evolution, Traditional Marketing vs Digital Marketing, Digital Marketing Landscape, Key Drivers. Digital Consumer & Communities, Generation Y, Netizens' expectations and influence in Digital Marketing[**?**].

UNIT–II:

Digital Users and Strategy: Overview of digital users in India and worldwide, Digital Marketing Strategy: Consumer Decision Journey, POEM Framework (Paid, Owned, Earned Media), Segmenting and Customizing Messages, Digital Advertising Market, Digital Skills, Digital Marketing Plan[**?**].

UNIT–III:

SEO, SEM & Social Media: Terminology in Digital Marketing, Search Engine Optimization (SEO), Pay-Per-Click (PPC), Social Media Marketing (SMM), Social Media Platforms (Facebook, LinkedIn, Twitter, Instagram, YouTube): Content strategy,

Targeting, Ad Campaigns. Overview of Google Analytics, Affiliate Marketing, Email Marketing, Mobile Marketing[**?**][**?**].

UNIT–IV:

Digital Advertising and Display Ads: Types of Digital Ads (Search, Display, Video, Native), Display Advertising – Concepts, Buying Models (CPC, CPM, CPL, CPA), Ad Formats, Ad Placement Techniques. Programmatic Advertising, AdWords, AdSense. Display metrics and ROI measurement techniques[**?**][**?**].

UNIT–V:

Content, Email and Analytics: Content Marketing strategy: Creation, Distribution, Blogging, Multimedia (video, podcasting, images). Email Marketing: Campaign setup, email list generation, A/B testing. Introduction to Web Analytics: Google Analytics, Traffic reports, Conversion evaluation. Mobile Marketing overview, basic legal/ethical issues[**?**][**?**].

Text Books:

1. Kamat and Kamat, *Digital Marketing*, Himalaya Publishing.
2. Damien Ryan, *Marketing Strategies for Engaging the Digital Generation*.
3. V. Ahuja, *Digital Marketing*, Oxford University Press.
4. S. Gupta, *Digital Marketing*, McGraw-Hill.
5. H. Annmarie, A. Joanna, *Quick win Digital Marketing*.

Reference Books:

1. Dodson, Ian, *The Art of Digital Marketing*, Wiley.
2. Owen Richards, *Digital Marketing Analytics*.
3. Pradeep Chopra, *Digital Marketing for Dummies*.

Web Resources:

- `https://onlinecourses.swayam2.ac.in/cec21_mg09/preview`
- `https://www.hubspot.com/resources/courses`
- `https://learndigital.withgoogle.com/digitalgarage`

Course Code	Course Name	Course Structure			
		L	T	P	C
P23XXXXX	CLOUD COMPUTING LAB	0	0	3	1.5

Internal Marks: 50 External Marks: 50

Course Objectives:

1. To introduce the various levels of services offered by cloud.
2. To give practical knowledge about working with virtualization and containers.
3. To introduce the advanced concepts such as serverless computing and cloud simulation.

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

1. Demonstrate various service types, delivery models and technologies of a cloud computing environment.
2. Distinguish the services based on virtual machines and containers in the cloud offerings.
3. Assess the challenges associated with a cloud-based application.
4. Discuss advanced cloud concepts such as serverless computing and cloud simulation.
5. Examine various programming paradigms suitable to solve real world and scientific problems using cloud services.

List of Experiments:

1. Lab on web services
2. Lab on IPC, messaging, publish/subscribe
3. Install VirtualBox/VMware Workstation with different flavours of Linux or Windows OS on top of Windows8 or above.
4. Install a C compiler in the virtual machine created using VirtualBox and execute Simple Programs.
5. Create an Amazon EC2 instance and set up a web-server on the instance and associate an IP address with the instance. In the process, create a security group allowing access to port 80 on the instance.
   OR
6. Do the same with OpenStack
7. Install Google App Engine. Create a hello world app and other simple web applications using python/java.
8. Start a Docker container and set up a webserver (e.g. apache2 or Python based Flask micro web framework) on the instance. Map the host directory as a data volume for the container.
9. Find a procedure to transfer the files from one virtual machine to another virtual machine. Similarly, from one container to another container.

