AVANTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

(Autonomous)

(Approved by A.I.C.T.E., New Delhi &Permanently Affiliated to JNTU-GV, Vizianagaram)
NAAC Accredited with A+ grade
Tamaram (V), Makavarapalem, Narsipatnam (RD), Anakapalle Dist,Pin-531113

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

ACADEMIC REGULATIONS

COURSE STRUCTURE AND SYLLABUS

For UG-R24

B.Tech – COMPUTER SCIENCE AND ENGINEERING

(Applicable for batches admitted from 2024-2025)

AVANTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY
(Autonomous)

Tamaram (V), Makavarapalem,Narsipatnam (RD), Anakapalle Dist,Pin-531113

AVANTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY
(Autonomous)
(Approved by A.I.C.T.E., New Delhi &Permanently Affiliated to JNTU-GV, Vizianagaram)
NAAC Accredited with A+ grade
Tamaram (V), Makavarapalem, Narsipatnam (RD), Anakapalle Dist,Pin-531113.
www.avanthienggcollege.ac.in, mail: principal@avanthienggcollege.ac.in

Academic Regulations 2024 (R24) for B. Tech (Regular-Full time)

(Effective for the students admitted into I year from the
Academic Year: 2024-2025 onwards)

1. Award of the B.Tech 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 be 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 are to be completed simultaneously with B.Tech. Programme.

2. Students, who fail to fulfill 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 present with specialization in the B.Tech

S.No.	Branch Code - Abbreviation	Branch
1.	02-EEE	Electrical and Electronics Engineering
2.	03-ME	Mechanical Engineering
3.	04-ECE	Electronics and Communication Engineering
4.	05-CSE	Computer Science and Engineering

5.	42-CSM	Computer Science and Engineering (Artificial Intelligence and Machine Learning)
6.	44-CSD	Computer Science and Engineering (Data Science)

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/ Institution 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 /Institution or to any other order of merit approved by the A.P. Government / Institution, subject to reservations as prescribed by the Government/ Institution from time to time.

5. Program related terms

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

Credit Definition:

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

a) **Semester:** A semester comprises 90 working days.
b) **Academic Year:** Two consecutive (one odd + one even) semesters constitute one academic year.
c) **Choice Based Credit System (CBCS):** The CBCS provides a choice for students to select from the prescribed courses.

6. Semester / Credits:

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

7. Structure of 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 (B.Tech. Degree programmes) are broadly classified as follows:

S. No.	Broad Course Classification	Course Category	Description
1.	Foundation Core Courses	Foundation courses	Includes Mathematics, Physics and Chemistry; fundamental engineering courses; Humanities, Social sciences and Management courses
2.	Professional Core Courses	Professional Core Courses (PC)	Includes subjects related to the parent discipline/department/branch of Engineering
3.	Open Elective Courses	Professional Elective Courses (PE)	Include selective 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
		Domains 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

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

ii. Each academic year of study is divided into two semesters.

iii. Minimum number of instruction days in each semester is 90 days

iv. There shall be mandatory student induction program for fresher's, with 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.

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

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

vii. Design Thinking for Innovation & Tinkering Labs is made mandatory as credit courses for all the undergraduate students.

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

ix. Professional Elective Courses, include the elective courses relevant to the chosen specialization/branch. Proper choice of professional elective courses can lead to students specializing in emerging areas within the chosen field of study.

x. 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.

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

xii. 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.

xiii. 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

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

xv. Undergraduate degree with Honors is introduced by the Institution for the students having good academic record.

xvi. 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.

xvii. 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.

xviii. 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, and mandatory courses with no credits shall be evaluated for 30 MID semester marks.

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

Theory Courses

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

i) For theory subject, the distribution shall be 30 marks for Internal Evaluation and 70 marks for the End-Examination.

ii) For practical subject, the distribution shall be 30 marks for Internal Evaluation and 70 marks for the End- Examination.

iii) If any course contains two different branch subjects, the syllabus shall be written in two parts with 2.5 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 questions / short answer questions (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 / multiple choice questions (MCQ's) for 10 marks. Subjective paper shall contain 3 questions and 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 questions of equal weightage of 10 marks. Any fraction shall be rounded off to the next higher mark.
- The objective paper shall be conducted either online or offline by the respective department 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 2.5 (I Unit, II Unit and III Unit half part) units of syllabus the second midterm examination shall be conducted for remaining 2.5 Units (III Unit half part, IV and V units).

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: (25 x 0.8) + (20 x 0.2) = 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: (25 x 0.8) + (0 x 0.2) = 20

b) 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. There shall be 2 short answer questions from each unit

(iii) In each of the questions from 2 to 6, there shall be either / or type questions of 10 marks each and each question have internal choice. 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.

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 regularity / 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.

- Procedure:20 marks
- Experimental work & Results: 30 marks
- Viva Voce: 20 marks.

In a practical subject consisting of two parts (Eg: Basic Electrical & Electronics Engineering Lab), the examination shall be conducted for 70 marks as a single laboratory in 3 hours. Mid semester examination shall be evaluated 30 marks in each part. 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.

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

Day-to-Day work shall be evaluated for 15 marks by the concerned subject teacher based on the reports/submissions prepared in the class. And there shall be two midterm examinations in a semester for duration of 2 hours each for 15 marks with weightage of 80% to better mid marks and 20% for the other. The subjective paper shall contain 3 either or type questions of equal weightage of 5 marks. There shall be no objective paper in mid semester examination. The sum of day-to-day evaluation and the mid semester marks will be the final sessional marks for the subject.

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.

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

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

ii) Out of the five skill courses, two shall be skill-oriented courses from the same domain of their main three skill courses, one shall be a soft skill course and the remaining two shall be skill-advanced courses from the same domain/Interdisciplinary/Job oriented.

iii) The course shall carry 100 marks and shall be evaluated through continuous assessments during the semester for 30 sessional marks and end examination shall be for 70 marks. Day-to-Day work in the class / laboratory shall be evaluated for 30 marks by the concerned teacher based on the regularity/ assignments / viva/ mid semester test. The end examination similar to practical examination pattern shall be conducted by the concerned teacher and an expert in the subject nominated by the principal.

iv) The Head of the Department shall identify a faculty member as coordinator for the course. A committee consisting of the Head of the Department, coordinator and a senior Faculty member nominated by the Head of the Department shall monitor the evaluation process. The marks / grades shall be assigned to the students by the above committee based on their performance.

v) The student shall be given an option to choose either the skill courses being offered by the 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.

vi) 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 Institute.

12. Massive Open Online Courses (MOOCs):

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

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 Institution shall allow up to a maximum of 20% of the total courses being offered in a particular programme i.e., maximum of 32 credits through MOOCs platform.

i) The AIET (A) shall offer credit mobility for MOOCs and give the equivalent credit weightage to the students for the credits earned through online learning courses.
ii) Student registration for the MOOCs shall be only through the respective department of the institution, it is mandatory for the student to share necessary information with the department.
iii) The Credit transfer policy will be applicable to the Professional & Open Elective courses only.
iv) The concerned department shall identify the courses permitted for credit transfer.
v) The Institution shall notify at the beginning of semester the list of the online learning courses eligible for credit transfer.
vi) The institution shall designate a faculty member as a Mentor for each course to guide the students from registration till completion of the credit course.
vii) The Institute shall ensure no overlap of MOOC exams with that of the Institute examination schedule. In case of delay in results, the Institute will re-issue the marks sheet for such students.
viii) Student pursuing courses under MOOCs shall acquire the required credits only after successful completion of the course and submitting a certificate issued by the competent authority along with the percentage of marks and grades.
ix) The institution shall submit the following to the examination section of the Institute:
(a) List of students who have passed MOOC courses in the current semester along with the certificate of completion.

(b) Undertaking form filled by the students for credit transfer.

x) The 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 Institute 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

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

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 / Institute 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 as 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 Institute.

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 show cased at the department for the benefit of all students and staff and the same is to be evaluated by the departmental Project Review Committee consisting of supervisor, a senior faculty and HOD for 30 marks. The external evaluation of Project Work is a Viva-Voce Examination conducted in the presence of internal examiner and external examiner appointed by the Institute and is evaluated for 140 marks.

The institute 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 Minor

To promote inter disciplinary 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.

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

ii) Two courses for 06 credits related to a Minor are to be pursued compulsorily for them in or degree, but may be 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.

iii) 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 within the 160 credits by opting for the courses offered through various verticals/tracks under Open Electives.

17. Guidelines for offering Honors

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

i) Honors are 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.

ii) A student shall earn additional 15 credits for award of B.Tech. (Honors) degree from same branch/ department/ discipline registered for major degree. This is in addition to the credits essential for obtaining the under graduate degree in Major Discipline (i.e., 160 credits).

iii) A student is permitted to register for Honors in IV semester after the results of III Semester are declared and students may be allowed to take maximum two subjects per semester pertaining to the Honors from V Semester onwards.

iv) The concerned head of the department shall arrange separate class work and timetable of the courses offered under Honors program.

v) Courses that are used to fulfill the student's primary major may not be double counted towards the Honors. Courses with content substantially equivalent to courses in the student's primary Major may not be counted towards the Honors.

vi) Students can complete the courses offered under Honors either in the college or in online platforms like SWAYAM with a minimum duration of 12 weeks for a 3-credit course and 8 weeks duration for a 2-credit course satisfying the criteria for credit mobility. If the courses under Honors are offered in conventional mode, then the teaching and evaluation procedure shall be similar to regular B. Tech courses.

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

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

ix) A student registered for Honors shall pass in all subjects that constitute the requirement for the Honors degree program. No class/division (i.e., second class, first class and distinction, etc.) shall be awarded for Honors degree programme.

x) If a student drops or is terminated from the Honors program, the additional credits so far earned cannot be converted into open or core electives; they will remain extra. However, such students will receive a separate grade sheet mentioning the additional courses completed by them.

xi) The Honors will be mentioned in the degree certificate as Bachelor of Technology (Honors) in XYZ. For example, B.Tech. (Honors) in Mechanical Engineering.

Enrolment into Honors:

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

ii) The enrolment of student into Honors is based on the CGPA obtained in the major degree program. CGPA shall be taken up to III semester incase of regular entry students and only III semester in case of lateral entry students. Students having 7 CGPA without any backlog subjects will be permitted to register for Honors.

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

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

v) Honors are to be completed simultaneously with a Major degree program.

Registration for Honors:

i) The eligible and interested students shall apply through the HOD of his / her parent department. The whole process should be completed within one week before the start of every semester. Selected students shall be permitted to register the courses under Honors.

ii) The selected students shall submit their willingness to the principal through his/her parent department offering Honors. The parent department shall maintain the record of student pursuing the Honors.

iii) The students enrolled in the Honors courses will be monitored continuously. An advisor/ mentor from parent department shall be assigned to a group of students to monitor the progress.

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

18. Attendance Requirements:

i) A student shall be eligible to appear for the Institution's / Institute's external examinations if he/she acquires a minimum of 40% attendance in each subject and 75% of attendance in aggregate of all the subjects. b) 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's Academic Committee.

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

iii) A stipulated fee shall be payable towards condonation of shortage of attendance to the Institute.

iv) Students whose shortage of attendance is not condoned in any semester are not eligible to take their end examination of that class and their registration shall stand cancelled.

v) A student will not be promoted to the next semester unless he/she satisfies the attendance requirements of the present semester. They may seek readmission for that semester from the date of commencement of class work.

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

vii) If the learning is carried out in blended mode (both offline & online), then the total attendance of the student shall be calculated considering the offline and online attendance of the student.

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

19. Promotion Rules:

The following academic requirements must be satisfied in addition to the attendance requirements mentioned in section 18.

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

ii) A student will be promoted from II to III year if he/she fulfils the academic requirement of securing 40% of the credits (any **decimal** fraction should be **rounded off** to **lower** digit) up to in the subjects that have been studied up to III semester.

iii) A student shall be promoted from III year to IV year if he/she fulfils the academic requirements of securing 40% of the credits (any **decimal** fraction should be **rounded off** to **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.

iv) When a student is detained due to lack of credits/ shortage of attendance he/she may be re-admitted when the semester is offered after fulfillment of academic regulations. In such case, he/she shall be in the academic regulations into which he/she is readmitted.

20. Grading:

As 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: **A**fter each course is evaluated for 100 marks, the marks obtained in each course will be converted to a corresponding

letter grade as given below, depending on the range in which the marks obtained by the student fall.

Structure of Grading of Academic Performance

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

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

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

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

SGPA: 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 under gone by a student, i.e.,

$$\mathbf{SGPA=\Sigma (C_i \times G_i) / \Sigma C_i}$$

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.

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

$$\mathbf{CGPA=\Sigma (C_i \times S_i) / \Sigma C_i}$$

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 whom 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, E and F.

Award of Class:

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

Class Awarded	CGPA Secured
First Class with Distinction	≥ 7.5 (Without any supplementary appearance)
First Class	≥ 6.5 < 7.5
Second Class	≥ 5.5 < 6.5
Pass Class	≥ 5.0 < 5.5

Note: Students who have written supplementary examinations to fulfill the credit requirement will not be awarded First Class with Distinction. For such students the highest degree that is awarded will be First Class Only.

CGPA to Percentage conversion Formula: (CGPA–0.75) x 10

21. With-holding of Results

If the candidate has any dues not paid to the Institute or if any case of indiscipline or malpractice is pending against him/her, the result of the candidate shall be with held 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 Institute 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/APSCHE 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 principal of the respective college shall forward such proposals submitted by the students to the Institute. An evaluation committee constituted by the Institute 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 fulfillment 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, subjected 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, subjected to Section **2** and they

will follow the academic regulations in to which they are readmitted.

25. Minimum Instruction Days for a Semester:

The minimum instruction days including internal 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 (including examinations and project reports) will be in English only.

27. Student Transfers:

Student transfers shall be as per the guidelines issued by the Government of Andhra Pradesh / JNTU-GV and the Institute from time to time.

28. General Instructions:

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

ii) Malpractices rules-nature and punishments are appended.

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

iv) In the case of any doubt or ambiguity in the interpretation of the above rules, the decision of the Principal / Dean-Academics of the institution is final.

v) The Institute 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.

vi) In the case of any doubt or ambiguity in the interpretation of the guidelines given, the decision of the Principal / Dean-Academics of the institution is final.

ACADEMIC REGULATIONS (R24) for B.Tech (LATERAL ENTRY SCHEME)

(Effective for the students getting admitted into II year through Lateral Entry Scheme from the Academic Year **2025-2026** onwards)

1. Award of the Degree

(a) Award of the B.Tech. Degree / B.Tech. Degree with a Minor if he/she fulfills 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**

A student will be declared eligible for the award of the B.Tech. 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 are to be completed simultaneously with B.Tech programme.

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

3. Minimum Academic Requirements

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

i. A student shall be deemed to have satisfied the minimum academic requirements and earned the credits allotted to each theory, practical, design, drawing subject or project if he secures not less than 35% of marks in the end examination and a minimum of 40% of marks in the sum total of the mid semester evaluation and end examination taken together.

ii. A student shall be promoted from III year to IV year if he/she fulfils the academic requirements of securing 40% of the credits (any decimal fraction should be rounded off to lower digit) in the subjects that have been studied up to V 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

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

ii) 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.

iii) 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 fulfillment 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).

****

(Dr. R Prasad Rao)
Dean(Academics) &
Member Secretary (AC)

(Dr.C P V N J Mohan Rao)
Chairman
Academic Council

AVANTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY
(Autonomous)
(Approved by A.I.C.T.E., New Delhi &Permanently Affiliated to JNTU-GV, Vizianagaram)
NAAC Accredited with A+ grade
Tamaram (V), Makavarapalem,Narsipatnam (RD), Anakapalle Dist,Pin-531113.
https://avanthienggcollege.ac.in/, E-mail: principal@avanthienggcollege.ac.in

Department of COMPUTER SCIENCE AND ENGINEERING

Proposed Course Structure

Program– B. Tech Computer Science and Engineering

Regulation-R24

(Applicable from the academic year 2024-2025 to 2026-2027)

Induction Programme

S.No	Course Title	Category	L-T-P-C
1	Physical Activities--Sports, Yoga and Meditation, Plantation	MC	0-0-6-0
2	Career Counseling	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 COMPUTER SCIENCE AND ENGINEERING

Program: B. Tech Computer Science and Engineering Regulation: R24

I Year I Semester- Course Structure

S.No	Category	Course Code	Course Title	Hours per Week			
				Lecture	Tutorial	Practical	Credits
1	BS	R24BS01	Linear Algebra and Calculus	3	0	0	3
2	HS	R24HS01	Communicative English	2	0	0	2
3	BS	R24BS02	Engineering Physics	3	0	0	3
4	ES	R24ES01	Basic Civil & Mechanical Engineering	3	0	0	3
5	ES	R24ES02	Problem Solving and Programming with C	3	0	0	3
6	BS	R24BS03	Engineering Physics Lab	0	0	2	1
7	HS	R24HS02	Communicative English Lab	0	0	2	1
8	ES	R24ES03	Problem Solving and Programming with C Lab	0	0	3	1.5
9	ES	R24ES04	IT Workshop	0	0	2	1
10	MC	R24MC01	Health and Wellness, Yoga, and Sports	0	0	1	0.5
			Total	14	00	10	19

Category	Courses	Credits
BS-Basic Science Course	3	7
ES-Engineering Science Course	4	8.5
HS-Humanities and Social Science including Management Course	2	3
MC-Mandatory Course	1	0.5
Total	10	19

Department of COMPUTER SCIENCE AND ENGINEERING

Program: B. Tech Computer Science and Engineering Regulation: R24

I Year II Semester- Course Structure

S.No	Category	Course Code	Course Title	Hours per Week			Credits
				Lecture	Tutorial	Practical	
1	BS	R24BS04	Differential Equations and Vector Calculus	3	0	0	3
2	BS	R24BS05	Applied Chemistry	3	0	0	3
3	ES	R24ES06	Engineering Graphics	1	0	4	3
4	ES	R24ES05	Basic Electrical and Electronics Engineering	3	0	0	3
5	PC	R24CSPC01	Data Structures through C	3	0	0	3
6	BS	R24BS06	Applied Chemistry Lab	0	0	2	1
7	ES	R24ES07	Basic Electrical and Electronics Engineering Lab	0	0	3	1.5
8	PC	R24CSPC02	Data Structures through C Lab	0	0	3	1.5
9	ES	R24ES08	Engineering Workshop	0	0	3	1.5
10	MC	R24MC02	NSS/NCC/Scouts &Guides/Community Service	0	0	1	0.5
			Total	13	00	16	21

Category	Courses	Credits
BS- Basic Science Course	3	7
ES-Engineering Science Course	4	9
PC-Professional Core Course	2	4.5
MC-Mandatory Course	1	0.5
Total	10	21

Chairperson
Board of Studies (CSE)

Linear Algebra and Calculus
I B.TECH- I SEMESTER (Common to all Branches)

Course Title: **Linear Algebra and Calculus**	Course Code: **R24BS01**
Teaching Scheme (L:T:P): 3:0:0	Credits: 3
Type of Course: Lecture + Tutorial	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites: Linear algebra is a prerequisite for calculus, and that you should have a deep understanding of linear algebra before moving on to calculus.	

COURSE OVERVIEW:

A course on linear algebra and calculus typically covers fundamental concepts like vectors, matrices, linear systems, differentiation, and integration.

COURSE OBJECTIVES:

The objectives of this course are to:

1. To equip the students with standard concepts and tools of mathematics to handle various real-world problems and their applications.
2. To enable the students to apply linear algebra to solve engineering problems.
3. To enable the students to apply calculus to solve engineering problems.

COURSE OUTCOMES:

CO#	Course Outcomes
CO1	Develop matrix algebra techniques that are needed by engineers for practical applications.
CO2	To find the eigen values and eigen vectors and solve the problems by using linear transformation.
CO3	Apply the knowledge of mean value theorems, solve inequality.
CO4	Familiarize with functions of several variables which is useful in optimization.
CO5	Familiarize with double and triple integrals of functions of several variables in two and three dimensions.

COURSE CONTENT (SYLLABUS)

UNIT-I: Matrices and Linear System of Equations **10 Hours**

Matrices: Vector Space, Linear independent, dependent (only definitions).

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. **COs-CO1**

Self-Learning Topic: Encoding and Decoding messages by using matrices

UNIT- II: Linear Transformation and Orthogonal Transformation **10 Hours**

Eigen values and Eigen vectors and their properties(without proof), 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. **COs-CO2**

Self-Learning Topic: Google's page rank Algorithm.

UNIT–III: Calculus **10 Hours**

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 on the above theorems. **COs-CO3**

Self-Learning Topic: Application of mean value theorems

UNIT- IV: Partial differentiation and Applications **10 Hours**

Partial derivatives, total derivatives, chain rule, change of variables, Taylor's and Maclaurin's series expansion of functions of two variables. Jacobian, maxima and minima of functions of two variables, method of Lagrange multipliers. **COs-CO4**

Self-Learning Topic: Jacobian of implicit functions.

UNIT-V: Multiple Integrals **10 Hours**

Double integrals - change of variables (Cartesian and Polar coordinates), change of order of integration, Cylindrical and Spherical coordinates, triple integrals. Finding areas (by double integrals) and volumes (by double integrals and triple integrals). **COs-CO5**

Self-Learning topic: Calculating Centers of Mass and Moment of inertia

Text Books:

1. B.S.Grewal, Higher Engineering Mathematics, 44/e, Khanna publishers, 2017.
2. Erwin Kreyszig, Advanced Engineering Mathematics, 10/e, John Wiley & Sons, 2018.

Reference Books:

1. Dennis G. Zill and Warren S. Wright , Advanced Engineering Mathematics, Jones and Bartlett, 2018.
2. Michael Green berg, Advanced Engineering Mathematics, 9th edition, Pearson edn.
3. George B. Thomas, Maurice D. Weir and Joel Hass, Thomas Calculus, 14/e, Pearson Publishers, 2018.
4. R. K. Jain and S. R. K. Iyengar, Advanced Engineering Mathematics, 5/e, Alpha Science
5. International Ltd., 2021 (9th reprint).
6. B.V. Ramana, Higher Engineering Mathematics, McGraw Hill Education, 2017.

Web References:

1. http://onlinecourses.nptel.ac.in
2. https://nptel.ac.in/courses/111105121
3. https://onlinecourses.nptel.ac.in/noc24_ma91/course
4. https://onlinecourses.nptel.ac.in/noc24_ma53/course
5. https://onlinecourses.nptel.ac.in/noc24_ma11/course

COMMUNICATIVE ENGLISH
I B.TECH- I SEMESTER (Common to CSE,CSE(AIML),CSE(DS))

Course Title: **COMMUNICATIVE ENGLISH**	Course Code: **R24HS01**
Teaching Scheme (L:T:P): **2 0 0**	Credits: 2
Type of Course: Lecture + Tutorial	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites: To excel in a *Communicative English* course, certain foundational skills and prerequisites are helpful such as Basic Grammar Knowledge, **Listening Skills,** Basic Vocabulary, Reading Comprehension, Confidence in Speaking, Writing Skills.	

COURSE OVERVIEW:

A *Communicative English* course is designed to develop students' proficiency in spoken and written English through practical and interactive learning methods. The course focuses on improving students' ability to communicate effectively in real-life situations, emphasizing both fluency and accuracy.

COURSE OBJECTIVES:

The objectives of this course are to:

1. To identify the English Communication Skills among the first year B.Tech students and to initiate measures to bridge the gap.
2. To enlighten the students on the necessity of cultivating good language habits through practising LSRW skills.
3. To explain them various topics of grammar and the importance of being grammatically correct in speech and writing.
4. To make them practise Phonetics and impart the nuances of fine speech.
5. To instruct them about the various types of format related to writing letters, paragraph, emails, essays and reports.
6. To make them appreciate English text and deepen their comprehension through reading of textual and non-detailed topics.

COURSE OUTCOMES:

CO#	Course Outcomes
CO1	To utilize the text, online resources, and other social, and real time situations with an aim to practice Communicative English
CO2	To apply grammatical knowledge for speaking, and writing purposes
CO3	To analyze and practice various devices of speech for effective conversation and presentations
CO4	Appraising the language competence of the learners and suggesting remedial action
CO5	To make the learners practice writing tasks which are relevant for job training and academic purposes.

COURSE CONTENT (SYLLABUS)

UNIT-I **12 Hours**

Lesson: HUMAN VALUES: A Power of a Plate of Rice by Ifeoma Okoye (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. **COs-C01**

Self learning topics : The Great Indian Scientists-Biography of CV Raman

UNIT-II **10 Hours**

Lesson: NATURE: Night of the Scorpion by Nissim Ezekiel (Indian and contemporary)

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. **COs-C02**

Self learning Topics : Seven Ages of Man by William Shakespeare.

UNIT-III **12 Hours**

Lesson: BIOGRAPHY: Steve Jobs

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

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

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

Writing: Summarizing, Note-making, paraphrasing

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

Vocabulary: Compound words, Collocations **COs-C03**

Self learning topics: Elon Musk

UNIT-IV **8 Hours**

Lesson: INSPIRATION: The Knowledge Society by APJ Abdul Kalam (Ignited minds)

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 and Resumes
Grammar: Reporting verbs, Direct & Indirect speech, Active & Passive Voice
Vocabulary: Words often confused, Jargons **COs-C04**
Self learning Topics: The writings of Sudha Murthy- "The day I stopped drinking milk"

UNIT-V **10 Hours**

Lesson: MOTIVATION: The Power of Intra personal Communication (An Essay)
Listening: Identifying key terms, understanding concepts and answering a series of relevant questions that test comprehension.
Speaking: Formal Oral Presentation topics from academic contexts
Reading: Reading comprehension.
Writing: Writings 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 **COs-C05**
Self learning Topics: Body Language (Allan Pease)

Textbooks:

1. Pathfinder: Communicative English for Undergraduate Students, 1st Edition,Orient BlackSwan, 2023 (Units 1,2,3 &5)
2. Empowering English by Cengage Publications, 2023
3. The Great Indian Scientists-Cengage Publications
4. English Essentials- Maruthi Publications.(Unit 4)

Reference Books:

1. P. Elian : A Hand book of English for Engineers and Technologists,
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. English for Engineers by Shyam Ji Dubey- Vikas Publishing House

Web References:

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
8. https://www.youtube.com/channel/UC4cmBAit8i_NJZE8qK8sfpA

ENGINEERING PHYSICS

I B.TECH- I SEMESTER (Common to CSE,CSE(AI&ML),CSE(DS)

Course Title: **ENGINEERING PHYSICS**	Course Code: **R24BS02**
Teaching Scheme (L:T:P): 3:0:0	Credits: 3
Type of Course: Lecture + Tutorial	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites: For *Engineering Physics*, which combines principles of physics with engineering applications, having a solid foundation in certain areas is crucial for understanding advanced concepts. Here's an outline of key prerequisites that can help students succeed in the course.	

COURSE OVERVIEW:

An *Engineering Physics* course is typically designed to bridge the gap between theoretical physics principles and engineering applications, providing students with a solid foundation to analyze and solve complex engineering problems.

COURSE OBJECTIVES:

The objectives of this course are to

1. bridge the gap between the physics in school at 10+2 level and UG level engineering courses.
2. identify the importance of the optical phenomenon i.e. interference and diffraction related to its engineering applications.
3. understand the mechanism of emission of light, utilization of lasers as coherent light sources for low and high energy applications.
4. enlightening the periodic arrangement of atoms in crystalline solids and classify various crystal systems.
5. explain the significant concepts of dielectric and magnetic materials that leads to potential applications in the emerging micro devices.
6. enlightenment of the concepts of quantum mechanics and to provide fundamentals of de-Broglie matter waves and the importance of free electron theory for metals.
7. understand the physics of semiconductors and identify the type of semiconductor using Hall effect.

COURSE OUTCOMES:

CO#	Course Outcomes
CO1	Analyze the intensity variation of light due to interference, diffraction and classify various types of lasers.
CO2	Identify various crystal systems and analyze the crystalline structure.
CO3	Summarize various types of polarization of dielectrics and classify the magnetic materials.
CO4	Explain fundamentals of quantum mechanics and apply to one dimensional motion of particles.
CO5	Outline the properties of charge carriers in semiconductors

COURSE CONTENT (SYLLABUS)

UNIT-I Wave Optics **12 Hours**

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

Diffraction: Introduction - Fresnel and Fraunhofer diffractions - Fraunhofer diffraction due to single slit- Fraunhofer diffraction due to N Slits -Diffraction Grating - Dispersive power and resolving power of Grating (Qualitative).

Lasers: Introduction–Characteristics of laser – Spontaneous and Stimulated emissions of radiation – Population inversion – Lasing action - Pumping mechanisms – Ruby laser –He-Ne laser-Applications of lasers. **COs – CO1**

Self-Learning Topics: Interference in thin films due to Transmission of light

UNIT-II Crystallography and X-ray diffraction **12 Hours**

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. **COs– CO2**

Self-Learning Topics: Effect of crystallite size on diffracted X-Ray intensity.

UNIT-III Magnetic and Dielectric Materials **12 Hours**

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.

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–dielectric loss. **COs– CO3**

Self-Learning Topics: Frequency dependence of polarization.

UNIT-IV Quantum Mechanics and Free electron theory **12 Hours**

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 - Fermi energy. **COs– CO4**

Self-Learning Topics: Density of states, Origin of energy bands in solids

UNIT-V Semiconductors **8 Hours**

Semiconductors: Formation of energy bands – classification of crystalline solids - Intrinsic semiconductors:– Fermi level – Extrinsic semiconductors- P-Type semiconductors- N-Type semiconductors- principle of operation and characteristics of P-N Junction diode - Drift and diffusion currents –Einstein's equation - Hall effect and its applications. **COs – CO5**

Self-Learning Topics: Zener diode, Solar cells

TEXT BOOKS:

1. "ATextbook of EngineeringPhysics"by M.N. Avadhanulu, P.G.Kshirsagar-S.Chand Publications, 2017.
2. "Engineering Physics" byD. K. Bhattacharya and PoonamTandon,Oxfordpress (2015).
3. "Engineering Physics"byR.KGaur.and S.LGupta.,-Dhanpat Rai publishers, 2012.

REFERENCE BOOKS:

1. Engineering Physics - B.K. Pandey and S. Chaturvedi, Cengage Learning.
2. The Principles of Quantum Mechanics, P. A. M. Dirac, fourth Edition (Oxford University Press,Oxford, 1958).
3. Physics-Resnick, Halliday, Krane, Fifth edison, Volume-1, Wiley student edition.
4. Engineering Physics - Dr.R. Swapna, Scientific International Publishing House.
5. Concepts of Modern Physics. Arthur Beiser, Tata McGraw-Hill, New Delhi (2010).
6. Engineering Physics" - Sanjay D. Jain, D. Sahasrabudhe and Girish, University Press.
7. Engineering Physics - M.R. Srinivasan, New Age international publishers (2009).

ONLINE RESOURCES:

Web References:

1. https://www.ebooksdirectory.com/
2. http://www.sciencedirect.com/Science
3. https://onlinecourses.nptel.ac.in/
4. https://www.link.springer.com/physics/

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

E-BOOKS:

1. https://www.ebooksdirectory.com/

BASIC CIVIL AND MECHANICAL ENGINEERING

IB.TECH-I SEMESTER (Common to CSE, CSE(AIML, CSE(DS))

Course Title: **Basic Civil and Mechanical Engineering**	Course Code: **R24ES01**
TeachingScheme(L:T:P): 3:0:0	Credits:3
Typeof Course:Lecture +Tutorial	
ContinuousInternalEvaluation: 30Marks	SemesterEndExam: 70Marks
Pre requisites: To excel in a *Basic Civil and Mechanical Engineering* course, students should have foundational knowledge and skills in both mathematics and physics. These prerequisites help students understand the principles that underlie civil and mechanical engineering applications.	

COURSEOVERVIEW:

- Basic Civil and Mechanical Engineeringcourse provides a broad foundation for all engineering disciplines, equipping students with a holistic understanding of the principles, design methods, and innovations shaping each branch, fostering interdisciplinary knowledge and skills.
- This introductory course covers fundamental concepts in Civil and Mechanical Engineering, emphasizing their roles in society and diverse applications. In Civil Engineering, students explore various disciplines, including structural, transportation, water resources, and environmental engineering, along with essential materials and construction techniques, surveying, and advancements in sustainable practices.
- The Mechanical Engineering segment introduces students to core sectors such as energy, manufacturing, and automotive, along with basic design principles and engineering materials. Key topics include thermal engineering, power cycles, IC engines, and power plant operations, as well as principles in manufacturing, CNC, 3D printing, and robotics. This course provides a foundational understanding of both fields, preparing students for more specialized study and practical applications in engineering.

COURSEOBJECTIVES:

Theobjectives of thiscourseareto

1. **Understand the Role of Civil Engineers**: Familiarize students with the roles and responsibilities of civil engineers in society and the various sub-disciplines within civil engineering.
2. **Construction Materials**: Provide knowledge about different construction materials such as cement, aggregates, bricks, concrete, steel, soil, stones and their applications in building construction.
3. **Transportation Engineering**: Offer insights into the importance of transportation engineering for national economic development and the fundamentals of highway pavements, harbor, tunnel airport and railway engineering.
4. **Water Resources and Environmental Engineering**: Cover the basics of water sources, water quality specifications, hydrology, rainwater harvesting, and water storage structures, emphasizing their importance in environmental sustainability.
5. **Scope and Importance of Mechanical Engineering**: Familiarize students with the scope and significance of mechanical engineering in various sectors, including energy, manufacturing, automotive, aerospace and marine industries.
6. **Engineering Materials and Manufacturing Processes**: Explain different engineering

materials and various manufacturing processes and computational manufacturing.

7. **Thermal Engineering**: Provide an overview of thermal engineering principles, including the working of boilers, IC engines, and power plants, and introduce concepts related to electric and hybrid vehicles.
8. **Mechanical Power Transmission Systems**: Describe different mechanical power transmission systems such as belt drives, chain drives, gear drives, and their applications.
9. **Basics of Robotics**: Introduce the basics of robotics, including joints, links, configurations, and applications, along with advancements in robotics technology.

COURSEOUTCOMES:

CO#	CourseOutcomes
CO1	Understand the role of civil engineers in various disciplines, the scope of each discipline, and the materials used in building construction and principles of surveying.
CO2	Describe the fundamentals of transportationengineering, waterresources, and environmental engineering, including highway pavements, water quality, hydrology, and water storage structures.
CO3	Understand and apply different manufacturing processes and engineering materials, including theirapplications,andbasic mechanical design principles.
CO4	Explain the basics of thermal engineering, including working principles of engines, power plants, and related thermal cycles, along with their applications.
CO5	Describe the working of different mechanical power transmission systems and the basics of robotics and their applications.

COURSECONTENT(SYLLABUS)

UNIT-I:

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, soils and stones. Introduction to Prefabricated construction Techniques.

Surveying: Objectives of Surveying, Horizontal Measurements, Angular Measurements, Introduction to Bearings Simple problems on bearings-Contour mapping.

Self-Learning Topic: Advancements in Prefabricated Construction Techniques

UNIT-II:

TransportationEngineering: ImportanceofTransportationinNation'seconomicdevelopment, Types of Highway Pavements, Flexible Pavements and Rigid Pavements, Simple Differences. Basics of Harbor, 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).

Self-Learning Topic: Sustainable Transportation Engineering

UNIT-III:

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.

Basic Mechanical Design Principles: Fundamentals of Mechanical Design- Introduction to the design process, understanding design requirements, and conceptual design, Design of Simple Machine Components - Design considerations for basic machine components like shafts, bearings, gears, and fasteners.

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

Self-Learning Topics: Sustainable Engineering Practices, Advancements in Smart Materials.

UNIT-IV:

Thermal Engineering–Working principle of Boilers

Cycles-Ottocycle,Dieselcycle,Refrigerationand air-conditioningcycles,

Engines-ICengines,2-Strokeand4-Strokeengines, SI/CIEngines,

Power plants – Working principle of Steam, Diesel, Hydro, Nuclear power plants, Introduction to Electric and Hybrid Vehicles.

Self-LearningTopics: AdvancedEngineTechnologies,ThermodynamicsinRenewableEnergy Systems.

UNIT-V:

Manufacturing Processes: Principles of Casting,Forming,joiningprocesses,

Computational Manufacturing: Introduction to CNCmachines,3Dprinting and Smart manufacturing.

Machining– Conventional & Non-Conventional,

Mechanical Power Transmission-Belt Drives, Chain, Rope drives, Gear Drives and their applications.

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

Self-Learning Topics: Additive Manufacturing Technologies, Innovations in Mechanical Power Transmission

TEXTBOOKS:

1. Basic Civil and Mechanical Engineering, by Ommi Srikanth,M.Sreenivasa Reddy S. Chand Publications
2. Internal Combustion Engines by V.Ganesan, By Tata McGraw Hill publications (India) Pvt. Ltd.
3. A Text bookofTheoryofMachinesbyS.S.Rattan,TataMcGrawHillPublications, (India) Pvt. Ltd.
4. An introduction to Mechanical Engg by JonathanWicker and Kemper Lewis, cengage learning India pvt. Ltd.

REFERENCEBOOKS:

1. AppuuKuttanKK, Robotics,I.K.InternationalPublishingHousePvt. Ltd. Volume-I
2. 3D printing & Additive Manufacturing Technology- L. Jyothish Kumar, Pulak M Pandey,

Springer publications
3. ThermalEngineeringbyMaheshMRathoreTataMcgrawHillpublications(India)Pvt. Ltd.
4. G.Shanmugam andM.S.Palanisamy,BasicCivilandtheMechanical Engineering,Tata Mcgraw Hill publications (India) Pvt. Ltd.

ONLINERESOURCES:

1. https://www.youtube.com/playlist?list=PLyqSpQzTE6M_SM0Lrnzk2dJFwElh0Ebhu
2. https://nptel.ac.in/courses/105101087
3. https://archive.nptel.ac.in/courses/105/105/105105110/
4. https://archive.nptel.ac.in/courses/112/105/112105125/
5. https://www.youtube.com/watch?v=-cr5vfV4YAI
6. https://nptel.ac.in/courses/112105266
7. https://archive.nptel.ac.in/courses/112/104/112104301/

Problem Solving and Programming with C
I B.TECH- I SEMESTER (Common to all Branches)

Course Title: **Problem Solving and Programming with C**	Course Code: **R24ES02**
Teaching Scheme (L:T:P): 3:0:0	Credits: 3
Type of Course: Lecture + Tutorial	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites: for learning C programming, a strong background in problem-solving skills and an understanding of data structures and algorithms.	

COURSE OVERVIEW:

1. To understand computer programming and its roles in problem solving.
2. To understand and develop well-structured programs using C language.

COURSE OBJECTIVES:

The objectives of this course are to:

1. To impart adequate knowledge on the need of programming languages and problem-solving techniques and develop programming skills.
2. To express algorithms and draw flowcharts in a language independent manner.
3. To enable effective usage of Operators &Control Structures.
4. To learn about the design concept of Arrays, Strings and Functions.
5. To understand Structures and Unions and their usage.
6. To assimilate about Pointers, Dynamic Memory Allocation and know the significance of Pre-processors, perform operations on files.

COURSE OUTCOMES:

CO#	Course Outcomes
CO1	Illustrate the fundamental concepts of computers and basic computer programming and problem-solving approach.
CO2	Understand the Control structures, Branching and Looping.
CO3	Make use of Arrays and Develop Programs on modular programming using functions and strings.
CO4	Demonstrate the ability to write programs using Structures and Unions.
CO5	Apply File handling operations.

COURSE CONTENT (SYLLABUS)

UNIT-I: Introduction to Programming and Algorithm for Problem Solving: 10 Hours

Introduction to Programming: The Basic Model of Computation, Algorithms, Flow-charts, Programming Languages, Compilation, Linking and Loading, Testing and Debugging, Documentation,

Algorithm for Problem Solving: Exchanging values of two variables, summation of a set of numbers, Decimal Base to Binary Base conversion, Reversing digits of an integer, GCD (Greatest Common Division) of two numbers, Test whether a number is prime, Organize numbers in ascending order, Find square root of a number, factorial computation, Fibonacci sequence, Evaluate 'sin x' as sum of a series, Reverse order of elements of an array, Find largest number in an array, Print elements of upper triangular matrix, multiplication of two matrices, Evaluate a Polynomial. **COs–CO1**
Self-Learning Topics: Compilation and Interpretation

UNIT- II: Introduction to the 'C' Programming **15 Hours**

Introduction: Character set, Variables and Identifiers, Built-in Data Types, Input/output statements, Variable Definition, Arithmetic operators and Expressions, Constants and Literals, Simple assignment statement, Basic input/output statement, Type Casting and Type def Simple 'C' programs.Storage Classes**:** Scope and extent, Storage Classes in a single source file: auto, extern and static, register, Storage Classes in multiple source files: extern and static.

Conditional Statements and Loops: Decision making within a program, Conditions, Relational Operators, Logical Connectives, if statement, if-else statement, Loops: while loop, do while, for loop, Nested loops, Infinite loops, Switch statement, Break statement, Go to statement. **COs-CO2**

Self-Learning Topics: Escape Sequences

UNIT – III: Arrays **15 Hours**

Arrays: One dimensional array: Array manipulation; Searching, Insertion, Deletion of an element from an array; Finding the largest/smallest element in an array; two dimensional arrays with examples.

Strings: Concepts, String Types, String Input / Output functions, String manipulation functions, Null terminated strings as array of characters, **COs–CO3**

Self-Learning Topics: String Pattern Matching

UNIT- IV: Functions&Pointers **15 Hours**

Functions: Top-down approach of problem solving, Modular programming and functions, Standard Library of C functions, Prototype of a function: Formal parameter list, Return Type, Function call, Block structure, passing arguments to a Function: call by reference; call by value, Recursive Functions, arrays as function arguments, Standard library string functions..

Pointers: Address operators, pointer type declaration, pointer assignment, pointer initialization, pointer arithmetic, functions and pointers, Arrays and Pointers, pointer arrays, pointers and structures, dynamic memory allocation. **COs– CO4**

Self-Learning Topics: How do you pass a structure to a function?

UNIT-V: Structures and Unions **10 Hours**

Structures and Unions: Structure variables, initialization, structure assignment, nested structure, structures and functions, structures, and arrays: arrays of structures, structures containing arrays, unions, Enumeration.

File Processing: Concept of a file, streams, text files and binary files, Differences between text and binary files, State of a file, Opening and Closing files, file input/output functions (standard library input/output functions for files), file status functions (error handling), Positioning functions **Cos-CO5**
Self-Learning Topics: Binary Files and operations on Binary files

TEXT BOOKS:

1. Byron S Gottfried "Programming with C" Second edition, Tata McGrawhill, 2007 (Paperback)
2. R.G. Dromey, "How to solve it by Computer", Pearson Education, 2008.
3. Kanetkar Y, "Let us C", BPB Publications, 2007.
4. 4. Hanly J R & Koffman E.B, "Problem Solving and Program design in C", Pearson Education, 2009.

REFERENCE BOOKS:

1. E. Balaguruswamy, "Programming with ANSI-C", Fourth Edition,2008, Tata McGraw Hill.
2. Venugopal K. R and Prasad S. R, "Mastering 'C'", Third Edition, 2008, Tata McGraw Hill.
3. B.W. Kernighan & D. M. Ritchie, "The C Programming Language", Second Edition, 2001, Pearson Education
4. ISRD Group, "Programming and Problem-solving Using C", Tata McGraw Hill, 2008.
5. Pradip Dey, Manas Ghosh, "Programming in C", Oxford University Press, 2007.

Web References:

1. http://www.c4learn.com/
2. http://www.geeksforgeeks.org/c/
3. http://nptel.ac.in/courses/122104019/
4. http://www.learn-c.org/
5. https://www.tutorialspoint.com/c programming/

ENGINEERING PHYSICS LAB

I B.TECH- I SEMESTER (Common to CSE,CSE(AI&ML),CSE(DS))

Course Title: **ENGINEERING PHYSICS LAB**	Course Code: **R24BS03**
Teaching Scheme (L:T:P): 0:0:2	Credits: 1
Type of Course: Practical	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites: The *Engineering Physics Lab* is designed to provide students with hands-on experience and practical knowledge of physics concepts applied in engineering. Through experiments and observations, students can bridge theoretical physics principles with real-world applications, enhancing their understanding of experimental physics and analytical skills.	

COURSE OVERVIEW:

To succeed in an *Engineering Physics Lab* course, certain foundational skills and knowledge are necessary for effective participation and understanding. Here are the key prerequisites:

1. Basic Physics Knowledge
2. Mathematics Skills
3. Measurement and Unit Conversions
4. Basic Laboratory Skills
5. Problem-Solving and Analytical Skills
6. Familiarity with Safety Practices
7. Basic Computing Skills

COURSE OBJECTIVES:

1. To study the concepts of optical phenomenon like interference, diffraction etc.,
2. To recognize the importance of energy gap in the study of conductivity and Hall effect in semiconductors
3. To study the parameters and applications of dielectric and magnetic materials by conducting experiments.

COURSE OUTCOMES:

CO#	Course Outcomes
CO1	Demonstrate the modern engineering physics Techniques and tools in real times applications in engineering studies.
CO2	Develop the laboratory skills in handling of electrical and optical instruments.
CO3	Conduct experiment Independently and In team to record the measurements
CO4	Compare the experimental results with standard values and estimate errors

COURSE CONTENT (SYLLABUS)

List of Experiments

1. Determination of radius of curvature of a given plano convex lens by Newton’sring;s method.
2. Determination of wavelengths of different spectral lines in mercury spectrum using diffraction grating in normal incidence configuration.

3. Determination of thickness of thin object by air wedge method
4. Determination of wavelength of Laser Source by diffraction gratting.
5. Determination of rigidity modulus of the material of the given wire using Torsional pendulum.
6. Magnetic field along the axis of a current carrying circular coil by Stewart & Gee's Method.
7. Determination of dispersive power of the prism.
8. Determination of acceleration due to gravity and radius of Gyration by using Compound pendulum.
9. Determination of energy gap of a semiconductor using p-n junction diode.
10. Determination of dielectric constant using charging an discharging method.
11. Sonometer: Verification of laws of stretched string.
12. Estimation of Planck's constant using photoelectric effect.
13. Study the variation of B versus H by magnetization the magnetic material (B-H curve)
14. Determination of frequency of electrically maintained tunning fork by Melde's experiment.
15. Determination of Hall voltage and Hall coefficient of a given semiconductor using Hall effect.
16. Determination of the resistivity of semiconductor by four probe method.
17. Determination of young's modulus for the given material of wooden scale by non- uniform bending (or double cantilever) method .
18. Determination of magnetic suseptibility by Kundt's tube method

REFERENCE BOOKS:

1. S. Balasubramanian, M.N.Srinivasan "A Text Book of Practical Physics"-S ChandPublishers,2017.
2 .J.Raja Gopalam Patnaik, "Physics Laboratory Manual for Undergraduate Students "Paramount Book Disributors 2023.

ONLINE RESOURCES:
Web References:

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

COMMUNICATIVE ENGLISH LAB
I B.TECH- I SEMESTER (Common to CSE, CSE (AIML), CSE (DS))

Course Title: **COMMUNICATIVE ENGLISH LAB**	Course Code: **R24HS02**
Teaching Scheme (L:T:P): **0 0 2**	Credits: 1
Type of Course: Practical	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites: To excel in a *Communicative English* course, certain foundational skills and prerequisites are helpful such as Basic Grammar Knowledge, **Listening Skills,** Basic Vocabulary, Reading Comprehension, Confidence in Speaking, Writing Skills.	

COURSE OVERVIEW:

A *Communicative English* course is designed to develop students' proficiency in spoken and written English through practical and interactive learning methods. The course focuses on improving students' ability to communicate effectively in real-life situations, emphasizing both fluency and accuracy.

COURSE OBJECTIVES:

The objectives of this course are to:

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. Students undergo training in basic communication skills to make them into confident communicators in all situations.

COURSE OUTCOMES:

CO#	Course Outcomes
CO1	Understand and recognize the various facets of English language ability with a focus on the four basic skills- namely -LSRW abilities.
CO2	Implement various activities for language learners to practise communication skills.
CO3	To enhance listening and speaking comprehension, analyze the sounds, stress, rhythm, intonation, and syllable division of English speech.
CO4	Assess the professionalism of students when taking part in group discussions, debates, JAM sessions, Presentations and Interviews.
CO5	Equipping oneself with Interview Skills and a range of Soft Skills for life and career.

COURSE CONTENT (SYLLABUS)

Week1:

1. To explain and guide the students in decoding the sounds of English.
2. List all the consonant sounds and vowel sounds in English

Week2:

1. What is a syllable and describe the syllable structure.
2. Define stress, functional stress and various rules of stress.

3. What is connected speech?

Week3:

1. What is Intonation and mention the various pitch movements like rise, fall, fall-rise or rise-fall?
2. What is connected speech?

Week4:

1. To equip students to speak in English language confidently without any inhibitions.
2. Why are majority of the companies conducting JAM session as a preliminary interview?
3. What are the key skills tested in JAM round?

Week5:

1. To help students learn and understand different functions of language like greeting, asking
2. For information, giving information, meetings, requests, exchanging dialogues in formal and informal contexts.
3. Introduce yourself and others, give instructions and directions

Week 6:

1. To help the students understand and work on the digital age connector for personal correspondence, business communication, etc.
2. Write about email etiquette.
3. Draft an email to the HR Manager of Wipro Technologies requesting to consider your application for the post of Software Engineer.

Week 7:

1. To update students about the importance of Resume, the various types and the essentials of an effective resume
2. Draft a resume for a software post in reputed organization.

Week 8:

1. To educate students about the various styles of writing formal letters.
2. What is a cover letter? What are the different types of cover letters?
3. Write a job application letter for any post of your choice in a reputed company?

Week 9:

1. To help students know the importance of an SOP in their professional advancements?
2. What is an SOP and what are the different kinds and parts of an SOP?
3. Prepare an SOP to apply for a Master's Programme in any University of your choice.

Week 10:

1. To educate and guide the students about presentation skills and its importance in the technical evolving world.
2. To inform explain students about the importance of body language in various personal and professional forums
3. To help students to present papers, PPT's in seminars, workshops, conferences, research projects, interviews, etc.

Week 11:

1. To help students to give effective PPT's in various academic and professional platforms.
2. Describe various aspects that make PPT more effective.

3. Make a PPT on any topic of your choice and present it to the class.

Week 12:

1. To foster, creative, critical thinking skills, analytical skills and problem solving skills.
2. Suggest a few tips for preparing a poster.
3. Prepare posters from or outside your curriculum.

List of Activities:

1.	Sounds of English (Vowels and Consonants)	**COs: CO1,CO2**
2.	Neutralization and Accent Rules	**COs: CO1,CO2**
3.	Improving communication skills /JAM.	**COs: CO3,CO4**
4.	Letter Writing and E-mail Writing	**COs: CO1,CO2**
5.	Cover letters and Resume Writing	**COs: CO1,CO2**
6.	Statement of Purpose.	**COs: CO1,CO2**
7.	Debates	**COs: CO4:CO5**
8.	Presentation skills- PPT and Poster	**COs: CO4:CO5**
9.	Group Discussions , types and practice	**COs: CO4,CO5**
10.	Interview skills – Mock interviews	**COs: CO4,CO5**

Reference Books:

1. Prof. M. Hari Prasad, Prof. Vijaya Babu, Prof. Padmaja Kalapala, Skill Craft – A Communicative English Laboratory Workbook, Maruthi Publications first Edition, 2023
2. Meenakshi Ramana, Sangeeta-Sharma, 4thEdition, Technical Communication, Oxford Press, 2022.
3. Grant Taylor: English Conversation Practice, 1st Edition, Tata ,Mc Graw-Hill Education India, 2001.
4. Hewing,s, Martin, Cambridge Academic English(B2), Cambridge University Press,2012.
5. T. Balasubramanyam, A Textbook of English Phonetics for Indian Students, 3rd Edition, Trinity, 2022.
6. Dr. ShaliniSharma's Body Language Your Success Mantra, S. Chand publications 2010.
7. Sunitha Mishra and C.Murali Krishna's Communication Skills for Engineers Pearson Education Edition 2009.

Suggested software:

- English Wordsworth –Language Lab- Wordsworth Software

Web References for:

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/ArnelsEverydayEnglish/featured
7. https://www.youtube.com/c/engvidAdam/featured
8. . https://www.youtube.com/c/EnglishClass101/featured
9. https://www.ted.com/watch/ted-ed
10. http://www.edest.org/

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

Problem Solving and Programming with C Lab
I B.TECH- I SEMESTER (Common to all Branches)

Course Title: **Problem Solving and Programming with C Lab**	Course Code: **R24ES03**
Teaching Scheme (L:T:P): **0 0 3**	Credits: 1.5
Type of Course:Practical	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites: **Problem Solving & Programming with C lab include: Understanding programming fundamentals,** Writing C programs, Applying programming techniques, Using algorithms, Using pseudocode and flowcharts	

COURSE OVERVIEW:

1. To understand computer programming and its roles in problem solving.
2. To understand and develop well-structured programs using C language.

COURSE OBJECTIVES:

The objectives of this course are to:

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

COURSE OUTCOMES:

CO#	Course Outcomes
CO1	Read, understand, and trace the execution of programs written in C language.
CO2	Select the right control structure for solving the problems .and demonstrate the application of arrays functions and strings
CO3	Develop Debug and Execute programs to demonstrate the applications of Pointers, Structures& Unions, and Files.

COURSE CONTENT (SYLLABUS)

Developing the following programs:

Week 1:

1. Write a C program using printf() and Scanf(). COs:CO1
2. Write a C program on swapping of two nos. COs:CO1
3. Write a C program using arithmetic Expressions. COs:CO1

Week 2:

4. Simple interest calculation COs:CO2
5. Finding compound interest COs:CO2
6. Area of a triangle using heron's formulae COs:CO2
7. Distance travelled by an object COs:CO2

Week 3:

8. Find the maximum of three numbers using conditional operator COs:CO2

9. Take marks of 5 subjects in integers, and find the total, average in float COs:CO2
10. Write a C program to shift/rotate using bit fields. COs:CO2
11. Finding the square root of a given number COs:CO2
12. Write a C program using if-else statement. COs:CO2

Week 4:

13. Write a C program to find the max and min of four numbers using if-else. COs:CO2
14. Write a C program to generate electricity bill. COs:CO2
15. Find the roots of the quadratic equation. ` COs:CO2
16. Write a C program to find the given year is a leap year or not. COs:CO2
17. Write a C program to simulate a calculator using switch case. COs:CO2

Week 5:

18. Find the factorial of given number using any loop. COs:CO2
19. Find the given number is a prime or not. COs:CO2
20. Compute sine and cos series. COs:CO2
21. Checking a number palindrome. COs:CO2
22. Construct a pyramid of numbers. COs:CO2

Week 6:

23. Write a C program on Linear Search. COs:CO3
24. Find the min and max of a 1-D integer array. COs:CO3
25. Perform linear search on1D array. COs:CO3
26. The reverse of a 1D integer array. COs:CO3

Week 7:

27. Find 2's complement of the given binary number. COs:CO3
28. Eliminate duplicate elements in an array. COs:CO3
29. Sort array elements using bubble sort. COs:CO3
30. Addition of two matrices. COs:CO3

Week 8:

31. Multiplication two matrices. COs:CO3
32. Write a C program using call by reference. COs:CO3
33. Write a C program to find factorial of n using recursion. COs:CO3
34. Write a C function to calculate NCR value COs:CO3
35. Concatenate two strings without built-in functions. COs:CO3

Week 9:

36. Write a C function to transpose of a matrix. COs:CO3
37. Write a C function to find the length of a string. COs:CO3
38. Reverse a string using built-in and without built-in string functions. COs:CO3
39. Write a C program to find the sum of a 1D array using malloc ( ). COs:CO3

Week 10:

40. Write a recursive function to find the lcm of two numbers. COs:CO3
41. Write a recursive function to find the sum of series. COs:CO3
42. Write a C program to swap two numbers using call by reference. COs:CO3
43. Write a C program using Pointers, Structures and Unions. COs:CO4
44. Write a C program to find the total, average of n students using structures. COs:CO4

Week 11:

45. Enter n students data using calloc() and display failed students list. COs:CO4
46. Read student name and marks from the command line and display the student details along with the total. COs:CO4
47. Write a C program to implement realloc(). COs:CO4
48. Write a C program to copy one structure variable to another structure of the same type. COs: CO4

Week 12:

49. Demonstrate Dangling pointer problem using a C program. COs: CO4
50. Write a C program to copy one string into another using pointer. COs: CO4
51. Write a C program to find no of lowercase, uppercase, digits and other characters using pointers. COs: CO4

Week 13:

52. Write a C program using Files operations. COs:CO5
    a. Sum and average of 3 numbers
    b. Conversion of Fahrenheit to Celsius and vice versa.
53. Write a C program to write and read text into a file. COs:CO5

Week 14:

54. Write a C program to write and read text into a binary file using fread() and fwrite() COs:CO5
55. Copy the contents of one file to another file. COs:CO5
56. Write a C program to merge two files into the third file using command-line arguments. COs:CO5

Week 15:

57. Find no. of lines, words and characters in a file. COs:CO5
58. Write a C program to print last n characters of a given file. COs:CO5

Scenario Based Case Study:

1. **Objective:** To develop a simple utility program to calculate the area of geometric shapes.