10. Find a procedure to launch virtual machine using try stack (Online OpenStack Demo Version)
11. Install Hadoop single node cluster and run simple applications like word count.
12. Utilize OpenVAS – Serverless computing framework and demonstrate basic event driven function invocation.
13. Simulate a cloud scenario using CloudSim and run a scheduling algorithm that is not present in CloudSim.

Text Books:

1. Rajkumar Buyya, Christian Vecchiola, Thamarai Selvi, ShivanandaPoojara, Satish N. Srirama, "Mastering Cloud Computing", 2nd edition, McGraw Hill, 2024.
2. Kai Hwang, Geoffery C. Fox, Jack J. Dongarra, "Distributed and Cloud Computing", Elsevier, 2012.

Reference Books:

1. Dan C Marinescu, "Cloud Computing, Theory and Practice", MK Elsevier, 2nd edition, 2018.
2. Rajkumar Buyya, James Broberg and Andrzej M. Goscinski, "Cloud Computing: Principles and Paradigms", Wiley, 2011.
3. Online documentation and tutorials from cloud service providers (e.g. AWS, Google App Engine)
4. Docker, Reference documentation, docs.docker.com/reference/
5. OpenFaaS, Serverless Functions Made Simple, docs.openfaas.com/

Course Code	Course Name	Course Structure			
		L	T	P	C
P23XXXXX	MACHINE LEARNING LAB	0	0	3	1.5

Internal Marks: 50 External Marks: 50

Course Objectives:

1. To learn about computing central tendency measures and data preprocessing techniques.
2. To learn about classification and regression algorithms.
3. To apply different clustering algorithms for a problem.

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

CO1: Perform basic statistics and data preprocessing. (L2)
CO2: Apply classification and regression algorithms. (L3)
CO3: Implement linear and logistic regression models. (L4)
CO4: Use clustering techniques for unsupervised learning. (L3)
CO5: Evaluate and tune machine learning models. (L4)

Software Required: Python / R / Weka
List of Experiments: Lab should cover the concepts studied in the course work, and sample list of Experiments:

1. Compute central tendency measures: Mean, Median, Mode; measure of dispersion: Variance, Standard Deviation.
2. Apply the following pre-processing techniques for a given dataset:
   - Attribute selection
   - Handling missing values
   - Discretization
   - Elimination of outliers
3. Apply KNN algorithm for classification and regression.
4. Demonstrate decision tree algorithm for a classification problem and perform parameter tuning for better results.
5. Demonstrate decision tree algorithm for a regression problem.
6. Apply Random Forest algorithm for classification and regression.
7. Demonstrate Naïve Bayes classification algorithm.
8. Apply Support Vector algorithm for classification.
9. Demonstrate simple linear regression algorithm for a regression problem.
10. Apply logistic regression algorithm for a classification problem.
11. Demonstrate multilayer perceptron algorithm for a classification problem.
12. Implement the K-means algorithm and apply it to selected data. Evaluate performance by measuring the sum of the Euclidean distance of each example from its class center. Test the performance of the algorithm as a function of the parameter K.

13. Demonstrate fuzzy c-means clustering.
14. Demonstrate expectation maximization-based clustering algorithm.

Course Code	Course Name	Course Structure			
		L	T	P	C
P23XXXXX	SOFT SKILLS (SKILL ENHANCEMENT COURSE)	0	0	3	1.5

Internal Marks: 30 External Marks: 70

Course Objectives:

1. To equip the students with the skills to effectively communicate in English.
2. To train the students in interview skills, group discussions and presentation skills.
3. To motivate the students to develop confidence.
4. To enhance the students' interpersonal skills.
5. To improve the students' writing skills.

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

1. Enhance analytical thinking and communication skills. (L2)
2. Develop self-management and etiquette skills. (L6)
3. Master grammar and business writing. (L2)
4. Improve job readiness and interview skills. (L2)
5. Strengthen interpersonal relationship skills. (L2)

UNIT – I

Analytical Thinking & Listening Skills: Self-Introduction, Shaping Young Minds - A Talk by Azim Premji (Listening Activity), Self – Analysis, Developing Positive Attitude, Perception.