Target Audience: Beginner programmers and students.

A basic utility program in C that can:

1. Calculate the area of a circle.

2. Calculate the area of a rectangle.

3. Calculate the area of a triangle.

The program should be easy to understand and serve as an educational tool for new programmers.

1. Program Design:

The utility program will be designed with a simple menu-driven interface allowing the user to select the shape for which they want to calculate the area. The program will then prompt the user to input the necessary dimensions and display the result.

This basic C program demonstrates fundamental programming concepts such as variables, functions, and control structures in a practical context. It provides a clear introduction to C

programming for beginners by solving a simple problem using these core concepts.

This case study outlines a straightforward approach to teaching and implementing basic C programming concepts effectively.

2. Students Marks Sum HackerRank Solution

You are given an array of integers, marks, denoting the marks scored by students in a class.

- The alternating elements marks0, marks2, marks4 and so on denote the marks of boys.
- Similarly,marks1 , marks3, marks5 and so on denote the marks of girls.

The array name, marks, works as a pointer which stores the base address of that array. In other words, marks contains the address where marks0 is stored in the memory.

3. Sorting Array of Strings HackerRank Solution

To sort a given array of strings into lexicographically increasing order or into an order in which the string with the lowest length appears first, a sorting function with a flag indicating the type of comparison strategy can be written. The disadvantage with doing so has to rewrite the function for every new comparison strategy.

A better implementation would be to write a sorting function that accepts a pointer to the function that compares each pair of strings. Doing this will mean only passing a pointer to the sorting function with every new comparison strategy.

Textbooks:

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

Reference Books:

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

IT Workshop

I B.TECH- II SEMESTER (Common to CSE,CSE(AIML),CSE(DS))

Course Title: **IT Workshop**	Course Code: **R24ES04**
Teaching Scheme (L:T:P): **0 0 2**	Credits: 1
Type of Course: Practical	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites: an IT workshop for engineering students: **Familiarity with hand tools, equipment, and machines, Computer skills.**	

COURSE OVERVIEW:

1. To understand computer programming and its roles in problem solving.
2. To understand and develop well-structured programs using C language.

COURSE OBJECTIVES:

The objectives of this course are to:

1. To assemble and disassemble a computer.
2. To solve hardware and software problems.
3. To learn about Networking of computers and use Internet facility for Browsing and Searching.
4. To develop project documentation using MS word
5. To work with various productivity tools including Excel, PowerPoint.
6. To work with different online repositories such as GITHUB, AI CHATBOT.

COURSE OUTCOMES:

CO#	Course Outcomes
CO1	Perform Hardware troubleshooting and Perform Hardware troubleshooting
CO2	Apply different way of hooking the PC on to the internet from home and Workplace.
CO3	Design word documents by learning word processing and Create presentations by using different styles and using AI Tools-Chat GPT and GITHUB

COURSE CONTENT (SYLLABUS)

PC Hardware & Software Installation **9 Hours**

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 it 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. Lab instructor should verify the installation and follow it up with a Viva. **COs-CO1**

Internet &World Wide Web **6 Hours**

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. **COs-CO2**

MS WORD **6 Hours**

Task 1: 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 2: 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 **6 Hours**

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, and 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

POWER POINT **6 Hours**

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.**Cos-CO3**

AI TOOLS – Chat GPT **6 Hours**

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."

Explore – GITHUB **6 Hours**

Task 1: Students should understand GITHUB and should possess accounts in it.
Task 2: Students should explore different repositories available in GITHUB and student should Create his/ her own simple repositories.
Task 3: Students should take simple experiments /presentations and upload them in their GITHUB Account.
Task 4: Students should understand how GITHUB Enterprise Cloud is used and also explore the GIT and GIT HUB resources. **COs-CO3**

Reference Books:

1. Comdex Information Technology course tool kit, Vikas Gupta, WILEY Dream tech, 2003
2. The Complete Computer upgrade and repair book, Cheryl A Schmidt, WILEY Dream tech, 2013, 3rd edition
3. Introduction to Information Technology, ITL Education Solutions limited, Pearson Education, 2012, 2nd edition
4. PC Hardware - A Handbook, Kate J. Chase, PHI (Microsoft)
5. IT Essentials PC Hardware and Software Companion Guide, David Anfins on and Ken Quamme. – CISCO Press, Pearson Education, 3rd edition
6. IT Essentials PC Hardware and Software Labs and Study Guide, Patrick Regan– CISCOPress, Pearson Education, 3rd edition
7. "Microsoft Word 2021: A Beginner’s Guide"by Steve Lambert.
8. "Excel 2021: A Comprehensive Guide"by Chris Benham.
9. "Microsoft PowerPoint 2021: A Beginner’s Guide" by Steve Lambert
10. GITHUB Quick Start Tutorials

HEALTH AND WELLNESS, YOGA AND SPORTS
I B.TECH- I SEMESTER (Common to all Branches)

Course Title: **HEALTH AND WELLNESS, YOGA AND SPORTS**	Course Code: **R24MC01**
Teaching Scheme (L:T:P): 0:0:1 Type of Course: Practical	Credits: 0.5
Continuous Internal Evaluation: 100 Marks	Semester End Exam: 0 Marks
Pre requisites: Prerequisites for courses in **Health and Wellness, Yoga, and Sports** can vary by institution and program. However, here's a general outline of common prerequisites and recommended qualifications for these types of courses.	

COURSE OVERVIEW:

Here's a general course overview for programs in **Health and Wellness**, **Yoga**, and **Sports**. Each area may have specific courses and focuses depending on the institution, but this will provide a foundational understanding of what to expect.

Health and Wellness

- **Overview**: This program focuses on promoting overall health, wellness strategies, and preventive health measures. It often includes the study of physical, mental, and social well-being.
- **Core Courses**:
  - **Introduction to Health and Wellness**: Basics of health concepts, wellness promotion, and lifestyle choices.
  - **Nutrition and Health**: Understanding dietary needs, nutritional guidelines, and the role of nutrition in health.
  - **Mental Health and Wellness**: Exploring psychological well-being, stress management, and mental health issues.
  - **Exercise Physiology**: The study of how physical activity affects the body and mind.
  - **Health Education and Promotion**: Strategies for promoting health within communities and organizations.
- **Practical Experience**: Some programs may include internships or fieldwork in health settings, community organizations, or wellness programs.

2. Yoga

- **Overview**: Yoga programs typically cover the physical, mental, and spiritual aspects of yoga practice. They can be geared toward practitioners or those looking to teach.
- **Core Courses**:
  - **Yoga Philosophy**: Study of the history and philosophy of yoga, including key texts and principles.
  - **Asana Practice**: Detailed exploration of yoga postures (asanas), including alignment, modifications, and variations.
  - **Meditation and Pranayama**: Techniques for breath control and meditation practices.
  - **Anatomy for Yoga**: Understanding the human body in relation to yoga practice, focusing on anatomy and physiology.
  - **Teaching Methodology**: Instruction on how to teach yoga classes, including class planning, communication skills, and sequencing.
- **Practical Experience**: Teaching practice sessions, observation of experienced teachers, and peer teaching.

3. Sports

- **Overview**: Sports programs often encompass a broad understanding of physical education, sports science, coaching, and athletic training.
- **Core Courses**:
  - **Introduction to Sports Science**: Overview of the key principles in sports science, including biomechanics and exercise physiology.
  - **Coaching Principles**: Theories and techniques related to effective coaching and athlete development.
  - **Sport Psychology**: Understanding the mental aspects of sports performance and strategies for enhancing motivation and focus.
  - **Exercise and Sport Nutrition**: Nutrition principles specifically tailored for athletes and active individuals.
  - **Sport Management**: Insights into the business side of sports, including marketing, finance, and event management.

COURSE OBJECTIVES:

The objectives of this course are to:

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:

CO#	Course Outcomes
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

COURSE CONTENT (SYLLABUS)

UNIT I

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

Activities:

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

UNIT II

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

Activities:

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

UNIT III

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

Activities:

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

Reference Books:

1. Gordon Edlin, Eric Golanty. Health and Wellness, 14th Edn. Jones & Bartlett Learning, 2022

2. T.K.V.Desikachar. The Heart of Yoga: Developing a Personal Practice

3. Archie J.Bahm. Yoga Sutras of Patanjali, Jain Publishing Company, 1993

4. Wiseman, John Lofty, SAS Survival Handbook: The Ultimate Guide to Surviving Anywhere Third Edition, William Morrow Paperbacks, 2014

5. The Sports Rules Book/ Human Kinetics with Thomas Hanlon. -- 3rd ed. Human Kinetics, Inc.2014

General Guidelines:

1. Institutes must assign slots in the Timetable for the activities of Health/Sports/Yoga.

2. Institutes must provide field/facility and offer the minimum of five choices of as many as Games/Sports.

3. Institutes are required to provide sports instructor / yoga teacher to mentor the students.

Differential Equations and Vector Calculus
IB.TECH-IISEMESTER(Common to all Branches)

Course Title: **Differential Equations and Vector Calculus**	Course Code: **R24BS04**
Teaching Scheme(L:T:P): 3:0:0	Credits:3
Type of Course: Lecture +Tutorial	
Continuous Internal Evaluation: 30Marks	Semester EndExam: 70Marks
Pre requisites:To succeed in Differential Equations and Vector Calculus, you'll need a strong foundation in several key areas of mathematics. Here are the typical prerequisites: Calculus I (Single-variable Calculus), Calculus II (Single-variable Calculus, continuation), Calculus III (Multivariable Calculus), Linear Algebra.	

COURSEOVERVIEW:

This course is often taken after completing Calculus I, II, and III, and Linear Algebra. It combines methods and applications of differential equations with essential topics in vector calculus, as used in fields like physics, engineering, and applied mathematics.

COURSEOBJECTIVES:

Theobjectives of thiscourseareto:

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

COURSEOUTCOMES:

CO#	CourseOutcomes
CO1	Solve the first order differential equations related to various engineering fields.
CO2	Model engineering problems as higher order differential equations and solve analytically.
CO3	Identify solution methods for partial differential equations that model physical processes.
CO4	Interpret the physical meaning of different operators such as gradient, curl and divergence.
CO5	Estimate the work done against a field, circulation and flux using vector calculus.

COURSECONTENT(SYLLABUS)

UNIT- I: Differential equations of first order and first degree **10 Hours**

Formation of differential equations, order, degree, separation of variables (only Review). 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 (RL and LC). **COs-CO1**

Self-Learning Topic: Mixed tank problems

UNIT- II: Higher order Linear differential equations with Constant Coefficients **10 Hours**

Definitions, homogenous and non-homogenous, complimentary function, particular integral $\left(e^{ax}, \sin ax, \cos ax, \text{Polynomial in } x, e^{ax}V(x), xV(x)\right)$, general solution, Wronskian, method of variation of parameters.

Applications: L-C-R Circuit problems **COs-CO2** Self-Learning Topic: Simple Harmonic motion

UNIT–III: Partial Differential Equations **10 Hours**

Introduction and formation of Partial Differential Equations by elimination of arbitrary constants and arbitrary functions, solution of first order linear equations using Lagrange's method. Homogenous Linear Partial differential equations with constant coefficients. **COs-CO3**

Self-Learning Topic: Method of Separation of Variables

UNIT- IV: Vector differentiation **10 Hours**

Vector, Scalar, dot product, cross product, unit vector, equation of a line passing through two points (Review only)

Scalar and vector point functions, vector operator del, del applies to scalar point function-Gradient, del applied to vector point function – Divergence and Curl, Vector Identities

Application: Scalar Potential **COs-CO4**

Self-Learning Topic: Equation of tangent plane and Normal plane.

UNIT-V: Vector integration **10 Hour**

Line integral – 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).

COs-CO5

Self-Learning Topic: Application of above theorems.

Text Books:

1. B.S.Grewal, Higher Engineering Mathematics, 44/e, Khanna publishers, 2017.
2. Erwin Kreyszig, Advanced Engineering Mathematics, 10/e, John Wiley & Sons, 2018.

Reference Books:

1. Dennis G. Zill and Warren S. Wright, Advanced Engineering Mathematics, Jones and Bartlett, 2018.
2. Michael Green berg, Advanced Engineering Mathematics, 9th edition, Pearson edn
3. George B. Thomas, Maurice D. Weir and Joel Hass, Thomas Calculus, 14/e, Pearson Publishers, 2018.
4. R. K. Jain and S. R. K. Iyengar, Advanced Engineering Mathematics, 5/e, Alpha Science
5. International Ltd., 2021 (9th reprint).
6. B.V. Ramana, Higher Engineering Mathematics, McGraw Hill Education, 2017.

Web References:

1. http://onlinecourses.nptel.ac.in
2. https://nptel.ac.in/courses/111105121
3. https://onlinecourses.nptel.ac.in/noc24_ma86/course

APPLIED CHEMISTRY

IB.TECH- II SEMESTER(Common to EEE, ECE, CSE, CSE(AIML), CSE(DS))

Course Title: **APPLIED CHEMISTRY**	Course Code: **R24BS05**
Teaching Scheme(L:T:P): **3 0 0**	Credits:3
Type of Course :Lecture +Tutorial	
Continuous Internal Evaluation: 30Marks	Semester End Exam: 70Marks
Pre requisites:	

COURSEOVERVIEW:

1. To understand computer programming and its roles in problem solving.
2. To understand and develop well-structured programs using C language.

COURSEOBJECTIVES:

Theobjectives of thiscourseareto:

1. To familiarize Applied Chemistry and its application.
2. To train the students on the principles and applications of electrochemistry and polymers
3. To elucidate the Structure and bonding of molecules
4. To impart Basic concepts of Semiconductors
5. To introduce modern engineering materials
6. To introduce instrumental methods, chromatographic technique

COURSEOUTCOMES:

CO#	CourseOutcomes
CO1	Describe Molecular orbital diagrams
CO2	Importance of Graphenes. Apply the principle of band diagrams in the application of Super conductors and semiconductors.
CO3	Compare the Materials of Construction for Battery and Electro Chemical Sensors.
CO4	Explain the Preparation, Properties and applications of thermos plastics and thermos setting plastics, Elastomers and conducting polymers.
CO5	Summarize the concepts of instrumental methods

COURSECONTENT(SYLLABUS)

UNIT- I:Structure and Bonding models **10 hours**

Fundamentals of Quantum mechanics, Schrodinger Wave equation, significance of Ψ and Ψ^2,particle in one dimensional box, molecular orbital theory–bonding in homo-and hetero nuclear diatomic molecules – energy level diagrams of O2 and CO, etc. π-molecular orbitals of butadiene and Benzene, alculation of bond order. **COs-CO1**

Self-Learning Topics: Energy Level Diagrams of N2 & CN Molecule.

UNIT- II: SeriesModern Engineering materials **10 hours**

Semiconductors – Introduction, Classification semiconductor devices P-N junction diode as a rectifier and transistor, applications. Super conductors-Introduction basic concept, Classification,

applications.
Super capacitors:Introduction,BasicConcept-Classification–Applications.
Nanomaterial: Introduction, classification, properties and applications of Fullerenes, carbon nano tubes and Graphenes **COs-CO2**

Self Learning Topics: Band Theory of Solids, Sol-Gel method.

TEXT BOOKS:

1. 'Data Structures and Algorithm Analysis in C' by Mark Allen Weiss, Pearson.
2. 'Introduction to Algorithms' by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, MIT Press.
3. 'Data Structures Using C' by Reema Thareja, Oxford University Press.

Reference Books:

1. Algorithms, Part I and II' by Robert Sedgewick and Kevin Wayne, Addison-Wesley.
2. Data Structures and Algorithms Made Easy' by Narasimha Karumanchi, CareerMonk Publications

Web References:

1. http://www.hackerrank.com/domains/datastructures
2. http://www.github.com/topics/data_structures_c
3. http://nptel.ac.in/courses
4. http://www.cslibrary.stanford.edu

Basic Electrical and Electronics Engineering
I B.TECH- II SEMESTER (Common to CSE, CSE(AIML), CSE(DS))

Course Title: **Basic Electrical and Electronics Engineering**	Course Code: **R24ES05**
Teaching Scheme(L:T:P): 3:0:0	Credits:3
Type of Course: Lecture	
Continuous Internal Evaluation: 30Marks	Semester End Exam: 70Marks
Pre requisites: Solid state physics, Linear algebra, calculus.	

COURSEOVERVIEW:

- This course introduces to the concepts and definitions of Ohms law, KCL, KVL, power and energy. By applying Kirchhoff's current and voltage laws to circuits in order to determine voltage, current and power in branches of any circuits excited by DC voltages and current sources. Apply simplifying techniques to solve DC circuit problems using basic circuit theorems and structured methods like node voltage and mesh current analysis.This course also introduces the construction and operating principle of AC machines, DC machines, Generators and Transformers.
- This course explores the evolution of electronics, characteristics of PN junction and Zener diodes, and bipolar junction transistors in various configurations. It includes rectifiers, power supplies, and amplifiers, focusing on circuit diagrams and frequency responses and covers number systems, Boolean algebra, and logic gates, along with simple combinational circuits like adders. It also introduces sequential circuits, including flip-flops and counters, and concludes with a block diagram of an electronic instrumentation system.

COURSEOBJECTIVES:

The objectives of this course are to

1. To expose to the field of electrical & electronics engineering.
2. To understand the importance of electrical safety.
3. To teach the fundamentals of semiconductor devices and its applications.
4. To teach the working process and analysis of different rectifying and Amplifying Circuits.
5. To teach the fundamental principles and rules of digital electronic circuits like gates, Sequential and Combinational Circuits.

COURSEOUTCOMES:

CO#	Course Outcomes
CO1	Understand the problem-solving concepts associated to AC and DC circuits
CO2	Remember the fundamental laws, construction and operation of AC and DC machines, instruments.
CO3	Understand different power generation mechanisms, Electricity billing concept and important safety measures related to electrical operations.
CO4	Understand the fundamental principles of electronic devices, analyzing the different rectifying and Amplifying Circuits.
CO5	Analyze and design different digital electronic circuits like gates, Sequential and Combinational Circuits and Understand the basic Electronic instrumentation system.

COURSECONTENT(SYLLABUS)

Part A-BASIC ELECTRICAL ENGINEERING

UNIT-I: DC & AC Circuits

DC Circuits: Electrical circuit elements (R, Land 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).

COs – CO1

Self-Learning Topics: Source Transformation

UNIT-II: Machines and Measuring Instruments

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

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

COs – CO2

Self-Learning Topics: Magnetic materials.

UNIT-III: Energy Resources, Electricity Bill & Safety Measures

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

Electricity bill: Power rating of house hold 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.

COs– CO3

Self-Learning Topics: Different types of electrical tools.

Part B:ELECTRONICSENGINEERING

UNIT-IV:SemiconductorDevices and BasicElectronicCircuits

Introduction - Evolution of electronics – Vacuum tubes to nano electronics - Characteristics of PN Junction Diode — Breakdown Effects in diodes — Zener Diode and its Characteristics. Bipolar Junction Transistor — CB, CE, CC Configurations and Characteristics — Introduction to Small Signal CE configuration. Rectifiersandpowersupplies:Blockdiagramdescriptionofadcpowersupply,Half-WaveRectifiers, Full-Wave Rectifiers, capacitor filter (no analysis). Amplifiers: Block diagram of PublicAddress system, Circuit diagram and working of common emitter (RC coupled) amplifier with itsfrequency response.

COs–CO4

Self-LearningTopics:Electroniccomponentsandcharacteristics,

DesignAmplifiercircuitatdifferentR,CValues

UNIT-V:DIGITALELECTRONICS and INSTRUMENTTAION

Overview of Number Systems, BCD codes, Excess-3 code, Gray code, Hamming code. BooleanAlgebra, Basic Theorems and properties of Boolean Algebra, Truth Tables and Functionality ofLogic Gates – NOT, OR, AND, NOR, NAND, XOR and XNOR. Simple combinational circuits– HalfandFullAdders.Introductiontosequentialcircuits,Flipflops,Registersandcounters(ElementaryTreatmentonly), Electronic Instrumentation: Block diagram of an electronic instrumentationsystem

COs–CO5

Self-LearningTopics:Developdigitalcircuitsusingminimumno. ofgates, design principles of electronic instruments.

TEXTBOOKS:

1. Basic Electrical Engineering, D.C.Kulshreshtha,Tata McGrawHill,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
4. R.L.Boylestad&LouisNashlesky,ElectronicDevices&CircuitTheory,PearsonEducation,2021.
5. R.P.Jain,ModernDigitalElectronics, 4thEdition, TataMcGrawHill, 2009

REFERENCE BOOKS :(Basic Electrical Engineering)

1. Basic Electrical Engineering, D.P.Kothari and I.J.Nagrath, McGrawHill, 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, Pearson Publications,2018, Second Edition.

REFERENCEBOOKS:ElectronicsEngineering

1. R.S.Sedha,ATextbookofElectronicDevicesandCircuits,S.Chand&Co,2010.
2. SantiramKal,BasicElectronics-Devices,CircuitsandITFundamentals,PrenticeHall,India,2002.
3. R.T.Paynter, Introductory Electronic Devices & Circuits – Conventional FlowVersion,Pearson

ONLINERESOURCES:

Web References: (Basic Electrical Engineering)

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

Web References: (Electronics Engineering)

1. https://archive.nptel.ac.in/courses/108/101/108101091/
2. https://www.tutorialspoint.com/basic_electronics/index.htm
3. https://www.tutorialspoint.com/digital_circuits/index.htm

E-BOOKS:

https://www.pdfdrive.com/basic-electronics-for-scientists-and-engineers-e28939124.html

Engineering Graphics

IB.TECH-IISEMESTER(Common to CSE, CSE(AIML, CSE(DS))

Course Title : **Engineering Graphics**	Course Code : **R24ES06**
Teaching Scheme(L:T:P): 1:0:4	Credits:3
Type of Course: Lecture +Practical	
Continuous Internal Evaluation:30Marks	Semester End Exam: 70Marks
Prerequisites:	

COURSEOVERVIEW:

- The Engineering Graphics and Drawing course provides essential skills in visualizing and representing three-dimensional objects on two-dimensional media. Through structured units, students learn fundamentals such as line work, lettering, dimensioning, and geometric construction. The syllabus includes constructing curves, understanding scales, and mastering orthographic and isometric projections, essential for accurately depicting objects in engineering design.
- Additionally, students gain experience with computer-aided drafting using AutoCAD, learning to create 2D and 3D drawings and perform basic transformations. This course provides a strong foundation for technical drawing, crucial for design, manufacturing, and communication in engineering.

COURSEOBJECTIVES:

The objectives of this course are to

1. Understand the fundamentals of engineering drawing, including lines, lettering, and dimensioning.
2. Develop skills in geometrical constructions, including regular polygons and curves.
3. Learn orthographic projection techniques, including projections of points, lines, and planes.
4. Understand how to project solids in simple positions and create sectional views.
5. Develop skills in converting isometric views to orthographic views and vice versa.
6. Apply computer-aided design (CAD) techniques using AutoCAD to create 2D and 3D drawings.
7. Understand the importance of reference planes and reference lines in orthographic projection.
8. Develop problem-solving skills in engineering drawing, including creating and interpreting drawings.

COURSEOUTCOMES:

CO#	Course Outcomes
CO1	Understand the basics of Engineering Graphics to construct the polygon, curves, and scales.
CO2	Draw the orthographic projections of points and straight lines inclined to both the planes.
CO3	Draw the projections of planes in various conditions.
CO4	Draw the projections of regular solids, with its axis inclined to one plane and sections of solids.
CO5	Visualize the 3D isometric views from 2D orthographic views and vice versa along with basic introduction to CAD.

COURSE CONTENT (SYLLABUS)

UNIT-I: **12 Hours**

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

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

Scales: Plain scales, diagonal scales and vernier scales. **COs-C01**

UNIT-II: **16 Hours**

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. **COs-C02**

UNIT-III: **10 Hours**

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. **COs-C03**

UNIT-IV: **16 Hours**

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.

Sections of Solids: Perpendicular and inclined section planes, Sectional views and True shape of sections for simple position only. **COs-C04**

UNIT-V: **12 Hours**

Conversion of Views: Conversion of isometric views to orthographic views and Conversion of orthographic views to isometric views for simple objects only.

Computer graphics: Creating 2D&3D drawings of objects including PCB and Transformations using Auto CAD (Not for end examination). **COs-C05**

TEXT BOOKS:

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

REFERENCE BOOKS:

1. Engineering Drawing,K.L.NarayanaandP.Kannaiah,Tata McGrawHill.
2. Engineering Drawing, M.B.ShahandB.C. Rana, Pearson EducationInc.
3. Engineering Drawing with an Introduction to AutoCAD, DhananjayJolhe,TataMcGrawHill.

ONLINE RESOURCES:

1. https://www.iitg.ac.in/rkbc/me111.htm
2. https://archive.nptel.ac.in/courses/112/105/112105294/

E-BOOKS:https://www.pdfdrive.com/textbook-of-engineering-drawing-e28918244.html

Data Structures through C

IB.TECH- IISEMESTER(Common to CSE, CSE (DS) & CSE (AI&ML))

Course Title: **Data Structures through C**	Course Code: **R24CSPC01**
Teaching Scheme(L:T:P): **3 0 0**	Credits:3
Type of Course: Lecture +Tutorial	
Continuous Internal Evaluation: 30Marks	Semester End Exam: 70Marks
Pre requisites: a basic understanding of programming concepts and some familiarity with C: Programming concepts, C knowledge, Code editors. and Mathematics.	

COURSEOVERVIEW:

1. To understand computer programming and its roles in problem solving.
2. To understand and develop well-structured programs using C language.

COURSEOBJECTIVES:

The objectives of this course are to:

1. Understanding of fundamental data structures and algorithms.
2. To understand importance of data structure in context of writing efficient program.
3. Create and manage linked lists to efficiently organize and manipulate data, emphasizing memory efficiency.
4. Implement and apply stacks to manage program flow and solve problems involving expression evaluation and backtracking.
5. Explore basic concepts of hashing and apply it to solve problems requiring fast data retrieval and management.
6. To solve problems using data structures such as binary trees, binary search trees, and graphs.

COURSEOUTCOMES:

CO#	CourseOutcomes
CO1	Explain the role of linear data structures in organizing and accessing data efficiently in algorithms
CO2	Develop programs using stacks to handle recursive algorithms, manage program states, and solve related problems.
CO3	Apply queue-based algorithms for distinguish between deques and priority queues, and apply them appropriately to solve data management challenges, and apply Hash based solutions for specific problems.
CO4	Design, implement, and apply linked lists for dynamic data storage, demonstrating understanding of memory allocation.
CO5	Implement operations on Binary tree, Demonstrate the representation and traversal techniques of graphs and their applications.

COURSECONTENT(SYLLABUS)

UNIT-I: Introduction to Linear Data Structures, Searching, and Sorting **10 Hours**

Introduction to Linear Data Structures: Definition and importance of linear data structures, Abstract data types (ADTs) and their implementation, Overview of time and space complexity analysis for linear data structures.

Searching Techniques: Linear & Binary Search.

Sorting Techniques: Insertion Sort, Selection Sort, Bubble Sort, Quick Sort, Merge Sort. **COs–CO1**

Self-Learning Topics: Analyse Time complexity of searching and sorting techniques

UNIT–II: Stacks and its Operations **10 Hours**

Stacks: Introduction to stacks: properties and operations, implementing stacks using arrays and linked lists, Applications of stacks in expression evaluation (Prefix, Postfix, Infix), backtracking, reversing list etc. **COs–CO2**

Self-Learning Topics: Types of Stacks

UNIT-III: Queues and its Operations **12 Hours**

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

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

Hash Tables: Hash table implementation, Hash Table Representation: hash functions, collision resolution-separate chaining, open addressing-linear probing, quadratic probing, double hashing and rehashing, extensible hashing. **COs–CO3**

Self-Learning Topics: Different types of Queues

UNIT- IV Linked List **14 Hours**

Linked Lists: Singly linked lists, representation and operations, Applications on Single Linked List- Polynomial Expressions, Sparse Matrix Expressions, Advantages and disadvantages of Single Linked list, doubly linked lists and circular linked lists, Comparing arrays and linked lists, Applications of linked lists. ` **COs–CO4**

Self-Learning Topics: Swap nodes in a single linked list (without swapping data).