Communication Skills: Verbal Communication; Non-Verbal Communication (Body Language)

UNIT – II

Self-Management Skills: Anger Management, Stress Management, Time Management, Six Thinking Hats, Team Building, Leadership Qualities

Etiquette: Social Etiquette, Business Etiquette, Telephone Etiquette, Dining Etiquette

UNIT – III

Standard Operation Methods: Basic Grammars, Tenses, Prepositions, Pronunciation, Letter Writing; Note Making, Note Taking, Minutes Preparation, Email & Letter Writing

UNIT-IV

Job-Oriented Skills: Group Discussion, Mock Group Discussions, Resume Preparation, Interview Skills, Mock Interviews

UNIT-V

Interpersonal Relationships: Introduction, Importance, Types, Uses, Factors affecting interpersonal relationships, Accommodating different styles, Consequences of interpersonal relationships

Text Books:

1. Barun K. Mitra, "Personality Development and Soft Skills", Oxford University Press, 2011.
2. S.P. Dhanavel, "English and Soft Skills", Orient Blackswan, 2010.

Reference Books:

1. R.S.Aggarwal, "A Modern Approach to Verbal & Non-Verbal Reasoning", S.Chand & Company Ltd., 2018.
2. Raman, Meenakshi & Sharma, Sangeeta, "Technical Communication Principles and Practice", Oxford University Press, 2011.

E-resources:

1. https://swayam-plus.swayam2.ac.in/courses/course-details?id=P_CAMBR_01

Course Code	Course Name	Course Structure			
		L	T	P	C
P23XXXXX	TECHNICAL PAPER WRITING & IPR (AUDIT COURSE)	2	0	0	0

Internal Marks: – External Marks: –

Course Objectives:

1. Plan and write clear technical reports using correct tense and transitions.
2. Draft reports with illustrations and edit for grammar and clarity.
3. Proofread, summarize, and present technical documents effectively.
4. Use advanced word processor tools to prepare technical reports.
5. Understand intellectual property types and the patenting process.

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

1. Write well-structured technical reports with appropriate language and format.
2. Prepare drafts using visuals and improve documents through effective editing.
3. Proofread, summarize, and confidently present technical information.
4. Utilize advanced word processing features for professional report preparation.
5. Explain intellectual property concepts and the patenting process in technology.

UNIT-I

Introduction: An introduction to writing technical reports, technical sentences formation, using transitions to join sentences, Using tenses for technical writing.

Planning and Structuring: Planning the report, identifying reader(s), Voice, Formatting and structuring the report, Sections of a technical report, Minutes of meeting writing.

UNIT-II

Drafting report and design issues: The use of drafts, Illustrations and graphics.

Final edits: Grammar, spelling, readability and writing in plain English: Writing in plain English, Jargon and final layout issues, Spelling, punctuation and Grammar, Padding, Paragraphs, Ambiguity.

UNIT-III

Proofreading and summaries: Proofreading, summaries, Activities on summaries.

Presenting final reports: Printed presentation, Verbal presentation skills, Introduction to proposals and practice.

UNIT-IV

Using word processor: Adding a Table of Contents, Updating the Table of Contents, Deleting the Table of Contents, Adding an Index, Creating an Outline,

Adding Comments, Tracking Changes, Viewing Changes, Additions, and Comments, Accepting and Rejecting Changes, Working with Footnotes and Endnotes, Inserting citations and Bibliography, Comparing Documents, Combining Documents, Mark documents final and make them read only, Password protecting Microsoft Word documents, Using Macros.

UNIT-V

Nature of Intellectual Property: Patents, Designs, Trade and Copyright.

Process of Patenting and Development: Technological research, innovation, patenting, development.

International Scenario: International cooperation on Intellectual Property

Text Books:

1. Kompal Bansal & Parshit Bansal, "Fundamentals of IPR for Beginner's", BS Publications, 1st Edition, 2016.
2. William S. Pfeiffer and Kaye A. Adkins, "Technical Communication: A Practical Approach", Pearson.
3. Ramappa, T., "Intellectual Property Rights Under WTO", S Chand, 2nd Edition, 2015.

Reference Books:

1. Adrian Wallwork, "English for Writing Research Papers", Springer, 2011.
2. Day R, "How to Write and Publish a Scientific Paper", Cambridge University Press, 2006.

E-resources:

1. https://www.udemy.com/course/reportwriting/
2. https://www.udemy.com/course/professional-business-english-and-technical-report-writing/
3. https://www.udemy.com/course/betterbusinesswriting/