UNIT-V: Introduction to Graphs and Trees **14 Hours**

Graphs: Basic Concepts, Representations of Graphs-Adjacency Matrix and using Linked list, Graph Traversals (DFS&BFS), Graph Applications.

Trees: Introduction to Trees, Properties of Binary Tree, Representation of Binary Tree using Arrays and Linked List, Binary Search Tree – Insertion, Deletion & Traversals, BTrees, B+ Trees.AVL Trees. **COs–CO5**

Self-Learning Topics: Red-Black Trees

TEXT BOOKS:

1. 'Data Structures and Algorithm Analysis in C' by Mark Allen Weiss, Pearson.
2. 'Introduction to Algorithms' by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, MIT Press.
3. 'Data Structures Using C' by Reema Thareja, Oxford University Press.

Reference Books:

1. Algorithms, Part I and II' by Robert Sedgewick and Kevin Wayne, Addison-Wesley.
2. Data Structures and Algorithms Made Easy' by Narasimha Karumanchi, CareerMonk Publications

Web References:

1. http://www.hackerrank.com/domains/datastructures
2. http://www.github.com/topics/data_structures_c
3. http://nptel.ac.in/courses
4. http://www.cslibrary.stanford.edu

APPLIED CHEMISTRY LAB

IB.TECH-II SEMESTER (Common to CSE, CSE (DS) & CSE (AI&ML))

Course Title: **APPLIED CHEMISTRY LAB**	Course Code: **R24BS06**
Teaching Scheme(L:T:P): 0:0:2	Credits:1
Type of Course: Practical	
Continuous Internal Evaluation:30Marks	Semester End Exam: 70Marks
Prerequisites:	

COURSE OVERVIEW:

To succeed in an *APPLIED CHEMISTRY Lab* course, certain foundational skills and knowledge are necessary for effective participation and understanding. Here are the key prerequisites:

1. Basic Chemistry Knowledge
2. Measurement
3. Basic Laboratory Skills
4. Problem-Solving and Analytical Skills
5. Familiarity with Safety Practices

COURSE OBJECTIVES:

1.Verify the fundamental concepts with experiments.
2 Learn and carry out some of the important experiments related to batteries and their properties.
3. Learn the preparation of engineering polymer materials like Bakelite
4. Know the fundamental principles of chemistry lab experiments which include volumetric analysis, dichrometry, conductometry and potentiometer

COURSE OUTCOMES:

CO#	Course Outcomes
CO1	Determine the cell constant an conductance of solutions. Determine redox potentials
CO2	Prepare advanced polymer Bakelite materials. Calculate strength of acid in Pb-Acid battery and Ferrous Iron by Dichrometry
CO3	Measure the strength of an acid present in secondary batteries. Moisture content in a coal sample.

COURSE CONTENT(SYLLABUS)

List of Experiments

1.Conductometric titration of strong acid vs. strong base
2. Conductometric titration of weak acid vs. strong base
3. Determination of cell constant and conductance of solutions
4. Potentiometry - determination of redox potentials and emfs
5. PH metric titration – determination of Strength of Strong acid vs Strong base
6. Determination of Strength of an acid in Pb-Acid battery
7. Determination of Hardness of Water
8. Determination of $KMnO_4$ Using Standard Oxalic Acid Solution
9. Adsorption of acetic acid by charcoal
10. Estimation of Ferrous Iron by Dichrometry
11.Preparation of a Bakelite
12.Preparation of nanomaterials by precipitation method.

REFERENCEBOOKS:

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

Basic Electrical and Electronics Engineering Lab
IB.TECH-II SEMESTER(Common toCSE, CSE(AIML), CSE(DS))

Course Title: **Basic Electrical and Electronics Engineering Lab**	Course Code: **R24ES07**
Teaching Scheme(L:T:P): 0:0:3	Credits:1.5
Type of Course: Practical	
ContinuousInternalEvaluation:30Marks	Semester End Exam: 70Marks
Pre requisites:Understanding of Circuit Components, Breadboard connections.	

COURSEOVERVIEW:

In this lab, the students of all engineering streams are trained on basic concepts of electrical engineering, such as DC circuits, AC circuits, Resonance for series RLC and Parallel RLC circuit, AC to DC conversion, measurement, Efficiency and voltage regulation of transformer, electrical machines, verification of basic laws and theorems.

COURSEOBJECTIVES:

Theobjectives of thiscourseareto impartknowledgeonthefundamentallaws&theoremsofelectricalcircuits,functionsofelectricalmachines andenergycalculations.

COURSEOUTCOMES:

CO#	Course Outcomes
CO1	Apply the theoretical concepts and operating principles to derive mathematical models for circuits, Electrical machines and measuring instruments; calculations for the measurement of resistance, power and power factor.
CO2	Apply the theoretical concepts to obtain calculations for the measurement of resistance, power and power factor.
CO3	Plot and discussthe characteristics of various electron devices/instruments.
CO4	Design suitable circuits and methodologies for the measurement of various electrical parameters; Household and commercial wiring.
CO5	Understand the usage of electronic measuring instruments.
CO6	Plot and discuss the characteristics of various electron devices.

ListofExperiments:

PartA-Basic Electrical Engineering

1. Verification of KCL and KVL **COs:CO1**
2. Verification of Super position theorem **COs:CO1**
3. Measurement of Resistance using Wheat stone bridge **COs:CO1**
4. Magnetization Characteristics of DC shunt Generator **COs:CO1**
5. Measurement of Power and Power factor using Single-phase watt meter **COs:CO2**
6. Verification of ohms law **COs:CO1**
7. Calculation of Electrical Energy for Domestic Premises **COs:CO3**

PartB:Basic Electronics Engineering

1. PlotV-I characteristics of PN Junction diode A)Forward bias B)Reverse bias**COs:CO3**
2. PlotV –I characteristics of Zener Diode and its application as voltage Regulator. **COs:CO3**
3. Implementation of half wave and full wave rectifiers **COs: CO3**

4. Plot Input & Output characteristics of BJT in CE and CB configurations **COs: CO3**
5. Frequency response of CE amplifier. **COs: CO3**
6. Simulation of RC coupled amplifier with the design supplied **COs: CO3**
7. Verification of Truth Table of AND, OR, NOT, NAND, NOR, Ex-OR, Ex-NOR gates Using ICs. **COs:CO2**
8. Verification of Truth Tables of S-R, J-K& D flip flops using respective ICs. **COs: CO2**

REFERENCEBOOKS:

1. BasicElectricalEngineering,D.C.Kulshreshtha,TataMcGrawHill,2019,FirstEdition
2. Power System Engineering, P.V. Gupta, M.L. Soni, U.S. Bhatnagar and A.Chakrabarti,Dhanpat Rai & Co, 2013
3. FundamentalsofElectricalEngineering,RajendraPrasad,PHIpublishers,2014,ThirdEdition
4. R. L. Boylestad & Louis Nashlesky, Electronic Devices & Circuit Theory, Pearson Education, 2021.
5. R. P. Jain, Modern Digital Electronics, 4th Edition, Tata Mc Graw Hill, 2009
6. R. T. Paynter, Introductory Electronic Devices & Circuits – Conventional Flow Version, Pearson Education, 2009.

Data Structures through C Lab

IB.TECH- IISEMESTER(Common to CSE, CSE (DS) & CSE (AI&ML))

Course Title: **Data Structures through C Lab**	Course Code: **R24CSPC02**
Teaching Scheme(L:T:P): **0 0 3**	Credits: 1.5
Type of Course: Practical	
Continuous Internal Evaluation:30Marks	Semester End Exam: 70Marks
Pre requisites: **Data Structures through C Lab, you should have a good understanding of the following topics:** Software development,Programming languages,Time and space complexity analysis.	

COURSEOVERVIEW:

1. Basics of data structures including their fundamentals building blocks: arrays and linked list.
2. To solve problems using linear data structures such as linear lists, stacks, queues.
3. To solve problems using searching and sorting techniques.
4. To be familiar with non-linear data structures such as trees.

COURSEOBJECTIVES:

The objectives of this courseareto:

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

COURSEOUTCOMES:

CO#	Course Outcomes
CO1	Explain the role of linear data structures in organizing and accessing data efficiently in algorithms.
CO2	Develop programs using stacks to handle recursive algorithms, manage program states, and solve related problems& Queues.
CO3	Design, implement, and apply linked lists for dynamic data storage, demonstrating understanding of memory allocation, Graphs & Trees.

COURSECONTENT(SYLLABUS)

Developing the following programs:

Week 1:

1. Implement an array and perform operations such as insertion, deletion, and traversal.
2. Given an M*N integer matrix, if an element is 0, set its entire row and column to 0's.
3. Write a program that uses both recursive and non-recursive functions to perform linear search operations for a key value in the given list of integers.

Week 2:

4. Develop a program to perform quick sort, merge sort.
5. Create a program to perform bubble sort and selection sort.
6. Given an unsorted array of integers, find the length of the longest consecutive elements sequence.

Week 3:

7. Implement a stack using arrays.
8. Delete middle element of a stack
9. Develop a program to evaluate a postfix expression using stack.
10. Convert the prefix expression to postfix expression using the Stacks

Week 4:

11. Implement Queue operations using Arrays..
12. Create a circular queue and perform operations like insertion and deletion.

Week 5:

13. Develop a scenario-based program to simulate a ticket booking system using queues.
14. Implement a linked list and perform operations such as insertion, deletion, and traversal.
15. Detect the cycle in the given linked list.

Week 6:

16. Given two singly linked lists, determine if they intersect and return the intersecting node
17. Implement operations on Circular Linked List.
18. Remove Nth Node from end of List

Week 7:

19. Write a program that uses functions to perform the operations on Doubly Linked List.
20. Write a program that reverse the given linked list without using new list or array.
21. Delete last occurrence of an item from linked list.

Week 8:

22. Implement a queue using arrays and linked lists.

Week 9:

23. Create a scenario-based program to manage a library system using linked lists.

Week 10:

24. Create a program to find the height of a binary tree.
25. Create a program to find sub tree of another tree.

Week 11:

26. Develop a program to check if two binary trees are identical.

Week 12:

27. Develop a binary tree and perform in-order, pre-order, and post-order traversal.

Week 13:

28. Implement a binary search tree (BST) and perform operations such as insertion, deletion, and searching.

Week 14:

29. Implement graph data structure and perform depth-first search (DFS) and breadth-first search (BFS).

Week 15:

30. Implement a scenario-based program for a social network using graph data structures.

1. Delete duplicate-value nodes from a sorted Linked list| Linked List

You are given the pointer to the head node of a sorted linked list, where the data in the

nodes is in ascending order. Delete nodes and return a sorted list with each distinct value in the original list. The given head pointer may be null indicating that the list is empty.

Example:

head refers to the first node in the list.

1->2->2->3->3->3->3->NULL

Remove 1 of the 2 data values and return head pointing to the revised list.

Function Description

Complete the *removeDuplicates* function in the editor below.

removeDuplicates has the following parameter:

- *SinglyLinkedListNode pointer head:* a reference to the head of the list

Returns

- *SinglyLinkedListNode pointer:* a reference to the head of the revised list

2. Sparse Arrays| Array

There is a collection of input strings and a collection of query strings. For each query string, determine how many times it occurs in the list of input strings. Return an array of the results.

Example

stringList=[‘ab’,’ab’,’abc’]

queries=[‘ab’,’abc’,’bc’]

There are 2 instances of 'ab', 1 of 'abc' and 0 of 'bc'. For each query, add an element to the return array, results=[2,1,0] .

Function Description

Complete the function *matchingStrings* in the editor below. The function must return an array of integers representing the frequency of occurrence of each query string in *stringList*.

matchingStrings has the following parameters:

- *string stringList[n]* - an array of strings to search
- *string queries[q]* - an array of query strings

Returns

- *int[q]:* an array of results for each query

3. Mini-Max Sum

Given five positive integers, find the minimum and maximum values that can be calculated by summing exactly four of the five integers. Then print the respective minimum and maximum values as a single line of two space-separated long integers.

Example

arr=[1,3,5,7,9]

The minimum sum is 1+3+5+7=16 and the maximum sum is 3+5+7+9=24.

The function prints:16 and 24

Textbooks:

Data Structures and Algorithm Analysis in C' by Mark Allen Weiss, Pearson.

'Introduction to Algorithms' by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, MIT Press.

'Data Structures Using C' by Reema Thareja, Oxford University Press.

Reference Books:

1. Algorithms, Part I and II' by Robert Sedgewick and Kevin Wayne, Addison-Wesley.
2. Data Structures and Algorithms Made Easy' by Narasimha Karumanchi, CareerMonk Publications

ENGINEERING WORKSHOP

IB.TECH-II SEMESTER(Common to CSE, CSE(AIML), CSE(DS))

Course Title : **ENGINEERING WORKSHOP**	Course Code: **R24ES08**
Teaching Scheme(L:T:P): 0:0:3	Credits:1.5
Type of Course: Practical	
Continuous Internal Evaluation: 30 Marks	Semester End Exam: 70 Marks
Pre requisites:	

COURSEOVERVIEW:

- The Engineering Workshop Lab introduces students to essential hands-on skills across multiple trades, fundamental for understanding material manipulation, joining techniques, and assembly processes. Through eight core experiments, students learn woodworking, sheet metal fabrication, fitting, foundry molding, welding, electrical wiring, plumbing, and blacksmithing. These exercises build practical knowledge in creating structures, forming metal parts, achieving precision fits, casting molds, and assembling electrical and plumbing systems. This workshop lays a solid foundation for understanding and applying basic engineering processes, crucial for practical problem-solving and project execution in various engineering fields.

COURSEOBJECTIVES:

Theobjectives of thiscourseareto

- Describe how different tools are used in home wiring, tin smiting, blacksmithing, carpentry, and fitting.

COURSEOUTCOMES:

CO#	CourseOutcomes
CO1	Identify workshop tools and their operational capabilities. Practice on manufacturing of components using workshop trades including carpentry, fitting, sheet metal
CO2	Practice on manufacturing of components using workshop trades including foundry and welding.
CO3	Apply fitting operations in various applications and engineering knowledge for Plumbing, House Wiring Practice, and Making square rod and L-bend from the round rod in black smithy

COURSECONTENT(SYLLABUS)

1. **Wood Working**
   a) Half Lap joint
   b) Mortise and Tenon joint
   c) Corner Dovetail joint or Bridle joint
2. **Sheet MetalWorking**
   a) Tapered tray
   b) Conical funnel
   c) Elbow pipe
   d) Brazing
3. **Fitting**

a) V-fit
b) Dovetail fit
c) Semi-circular fit
d) Bicycle tire puncture and change of two-wheeler tyre

4.Foundry Trade: Preparation of Green Sand Moulds
a) Single piece pattern
b) Double piece pattern

5.Welding Shop: Arc welding Practice
a) Lap joint
b) Butt joint

6. Electrical Wiring
a) Parallel and series connection
b) Two-way switch connection
c) Tube light connection
d) Soldering of wires

7.Plumbing
a) Prepare Pipe joint with coupling for 1 inch diameter
b) Prepare Pipe joint with coupling for 1.5inch diameter

8.Black smithy
a) Round rod to Square
b) Round rod to S-Hook

TEXTBOOKS:

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

REFERENCEBOOKS:

1. Elements of Workshop Technology, Vol. I by S. K. Hajra Choudhury & Others, MediaPromoters andPublishers,Mumbai.2007,14thedition
2. WorkshopPracticebyH.S.Bawa,Tata-McGrawHill,2004.
3. Wiring Estimating, Costing and Contracting; Soni P.M. & Upadhyay P.A.; Atul Prakashan2021-22

ONLINE RESOURCES:

1. https://youtube.com/playlist?list=PLzkMouYverALpuDJ4g4TiICc6_vLcS1Ny&si=YGrVJY8uB0tHy_iQ

E-BOOKS:

1. https://www.pdfdrive.com/workshop-processes-practices-and-materials-third-edition-d158706794.html
2. https://www.pdfdrive.com/introduction-to-basic-manufacturing-processes-and-workshop-e217530.html
3. https://www.pdfdrive.com/workshop-technology-e55714020.html

NSS/NCC/SCOUTS & GUIDES/COMMUNITY SERVICE
IB.TECH-IISEMESTER(Common to all Branches)

Course Title: **NSS/NCC/SCOUTS & GUIDES/COMMUNITY SERVICE**	Course Code:**R24MC02**
Teaching Scheme(L:T:P): 0:0:1	Credits:0.5
Type of Course: Practical	
Continuous Internal Evaluation:100Marks	Semester End Exam: 0Marks
Pre requisites: Prerequisites for programs related to **NSS (National Service Scheme)**, **NCC (National Cadet Corps)**, **Scouts and Guides**, and **Community Service** can vary by institution and specific program. However, here are general guidelines and common prerequisites for each: National Service Scheme (NSS), National Cadet Corps (NCC), Scouts and Guides, Community Service Programs.	

COURSEOVERVIEW:

Here's a general course overview for programs related to NSS (National Service Scheme)**,** NCC (National Cadet Corps)**,** Scouts and Guides**, and** Community Service**. Each program focuses on different aspects of personal development, leadership, and community engagement.**

1. National Service Scheme (NSS)

- **Overview**: NSS is a voluntary program aimed at fostering a sense of social responsibility and community service among students. It emphasizes the importance of personal and community development.
- **Core Components**:
  - **Community Service Projects**: Participation in various community development activities, such as health camps, environmental awareness programs, and literacy initiatives.
  - **Workshops and Seminars**: Educational sessions on social issues, health, hygiene, and community development strategies.
  - **Leadership Development**: Training sessions focused on leadership skills, teamwork, and effective communication.
  - **Camps and Activities**: Organizing and participating in camps, rallies, and other events that promote social awareness and civic responsibility.

2. National Cadet Corps (NCC)

- **Overview**: NCC is a youth development movement that aims to develop character, discipline, leadership, and a spirit of adventure among young people.
- **Core Components**:
  - **Military Training**: Basic training in drill, weapons handling, and military tactics, combined with emphasis on discipline and teamwork.
  - **Adventure Activities**: Participation in activities such as trekking, mountaineering, and camping to foster adventure skills and resilience.
  - **Community Service**: Involvement in social service initiatives and community development projects.
  - **Leadership and Management Skills**: Training sessions focused on leadership, communication, and management, preparing cadets for future responsibilities.

3. Scouts and Guides

- **Overview**: Scouts and Guides programs promote personal development, leadership skills, and community service among young people through various outdoor and indoor activities.

- **Core Components**:
  - **Skill Development**: Learning practical skills such as first aid, navigation, and survival skills.
  - **Community Projects**: Engaging in community service projects and environmental conservation efforts.
  - **Outdoor Activities**: Camping, hiking, and other outdoor adventures that promote teamwork, resilience, and a love for nature.
  - **Values and Ethics**: Education on values such as integrity, respect, and service, aligning with the principles of scouting.

4. Community Service Programs

- **Overview**: Community service programs are designed to engage individuals in volunteer work that benefits their communities, fostering civic responsibility and social awareness.
- **Core Components**:
  - **Volunteer Projects**: Participation in various service projects, such as assisting in local shelters, food banks, environmental clean-ups, and educational initiatives.
  - **Skill-Building Workshops**: Workshops on leadership, teamwork, and project management to enhance volunteers' capabilities.
  - **Awareness Campaigns**: Engaging in campaigns to raise awareness about social issues, such as health, education, and the environment.
  - **Reflection and Evaluation**: Opportunities to reflect on experiences, discuss challenges, and evaluate the impact of their service.

COURSEOBJECTIVES:

Theobjectives of thiscourseareto:

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

COURSEOUTCOMES:

CO#	CourseOutcomes
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.

COURSECONTENT(SYLLABUS)

UNIT I Orientation

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

Activities:

i) Conducting –ice breaking sessions-expectations from the course-knowing personal talents and skills

ii) Conducting orientations programs for the students –future plans-activities-releasing road map etc.

iii) Displaying success stories-motivational biopics- award winning movies on societal issues etc.

iv) Conducting talent show in singing patriotic songs-paintings- any other contribution.

UNIT II Nature & Care

Activities:

i) Best out of waste competition.

ii) Poster and signs making competition to spread environmental awareness.

iii) Recycling and environmental pollution article writing competition.

iv) Organising Zero-waste day.

v) Digital Environmental awareness activity via various social media platforms.

vi) Virtual demonstration of different eco-friendly approaches for sustainable living.

vii) Write a summary on any book related to environmental issues.

UNIT III Community Service

Activities:

i) 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.

ii) Conducting awareness programs on Health-related issues such as General Health,Mental health, Spiritual Health, HIV/AIDS,

iii) Conducting consumer Awareness. Explaining various legal provisions etc.

iv) Women Empowerment Programmes- Sexual Abuse, Adolescent Health and Population Education.

v) 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.

AVANTHI INSTITUTE OF ENGINEERING & TECHNOLOGY

(Autonomous)

(Approved by A.I.C.T.E., New Delhi & Permanently Affiliated to JNTU-GV, Vizianagaram)

NAAC "A+" Accredited Institute

Tamaram (V), Makavarapalem, Narsipatnam (RD), Anakapalle Dist, Pin-531113. www.avanthienggcollege.ac.in,

mail: principal@avanthienggcollege.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Course Structure

Program– B. Tech Computer Science and Engineering

Regulation-R24

(Applicable from the academic year 2024-2025 to 2026-2027)

Program: **B.Tech Computer Science and Engineering** **Regulation: R24**

II Year I Semester-Course Structure

S.No	Category	Course Code	Course Title	Hours per Week			Credits
				Lecture	Tutorial	Practical	
1	ES	R24CSES09	Mathematical Foundations of Computer Science	3	0	0	3
2	PC	R24CSPC03	Design and Analysis of Algorithms	3	0	0	3
3	PC	R24CSPC04	Object Oriented Programming using JAVA	3	0	0	3
4	PC	R24CSPC05	Digital Logic and Computer Organization	3	0	0	3
5	PC	R24CSPC06	Software Engineering and Project Management	3	0	0	3
6	PC	R24CSPC07	Object Oriented Programming using JAVA Lab	0	0	3	1.5
7	PC	R24CSPC08	Algorithms For Efficient coding Lab	0	0	3	1.5
8	SC	R24CSSC01	Full Stack Development	0	0	3	1.5
9	HS	R24CSHS05	Soft skills& Verbal Ability	0	0	2	1
10	MC	R24MC04	Indian Traditional Knowledge	2	0	0	-
			Total	17	00	11	20.5

Category	Courses	Credits
ES- Engineering Science Course	1	03
PC-Proficitional Core Course	6	15
HS- Humanity Science and Management Course	1	01
SC- Skill Oriented Course	1	1.5
MC-Mandatory Course	1	00
Total	10	20.5

R24CSES09 Mathematical Foundations of Computer Science 3 0 0 3

(Common to CSE, CSE (DS) & CSE (AI&ML))

Course Objectives:

The course is designed with the objective to:

- Apply the knowledge in equivalence formulas, tautological implications in finding normal forms, and theory of inference in Statement Calculus and predicates, and explain Mathematical Induction principle and apply the same
- Develop the skill in equivalences and inference theory in Predicate Calculus
- Learn the properties of relations, Posets, Lattices, sets and to apply in problem solving skills
- Acquire the knowledge and skill to draw graphs and related structures and to apply the knowledge skill in solving problem
- Enhance skill and knowledge to draw of trees and solve minimum cost spanning tree problems and solve and formulate, generating functions and recurrence relations.

Course Outcomes

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

Course Code	Course Outcomes (COs)	Mapping with POs and PSOs										
		PO1	PO2	PO3	PO4	PO5	PO6	PO10	PO11	PSO1	PSO2	BT
R24ES09.1	Applyequivalenceformulas,tautologicalimplicationsinfindingnormalforms,andtheory of Inference in Statement Calculus and predicates	3	3	2	2	3	2	3	2	3	2	L2
R24ES09.2	Apply skill in equivalences and inference theory in Predicate Calculus.	3	3	3	3	3	3	3	2	3	2	L3
R24ES09.3	Students gain the knowledge of the properties of relations, Posets, Lattices, Sets and apply the same in solving the problems.	3	3	3	3	3	2	3	2	3	3	L3
R24ES09.4	solve Generating functions and recurrence relations	3	3	3	2	3	3	3	3	3	2	L3

R24ES09.5	Identify the properties of graphs and related structures and solve the related problems. Identify the properties of Trees and solve minimum cost spanning tree problems	3	3	3	2	3	3	-	-	-	-	L3

SYLLABUS

UNIT-I **14 hours**

Mathematical logic: Fundamentals (statements and notations, connectives, Truth tables), Tautologies, Equivalence of formulas, Tautological implications, Normal forms, Theory of Inference for statement calculus. **CO's-CO1**

Self-Learning Topics:

- Build and analyze truth tables manually.
- Practice rewriting expressions into CNF/DNF.
- Memorize common logical equivalences.

UNIT-II **14 hours**

Predicate Calculus: Predicate logic, statement functions, variables and quantifiers, free and bound variables. Inference Theory of the Predicate Calculus: Logical implication involving quantifiers, Statements with more than one variable. **CO's-CO2**

Self-Learning Topics:

- Memorize Inference Rules.
- Practice symbolic representation of logical statements with quantifiers

UNIT-III **15 hours**

Set Theory: Operations on Sets, Principle of Inclusion-Exclusion.

Relations: Relations, Properties of Relations, Equivalence relations, Operations on relations, Representation of relations, partial orders, Hasse Diagram, Lattices, properties of Lattices, Special types of Lattices (Proofs not required). **CO's-CO3**

Self-Learning Topics:

- Practice solving problems on inclusion-exclusion
- Memorize the properties of relations

UNIT-IV **14 hours**

Recurrence relations: Generating Function of Sequences, calculating coefficient of Generating function, Partial Fractions. First order Linear Homogeneous and Non- Homogeneous recurrence relations, Method of generating functions**,** Method of characteristic root**.**COs-CO4

Self-Learning Topics:

- Solve recurrence problems using both methods

UNIT-V **13 hours**

Graph Theory: Basic Concepts of Graphs, Matrix representation of graphs: Adjacency Matrices, Incidence Matrices, Isomorphic Graphs, Paths and Circuits, Eulerian & Hamiltonian graphs, Planar Graphs, Graph coloring.

Trees: Introduction, Spanning Trees, BFS, DFS, Minimum cost spanning trees (Prim's & Kruskal's)

Self-Learning Topics:

- Memorize all types of graphs and their properties
- Practice the problems on Isomorphic,Hamiltonian and Eulerian graphs
- Practice BFS, DFS, Prim's, Kruskal

Board of Studies : Computer Science and Engineering

Approved in BoS No : 02, --, April 2025

Approved in ACM No: 02

Text Books:

1. Trembly J.P. and Manohar. P, "Discrete Mathematical Structures with applications tocomputer science,"Tata McGraw Hill, New Delhi, 2017
2. Kolman B, Busoy R.C, Ross S.C, Discrete Mathematical Structures,5thEdition,PrenticeHall, 2004.
3. D.S. Chandra sekharaiah, "Mathematical Foundation of Computer Science" PrismPublications 2009.

Reference Books:

1. J.L. Mott, A. Kandel, T. P Baker, "Discrete Mathematics for Computer Scientists andMathematicians," Prentice Hall India, 2nd Edition2015
2. Kenneth H. Rosen, "Discrete Mathematics and its Applications," Tata McGraw Hill, NewDelhi, 7th edition, 2017
3. V. Krishnamurthy, "Combinatory: Theory and Applications", East-West Press. SeymourLipschutz, M.Lipson, "Discrete Mathematics" Tata McGraw Hill, 2005.

Reference Links:

1. https://nptel.ac.in/courses/106106094
2. https://nptel.ac.in/courses/106103205
3. https://nptel.ac.in/courses/106106183

Chairperson
Board of Studies (CSE)

R24CSPC03| Design and Analysis of Algorithms 3 0 0 3

(Common to CSE, CSE (DS), CSE (AI&ML))

Course Objectives:

1. To understand and apply the algorithm analysis techniques
2. To critically analyze the efficiency of alternative algorithmic solutions for the same problem
3. To understand different algorithm design techniques
4. To understand the limitations of Algorithmic power

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

Course Code	Course Outcomes	Mapping with POs and PSOs							Dok
		PO1	PO2	PO3	PO4	PS01	PS02	PS03	
R24CSPC03.1	Review the fundamentals of algorithmic problem solving and analyzing efficiency of algorithms	3	3	1	0	2	3	3	L1,L2,L4
R24CSPC03.2	Apply mathematical formulation, complexity analysis and methodologies	3	3	1	0	2	3	3	L2,L3
R24CSPC03.3	Compare the time complexities of various algorithms	2	3	0	1	3	2	3	L3,L4
R24CSPC03.4	Critically analyze the different algorithm design techniques for a given problem	2	3	0	2	2	3	3	L3,L4
R24CSPC03.5	Illustrate NP class problems and formulate solutions using standard approach	1	2	0	1	2	3	3	L4,L5

SYLLABUS

UNIT-I: Introduction of an Algorithm **15 Hours**

Notion of an Algorithm – Fundamentals of Algorithmic Problem Solving –Important Problem Types – Fundamentals of the Analysis of Algorithmic Efficiency –Asymptotic Notations and their properties - Analysis Framework –Empirical analysis - Mathematical analysis for Recursive and Non-recursive algorithms –Visualization.

Self-Learning Topics: Amortized analysis **COs–CO1**

UNIT–II: Brute Force and Divide-and-Conquer **13 Hours**

Brute Force – Computing an – String Matching - Closest-Pair and Convex-Hull Problems - Exhaustive Search - Travelling Salesman Problem – Knapsack Problem - Assignment Problem.

Divide and Conquer Methodology – Binary Search – Merge sort – Quick sort –Heap Sort - Multiplication of Large Integers – Closest-Pair and Convex – Hull Problems.

Self-Learning Topics: Knuth-Morris-Pratt Algorithm **COs–CO2**

UNIT-III: Dynamic Programming and Greedy Techniques **14 Hours**

Dynamic programming – Principle of optimality - Coin Changing Problem -Computing a Binomial Coefficient – Floyd's Algorithm – Multi Stage Graph -Optimal Binary Search Trees – Knapsack Problem and Memory functions

Greedy Technique – Container Loading Problem - Prim's Algorithm and Kruskal's Algorithm – 0/1 Knapsack Problem - Optimal Merge pattern -Huffman Trees.

Self-Learning Topics: Travelling Sales Person Problem **COs–CO3**

UNIT- IV Iterative Improvement **14 Hours**

The Simplex Method - The Maximum-Flow Problem – Maximum Matching in Bipartite Graphs - Stable Marriage problem

Self-Learning Topics: Hill Climbing **COs–CO4**

UNIT-V: Coping with the Limitations of Algorithm Power **14 Hours**

Lower - Bound Arguments - P, NP, NP - Complete and NP-Hard Problems-Backtracking – n-Queen Problem - Hamiltonian Circuit Problem – Subset Sum Problem. Branch and Bound – LIFO Search and FIFO Search –Assignment Problem – Knapsack Problem – Travelling Salesman Problem.

Self-Learning Topics: Approximation Algorithms for NP-Hard Problems – Travelling Salesman Problem – Knapsack Problem **COs–CO5**

Board of Studies : Computer Science and Engineering
Approved in BOS No: 02, --, May, 2025
Approved in ACM No: 02

Expert Talk (To be delivered by SMEs from Industries)

	COs	**POs / PSOs**
1. Branch and Bound Algorithmic Approach	CO5	PO1,PO2,PO4,PSO1,PSO2,PSO3

Text Books:

1. Anany Levitin, "Introduction to the Design and Analysis of Algorithms", Third Edition, Pearson Education, 2012.
2. Ellis Horowitz, Sartaj Sahni and Sanguthevar Rajasekaran, "Computer Algorithms/ C++", Second Edition, Universities Press, 2007

Reference Books:

1. Thomas H.Cormen, Charles E.Leiserson, Ronald L. Rivest and Clifford Stein, "Introduction to Algorithms", Third Edition, PHI Learning Private Limited, 2012.
2. Alfred V.Aho, John E. Hopcroft and Jeffrey D. Ullman, "Data Structures and Algorithms", Pearson Education, Reprint 2006.
3. Harsh Bhasin, "Algorithms Design and Analysis", Oxford university press,2016.
4. S. Sridhar, "Design and Analysis of Algorithms", Oxford university press,2014

Web References:

1. https://nptel.ac.in/courses/106101060
2. https://www.cse.iitm.ac.in/course_details.php?arg=OTI
3. https://swayam.gov.in/nd1_noc19_cs47/previ

Internal Assessment Pattern

Cognitive Level	**Internal Assessment #1(%)**	**Internal Assessment #2(%)**

L1	35	-
L2	30	-
L3	20	30
L4	15	40
L5	-	30
TOTAL (%)	**100**	**100**

Sample Short and Long Answers questions of Various Cognitive Levels

L1: Remember

1. What is an algorithm, and why is it important in computer science?
2. Name two examples of important problem types in algorithm design.
3. What does asymptotic notation analyze, and what are three common types of asymptotic notations?
4. Define dynamic programming and provide a short description of its principle of optimality.
5. What is the difference between NP and NP-Hard problems?

L2: Understand

1. Explain the purpose of empirical analysis versus mathematical analysis when evaluating algorithm efficiency.
2. Describe how the divide-and-conquer methodology works and provide an example of an algorithm that uses this approach.
3. Discuss the advantages and disadvantages of using a brute force approach versus a more efficient algorithm like quick sort.
4. What role does visualization play in understanding and analyzing algorithms?
5. How does the Simplex Method solve linear programming problems?

L3: Apply

1. Given an algorithm to sort a list of numbers, analyze its time complexity using asymptotic notation to demonstrate its efficiency.
2. Implement a brute-force solution for the string matching problem, and analyze the time complexity of your approach.
3. Use dynamic programming to solve the Coin Changing Problem and explain your solution step-by-step.
4. Apply the concept of the maximum-flow problem to a specific network example and demonstrate how to calculate the maximum flow.
5. Create a greedy algorithm to tackle the 0/1 Knapsack Problem and explain the logic behind your selection at each step.

L4: Analyze

1. Compare the performance of the naive recursive solution versus a dynamic programming solution for the Fibonacci sequence in terms of time and space.
2. Analyze why the Travelling Salesman Problem is categorized as NP-Complete and discuss

its implications for algorithm design.

3. Evaluate the efficiency of various algorithms (like Merge Sort and Quick Sort) for sorting a large dataset under different conditions.
4. Investigate the effectiveness of the branch-and-bound method compared to backtracking for solving combinatorial optimization problems.
5. Examine the potential trade-offs in solving an optimization problem using iterative improvement methods versus exact algorithms.

L5: Evaluating

1. Critically assess how well amortized analysis can predict the performance of data structures over time.
2. Evaluate the practical applications of approximation algorithms for NP-Hard problems, citing specific examples.
3. Discuss the implications of classifying problems into different categories (P, NP, NP-Complete, and NP-Hard) in the context of computer science and real-world applications.
4. Assess the effectiveness of dynamic programming techniques in practical scenarios compared to greedy algorithms.
5. Evaluate the significance of understanding algorithmic limitations in practical computing, particularly regarding the P vs NP debate.

Chairperson
Board of Studies (CSE)

R24CSPC04 Object Oriented Programming using JAVA 3 0 0 3

(Common to CSE, CSE (DS) & CSE (AI&ML))

Course Objectives:

The main objectives of the course is to

- Understand the basic ideas of Object-Oriented Programming (OOP) and implement them using Java through classes, objects, inheritance, and polymorphism.
- Understand and implement basic Java programming using data types, operators, control statements, and arrays to solve simple problems.
- Implement programs using Java features like constructors, method overloading, exception handling, and multithreading for better program design.
- Understand and use Java's built-in libraries like collections, file handling, and interfaces to create useful and organized programs.
- Implement graphical programs using JavaFX and understand how to create simple network-based applications using Java networking.

Course Outcomes

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

Course Code	Course Outcomes	Mapping with POs and PSOs							BT
		PO 1	PO 2	PO 3	PO 5	PO 11	PS0 1	PS0 2	
R24CSPC04.01	Implement object orientated programming strategies and Contrast classes and objects.	3	2	2	3	2	3	3	L2, L3
R24CSPC04.02	Analyze Inheritance and Dynamic Method Dispatch	3	3	2	2	2	2	3	L2, L3, L4
R24CSPC04.03	Demonstrate various classes in different packages and can design own packages	3	2	3	3	2	2	3	L2, L3
R24CSPC04.04	Manage Exceptions and Apply Threads	3	3	3	3	2	2	3	L3, L4
R24CSPC04.05	Create GUI screens along with event handling and write network programs	3	2	3	3	3	2	3	L5, L6

SYLLABUS

UNIT-I: 15 Hours

INTRODUCTION TO OBJECTS & CLASSES: What is Object Oriented Programming? Object Orientation as a New Paradigm, An Overview of Java: Process Oriented Vs Object Oriented

Programming, OOP Principles, Java BuzzWords, The Byte Code, A First Simple Program. Class Fundamentals with Variables and Methods, Declaring objects for accessing variables and methods. Data Types and Variables, Operators and Expressions, Control Statements, Type Conversion and Casting, Lexical Issues in Java, command line arguments,

ARRAYS: Introduction, Declaration and Initialization of Arrays, Single Dimension, Operations on Array Elements, Sorting of Arrays, Search for Values in Arrays, Multi Dimension. **CO's-CO1**

Self-learning topics:

- Difference between JDK, JRE, and JVM
- History and evolution of Java
- Best practices for writing Java code (naming conventions, comments, code formatting)
- Enhanced for-loop and foreach in Java

UNIT-II: **13 Hours**

CONSTRUCTORS: Default and Parameterized, this keyword and Garbage Collection, Final and Static Keywords, Overloading Methods, Overloading Constructors, Using objects as Parameters, Returning objects, Strings, String methods, String Buffer and String Builder.

INHERITANCE: Inheritance Basics, Types of Inheritance, Using Super keyword for constructors, Super to call variables and methods, Method Overriding, Dynamic Method Dispatch. **CO's-CO2**

Self-learning topics:

- Constructor chaining and private constructors
- Immutable classes and the final keyword usage in depth
- StringBuilder vs StringBuffer: performance comparison

UNIT-III: **15 Hours**

PACKAGES AND INTERFACES: Defining a Package, importing a package, Package Example, Access Protection, An Access Example, Abstract classes, Interfaces: Defining and Implementing Interfaces

EXPLORING java.lang: Wrapper classes, Object, Math, Runtime

EXPLORING java.util: The collection framework: ArrayList, HashSet HashMap, String Tokenizer, Calendar, Random, Scanner

EXPLORING java.io: File class, Byte Streams, Character Streams, File Input Stream, FileOutputStream, FileReader and FileWriter classes. **CO's-CO3**

Self-learning topics:

- Creating custom packages and JAR files
- Interface vs Abstract class: When to use what?
- Iterators and ListIterator in Java Collections

- Serialization and Deserialization using java.io

UNIT-IV: **14 Hours**

EXCEPTION HANDLING: Exception Handling Fundamentals, Exception Types, throw, throws and finally, Creating your own exceptions, Chained Exceptions.

MULTITHREADED PROGRAMMING: Java Thread Model, The Main thread, Two ways of Creating a Thread, Creating Multiple Threads, isAlive(), join(), Thread Priority, Synchronization, Inter Thread Communication. **COs-CO4**

Self-learning topics:

- Try-with-resources statement in exception handling
- Uncaught exceptions and the default handler
- Daemon threads and thread groups

UNIT-V: **12 Hours**

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

EXPLORING JAVA.NET: Socket, ServerSocket, InetAddress, DataGramSocket, URL, Client Server Program using Sockets. **COs-CO5**

Self-learning topics:

- Creating custom controls in JavaFX
- Building forms and handling validation in JavaFX
- Introduction to Java NIO (New I/O)
- Understanding protocols (TCP vs UDP) in Java networking

Board of Studies : Computer Science and Engineering
Approved in BoS No : -- May 2025
Approved in ACM No: 02

Text Books:

1. Herbert Schildt, Java The complete reference, 11th Edition, McGrawHill, 2019
2. Timothy budd, An introduction to object-oriented programming, 3rd Edition, Pearson, 2009.
3. JAVA 9 for Programmers, Paul Deitel, Harvey Deitel, 4th Edition, Pearson.

Reference Books:

1. Cay S. Horstmann, Core Java Volume I–Fundamentals, 11th Edition, Pearson 2019
2. Y. Daniel Liang Introduction to Java Programming Comprehensive Version, 10th Edition,

Pearson, 2015.

3. Bruce Eckel, Thinking in Java, 4th Edition, Prentice Hall, 2006

ReferenceLinks:

1. http://en.wikibooks.org/wiki/Java_Programming-JavaLearningWikiBook
2. http://www.javabeginner.com-JavaBeginnerTutorial

Chairperson
Board of Studies (CSE)

R24CSPC05 **Digital Logic and Computer Organization** **3 0 0 3**

(Computer Science and Engineering)

Course Objectives:

The main objectives of the course is to

- Provide students with a comprehensive understanding of digital logic design principles and computer organization fundamentals
- Apply Boolean algebrs and simplify logic expressions an can design sequential circuits.
- Design combinational circuits and perform arthemetic operations using binary address, substractors, multiplies and code converter.
- Explain input/output (I/O) systems and their interaction with the CPU, memory, andperipheral devices
- Explore secondary storage mechanism and their role in memory hierarchy.

Course Outcomes

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

Course Code	Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	BT
R24CSPC 10.01	Understand Significance of Number Systems, Conversions, and BinaryCodes.	3	2	2	1	2	-	-	1	1	1	-	3	2	L2, L3
R24CSPC 10.02	Apply Different Simplification Methods For Minimizing Boolean Functions, The Operation Of Flip-Flops, Registers, And Counters	3	3	2	2	2	-	-	1	1	1	-	3	2	L1, L4
R24CSPC 10.03	Illustrate Knowledge On Design Of Various Combinational Circuits.	3	3	3	2	2	-	-	1	1	2	1	3	2	L1, L3
R24CSPC 10.04	Understand the Basic Structure of Computer and Memory Organization.	3	3	2	2	3	1	1	1	2	2	2	2	1	L2, L4
R24CSPC 10.05	Discuss The Basic Structure And Organization Of Computers.	3	3	2	2	3	1	2	1	1	2	2	2	1	L5, L6

SYLLABUS

UNIT I: **15 Hours**

Number Systems and Codes: Decimal, Binary, Octal, and Hexa-decimal number systems and their conversions, ASCII code, Excess-3 code, Gray code, Complement representation of negative numbers: Signed Magnitude, One's complement method, Two's complement method, Binary

Arithmetic.

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

Self Learning Concepts: **CO's-CO1**

Basic Gates**:** AND, OR, NOT, XOR Learn 2, 3, 4-variable K-Maps, Multiplexers (MUX): Select one input among many

UNIT II: **15 Hours**

Boolean algebra: Boolean operations, Boolean functions, algebraic manipulations, min-terms and max-terms, sum-of-products and product-of-sum representations, two-input logic gates, NAND/NOR implementations, Minimization of Boolean functions using Karnaugh map, don't-care conditions, prime implicants, Tabular Method **CO's-CO2**

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

Self Learning Concepts:Shift Registers: Serial-In Serial-Out (SISO), Serial-In Parallel-Out (SIPO), etc

UNIT III: **15 Hours**

Combinational Logic Design: Analysis of combinational circuits, Design Procedure – Binary Adder, Subtractor, BCD Adder, multiplier, comparator, decoders, encoders, multiplexers, demultiplexers, Code Converters

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 **CO's-CO3**

Self Learning Concepts: Use Half Adder (1-bit) and Full Adder (with carry), simulate arithmetic circuits and MUX/DEMUX.

UNIT IV: **15 Hours**

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

Computer Organization

Organization and Architecture, Structure and Function, Computer Components, Bus Interconnection, Processor Organization, Register Organization. Instruction codes, Computer instructions, Memory reference instructions, Instruction Cycle. Stack organization, instruction formats, addressing modes, program control. **CO's-CO4Self**

Learning Concepts: Memorize types of registers: Program Counter (PC), Instruction Register (IR), Accumulator, MAR, MDR

UNIT V: **10 Hours**

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. **CO'S-CO5**

Self Learning Concepts: Learn the difference between volatile(e.g., RAM) **and** non-volatile (e.g., ROM, SSD) memory,

Board of Studies : Computer Science and Engineering

Approved in BoS No: 02, -- April, 2025

Approved in ACM No: 02

Text Books:

1. Computer Organization, Carl Hamacher, ZvonkoVranesic, SafwatZaky, 6th edition, McGraw Hill
2. M. Moris Mano, Computer Systems Architecture, 3rd Edition, Pearson Education, 2007.
3. William Stallings, Computer Organization and Architecture: Designing for Performance, 10th Edition, Pearson Education, 2016

Reference Books:

1. Computer Systems Architecture, M.Moris Mano, 3rdEdition, Pearson
2. Computer Organization and Design, David A. Paterson, John L.Hennessy,Elsevier
3. Fundamentals of Logic Design, Roth, 5thEdition, ThomsonM. Moris Mano, Computer Systems Architecture, 3rd Edition, Pearson Education, 2007.
4. John P. Hayes, Computer Architecture and Organization, 3rd Edition, WCB/McGraw-Hill.
5. M. Morris Mano and Michael D. Ciletti, Digital Design, 4th Edition, Pearson Education, 2013.
6. Vincent P. Heuring and Harry F. Jordan, Computer System Design and Architecture, 2nd Edition,Pearson Education, 2004.
7. Carl Hamacher, Computer Organization and Embedded Systems, 6th Edition, McGraw Hill Higher Education, 2002.

Online Learning Resources:

1. https://archive.nptel.ac.in/courses/106/105/106105163/
2. https://onlinecourses.nptel.ac.in/noc21_ee39/preview

Web References:

1. http://www.c4learn.com/
2. http://www.geeksforgeeks.org/c/
3. http://nptel.ac.in/courses/106105163/
4. http://www.learn-c.org/
5. https://www.tutorialspoint.com/c programming/

Chairperson
Board of Studies (CSE)

R24CSPC06 Software Engineering and Project Management 3 0 0 3

(Computer Science and Engineering)

Course Objectives:

1. To comprehend software development life cycle.
2. To gain knowledge of requirement engineering and SRS documents.
3. To understand software architecture styles.
4. To learn various software testing techniques and their applicability.
5. To apply and analyze project management life cycle.

Course Outcomes

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

Course Code	Course Outcomes (COs)	Mapping with POs and PSOs								DoK
		PO1	PO2	PO3	PO4	PO5	PO11	PSO1	PSO2	
R24CSPC05-01	Describe the principles of software engineering, life cycle models	3	2	2	-	1	1	2	-	L1,L2
R24CSPC05-02	Analyze the computing requirements to solve a given problem	2	3	2	2	2	1	3	-	L1,L2,L3
R24CSPC05-03	Demonstrate the importance of software modeling and modeling languages	2	2	3	2	3	1	2	-	L2,L3,L4
R24CSPC05-04	Illustrate the necessity of software testing and design test cases for a software	2	3	2	3	3	1	3	-	L4,L5,L6
R24CSPC05-05	Interpret Software maintenance and state the concepts of project Management.	2	2	2	1	2	2	-	2	L4,L5,L6

SYLLABUS

UNIT-I: 10 Hours

Introduction: Evolution, Software development projects, Exploratory style of software developments, Emergence of software engineering, Software Product and Process Characteristics, Software Process customization and improvement, Computer system engineering.

Software Life Cycle Models: Basic concepts, Waterfall model and its extensions, Rapid application development, Agile development model, Spiral model. **CO's–CO1**

Self-Learning Topics: Advanced Software Development Paradigms

UNIT II: 15

HoursRequirements Analysis and Specification: Requirements gathering and analysis, Software

Requirements Specification (SRS), Formal system specification, Axiomatic specification, Algebraic specification, Executable specification and 4GL, Petri Nets- Data Dictionary.**CO's–CO2**
Self-Learning Topics:Modern Requirements Engineering Approaches

UNIT – III: **15 Hours**

Design process and Design Concepts, Design Model– Design Heuristic, Architectural Design - Architectural styles, Architectural Design, Architectural Mapping using Data Flow- User Interface, Design: Interface analysis, Interface Design, Component level Design: Designing Class based components, traditional Components) **CO's–CO3**

Self-Learning Topics:Design Patterns (Gang of Four - GoF)

UNIT- IV: **10 Hours**

Coding And Testing: Coding, Code review, Software documentation, Software testing fundamentals, Internal and external views of Testing, Black-box testing, White-Box testing, , Debugging, Software **Implementation Techniques**: Coding practices- Refactoring, Maintenance and Reengineering-BPR model , Reengineering process model-Reverse and Forward Engineering, Program analysis tools, Unit Testing – Integration Testing – Validation Testing Q, Testing object-oriented programs, Smoke testing, and Some general issues associated with testing. .**CO's–CO4**

Self-Learning Topics:Test-Driven Development (TDD) and Behavior-Driven Development (BDD)

UNIT-V: **10Hours**

Software Maintenance & Software Project Measurement: Software Configuration Management (SCM), Software Change Management, Version Control, Change control and Reporting,Re-engineering, Reverse Engineering, Project Management Concepts, Project Scheduling and Tracking, Software Quality Assurance (SQA)

Software Reuse: reuse- definition, introduction, reason behind no reuse so far, Basic issues in any reuse program, a reuse approach, and Reuse at organization level. **CO's-CO5**

Self-Learning Topics: Adaptive and Preventive Maintenance Strategies

Board of Studies : Computer Science and Engineering

Approved in BoS No : 02, --, April, 2025

Approved in ACM No: 02

Expert Talk (To be Delivered by SMEs from Industries) **CO's/ PO's / PSO's**

1. Agile Software Development in the Real World: Challenges and Best Practices" Industry adoption of Agile, Scrum, and DevOps with practical case studies.CO1–CO5, PO1,PO2,PO3,PO12,PSO1,PSO2
2. Software Architecture and Design Patterns for Scalable Applications" Focus on modern design principles used in scalable and maintainable software CO1–CO5, PO1, PO2,PO3,PO12,PSO1,PSO2

Text Books:

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

Reference Books:

1. Software Engineering, Ian Sommerville,10th Edition, Pearson.
2. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press.
3. Software Project Management" by Bob Hughes, Mike Cotterell, and Rajib Mall.
4. Managing Software Projects" by Gopalaswamy Ramesh.

Web References:

1. https://www.computer.org/education/bodies-of-knowledge/software-engineering
2. https://www.geeksforgeeks.org/software-engineering/
3. https://www.pmi.org/pmbok-guide-standards
4. https://www.mindtools.com/pages/main/newMN_PPM.htm

Internal Assessment Pattern

Cognitive Level	Internal Assessment #1(%)	Internal Assessment #2(%)
L1	35%	--
L2	40%	--
L3	25%	25%
L4	--	35%
L5	--	25%
L6	--	15%
Total (%)	100%	100%

Sample Short and Long Answers questions of Various Cognitive Levels

L1: Remember

1. What is the purpose of software engineering?
2. Define the term "software life cycle."
3. List the characteristics of a software product.
4. Name any four software life cycle models.
5. What is the waterfall model?
6. What is the purpose of a Software Requirements Specification (SRS)?
7. Define formal system specification.
8. What is a data dictionary?
9. List the types of formal specification techniques.
10. What are Petri Nets used for in system modeling?

L2: Understand

1. Explain the difference between software product and software process.
2. Describe how exploratory software development differs from planned development.

3. Compare traditional software development with agile development.
4. Explain why customization of the software process is important.
5. Describe the evolution of software engineering.
6. Explain the role of requirements gathering in software development.
7. Describe the characteristics of a good SRS document.
8. Differentiate between axiomatic and algebraic specifications.
9. Explain how Petri Nets represent system behavior.
10. Describe the use of 4GLs in executable specifications.
11. Explain the importance of architectural design in software engineering.
12. Describe the purpose of design heuristics in the design process.
13. Discuss the role of data flow diagrams in architectural mapping.
14. Differentiate between class-based and traditional components.
15. Explain the concept of interface analysis in UI design.

L3: Apply

1. Write a simple example of an algebraic specification for a stack ADT.
2. Apply Petri Nets to model a simple producer-consumer system.
3. Develop a basic data dictionary for a student registration system.
4. Create an SRS for a library management system.
5. Use axiomatic specification to define a simple banking operation.
6. Apply design heuristics to improve an existing system design.
7. Use a data flow diagram to map the architecture of a student management system.
8. Develop a simple interface design for an online bookstore.
9. Design class-based components for a ride-sharing application.
10. Implement a basic architectural model using a layered style.

L4: Analysing

1. Analyze the relationship between software architecture and system performance.
2. Compare and contrast different architectural styles (e.g., layered vs. client-server).
3. Identify the strengths and weaknesses of traditional components in modern software systems.
4. Break down the steps involved in interface design for a mobile application.
5. Examine the impact of poor design heuristics on software maintainability.
6. Analyze the impact of poor documentation on software maintenance.
7. Compare white-box testing and black-box testing with respect to coverage.
8. Differentiate between reverse engineering and reengineering.
9. Examine how object-oriented features affect testing strategies.

10. Identify and explain common issues associated with testing large-scale systems.
11. Analyze the relationship between software maintenance and configuration management.
12. Compare re-engineering and reverse engineering processes.
13. Differentiate between corrective, adaptive, and perfective maintenance.
14. Identify the barriers to software reuse in large organizations.
15. Examine the role of SQA in reducing post-deployment defects.

L5: Evaluating

1. Evaluate the effectiveness of smoke testing in agile environments.
2. Justify the use of program analysis tools in improving code quality.
3. Assess the benefits of incorporating refactoring in continuous integration.
4. Critically evaluate various testing strategies for object-oriented software.
5. Review and recommend practices for effective debugging in real-time applications.
6. Evaluate the effectiveness of SCM tools in large-scale software projects.
7. Justify the need for strict change control in mission-critical applications.
8. Assess the impact of poor version control on project success.
9. Critique the current industry practices on software reuse.
10. Evaluate a project tracking system in terms of its ability to meet deadlines and budget.

L6: Creating

1. Design a comprehensive test plan for an e-commerce application.
2. Create a code review checklist tailored for a distributed development team.
3. Develop a test suite for validating a banking application using black-box techniques.
4. Construct a model that integrates reverse engineering and forward engineering in legacy system migration.
5. Propose a set of guidelines for testing object-oriented programs effectively.
6. Design a version control and change management strategy for a distributed team.
7. Develop an SQA plan for a medical software application.
8. Create a reuse strategy for an educational software company.
9. Construct a project management schedule using available software tools for a mid-sized web application.
10. Propose an organization-wide reuse program, addressing its key challenges and success factors.

Chairperson
Board of Studies (CSE)

R24CSPC07 Object Oriented Programming using JAVA Lab 0 0 3 1.5
(Common to CSE, CSE (DS) & CSE (AI&ML))

Course Objectives:

1. To develop skills to design and analyze the applications with respect to java programming.
2. To strengthen the ability to identify and apply the suitable object oriented concept for the given real world problem.

Course Outcomes

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

Course Code	Course Outcomes	Mapping with POs and PSOs						BT
		PO1	PO2	PO3	PO5	PS01	PS02	
R24CSPC06.1	Write basic Java programs using arrays, control structures, and demonstrate the use of classes and objects.	3	2	2	1	3	2	L2, L3
R24CSPC06.2	Apply object-oriented concepts such as inheritance and packages, and implement applications using built-in and user-defined packages.	2	2	3	3	2	3	L3, L4
R24CSPC06. 3	Develop Java applications with exception handling, multithreading, graphical user interfaces, and basic networking.	2	2	3	3	1	3	L3, L4, L6

Board of Studies: Computer Science and Engineering

Approved in BOS No: 02, --, April, 2025

Approved in ACM No: 02

Developing the following programs:

1. a) Implement the following programs using command line arguments and Scanner class
   i) Accept two strings from the user and print it on console with concatenation of "and" in the middle of the strings. **CO's-CO1**
   ii) To find the perimeter and area of a circle given a value of radius. **CO's-CO1**

   b) Write a program using classes and objects in java? **CO's-CO1**
2. a) Write a program to call default constructor first and then any other constructor in the class? **CO's-CO1**

   b) Write a program that accepts an array of integers and print those which are both odd and prime. If no such element in that array print "Not found". **CO's-CO1**

c) Write a program to accept contents into an Integer Array and print the frequency of each number in the order of their number of occurrences. **CO's-CO1**

d) Write a program that accepts an 'm x n' double dimension array, where 'm' represents financial years and 'n' represents Ids of the items sold. Each element in the array represents number of items sold in a particular year. Identify the year and id of the item which has more demand. **CO's-CO2**

3. a) Create a class Box that uses a parameterized constructor to initialize the dimensions of a box. The dimensions of the Box are width, height, depth. The class should have a method that can return the volume of the box. Create an object of the Box class and test the functionalities. **CO's-CO2**

   b) Create a new class called Calculator with the following methods:

   A static method called power Int(int num1,int num2) This method should return num1 to the power num2.

   A static method called power Double(double num1,double num2). This method should returnnum1 to the power num2.

   Invoke both the methods and test the functionality. Also count the number of objects created. **CO's-CO2**

4. a) Accept a String and a number 'n' from user. Divide the given string into substrings each of size 'n' and sort them lexicographically. **CO's-CO1**

   b) Accept an array of strings and display the number of vowels and consonants occurred in each string. **CO's-CO1**

   c) Accept two strings from the user and determine if the strings are anagrams or not. **CO's-CO1**

5. a) Create a multilevel inheritance for classes vehicle, brand and cost. The vehicle class determines the type of vehicle which is inherited by the class brand which determines the brand of the vehicle. Brand class is inherited by cost class, which tells about the cost of the vehicle. Create another class which calls the constructor of cost class and method that displays the total vehicle information from the attributes available in the super classes. **CO's-CO2**

   b) Create an inheritance hierarchy of Figure_3D, Cylinder, Cone, Sphere etc. In the base class provides methods that are common to all Figure_3Ds and override these in the derived classes to perform different behaviors, depending on the specific type ofFigure_3D. Create an array of Figure_3D, fill it with different specific types ofFigure_3Ds and call your base class methods. **CO's-CO2**

6. a) Design a package to contain the class Student that contains data members such as name,

roll number and another package contains the interface Sports which contains some sports information. Import these two packages in a package called Report which process both Student and Sport and give the report.**CO's-CO2**

b) Write a program that accepts values of different data types and convert them to corresponding wrapper classes and display using the vector. **CO's-CO1**

7. a) Write a program to generate a set of random numbers between two numbers x1 and x2,and x1>0. **CO's-CO1**

   b) Write a program to implement a new Array List class. It should contain add(), get(),remove(), size() methods. Use dynamic array logic.**CO's-CO2**

   c) Create an employee class containing at least 3 details along with Id, setters, and getters. Insert the employee objects dynamically key as employee id and value as its corresponding object into a HashMap. Perform Id based search operation on theHashMap.**CO's-CO2**

8. a) Write a program that reads file name from the user then displays information about that file, also read the contents from the file in byte stream to count the number of alphabets, numeric values, and special symbols. Write these statistics into another file using byte streams **CO's-CO3**

   b) Write a program that reads a CSV file containing a super market data containing product ID, Name, Cost and Quantity of sales and calculate the total revenue of the supermarket also sort the products in the order of their demand. **CO's-CO3**

   c) Write a program that reads a text file containing some technical content and identify the technical terms and sort them alphabetically. **CO's-CO3**

   Note: use a file containing stop words (general English and Grammar terms as many as possible)

9. a) Write a program that reads two numbers from the user to perform integer division into Num1 and Num2 variables. The division of Num1 and Num2 is displayed if they are integers. If Num1 or Num2 were not an integer, the program would throw a Number Format Exception. If Num2 were Zero, the program would throw an Arithmetic Exception.

   CO's-CO3

   b) Create a user defined exception. **CO's-CO3**

10. a) Write a program that creates 3 threads by extending the Thread class. First thread displays "Good Morning" every 1 sec, the second thread displays "Hello" every 2seconds and the third displays Welcome" every 3 seconds. (Repeat the same by implementing Runnable). **CO's-CO3**

    b) Write a program to illustrate Thread synchronization. **CO's-CO3**

11. a) Create a JApplet that displays a message which is scrolling from left to right.**CO's-CO3**

b) Write a program that displays a sample registration page using Swing controls use appropriate layout managers. **CO's-CO3**

c) Write a program for handling mouse events with adapter classes. **CO's-CO3**

12. a) Create an interface containing 3 radio buttons named line, rectangle, and oval. Based on the radio button selected, allow user to draw lines, rectangles, or ovals as per the locations selected by the user. **CO's-CO3**

    b) Write a program to create a Table inside a JFrame. **CO's-CO3**

    c) Create an interface that illustrates JFile Chooser class and read CSV file containing employee data of various departments and display the records department wise on the interface. **CO's-CO3**

13. a) Check all the fields filled or not, display success dialogue if all fields are filled with the help of ActionListener for program **CO's-CO3**

    b) Display respective error dialogue if a field is empty. **CO's-CO3**

14. Write a program to create three JSliders where each represents colors RED, GREEN and BLUE. Each slider has a value from 0 to 255. The background color of the applet is set based on the values retrieved from each slider to form a color using the color class constructor. On sliding any slider, the background color of the applet changes. **CO's-CO3**
15. Complete the code to develop an ADVANCED CALCULATOR that emulates all the functions of the GUI Calculator as shown in the image. **CO's-CO3**
16. Write a program that implements a simple client/server application. The client sends data to a server. The server receives the data, uses it to produce a result, and then sends the result back to the client. The client displays the result on the console.

    For ex: The data sent from the client is the radius of a circle, and the result produced by the server is the area of the circle. **CO's-CO3**

Case Studies:

1. Grading a Multiple-choice Test for students **CO's-CO1**
2. Create a Person class containing basic details like Name, Gender, Mobile and Email. Based on that create and manage the objects that are related to student and employee Classes **CO's-CO2**
3. Create a package called Banking containing classes and interfaces related to various banking operations such as withdrawal, deposits, loans and insurance etc. Create two classes related to any two specific banks that uses this package. **CO's-CO2**
4. Write a program that implements simple chat application using GUI. **CO's-CO3**

Textbooks:

1. Herbert Schildt, Java The complete reference, 11th Edition, McGrawHill, 2019
2. Timothy budd, An introduction to object-oriented programming, 3rd Edition, Pearson Education, 2009.

Reference Books:

1. Cay S. Horstmann, Core Java Volume I–Fundamentals, 11th Edition, Pearson 2019
2. Y. Daniel Liang Introduction to Java Programming Comprehensive Version, 10th Edition, Pearson, 2015.
3. Bruce Eckel, Thinking in Java, 4th Edition, Prentice Hall, 2006

Web References:

1. https://onlinecourses.nptel.ac.in/noc19_cs84/preview

Chairperson
Board of Studies(CSE)

R24CSPC08| Design and Analysis of Algorithms Lab 0 0 3 1.5
(Common to CSE, CSE (DS) & CSE (AI&ML)

Course Objectives:

The course aims to give students hands – on experience and train them to learn Design and Analysis Algorithm

1. To learn Prolog Program.
2. To Implement in prolog and its working environment
3. To Implement N-Queen problem and puzzle problem using Prolog
4. To Analyze the problem using BFS and DFS algorithm

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

Course Code	Course Outcomes	Mapping with POs and PSOs							Dok
		PO 1	PO 2	PO 3	PO 5	PS O1	PS O2	PS O3	
R24CSPC07.1	Understand of algorithm design techniques applicable across various tasks and ensuring foundational knowledge in problem-solving.	3	3	3	3	3	3	3	L3
R24CSPC07.2	Implement of the algorithms in both sequential and parallel paradigms, relevant for weeks involving sorting and graph traversals.	3	3	3	1	3	3	3	L4
R24CSPC07.3	Handle real-world computational problems.	3	3	3	1	3	3	3	L5

Board of Studies: Computer Science and Engineering
Approved in BOS No: 02, --, May, 2025
Approved in ACM No: 02

Developing the following programs

Week 1:

1. Sort a given set of elements using the Quick sort method and analyze time. **CO's-CO1**

Week 2:

2. Implement a parallelized Merge Sort algorithm using Open MPI**CO's-CO1**

Week 3:

3. Implement binary tree traversal techniques using recursion and non-recursion.**CO's-CO1**

Week 4:

4. Print reachable nodes in a digraph using BFS; check connectivity using DFS.**CO's-CO2**

Week 5:

5. Determine articulation points and biconnected components in a graph. **CO's-CO2**

Week 6:

6. Implement algorithms for minimum cost spanning trees (Prim's and Kruskal's).**CO's-CO2**

Week 7:

7. Find shortest paths in a graph using Dijkstra's algorithm; study of Prolog.**CO's-CO2**

Week 8:

8. Write simple fact for the statements using prolog**CO's-CO2**

Week 9:

9. Write predicates one convert's centigrade temperature to Fahrenheit,other checks if a temperature is below freezing.**CO's-CO2**

Week 10:

10. Write a program to solve 4-Queen problem.**CO's-CO3**

Week 11:

11. Write a program to solve 8-puzzle problem.**CO's-CO3**

Week 12:

12. Write a program to solve any problem using Breadth First Search.**CO's-CO3**

Week 13:

13. Write a program to solve any problem Depth First Search**CO's-CO3**

Week 14:

14. Write a program to solve Travelling salesman Problem**CO's-CO3**

Week 15 and Week 16:

15. Develop a tool to analyze social networks using graph algorithms (e.g., finding the shortest path, clustering, and connectivity). Input social network data and visualize the connections.

CO's-CO3

Text Books:

1. AnanyLevitin, "Introduction to the Design and Analysis of Algorithms", Third Edition, Pearson Education, 2012.
2. Ellis Horowitz, Sartaj Sahni and Sanguthevar Rajasekaran, "Computer Algorithms/ C++", Second Edition, Universities Press, 2007

Reference Books:

1. Thomas H.Cormen, Charles E.Leiserson, Ronald L. Rivest and CliffordStein, "Introduction to Algorithms", Third Edition, PHI Learning PrivateLimited, 2012.
2. S. Sridhar, "Design and Analysis of Algorithms", Oxford university press,2014

Web References:

1. https://nptel.ac.in/courses/106101060
2. https://www.cse.iitm.ac.in/course_details.php?arg=OTI
3. https://swayam.gov.in/nd1_noc19_cs47/previ

Chairperson
Board of Studies (CSE)

R24CSSC01 **Full Stack Development** **0 0 3 1.5**

Course Objectives:

1. To implement the client side of the web application using java-script.
2. To understand Java script on the desktop using Node JS.
3. To develop a web application using Node JS and Express.
4. To implement a SPA using React.
5. To develop a full stack single page application using React ,Node JS, and a Database (MongoDB or SQL).

Course Outcomes

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

Mapping with POs and PSOs																BT
Course Code	**Course Outcomes (COs)**	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**	
24SC02.1	Implement the client side of the web application.	2	1	2	1	3	1	0	2	3	2	1	-	2	1	L1, L2
24SC02.2	Deploy server side applications using NodeJS. Use express framework in web development	2	2	3	2	3	1	0	2	2	2	3	-	3	2	L2, L3
24SC02.3	Implement and architect database systems. Develop a full stack single page applications	3	2	3	2	3	2	1	3	3	3	3	-	3	2	L2, L3

Board of Studies: Computer Science and Engineering

Approved in BOS No: 02, --, April, 2025

Approved in ACM No: 02

Experiments covering the Topics:

- Lists, Links and Images
- HTML Tables, Forms and Frames
- HTML5and Cascading Style Sheets, Types of CSS
- Selector forms
- CSS with Color, Background, Font, Text and CSS Box Model
- Applying JavaScript-internal and external, I/O, Type Conversion
- JavaScript Conditional Statements and Loops, Pre-defined and User-defined Objects
- JavaScript Functions and Events

Developing the following programs:

Week-1:

1. Write a HTML program, to explain the working of lists. **CO's:CO1**
2. Write a HTML program, to explain the working of hyperlinks using <a> tag and href, target Attributes. **CO's:CO1**
3. Create a HTML document that has your image and your friend's image with a specific height and width. Also when clicked onthe images it should navigate to their respective profiles. **CO's:CO1**
4. To create Form validation using JavaScript. **CO's:CO1**

Week-2:

1. WriteaHTMLprogram,toexplaintheworkingoftables.(usetags:<table>,<tr>,<th>, <td>and attributes:border,rowspan, colspan) **CO's:CO2**
2. Write a HTML program, to explain the working of forms by designing Registration form. **CO's:CO2**
3. Get data using Fetch API from an open-source end point and display the contents in the form of a card. **CO's:CO2**

Week-3:

1. WriteaHTMLprogram,thatmakesuseof<article>,<aside>,<figure>,<figcaption>,<footer>,<header>,<main>,<nav>,<section>, ,tags. **CO's:CO2**
2. Write a HTML program, to embed audio and video into HTML webpage. **CO's:CO2**
3. 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).**CO's:CO2**

Week-4:

1. Write a program to apply different types of selector forms
   i. Simple selector(element, id,class,group,universal)
   ii. Combinator selector(descendant, child,adjacentsibling,generalsibling)
   iii. Pseudo-classs elector
   iv. Pseudo-element selector
   v. Attribute selector **CO's:CO2**

Week-5:

1. Write a program to demonstrate the various ways you can reference a color in CSS. **CO's:CO2**
2. Write a CSS rule that places a back ground image half way down the page,tilting it horizontally. The image should remain in place when the user scrolls up or down. C**O's:CO2**

3. Write a program using the following terms related to CSS fontandtext:
   i. font-size ii.font-weight iii.font-style
   iv.text-decoration v.text-transformation vi.text-alignment **CO's:CO2**
4. Write a program, to explain the importance of CSS Box modelusing
   i. Content ii.Border iii.Margin iv.Padding **CO's:CO2**

Week-6:

1. Write a program to embed internal and external Java Script in a webpage. **CO's:CO3**
2. Write a program to explain the different ways for taking input & displaying output..
   CO's:CO3
3. 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
   CO's:CO3

Week-7:

1. Write a program using document object properties and methods.
2. Write a program using window object properties and methods.
3. Write a program using array object properties and methods.
4. Write a program using math object properties and methods.
5. Write a program using string object properties and methods. **CO's:CO3**

Week-8:

1. Write a program to display week days using switch case.
2. Write a program to print1to10 numbers using for, while and do-while loops.
3. Write a program to print data in object using for-in, for-each and for-of loops Develop a program to determine whether a given number is an 'ARMSTRONG NUMBER' or not.
4. 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) **CO's:CO3**

Week-9:

1. 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 **CO's:CO3**

Week-10:

1. Design a HTML having a textbox 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 **CO's:CO3**

Week-11:

1. 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(onlynumbersandlength10digits)
   III. E-mail(shouldcontainformatlikexxxxxxx@xxxxxx.xxx). **CO's:CO3**

Week-12:

1. Create a form and validate the contents of the form using JavaScript. **CO's: CO3**
2. Get data using Fetch API from an open-source end point and display the contents in the form of a card.. **CO's: CO3**
3. Create a Node JS server that serves static HTML and CSS files to the user without using Express. **CO's: CO3**

Week-13:

1. Create a Node JS server using Express that stores data from a form as a JSON file and displays it in another page. The redirect page should be prepared using Handlebars. **CO's:CO3**
2. Create a Node JS server using Express that creates, reads, updates and deletes students' details and stores them in MongoDB database. The information about the user should be obtained from a HTML form. Conversion of Fahrenheit to Celsius and vice versa. **CO's:CO3**

Week-14:

1. Create a Node JS server that creates, reads, updates and deletes event details and stores them in a MySQL database. The information about the user should be obtained from a HTML form. **CO's:CO3**
2. Create a counter using React JS **CO's:CO3**
3. Create a To do application using ReactJS. Store the data to a JSON file using a simple Node JSserver and retrieve the information from the same during page reloads. **CO's:CO3**

Week-15:

1. Create a simple Signup and Login mechanism and authenticate the user using cookies. The user information can be stored in either MongoDB or MySQL and the server should be built using Node JS and Express Framework. **CO's:CO3**
2. Create and deploy a virtual machine using a virtual box that can be accessed from the host computer using SSH. Write a C program to print last n characters of a given file. **CO's:CO3**
3. Create a docker container that will deploy a Node JSping server using the Node JS image. **CO's:CO3**

Textbooks:

1. JohnDean, WebProgrammingwithHTML5, CSSandJavaScript,Jones&Bartlett Learning, 2019.

Reference Books:

1. Programming the World WideWeb, 7thEdition, RobetWSebesta, Pearson, 2013.
2. ProMERNStack: FullStackWebAppDevelopmentwithMongo, Express, React, and Node, Vasan Subramanian, 2nd edition, APress, O'Reilly.

Online Learning Resources:

1. https://www.w3schools.com/html
2. https://www.w3schools.com/css
3. https://www.w3schools.com/js/
4. https://www.w3schools.com/nodejs
5. https://www.w3schools.com/typescript

Chairperson
Board of Studies (CSE)

Course Title: SOFT SKILLS & VERBAL ABILITY	**CourseCode:** R24CSHS05
Teaching Scheme(L:T:P):0:1:2	**Credits:** 02
Type of Course: Tutorials + practical's	
Continuous Internal Evaluation: 30Marks	**Semester End Exam:** 70Marks
Prerequisites: To succeed in Soft Skills and Verbal ability, students should have a basic understanding of effective communication principles, including verbal and non-verbal communication. Familiarity with teamwork and collaboration techniques is essential for group activities and presentations. A positive attitude and willingness to receive feedback help in personal development. Time management and problem-solving abilities support efficient task handling. Basic proficiency in English and presentation tools also enhances overall performance in soft skills training	

Course objective:

1. Enhance proficiency in English grammar, vocabulary, and reasoning skills for recruitment exams.
2. Develop effective communication skills for group discussions, resume building, and interviews.
3. Equip students with techniques for reading comprehension, logical reasoning, and professional presentation.
4. Prepare students for successful career placements through improved language and soft skills.

Course Outcomes:

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

Course Outcomes	**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 8**	**PO 9**	**PO1 0**	**PO1 1**	**PSO 1**	**BT LEVE L**
CO1: Demonstrate strong command over English grammar, vocabulary, and reasoning skills	3	3	2	2	1	1	1	1	2	1		L3
CO2: Effectively communicate in group discussions, create impactful resumes, and excel in interviews	2	1	1	1	2	2	2	3	3	2		L4
CO3: Develop critical thinking	3	3	2	2	1	1	1	1	2	1		L3

and problem-solving abilities for recruitment exams												
CO4: Be well-prepared for career placements with enhanced professional communication and soft skills	2	2	1	1	2	2	2	3	3	2		L3

Unit I – English Grammar and Usage (10 Hours)

***This unit focuses on core grammar concepts frequently tested in company recruitment exams. Topics include*:**

Parts of Speech, Tenses and Subject-Verb Agreement, Articles and Prepositions, Sentence Correction and Spotting Errors, Active and Passive Voice, Direct and Indirect Speech

Learning Outcome: Students will demonstrate accurate grammar usage and error detection skills in various sentence structures.

Vocabulary Development and Application (10 Hours)
This unit enhances vocabulary required for business communication and aptitude tests. Topics include:

Synonyms and Antonyms, One-word Substitution, Idioms and Phrases, Confusing Word Pairs, Phrasal Verbs and Collocations

Learning Outcome: Students will improve their vocabulary strength and apply words appropriately in verbal and written contexts.

Unit II – Reading Comprehension Skills (5 Hours)
Students will learn techniques to understand, interpret, and analyze passages. Focus areas:

Main Idea and Supporting Details, Inference-Based Questions, Vocabulary in Context, Tone and Author's Perspective

Learning Outcome: Students will effectively comprehend and answer questions based on unseen passages within time constraints.

Verbal Reasoning and Logic-Based Language Skills (5 Hours)
This unit covers logical verbal questions commonly seen in recruitment exams:

Sentence Completion, Cloze Tests, Para Jumbles / Sentence Rearrangement, Statement and Conclusion / Assumptions.

Learning Outcome: Students will develop reasoning skills to solve pattern-based language puzzles.

Unit III – Group Discussion Skills (10 Hours):

This unit develops students' ability to communicate effectively in a group setting. It includes understanding the GD format, evaluation criteria, and participation strategies. Sessions will train students on body language, tone modulation, handling abstract and controversial topics, and presenting logical arguments. Multiple GD simulations will be conducted with personalized feedback to improve spontaneity and structure in speaking.

Learning Outcome: students will be able to communicate their ideas clearly, listen actively, contribute effectively to discussions, and demonstrate leadership and teamwork while maintaining professionalism and respect for diverse opinions.

Unit IV – Resume Preparation and Personal Branding (10 Hours):

This unit guides students in preparing an impactful, professional resume suited for technology and consulting sectors. Key areas include formatting, project and internship presentation, using effective language, and highlighting strengths and certifications. Students will also learn to optimize their LinkedIn profiles and online presence to reflect a professional digital identity.

Learning Outcome: Students will be able to create a professional, well-structured resume that highlights their skills and experiences, and build a strong personal brand to effectively present them in the job market

Unit V – Interview Preparation (10 Hours):

This unit addresses all aspects of interview readiness. It covers commonly asked HR and technical questions, behavioral questions using the STAR (Situation, Task, Action, Result) method, and communication strategies during online interviews. Students will receive training in grooming, attire, voice modulation, and confidence building

Learning Outcome: students will be able to confidently handle both technical and HR interviews, presenting themselves professionally and effectively communicating their skills and experiences

Text Books:

1. **Wren, P. C., and H. Martin,***High School English Grammar and Composition,* S. Chand Publishing, 1990.
2. **Lewis, N.,***Word Power Made Easy,* Goyal Publishers, 1993.

3. **Aggarwal, R. S.,***A Modern Approach to Verbal & Non-Verbal Reasoning,* S. Chand Publishing, 2017.
4. **Bakshi, S. P.,***Objective General English,* Arihant Publications, 2018.

E-Resources :

1. **Grammarly,***AI-powered writing assistant,* Grammarly, https://www.grammarly.com/
2. **IndiaBIX,***Online Aptitude & Reasoning Practice,* IndiaBIX, https://www.indiabix.com/.
3. **AmbitionBox,***Interview Experiences and Reviews,* AmbitionBox, https://www.ambitionbox.com/.
4. **Canva,***Online Resume Builder and Templates,* Canva, https://www.canva.com/resumes/templates
5. **Testbook,***Testbook: Online Mock Tests and Practice Papers,*https://testbook.com/.

Chairperson
Board of Studies (CSE)

Course Title: Indian Traditional Knowledge	**CourseCode:R24MC04**
Teaching Scheme(L:T:P): 2:0:0	**Credits:** -
Type of Course: Lecture	
Continuous Internal Evaluation: 30Marks	**Semester End Exam:** 70Marks
Pre requisites: To succeed in Indian Traditional Knowledge, students should have a basic understanding of cultural heritage, environmental sustainability, and indigenous practices. Familiarity with legal and constitutional frameworks, especially related to biodiversity and forest rights, is essential.	

Course Objectives:

To facilitate the students with the concepts of Indian traditional knowledge and to make them understand the Importance of roots of knowledge system

- The course aim of the importing basic principle of third process reasoning and inference sustainability is at the course of Indian traditional knowledge system.
- To understand the legal framework and traditional knowledge and biological diversity act 2002 and geographical indication act 2003.
- The courses focus on traditional knowledge and intellectual property mechanism of traditional knowledge and protection.
- To know the student traditional knowledge in different sector.

Course Outcomes:

After completion of the course, students will be able to:

Course Outcomes	PO1	PO2	PO6	PO7	PO8	PSO1	BT LEVEL
CO1: Understand the concept of Traditional knowledge and its importance	2	0	2	2	2		L2
CO2: Know the need and importance of protecting traditional knowledge	2	2	3	2	3		L2
CO3: Know the various enactments related to the protection of traditional knowledge	2	2	3	2	3		L1
CO4: Understand the concepts of Intellectual property to protect traditional knowledge	3	2	2	2	2		L2

UNIT-I

Introduction to traditional knowledge: Define traditional knowledge, nature and characteristics, scope and importance, kinds of traditional knowledge, the physical and social contexts in which traditional knowledge develop, the historical impact of social change on traditional knowledge

systems. Indigenous Knowledge (IK), characteristics, traditional knowledge vis-à-vis indigenous knowledge, traditional knowledge Vs western knowledge traditional knowledge vis-à-vis formal knowledge. **COs- CO1**

UNIT-II

Protection of traditional knowledge: the need for protecting traditional knowledge Significance of TK Protection, value of TK in global economy, Role of Government to harness TK.

COs- CO1, CO2

UNIT-III

Legal framework and TK: A: The Scheduled Tribes and Other Traditional Forest Dwellers (Recognition of Forest Rights) Act, 2006, Plant Varieties Protection and Farmers Rights Act, 2001 (PPVFR Act);B:The Biological Diversity Act 2002 and Rules 2004, the protection of traditional knowledge bill, 2016. Geographical indications act 2003. **COs- CO2, CO3**

UNIT-IV

Traditional knowledge and intellectual property: Systems of traditional knowledge protection, Legal concepts for the protection of traditional knowledge, Certain non IPR mechanisms of traditional knowledge protection, Patents and traditional knowledge, Strategies to increase protection of traditional knowledge, global legal FORA for increasing protection of Indian Traditional Knowledge. **COs- CO3, CO4**

UNIT-V

Traditional knowledge in different sectors: Traditional knowledge and engineering, Traditional medicine system, TK and biotechnology, TK in agriculture, Traditional societies depend on it for their food and healthcare needs, Importance of conservation and sustainable development of environment, Management of biodiversity, Food security of the country and protection of TK.

COs- CO3, CO4

Reference Books:

1. Traditional Knowledge System in India, by Amit Jha, 2009.
2. Traditional Knowledge System and Technology in India by Basanta Kumar Mohanta and

Vipin Kumar Singh, Pratibha Prakashan 2012.

3. Traditional Knowledge System in India by Amit Jha Atlantic publishers, 2002
4. "Knowledge Traditions and Practices of India" Kapil Kapoor, Michel Danino

e-Resources:

1. https://www.youtube.com/watch?v=LZP1StpYEPM
2. http://nptel.ac.in/courses/121106003/

Chairperson
Board of Studies (CSE)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Program: B. Tech Computer Science and Engineering **Regulation: R24**

II Year II Semester- Course Structure

S.No	Category	Course Code	Course Title	Hours per Week			
				Lecture	Tutorial	Practical	Credits
1	BS	R24CSBS07	Probability & Statistics	3	0	0	3
2	PC	R24CSPC09	Formal Languages and Automata Theory	3	0	0	3
3	PC	R24CSPC10	Database Management Systems	3	0	0	3
4	HS	R24HS03	Universal Human Values-Understanding Harmony & Human Ethical Conduct	2	0	0	2
5	PC	R24CSPC11	Operating Systems	3	0	0	3
6	PC	R24CSPC12	Database Management Systems Lab	0	0	3	1.5
7	PC	R24CSPC13	Operating Systems Lab	0	0	3	1.5
8	PC	R24CSPC14	Advanced Coding Lab–I	0	0	3	1.5
9	SC	R24CSSC02	Python Programming	0	1	2	2
10	HS	R24HS04	Quantitative aptitude & Logical Reasoning	0	0	2	1
11	MC	R24MC03	Environmental Science	2	0	0	-
			Total	**16**	**01**	**13**	**21.5**
Mandatory Community Service Project Internship of 08 weeks duration during summer vacation							

Category	Courses	Credits
BS- Basic Sciences Course	1	03
PC-Proficitional Core Course	6	13.5
SC- Skill Oriented Course	1	2
HS- Humanity Science and Management Course	2	03
MC-Mandatory Course	1	--
Total	11	21.5

Chairperson
Board of Studies (CSE)

R24CSPC09 **Formal Languages and Automata Theory** **3 0 0 3**

(Common to CSE,CSE (DS),CSE (AI&ML))

Course Objectives:

The courseis designed with the objectiveto:

- Introduce languages, grammars, and computational models
- Understand the foundational principles of Automata Theory&Analyze and design Finite Automata (FA)
- Formulate Regular Expressions and Grammars and demonstrate their equivalence to Finite Automata
- Discussing regular expressions and regular languages
- Construct Context Free Grammars (CFG) to describe context-free languages, and simplify and normalize these grammars intoCNF and GNF.
- Design and analyze Pushdown Automata (PDA) to recognize CFL'S and understand their relationship with CFG
- Explain the capabilities and limitations of Turing Machines&Demonstrate decidability and un decidability for NP Hard problems

CourseoutComes

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

Course Code	Course Outcomes	Mapping with POs and PSOs						Dok
		PO 1	PO 2	PO 3	PO 4	PS01	PS02	
R24CSPC08.1	Comprehendible to understand the concept of abstract machine and their ability to recognize languages based on their computational power and the types of languages they can process	3	3	2	--	3	2	L1, L2
R24CSPC08.2	Understand regular expressions and finite automata	3	3	3	--	3	2	L1, L2 L3
R24CSPC08.3	Develop context free grammars for formal languages.	3	3	3	--	2	3	L1, L2, L3
R24CSPC08.4	Design pushdown automata for context free grammars	3	3	3	--	3	2	L4
R24CSPC08.5	Design Turing machine and Formulate decidability and un-decidability problems and P&NP problems	3	3	3	2	3	3	L4, L5,L6

SYLLABUS

UNIT-I:Finite Automata: **14 hours**

Need of Automata theory, Central Concepts of Automata Theory, Automation, FiniteAutomation, Transition Systems, Acceptance of a String, DFA, Design of DFAs, NFA, Design of NFA,Equivalence of DFA and NFA, Conversion of NFA into DFA, Finite Automata with Є-Transitions,Minimization of Finite Automata, Finite Automata with output-Mealy and Moore Machines, Applicationsand Limitation of Finite Automata.

Self Learning Concepts: Research how DFAs/NFAs are used in lexical analyzer, Network protocol Design & Control system. **CO's–CO1**

UNIT-II:Regular Expressions **12 hours**

Regular Expressions, Regular Sets, Identity Rules, Equivalence of two RE**,** Inter Conversion, Equivalence between FA and RE, Pumping Lemmaof Regular Sets, Grammars, Classification of Grammars, ChomskyHierarchy Theorem, Right and Left Linear Regular Grammars, Equivalence between RG and FA, InterConversion. **CO's–CO2**

Self Learning Concepts: Closure properties of Regular set

UNIT-III:Context Free Grammar **10 hours**

Formal Languages, Context Free Grammar, Leftmost and Rightmost Derivations, Parse Trees, AmbiguousGrammars, Simplification of Context Free Grammars, Normal Forms-CNF and GNF, Pumping Lemma CFG**,** Applications of Context Free Grammars. **COs–CO3**

Self Learning Concepts: Elimination of left recursion & left factoring

UNIT-IV:**Pushdown Automata** **12 hours**

Pushdown Automata, Definition, Model, Graphical Notation, Instantaneous Description, LanguageAcceptance of Pushdown Automata, Design of Pushdown Automata, Deterministic and Non –Deterministic Pushdown Automata, Equivalence of Pushdown Automata and Context Free Grammars,Conversion, Two Stack Pushdown Automata, Application of Pushdown Automata.

CO's–CO4

Self Learning Concepts: Investigate practical applications of PDAs in parsing such as in compiler & syntax analysis.

UNIT-V: Turning Machine **14 hours**

Turning Machine: Definition, Model, Representation of TMs-Instantaneous Descriptions, TransitionTables and Transition Diagrams, Language Acceptanceof a TM, Design of TMs, Types of TMs**,**UniversalTM,Decidable and Un-decidable Problems, Halting Problem of TMs, Post'sCorrespondence Problem, Modified PCP, Classes of P and NP, NP-Hard and NP-Complete Problems. **CO's–CO5**

Self Learning Concepts: Church's Thesis for Turning machine.

Board of Studies:Computer Science and Engineering

Approved in BoS No: 02,--April,2025

Approved in ACM No: 02

Text Books:

1) Introduction to Automata Theory, Languages and Computation, J. E. Hopcroft, R. Motwani and J. D. Ullman, 3rdEdition, Pearson, 2008.
2) Theory of Computer Science-Automata, Languages and Computation, K. L. P. Mishra and N.Chandrasekharan, 3rdEdition, PHI, 2007.

ReferenceBooks:

1) Elements of Theory of Computation, Lewis H.P. &Papadimition C.H., Pearson /PHI
2) Theory of Computation, V. Kulkarni, Oxford University Press, 2013.
3) Theory of Automata, Languages and Computation, Rajendra Kumar, McGraw Hill, 2014

Web References:

1) https://nptel.ac.in/courses/106103070
2) https://nptel.ac.in/courses/106106049

Internal Assessment Pattern

Cognitive Level	Internal Assessment #1(%)	Internal Assessment #2(%)
L1	35%	--
L2	40%	--
L3	25%	15%
L4	--	35%
L5	--	35%
L6	--	15%
Total (%)	**100%**	**100%**

Sample Short and Long Answers questions of Various Cognitive Levels

L1: Remember

1) What is Automata Theory and why is it needed?
2) What are Mealy and Moore Machines?
3) Define Regular Expressions and Regular Sets.
4) List the Identity Rules for Regular Expressions.
5) Define Chomsky Normal Form (CNF) and Greibach Normal Form (GNF).
6) State the Pumping Lemma for CFG.
7) What is an Instantaneous Description of a PDA?
8) Differentiate between Deterministic and Non-deterministic PDA.
9) Define a Turing Machine..

10) Define Decidable and Undecidable Problems

L2: Understand

1) Describe the differences between a DFA and an NFA.
2) Explain the significance of minimizing Finite Automata.
3) Explain the equivalence of two Regular Expressions.
4) Describe the inter-conversion between Finite Automata and Regular Expressions.
5) Explain how Parse Trees represent the derivation of a string.
6) Describe the steps involved in simplifying Context Free Grammars.
7) Describe the relationship between PDA and Context Free Grammars.
8) Explain the process of converting a CFG to a PDA and vice versa.
9) Describe the different types of Turing Machines.
10) Describe the classes of P and NP.

L3: Apply

1) Design a DFA to accept strings with a specific pattern (e.g., strings containing "ab").
2) Construct a Mealy Machine to perform a specific sequence transformation.
3) Convert a given Finite Automaton to a Regular Expression.
4) Use the Pumping Lemma to prove whether a given set is regular or not.
5) Convert a given CFG to CNF. And Convert a given CFG to GNF.
6) Use the Pumping Lemma to prove whether a language is Context Free or no
7) Design a PDA to accept a given Context Free Language.
8) Convert a given CFG to an equivalent PDA.
9) Design a Turing Machine to perform a specific computation (e.g., addition, subtraction).
10) Represent a Turing Machine using Instantaneous Descriptions.

L4: Analysing

1) Analyze the time and space complexity of DFA vs. NFA.
2) Differentiate between the capabilities and limitations of Finite Automata
3) Analyze the equivalence of different Regular Expressions.
4) Analyze the given grammar and classify it according to Chomsky Hierarchy.
5) Analyze a given grammar and determine if it is ambiguous.
6) Compare the advantages and disadvantages of CNF and GNF
7) Analyze the limitations of Pushdown Automata.
8) Compare the complexity of Deterministic and Non-deterministic PDA.
9) Design a Turing Machine to perform a specific computation (e.g., addition, subtraction).
10) Represent a Turing Machine using Instantaneous Descriptions.
11) Show that a given problem is decidable or undecidable.

L5: Evaluating

1) Evaluate whether a given language can be recognized by a Finite Automaton.
2) Compare different designs of Finite Automata for the same language in terms of efficiency.
3) Justify the need for NFA when DFA can recognize the same language.
4) Evaluate the limitations of Regular Expressions in expressing certain languages.
5) Determine whether a given language can be described by a Regular Expression.
6) Evaluate whether a given language can be generated by a Context Free Grammar.
7) Assess the impact of ambiguity in a grammar.
8) Evaluate whether a given language can be generated by a Context Free Grammar.
9) Assess the impact of ambiguity in a grammar.
10) Evaluate the impact of the Halting Problem on computation.
11) Assess the theoretical limits of computation.
12) Justify the importance of P vs. NP problem.

L6: Creating

1) Design a Context Free Grammar for a given language.
2) Develop a parser for a simple Context Free Grammar.
3) Design a Pushdown Automaton with additional features.
4) Develop an application using PDA for tasks like expression evaluation or syntax checking

Chairperson
Board of Studies (CSE)

R24CSPC10 DATABASE MANAGEMENT SYSTEMS 3 0 0 3

(Common to CSE,CSE (DS),CSE (AI&ML))

Course Objectives:

- Train in the fundamental concepts of database management systems, database modelling and design, SQL, PL/SQL, and System implementation techniques.
- Enable students to model ER diagram for any customized applications.
- Provide knowledge on concurrency techniques.
- Understand normalization theory and apply such knowledge to the normalization of a database.
- To learn the principles of systematically designing and using large scale Database Management Systems for various applications.

Course Outcomes

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

Course Code	Course Outcomes (COs)	Mapping with POs and PSOs								BT
		PO1	PO2	PO3	PO4	PO5	PO11	PSO1	PSO2	
R24CSPC09.1	Understand basic concepts of database systems and data models.	3	2	-	1	2	2	3	-	L 1, L2
R24CSPC09.2	Design ER diagrams and apply conceptual modeling for real-world problems like Railway & Hospital systems.	3	3	3	1	2	2	3	1	L1, L2, L3
R24CSPC09.3	Apply relational model concepts and SQL operations to implement simple databases.	3	2	2	1	3	2	3	1	L 2, L3
R24CSPC09.4	Perform schema refinement using normalization techniques.	3	2	2	2	3	2	3	-	L3, L4
R24CSPC09.5	Explain transaction management concepts and implement concurrency control and recovery methods.Understand indexing techniques like B+ Trees and Hash-based indexing for efficient data retrieval.	3	3	2	3	3	2	3	-	L4, L5,L6

SYLLABUS

UNIT I: 15 Hours

Introduction: Database system, Characteristics (Database Vs File System), Database Users, 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.

Entity Relationship Model: Introduction, Representation of entities, attributes, entity set, relationship, relationship set, constraints, sub classes, super class, inheritance, specialization, generalization using ER Diagrams **CO's–CO1**

Self Learning Topics: Database Applications(Real-world applications: Banking, E-commerce, Healthcare, Education,and Social Media)

UNIT II:15 Hours

Relational Model: Introduction to relational model, concepts of domain, attribute, tuple, relation, importance of null values, constraints (Domain, Key constraints, integrity constraints) and their importance, Relational Algebra, Relational Calculus.

BASIC SQL: Simple Database schema, data types, table definitions (create, alter), different DML operations (insert, delete, update).**CO's–CO2**

Self Learning Topics: Structure of a Relational Database

UNIT III:15 Hours

SQL: Basic SQL querying (select and project) using where clause, arithmetic & logical operations, SQL functions(Date and Time, Numeric, String conversion).Creating tables with relationship, implementation of key and integrity constraints, nested queries, sub queries, grouping, aggregation, ordering, implementation of different types of joins, view(updatable and non-updatable), relational set operations.**CO's–CO3**

Self-Learning Topics: Advanced Grouping and Aggregation (GROUP BY with ROLLUP, CUBE, and GROUPING SETS, Filtering groups using HAVING)

UNIT IV:10 Hours

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

CO's–CO4

Self-Learning Topics: Decomposition Deep Dive (Comparing Lossless vs. Lossy Decomposition,

Dependency-Preserving Decomposition: When to compromise)

UNIT V:10 Hours

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

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

CO's–CO5

Self-Learning Topics: Locking & Deadlock Handling – Strategies for handling concurrent transactions.

Board of Studies: Computer Science and Engineering

Approved in BOS No: 02, --, April, 2025

Approved in ACM No: 02

Text Books:

1. Introduction to Database Systems, CJ Date, Pearson.
2. Data base Management Systems, Raghurama Krishnan, Johannes Gehrke, TATA Mc Graw Hill3rd Edition.

Reference Books:

1. Data base Systems design, Implementation, and Management, Peter Rob & Carlos Coronel7thEdition.
2. Fundamentals of Database Systems, Elmasri Navrate Pearson Education.
3. Database Systems - The Complete Book, H G Molina, J D Ullman, J Widom Pearson.
4. Data base System Concepts,5/e, Silberschatz, Korth, TMH.

Web References:

1. http://www.c4learn.com/
2. http://www.geeksforgeeks.org/c/
3. http://nptel.ac.in/courses/122104019/
4. http://www.learn-c.org/
5. https://www.tutorialspoint.com/c programming/

Chairperson
Board of Studies (CSE)

R24HS03 Universal Human Values Understanding Harmony and Ethical Human Conduct 2 0 0

(Common to CSE, CSE (DS) & CSE (AI&ML))

Course Objectives:

The main objectives of the course is to

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

Course Outcomes

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

Course Code	Course Outcomes	Mapping with POs and PSOs								BT
		PO 1	PO 2	PO 6	PO 7	PO 8	PS 09	PS 11	PSO1	
R24HS03.01	Define the terms like Natural Acceptance, Happiness and Prosperity.	1	2	2	3	--	2	3		L1, L2
R24HS03.02	Analyze the coexistence of the self and the body, distinguish between their needs, and apply practices that promote self regulation and overall well-being.	1	2	3	3	2	2	3		L4
R24HS03.03	Relate human values with human relationship and human society.	1	2	2	3	2	2	3		L4
R24HS03.04	Justify the need for universal human values and harmonious existence.	1	2	3	3	3	2	3		L5
R24HS03.05	Develop as socially and ecologically responsible engineers.	2	2	3	3	3	2	3		L3, L6

SYLLABUS

UNIT-I: Introduction to Value Education (6 lectures and 3 tutorials for practice session)

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 Lecture4: 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 **COs- CO1, CO2, CO3**

UNIT-II: Harmony in the Human Being (6 lectures and 3 tutorials for practice session)

Lecture 7: Understanding Human being as the Co-existence of the self and the body. 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 **COs- CO2, CO3**

UNIT-III: Harmony in the Family and Society (6 lectures and 3 tutorials for practice session)

Lecture 13: Harmony in the Family – the Basic Unit of Human Interaction Lecture 14: 'Trust' – the Foundational Value in Relationship Tutorial 7: Practice Session PS7 Exploring the Feeling of Trust Lecture 15: 'Respect' – as the Right Evaluation Tutorial 8: Practice Session PS8 Exploring the Feeling of Respect Lecture 16: Other Feelings, Justice in Human-to-Human Relationship Lecture 17: Understanding Harmony in the Society Lecture 18: Vision for the Universal Human Order Tutorial 9: Practice Session PS9 Exploring Systems to fulfil Human Goal **COs- CO2, CO4, CO5**

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

Lecture 19: Understanding Harmony in the Nature Lecture 20: Interconnectedness, self-regulation and Mutual Fulfilment among the Four Orders of Nature Tutorial 10: Practice Session PS10 Exploring the Four Orders of Nature Lecture 21: Realizing Existence as Co-existence at All Levels Lecture 22: The Holistic Perception of Harmony in Existence Tutorial 11: Practice Session PS11 Exploring Co-existence in Existence. **COs- CO2, CO5**

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

Lecture 23: Natural Acceptance of Human Values Lecture 24: Definitiveness of (Ethical) Human Conduct Tutorial 12: Practice Session PS12 Exploring Ethical Human Conduct Lecture 25: A Basis for Humanistic Education, Humanistic Constitution and Universal Human Order Lecture 26: Competence in Professional Ethics Tutorial 13: Practice Session PS13 Exploring Humanistic Models in Education Lecture 27: Holistic Technologies, Production Systems and Management Models-Typical Case Studies Lecture 28: Strategies for Transition towards Value-based Life and Profession Tutorial 14: Practice Session PS14 Exploring Steps of Transition towards Universal Human Order. **COs- CO3, CO5**

Practice Sessions for UNIT I – Introduction to Value Education

PS1 Sharing about Oneself

PS2 Exploring Human Consciousness

PS3 Exploring Natural Acceptance

Practice Sessions for UNIT II – Harmony in the Human Being

PS4 Exploring the difference of Needs of self and body

PS5 Exploring Sources of Imagination in the self

PS6 Exploring Harmony of self with the body

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

PS7 Exploring the Feeling of Trust

PS8 Exploring the Feeling of Respect

PS9 Exploring Systems to fulfil Human Goal

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

PS10 Exploring the Four Orders of Nature

PS11 Exploring Co-existence in Existence

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

PS12 Exploring Ethical Human Conduct

PS13 Exploring Humanistic Models in Education

PS14 Exploring Steps of Transition towards Universal Human Order

Readings:
Textbook and Teachers Manual

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

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

Reference Books

1. 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.

Chairperson
Board of Studies (CSE)

R24CSPC11 **Operating Systems** **3 0 0 3**

(Computer Science and Engineering)

Course Objectives:

1. To understand operating system overview, functions and services.
2. To analyze process concept, inter process communication and synchronization.
3. Illustrate different conditions for deadlock and their possible solutions.
4. To understand secondary storage management.
5. To understand protection and security.

Course Outcomes

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

Mapping with POs and PSOs															Dok
Course Code	**Course Outcomes (COs)**	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**	
R24CSPC11.1	Describe operating system overview, functions, and services	3	-	-	-	2	-	-	-	-	-	2	3	2	L1, L2
R24CSPC11.2	Analyze process concept, inter-process communication, and synchronization	3	3	-	2	3	-	-	2	2	-	2	3	2	L2, L3
R24CSPC11.3	Analyze process synchronization mechanisms and dead lock handling techniques.	3	3	-	2	3	-	-	-	-	-	2	3	3	L3, L4
R24CSPC11.4	Describe methods for mapping secondary storage management.	3	-	-	2	3	-	-	-	-	-	2	3	2	L4, L5
R24CSPC11.5	Evaluate protection mechanism and security	3	2	3	2	3	2	3	-	-	2	3	-	-	L5, L6

SYLLABUS

UNIT I: **15 Hours**

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

System Structures: Operating System Services, system calls, Types of System Calls, system programs, Operating system structure, Operating system Design and Implementation, Building and Booting an Operating System, Operating system debugging**.** **CO's–CO1**

Self-Learning Topics: Evaluation of Operating systems.

UNIT II: **15 Hours**

Processes: Process Concepts, Process states, Process scheduling, Operations on processes, Inter-process communication,

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

CPU Scheduling: Basic concepts, Scheduling-Criteria, CPU Scheduling Algorithms, Algorithms Evaluation, Thread Scheduling, Multiple processor scheduling. **CO's–CO2**

Self-Learning Topics: Multicore and Multithreading.

UNIT III: **15 Hours**

Process Synchronization: Critical-Section Problem, Peterson's Solution, Synchronization Hardware, Mutex Locks, Semaphores, Monitors, Classic Problems of Synchronization, Synchronization examples

Deadlock: System Model, Deadlock Characterization, Deadlock Prevention, Detection and Avoidance, Recovery from Deadlock. **CO's–CO3**

Self-Learning Topics: Mutual Exclusion: Hardware Support

UNIT IV: **15 Hours**

Memory-Management Strategies: Introduction, Contiguous Memory Allocation, Paging, Page-Table Structure, Swapping, Segmentation.

Virtual Memory Management: Virtual Memory, Demand Paging, Copy-on-write, Page-Replacement Algorithms, Frames Allocation, Thrashing.

File System Interface: File Concepts, Access Methods, Directory Structure, File system Implementation: File-system structure, File-system Operations, Directory implementation, File System Mounting, File Sharing and Protection, Allocation Methods, Free-Space Management. **COs–CO4**

Self-Learning Topics: Memory Partitioning and Hardware Control Structures.

UNIT V: **10 Hours**

Mass Storage: Overview Mass-Storage Structure, Disk Structure, Disk Attachment, Disk Scheduling, RAID Structure

Protection: Goals of Protection, Principles of Protection, Domain of Protection Access Matrix, Implementation of Access Matrix.

Security: Security Problem, Program Threats, System and Network Threats **CO's-CO5**

Self-Learning Topics: Disk Cache and I/O Buffering

Board of Studies : Computer Science and Engineering

Approved in BoS No: 02, --, April, 2025

Approved in ACM No: 02

Expert Talk (To be Delivered by SMEs from Industries) **CO's/ PO's / PSO's**

1. Operating Systems in Cloud Computing (Virtualization Techniques in Industry Cloud Platforms (AWS, Azure, GCP), Containerization (Docker, Kubernetes) vs Virtual Machines, Operating System Abstractions in Cloud Architecture) CO1–CO5, PO1, PO2, PO3, PO11, PSO1, PSO2
2. Real-Time Operating Systems (RTOS) in Industry Applications (RTOS in Automotive Systems (ADAS, AUTOSAR), RTOS for Industrial Automation and Roboticists vs General Purpose OS in Embedded Systems) CO1–CO5, PO1, PO2, PO3, PO11, PSO1, PSO2

3. Operating Systems in Cybersecurity (OS Hardening Techniques in Enterprises, Secure Boot and Trusted Execution Environments, Role of OS in Endpoint Security Solutions) CO1–CO5, PO1, PO2, PO3, PO11, PSO1, PSO2

Text Books:

1. Operating System Concepts, Abraham Silberchatz, Peter B. Galvin, Greg Gagne, 10th Edition, John Wiley
2. Operating Systems- A Concept based Approach, D. M. Dhamdhere, 2nd Edition, Tata McGraw-Hill
3. Operating System Concepts, Abraham Silberchatz, Peter B. Galvin, Greg Gagne, 7th Edition, John Wiley

Reference Books:

1. Operating Systems-Internal and Design Principles, Stallings, 5th Edition, Pearson Education/PHI, 2005
2. Operating Systems -Internals and Design Principles, Stallings W, 9th edition, Pearson, 2018
3. UNIX: Concepts and Applications, Sumit Abha Das, 4th Edition, Tata McGraw-Hill
4. Operating System: A Design-Oriented Approach, Charles Patrick Crowley

Web References:

1. https://archive.nptel.ac.in/courses/106/102/106102132/
2. https://edurev.in/courses/23057_Operating-System
3. https://www.tutorialspoint.com/operating_system

Internal Assessment Pattern

Cognitive Level	Internal Assessment #1(%)	Internal Assessment #2(%)
L1	35	--
L2	40	--
L3	25	25
L4	--	35
L5	--	25
L6	--	15
Total (%)	**100**	**100**

Sample Short and Long Answers questions of Various Cognitive Levels

L1: Remember

1. What is an operating system (OS)?
2. What is meant by an operating system's "kernel"?
3. Define "System Programs" in the context of operating systems.
4. What are system calls in an operating system?
5. Define multithreading.
6. What are the basic criteria for CPU scheduling?
7. What is the critical-section problem in process synchronization?
8. What are mutex locks, and how do they function in process synchronization?
9. Define deadlock in a system.
10. What is a page table, and what role does it play in memory management?
11. What is segmentation in memory management?
12. What are the key components of a mass-storage structure?
13. Define the disk structure and its main components.

L2: Understand

1. Explain the role of the operating system in managing hardware resources.
2. Compare the difference between free and open-source operating systems and proprietary operating systems.
3. Describe the process of booting an operating system from start to finish.
4. Explain the different states that a process can be in and how processes transition between these states.
5. How does inter-process communication (IPC) help processes communicate and synchronize?
6. Explain the difference between a process and a thread.
7. Describe the benefits and challenges of multithreading in modern operating systems.
8. Describe how Peterson's solution works to ensure mutual exclusion.
9. How does deadlock occur in a system? Explain the deadlock characterization in terms of the four necessary conditions.
10. Describe the structure of a page table and its role in translating virtual addresses to physical addresses.
11. Explain how the file system structure is organized to store data efficiently.
12. Describe how the directory structure in a file system helps in managing files.
13. Describe how disk scheduling algorithms manage read/write operations and improve efficiency.

L3: Apply

1. In what way does an operating system structure affect its performance?
2. Demonstrate how system services manage the hardware components of a computer system.
3. How can an operating system be debugged to ensure it works correctly?
4. Given a scenario with multiple processes, how would you apply a CPU scheduling algorithm (e.g., FCFS, SJF) to allocate CPU time?
5. In a system with multiple threads, demonstrate how semaphores can be used to solve the Producer-Consumer problem.
6. Given a scenario of resource allocation, explain how you would detect deadlock using a resource allocation graph.
7. Given a scenario with file system fragmentation, how would you apply free-space management techniques to handle the issue?
8. Apply demand paging in a system with limited physical memory. How would you ensure that the page-replacement algorithm minimizes page faults?
9. Given a system with limited physical memory, demonstrate how paging can be used to manage the memory.
10. Apply a disk scheduling algorithm (e.g., FCFS, SSTF) to a series of disk I/O requests and explain the resulting order of operations.
11. Given a security policy, how would you implement the access matrix model to control access to resources?
12. Given a security threat (e.g., buffer overflow), apply appropriate measures to secure the system against this type of attack.

L4: Analysing

1. Analyze how the structure of an operating system (monolithic vs. microkernel) impacts system performance and security.
2. Analyze how computing environments (such as cloud-based and embedded systems) influence the design and functionality of an operating system.
3. Examine how the operating system handles process synchronization using inter-process communication (IPC) mechanisms like message passing and shared memory.
4. Compare and contrast the different CPU scheduling algorithms (e.g., Round Robin, Shortest Job First, and Priority Scheduling). What are their strengths and weaknesses?
5. Analyze a system where deadlock has occurred. Based on the resource allocation graph, identify the processes involved in the deadlock.
6. Compare and contrast the effectiveness of Peterson's solution and hardware-based synchronization mechanisms in solving the critical-section problem.
7. Analyze how segmentation can affect the allocation of memory for processes with different

memory requirements.

8. Compare the different file access methods (sequential, direct, and indexed) and their suitability for various types of applications.
9. Compare the efficiency of different disk scheduling algorithms (e.g., FCFS, SSTF, SCAN, C-SCAN) in terms of average seek time and throughput.
10. Analyze the advantages and disadvantages of using different RAID levels (e.g., RAID 0, RAID 1, RAID 5) in terms of performance, redundancy, and cost.
11. Break down how the principles of protection are applied in a multi-user system to ensure fairness and security.
12. Analyze the security vulnerabilities that can arise from program threats versus network threats and recommend strategies to mitigate them.

L5: Evaluating

1. Evaluate the impact of free and open-source software on the development of modern operating systems.
2. Based on a particular computing environment (e.g., mobile vs. desktop OS), assess the suitability of specific operating systems for that environment.
3. Evaluate the efficiency of CPU scheduling algorithms in a real-time operating system.
4. Critically evaluate the fairness of the Round Robin scheduling algorithm in a system with diverse process priorities &evaluate the efficiency of the Shortest Job First (SJF) scheduling algorithm in a system with mixed process types (short and long processes).
5. Evaluate the advantages and disadvantages of using semaphores over mutex locks in process synchronization.
6. Assess the effectiveness of deadlock avoidance techniques in real-time systems. Which strategy would you recommend for a highly dynamic environment?
7. Evaluate the use of different page-replacement algorithms (e.g., FIFO, LRU, Optimal) in terms of efficiency and system performance.
8. Evaluate the effectiveness of demand paging versus preloading all necessary pages into memory at the beginning of a program's execution.
9. Evaluate how file protection mechanisms (such as access control lists) can be integrated into a file system to prevent unauthorized access.
10. Evaluate a system's vulnerability to program, system, and network threats and propose a security enhancement plan.
11. Evaluate the security strengths and weaknesses of the access matrix model in terms of scalability and flexibility.

L6: Create

1. Design a new microkernel-based operating system for a smart IoT device. Describe the architecture, essential system services, and justify your design choices with respect to security and performance.
2. Create a CPU scheduling simulator that allows the user to switch between different algorithms (FCFS, SJF, RR, Priority). Include multithreading support and compare the performance for different workloads.
3. Develop a solution to a new synchronization problem in a multi-robot warehouse environment using semaphores or monitors. Justify how your solution avoids deadlocks and ensures resource fairness.
4. Design a hybrid memory management scheme for a custom OS that combines paging and segmentation to optimize memory usage in embedded systems. Provide diagrams and justify your approach.
5. Develop a security framework for a cloud-based file storage system that integrates protection principles, access control, and encryption. Illustrate how your framework defends against both system and network threats.

Chairperson
Board of Studies (CSE)

R24CSPC12 DATABASE MANAGEMENT SYSTEMS LAB 0 0 3 1.5

(Common to CSE, CSE (DS), CSE (AI &ML))

Course Objectives:

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
4. Programming PL/SQL including procedures, functions, cursors and triggers

Course Outcomes

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

Course Code	Course Outcomes	Mapping with POs and PSOs													
		PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PSO1**	**PSO2**	**BT**
R24CSPC 12.1	Utilizing Data Definition Language (DDL), Data Manipulation Language (DML), and Data Control Language (DCL) commands effectively within a database environment	3	2	2	1	3	1	1	1	2	1	2	3	1	L1, L2
R24CSPC 12.2	Constructing and execute queries to manipulate and retrieve data from databases	3	3	2	2	3	1	1	1	2	1	2	3	1	L2, L3
R24CSPC 12.3	Develop application programs using PL/SQL. Establishdatabase connectivity through JDBC (Java Database Connectivity)	2	2	3	2	3	1	1	2	2	2	2	2	3	L3

Board of Studies: Computer Science and Engineering

Approved in BOS No: 02, --, April, 2025

Approved in ACM No: 02

Developing the following programs:

Week 1: **CO's:CO1**

1. Creation, altering and dropping of tables and inserting rows into a table (use constraints while creating tables) examples using SELECT command.

Week 2: **CO's:CO1**

2. Queries (along with sub Queries) using ANY, ALL, IN, EXISTS, NOTEXISTS, UNION,

INTERSECT, Constraints. Example:- Select the roll number and name of the student who secured fourth rank in the class.

Week 3: **CO's:CO1**

3. Queries using Aggregate functions (COUNT, SUM, AVG, MAX and MIN), GROUPBY, HAVING and Creation and dropping of Views.

Week 4: **CO's:CO1**

4. Queries using Conversion functions (to_char, to_number and to_date), string functions (Concatenation, lpad, rpad, ltrim, rtrim, lower, upper, initcap, length, substr and instr), date functions (Sysdate, next_day, add_months, last_day, months_between,least, greatest, trunc, round, to_char, to_date)

Week 5: **CO's:CO1**

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 table and printed for those who secured first class and an exception can be raised if no records were found)

   ii. Insert data into student table and use COMMIT, ROLLBACK and SAVEPOINT in PL/SQL block.

Week 6: **CO's:CO2**

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.

Week 7: **CO's:CO2**

7. Program development using WHILE LOOPS, numeric FOR LOOPS, nested loops using ERROR Handling, BUILT –IN Exceptions, USE defined Exception RAISEAPPLICATION ERROR.

Week 8: **CO's:CO2**

8. Programs development using creation of procedures, passing parameters IN and OUT of PROCEDURES.

Week 9: **CO's:CO2**

9. Program development using creation of stored functions, invoke functions in SQL Statements and write complex functions.

Week 10: **CO's:CO2**

10. Develop programs using features parameters in a CURSOR, FOR UPDATECURSOR, WHERE CURRENT of clause and CURSOR variables.

Week 11: **CO's:CO3**

11. Develop Programs using BEFORE and AFTER Triggers, Row and Statement Triggers and

INSTEAD OF Triggers

Week 12: **CO's:CO3**

12. Create a table and perform the search operation on table using indexing and no indexing techniques.

Week 13: **CO's:CO3**

13. Write a Java program that connects to a database using JDBC

Week 14: **CO's:CO3**

14. Write a Java program to connect to a database using JDBC and insert values into it

Week 15: **CO's:CO3**

15. Write a Java program to connect to a database using JDBC and delete values from it

Reference Books:

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
4. RamezElmasri, Shamkant, B. Navathe, "Database Systems", Pearson Education, 6th
5. Edition, 2013.
6. Database Principles Fundamentals of Design Implementation and Management, 10th
7. edition, Corlos Coronel, Steven Morris, Peter Robb, Cengage Learning, 2022

Online Learning Resources:

1. http://www.scoopworld.in
2. http://vlabs.iitb.ac.in/vlabs-dev/labs/dblab/index.php

Chairperson
Board of Studies(CSE)

R24CSPC12 **Operating Systems Lab** **0 0 3 1.5**

(Computer Science and Engineering)

Course Objectives:

1. To understand the basic system calls related to process
2. To implement various CPU scheduling algorithms
3. To implement memory management and virtual memory techniques
4. To implement file management and disk scheduling algorithms

Course Outcomes

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

Mapping with POs and PSOs					DoK
Course Code	**Course Outcomes (COs)**	**PO1**	**PO2**	**PO3**	
R24CSPC12.1	Know the basic system calls related to process	3	2	1	L2
R24CSPC12.2	Simulate CPU scheduling algorithms, disk scheduling algorithms	3	3	2	L3
R24CSPC12.3	Understand memory management and virtual memory techniques, file management techniques	3	3	3	L3

Board of Studies: Computer Science and Engineering

Approved in BOS No: 02, --, April, 2025

Approved in ACM No: 02

Developing the following programs:

Week 1:

1. Practicing of Basic UNIX Commands. **CO's:CO1**

Week 2:

2. Write programs using the following UNIX operating system calls fork, exec, getpid, exit, wait, close, stat, opendir and readdir **CO's:CO1**

Week 3:

3. Simulate the following CPU scheduling algorithms a) FCFS b) SJF c) Priority d) Round Robin **CO's: CO2**

Week 4:

4. Control the number of ports opened by the operating system with a) Semaphore b) Monitors. **CO's: CO2**

Week 5:

5. Write a program to solve producer-consumer problem using Semaphores. **CO's: CO2**

Week 6:

6. Implement the following memory allocation methods for fixed partition a) First fit b) Worst fit c) Best fit **CO's:CO3**

Week 7:

7. Simulate the following page replacement algorithms a) FIFO b) LRU c) LFU **CO's:CO3**

Week 8:

8. Simulate Paging Technique of memory management. **CO's:CO3**

Week 9:

9. Implement Bankers Algorithm for Dead Lock avoidance and prevention **CO's: CO2**

Week 10:

10. Simulate the following file allocation strategies a) Sequential b) Indexed c) Linked **CO's:CO5**

Week 11:

11. Simulate the following Disk scheduling algorithms. a) FCFS b) SSTF c) SCAN **CO's:CO4**

Week 12:

12. Simulate the following Disk scheduling algorithms. a) C-SCAN b) LOOK c) C-LOOK **CO's:CO4**

Textbooks:

1. Operating System Concepts, Abraham Silberchatz, Peter B. Galvin, Greg Gagne, 7th Edition, John Wiley
2. Operating Systems- A Concept based Approach, D. M. Dhamdhere, 2nd Edition, Tata McGraw-Hill

Reference Books:

1. Operating Systems-Internal and Design Principles, Stallings, 5th Edition, Pearson Education/PHI, 2005
2. UNIX: Concepts and Applications, Sumitabha Das, 4th Edition, Tata McGraw-Hill
3. Operating System: A Design-Oriented Approach, Charles Patrick Crowley
4. Operating Systems: A Modern Perspective, Gary J. Nutt
5. Design of the Unix Operating Systems, Maurice J. Bach

Chairperson
Board of Studies (CSE)

R24CSPC13 **Advanced Coding Lab** **0 0 3 1.5**

(Computer Science and Engineering)

Course Objectives:

1. The objective of this course is to provide students with hands-on experience and advanced knowledge in programming through the use of multiple languages, including C (Data Structures (DS)), and JAVA.
2. The course aims to develop problem-solving skills, enhance algorithmic thinking, and train students in the practical implementation of core programming concepts across different paradigms and environments.

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

Course Code	Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	Dok
R24CSPC 13.1	Apply logical techniques such as pattern recognition, sequencing, and classification to solve problems.	3	3	2	2	2	-	-	-	1	1	-	3	2	L3,L 4
R24CSPC 13.2	Implement and manage linear and nonlinear data structures (like stacks, queues, linked lists, trees, and graphs).	3	3	3	2	3	-	-	-	2	1	1	3	2	L4,L 6
R24CSPC 13.3	Solve complex real-world problems using object-oriented programming concepts in Java.	3	3	3	3	3	2	1	1	3	3	3	2	3	L5,L 6

Board of Studies: Computer Science and Engineering

Approved in BOS No: 02, --, April, 2025

Approved in ACM No: 02

Developing the following programs:

C AND DATA STRUCTURES

WEEK-1 **CO's: CO1**

1. The given function has a string (str) and two characters, ch1 and ch2. Execute the function in such a way that str returns to its original string, and all the events in ch1 are replaced by ch2, and vice versa.

Replace character (Char str1, Char ch1, Int 1, Char ch2)

Assumption

This string has only alphabets that are in lower case**.**

Example

Input:

str: apples

ch1: a

ch2: p

Output:

paales

Explanation: All the 'a' in the string is replaced with 'p'. And all the 'p's are replaced with 'a'.

WEEK-2 **COs: CO1**

2. In an enchanted garden, flowers bloom in various colors, and each color has a specific point value. Your task is to calculate the total points based on the flowers picked by a gardener.

   Input Format

   A list of integers representing the points for each flower.

   Output Format

   An integer representing the total points earned from the flowers.

   Example

   Input: [5, 3, 8, 2]

   Output: 18

WEEK-3 **CO's: CO2**

3. **Count nodes in circular linked list**

Algorithm for count nodes in circular linked list

A-Accept the head pointer in the function

B-Declare the count variable which maintains the count of nodes and initialize it to zero.

C-Traverse the link till next node is the head node

D- Define a Node class which represents a node in the list. It has two properties data and next which will point to the next node.

E-Define another class for creating the circular linked list and it has two nodes: head and tail. It has two methods: add () and display ().

WEEK-4 **CO's: CO2**

4. Implement a binary search tree (BST) and perform operations such as insertion, deletion, and searching.

WEEK-5 **CO's: CO2**

5. Implement a scenario-based program for a social network using graph data structures.

WEEK-6 **CO's: CO1**

6. Develop a scenario-based program to simulate a ticket booking system using queues.

WEEK-7 **CO's: CO2**

7. Implement graph data structure and perform depth-first search (DFS) and breadth-first search (BFS).

WEEK-8 **CO's : CO1**

8. The function accepts string str of size n as an argument. Implement the function which returns 1 if given string str is valid password else 0.
   str is a valid password if it satisfies the below conditions.

   - – At least 4 characters
   - – At least one numeric digit
   - – At Least one Capital Letter
   - – Must not have space or slash (/)
   - – Starting character must not be a number

   Assumption:

   Input string will not be empty.

   Example:

   Input 1:

   aA1_67

   Input 2:

   a987 abC012

   Output 1:

   1

   Output 2:

   0

WEEK-9 **CO's: CO1**

9. The function accepts an integer array 'arr', its length and two integer variables 'num' and 'diff'. Implement this function to find and return the number of elements of 'arr' having an absolute difference of less than or equal to 'diff' with 'num'.
   Note: In case there is no element in 'arr' whose absolute difference with 'num' is less than or equal to 'diff', return -1.

   Example:

   Input:

   - arr: 12 3 14 56 77 13
   - num: 13
   - diff: 2

   Output: 3

WEEK-10 **CO's: CO1**

10. Write a function to validate if the provided two strings are anagrams or not. If the two strings are anagrams, then return 'yes'. Otherwise, return 'no'.

    Input:

    Input 1: 1st string

    Input 2: 2nd string

    Output:

    (If they are anagrams, the function will return 'yes'. Otherwise, it will return 'no'.)

    Example

    Input 1: Listen

    Input 2: Silent

    Output:

    Yes

Programming through JAVA

WEEK-11 **CO's: CO3**

11. **Problem Statement:**

    In a classroom some of the seats are already occupied by students and only a few seats are available in the classroom. The available seats are assumed as **r** and **n** number of students is looking for the seat. We need to find in how many different permutations n number of students can sit on r number of chairs.

Algorithm

- Input number of students in n
- Input number of seats in r
- Use permutation formula { factorial(n) / factorial(n-r) }
- Print Output

WEEK-12 **CO's: CO3**

12. Find all possible Palindromic Partitions of the given String in Java.

WEEK-13 **CO's: CO3**

13. Write a Java program to remove a specific element from an array.

WEEK-14 **CO's: CO3**

14. Develop a Java application that demonstrates comprehensive file handling operations. The application should perform the following tasks:
    - Create a new file
    - Read the contents of the file and display it on the console.

- Append additional text to the existing content of the file.

  Delete the original file

WEEK-15 **CO's: CO3**

15. Can a number be expressed as a sum of two prime numbers using java

Here, we will discuss program to check whether a number be expressed as a sum of two prime numbers using java. we will ask the user to enter a positive integer and check whether that number can be expressed as the sum of two prime numbers. If that number can be expressed as sum of two prime numbers then print the numbers in the output and else print the statement that the number cannot be expressed as a sum of two prime numbers.

WEEK-16 **CO's: CO3**

16. Write a Java program to compare two files lexicographically.

According to Wikipedia:

In mathematics, the lexicographic or lexicographical order (also known as lexical order, dictionary order, alphabetical order or lexicographic (al) product) is a generalization of the way the alphabetical order of words is based on the alphabetical order of their component letters. This generalization consists primarily in defining a total order over the sequences (often called words in computer science) of elements of a finite totally ordered set, often called alphabet.

Textbooks:

1. Data structures Using C Second Edition,Reema Thareja.
2. Programming Problem Solving,Reema Thareja.
3. Herbert Schildt, Java The complete reference, 11th Edition, McGrawHill, 2019.
4. Timothy budd, An introduction to object-oriented programming, 3rd Edition, Pearson, 2009.

Reference Books:

1. Lets Us C, Yeswanth Kanethkar.
2. Data Structures,S.Chand And Company Limited,Dr.KV Sambasivarao.
3. Y. Daniel Liang Introduction to Java Programming Comprehensive Version, 10th Edition, Pearson, 2015.
4. Bruce Eckel, Thinking in Java, 4th Edition, Prentice Hall, 2006.

Chairperson
Board of Studies (CSE)

R24CSSC01 **Python Programming** **0 1 2 2**

(Computer Science and Engineering)

Course Objectives:

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

Course Outcomes

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

Course Code	Course Outcomes	Mapping with POs and PSOs			
		PO1	PO2	PO3	Dok
R24ES03.1	Introduce core-programming concepts of Python programming language.	3	3	3	L1
R24ES03.2	Develop, run and manipulate Python programs using Functions, Core data structures like Lists, Dictionaries, and use of Strings Handling methods	3	3	3	L2
R24ES03.3	Demonstrate about Python data structures like Tuples, Sets and dictionaries	3	3	3	L2
R24ES03.4	Develop, run and manipulate Python programs using File Operations and concepts of object-oriented programming	3	3	3	L3
R24ES03.5	Understand Data Science, Numpy, Pandas and working with XML, JSON and other file formats.	3	3	3	L3

Board of Studies : Computer Science and Engineering
Approved in BOS No: 02, --, April, 2025
Approved in ACM No: 02

Developing the following programs:

UNIT-I: **CO's:CO1**

History of Python Programming Language, Thrust Areas of Python, Installing Anaconda Python Distribution, Installing and Using Jupyter Notebook,Features, Limitations, advantages, and applications of python.

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.
7. Write how to Handle specific exceptions – like division of a number by zero

UNIT-II: **CO's:CO2**

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,Lambda Functions: Syntax and usage of lambda functions..

Strings: Creating and Storing Strings, Basic String Operations, Accessing Characters in String by Index Number, String Slicing and Joining, String Methods, Formatting Strings, Regular Expressions (Regex)

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 Python program to square each element in a list using a lambda function.
2. Write a program to define a function with multiple return values.
3. Write a program to define a function using default arguments.
4. Write a program to find the length of the string without using any library functions.

5. Write a program to check if the substring is present in a given string or not.
6. Write a program to perform the given operations on a list:
    i. Addition
    ii. Insertion
    iii. Slicing
7. Write a program to perform any 5 built-in functions by taking any list.

UNIT-III: CO's:CO3

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, Defaultdict and OrderedDict (from collections).

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.
6. Write a program using defaultdict to count the frequency of each fruit in the given list:['apple', 'banana', 'apple', 'orange', 'banana', 'apple']

UNIT-IV: CO's:CO4

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: **CO's:CO5**

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. Gowri shankar 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 Links:

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

Web References:

1. https://www.geeksforgeeks.org/python-programming-language-tutorial/
2. https://www.python.org/about/gettingstarted/
3. https://www.w3schools.com/python/

Sample Short and Long Answers questions of Various Cognitive Levels

L1: Remember

1. Define identifiers and give two examples.
2. List any three features of Python.
3. What is the use of the type() function.
4. Define a function. What is a lambda function?
5. What are default and keyword arguments?
6. List any four string methods in Python.
7. Define a dictionary. How is it different from a list?
8. What is a frozenset?
9. What is the syntax of zip()?
10. What is a constructor in Python?
11. List any three file handling functions.
12. Define encapsulation.
13. What is a NumPy array?
14. Define a DataFrame in pandas.
15. Mention any two data visualization functions from matplotlib.

L2: Understand

1. Explain the significance of indentation in Python.
2. Differentiate between is and == with examples.
3. Explain the flow of execution for the if...elif...else structure.
4. Differentiate between *args and **kwargs.
5. Explain string slicing and joining with examples.
6. Explain the scope and lifetime of a variable in Python
7. Explain the relation between tuples and dictionaries.
8. Describe the purpose of defaultdict and OrderedDict.
9. Differentiate between class and object.
10. Explain the difference between class attributes and instance attributes.
11. Explain the difference between .ndim, .shape, and .size in NumPy.
12. Describe the process of reading JSON data in Python.

L3: Apply

1. Write a Python program to swap two variables without a temporary variable.
2. Write a program to print the largest of three numbers using conditional statements.
3. Write a Python program using a lambda function to square each element of a list.
4. Write a function that returns both sum and product of two numbers.
5. Write a program to check if a substring exists in a string.
6. Write a program to count vowels in a string using only string and set operations.

7. Write a program to check if a given key exists in a dictionary.
8. Write a program to concatenate two tuples.
9. Write a Python program to read from a file and print its contents in reverse line order.
10. Write a class representing a rectangle with methods to calculate area and perimeter.
11. Write a Python program to create a NumPy array and perform slicing.
12. Create a dictionary with lists and convert it into a pandas DataFrame.

L4: Analysing

1. Compare the use of for and while loops with scenarios.
2. Analyze how operator precedence affects the outcome of expressions in Python.
3. Analyze how positional arguments differ from keyword arguments in function calls.
4. Compare string formatting methods: %, .format(), and f-strings.
5. Compare the use of sets and dictionaries in Python.
6. Analyze the output when applying slicing on a tuple with an example.
7. Compare inheritance and polymorphism with suitable class examples.
8. Analyze the structure and behavior of a file object during read and write operations.
9. Analyze how data is structured differently in NumPy and pandas.
10. Compare JSON and XML data formats with Python parsing tools.

L5: Evaluating

1. Evaluate the use of dynamic typing in Python. What are its pros and cons?
2. Assess the effectiveness of Anaconda and Jupyter Notebook for beginner programmers
3. Evaluate the usefulness of using regular expressions for string validation.
4. Justify the use of built-in functions over writing manual logic
5. Evaluate the use of immutable data types (tuples, frozensets) in large applications.
6. Assess the performance difference between regular dict and defaultdict when counting frequencies.
7. Evaluate the usefulness of OOP in Python compared to procedural programming.
8. Judge whether using binary files or text files is better for storing structured data.
9. Evaluate the benefits of using pandas over lists/dictionaries for data manipulation.
10. Assess the effectiveness of functional programming in handling large-scale data pipelines.

Chairperson
Board of Studies (CSE)

Course Title: Quantitative Aptitude & Logical reasoning	Course Code: R24HS04
Teaching Scheme(L:T:P): 0:1:2	**Credits:**2
Type of Course: Tutorial+ Practical	
Continuous Internal Evaluation: 30Marks	**Semester End Exam:** 70Marks
Prerequisites: To succeed in the Quantitative Aptitude & Logical reasoning course, students should have a basic understanding arithmetic, algebra, and geometry from school-level mathematics. Analytical thinking and English comprehension skills for interpreting logical patterns and problems.	

Course objective:

- Build a strong foundation in quantitative aptitude and logical reasoning.
- Enhance problem-solving skills for topics like percentages, profit and loss, time and work, and logical reasoning puzzles.
- Improve speed, accuracy, and critical thinking for efficient problem-solving.
- Prepare students for competitive exams and real-world applications of math and logic.

Course outcome:

COs	PO1	PO2	PO3	PO4	PO5	PO11	PSO1	BT LEVEL
CO1: Mastery of Key Concepts – Understand number systems, percentages, time/work, profit/loss, and series completion.	3	2	1	2	1	-	1	L2
CO2: Improved Problem-Solving Skills – Solve mathematical and logical problems with speed and accuracy.	3	3	2	3	1	-	1	L3
CO3: Enhanced Analytical Thinking – Develop critical thinking for reasoning puzzles and real-life challenges.	2	3	2	3	2	-	1	L4
CO4: Competitive Exam Preparedness – Be well-prepared for exams requiring aptitude and reasoning.	3	2	1	2	2	1	2	L3

Aptitude:

Unit 1: Number System: Speed Maths, Numbers, Factors, Prime & Co-Primes, LCM, HCF, Divisibility rules, finding unit place digit and last two digits of an expression.

Averages and Ages: Average of different groups, change in averages by adding, deleting and replacement of objects, problems on ages.

Ratio, Proportion and Variations: Definition of Ratio, Definition of Proportion, Types of ratios, Types of proportions, mixture model, age model, salary model questions, Direct and indirect

proportion. Allegation and mixtures: Allegation rule. **COs- CO1**

Unit 2: Percentages: Converting fractions and decimal into percentages, successive percentage, populations, expenditure and savings.

Profit and loss: Relation between Cost price and Selling price, Discount and Marked price, Gain or Loss percentages on selling price

Simple and Compound Interest: Problems on Interest (I), Amount (A), Principal (P)and Rate of Interest(R), Difference between the simple interest and compound interest for 2 and 3years.

COs- CO1, CO2

Unit 3: Time and Work: Men and Days, Work and Wages, Hours and Work, Alternate days concept, Chain rule.

Time and Distance: Difference between the average and relative speeds, reaching the destination late and early, Stopp age time per hour, time and distance between two moving bodies.

Trains, Boats and Streams: Train crossing man, same and opposite directions, Speed ofboat and stream. **COs- CO2, CO3**

Logical Reasoning

Unit 4: Series completion: Number series, Alphabet series and letter series.

Blood Relations: Defining the various relations among the members of a family, Solving Blood Relation Puzzles by using symbols and notations. Problems on Coded relations.

Coding and Decoding: Letter coding, Number coding, Number to letter coding, Matrix coding, Substitution, Mixed letter coding, Mixed number coding, deciphering individual letter codes by analysis.

Direction sense test: Sort of directions in puzzles distance between two points, problems on shadows, Application of triangular triplets. **COs- CO3, CO4**

Unit 5: Clocks: Relation between minute-hour hands, anglevs. time, exceptional cases in clocks

Calendars: Definition of a Leap Year, Finding the odd days, finding the day of any random calendar date, repetition of calendar years. **COs- CO3, CO4**

Text Books:

1. R.S.Aggarwal“QuantitativeAptitude”,Reviseded.,SChandpublication,2017 ISBN:8121924987
2. R.S.Aggarwal “Verbal- Non verbal Reasoning”,Reviseded.,SChandpublication,2017

E- resources:

1. https://www.indiabix.com/aptitude/questions-and-answers/
2. https://www.tutorialspoint.com/quantitative_aptitude/
3. https://www.careerbless.com/aptitude/qa/home.php

Chairperson
Board of Studies (CSE)

Course Title: ENVIRONMENTAL SCIENCE	**Course Code:** R24MC03
Teaching Scheme(L:T:P):2:0:0	**Credits:-**
Type of Course: Lecture	
Continuous Internal Evaluation: 30Marks	**Semester End Exam:** 70Marks
Pre requisites: To succeed in the Environmental Science course, students should have a basic knowledge of high school-level biology, physics, and chemistry. Awareness of environmental issues like pollution, climate change, and conservation is helpful. Interest in sustainability and nature will enhance understanding and engagement.	

Course Objectives:

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

Course Outcomes:

After completion of the course, students will be able to

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO11	PSO1	BT LEVEL
CO1: Understand the scope, importance, and multidisciplinary nature of environmental studies, and analyze the exploitation of natural resources.	2	3	1	2	2	1	2	2	L2,L4
CO2: Describe the structure, function, and energy flow in ecosystems, and understand the importance of biodiversity and its conservation.	3	2	2	3	2	1	3	1	L2,L3
CO3: Evaluate the causes, effects, and control measures of different types of environmental pollution, and understand the strategies for solid waste management.	3	3	2	3	3	2	2	1	L4,L5
CO4: Examine the concept of sustainable development, urban environmental issues, and the role of	3	2	3	3	3	2	3	2	L4,L5

environmental ethics and legislation in protecting the environment.									
CO5: Analyze the relationship between human population growth and environmental degradation, and evaluate the role of population management and health programs in sustainable development.	3	2	2	2	3	2	3	1	L4,L5

SYLLABUS

UNITI

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

Natural Resources : Renewable and non-renewable resources – Natural resources and associated problems – Forest resources – Use and over – exploitation, deforestation, case studies–Timberextraction–Mining,damsandothereffectsonforestandtribalpeople–Water resources – Use and over utilization of surface and ground water – Floods, drought, conflicts over water, dams – benefits and problems – Mineral resources: Use and exploitation, environmentaleffectsofextractingandusingmineralresources,casestudies–Foodresources: World food problems, changes caused by agriculture and overgrazing, effects of modern agriculture, fertilizer-pesticide problems, water logging, salinity, case studies. – Energy resources. **COs- CO1**

UNITII

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

a. Forest ecosystem.

b. Grass land ecosystem

c. Desert ecosystem.

d. Aquatic ecosystems(ponds, streams, lakes, rivers, oceans, estuaries)

BiodiversityanditsConservation: Introduction0Definition:genetic,speciesandecosystem diversity – Bio-geographical classification of India –Value of biodiversity: consumptive use, Productive use, social, ethical, aesthetic and option values – Biodiversity at global, National and local levels – India as a mega-diversity nation – Hot-sports of biodiversity – Threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts – Endangered and endemic species of India – Conservation of biodiversity: In-situ and Ex-situ conservation of biodiversity. **COs- CO2**

UNITIII

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

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

Solid Waste Management: Causes**,** effects and control measures of urban and industrial wastes – Role of an individual in prevention of pollution – Pollution case studies – Disaster management: floods, earthquake, cyclone and landslides. **COs- CO3**

UNITIV

Social Issues and the Environment: From Unsustainable to Sustainable development – Urban problems related to energy – Water conservation, rain water harvesting, water shed management – Resettlement and rehabilitation of people; its problems and concerns.

Case studies – Environmental ethics: Issues and possible solutions – Climate change, global warming, acid rain, ozone layer depletion, nuclear accidents and holocaust. Case Studies – Wasteland reclamation. – Consumerism and waste products.

Environment Protection Act. – Air (Prevention and Control of Pollution) Act. – Water (Prevention and control of Pollution) Act – Wildlife Protection Act – Forest Conservation Act – Issues involved in enforcement of environmental legislation – Public awareness.

COs- CO4

UNITV

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

Field Work: Visit to a local area to document environmental assets River/forest grassland/hill/mountain – Visit to a local polluted site-Urban/Rural/Industrial/Agricultural Study of common plants, insects, and birds – river, hill slopes, etc. **COs- CO5**

Textbooks:

1. Textbook of Environmental Studies for Undergraduate Courses ErachBharucha for University Grants Commission, Universities Press.
2. Palaniswamy, "Environmental Studies", Pearson education
3. S.AzeemUnnisa, "Environmental Studies"Academic Publishing Company
4. K.RaghavanNambiar, "Textbook of Environmental Studies for Undergraduate Courses as per UGC model syllabus", Scitech Publications (India), Pvt. Ltd.

References:

1. DeekshaDave and E.SaiBabaReddy, "Textbook of Environmental Science", Cengage Publications.
2. M.AnjiReddy, "Textbook of Environmental Sciences and Technology", BS Publication.
3. J.P.Sharma, Comprehensive Environmental studies, Laxmi publications.
4. J.Glynn Henry and Gary W.Heinke, "Environmental Sciences and Engineering", Prentice Hall of India Private limited
5. G.R.Chatwal, "A Textbook of Environmental Studies "Himalaya Publishing House
6. Gilbert M. Masters and Wendell P. Ela, "Introduction to Environmental Engineering and Science, Prentice Hall of India Private limited.

Chairperson
Board of Studies (CSE)