4 Year B.Tech Program
of
Computer Science and Engineering

R-24 Regulations & Scheme
(w.e.f. 2024-2025)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

BAPATLA ENGINEERING COLLEGE :: BAPATLA

(AUTONOMOUS UNDER ACHARYA NAGARJUNA UNIVERSITY)

(SPONSORED BY BAPATLA EDUCATION SOCIETY)

BAPATLA - 522102 GUNTUR DISTRICT, A.P.

www.becbapatla.ac.in

INSTITUTE VISION

To build centers of excellence, impart high quality education and instill high standards of ethics and professionalism through strategic efforts of our dedicated staff, which allows the college to effectively adapt to the ever-changing aspects of education.
To empower the faculty and students with the knowledge, skills and innovative thinking to facilitate discovery in numerous existing and yet to be discovered fields of engineering, technology and interdisciplinary endeavours.

INSTITUTE MISSION

Our Mission is to impart the quality education at par with global standards to the students from all over India and in particular those from the local and rural areas.
We continuously try to maintain high standards so as to make them technologically competent and ethically strong individuals who shall be able to improve the quality of life and economy of our country.

PROGRAM OUTCOMES

PO1-Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2-Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3-Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4-Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5-Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO6-The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7-Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO8-Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO9-Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO10-Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO11-Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Academic Rules & Regulations
(R24 Regulations)
Regulations for Four Year Bachelor of Technology (B.Tech)
Degree Program for the Students Admitted from the Academic Year 2024-25

1. Admissions

The sanctioned intake in a particular B.Tech program comprises of Category-A (presently 70%) and Category-B (30%) seats which is supplemented with supernumerary (10%) EWS seats. Admissions for the Category-A seats and the supernumerary seats shall be made by the Andhra Pradesh (A.P.) State Government based on the merit rank obtained by the student in the common entrance examination conducted. Admissions for the remaining Category-B seats shall be made by the college in accordance with the guidelines issued by the A.P. State Government.

2. Medium of Instruction and Examination:

The medium of instruction of the entire B.Tech undergraduate program in Engineering and Technology and the examinations will be in English only.

3. Minimum Instruction Days:

A semester comprises of 90 working days and the year is divided into two semesters.

4. Award of B.Tech. Degree:

A student will be declared eligible for the award of the B.Tech degree if he / she fulfills the following academic regulations:

a. The student pursues a program of study in B.Tech for four academic years and in not more than eight academic years. A lateral entry student pursues a program of study for 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 not be counted in the maximum time permitted for graduation.

b. The student registers for 160 credits and secures all 160 credits. However, a lateral entry student registers for 121 credits and secures all the 121 credits from III semester to VIII semester of regular B. Tech program.

c. **Award of B. Tech degree with Minor:**

The student secures an additional 16 credits from minor stream chosen and fulfills all the requisites of a B.Tech program i.e. secures 160 (Regular program) / 121 (Lateral Entry program) credits.
Minor is to be completed simultaneously with B.Tech program. Registering for a Minor degree is optional.

d. **Award of B.Tech degree with Honors:**

The student secures an additional 16 credits fulfilling all the requisites of B.Tech program i.e. secures 160 (Regular program) / 121 (Lateral Entry program) credits. Registering for Honors is optional and is to be completed simultaneously with B.Tech program.

Students can register either for Honors stream or Minor stream.

4.1 Guidelines for offering a Minor in a discipline:

Minor in a discipline concept is introduced in the curriculum for all conventional B. Tech programs in which it offers a Major Program (B.Tech degree). The main objective of Minor in a discipline is to provide additional learning opportunities for academically motivated students, and it is an optional added feature of the B. Tech. program.

a. Students who are desirous of pursuing their special interest areas other than the chosen discipline of Engineering may opt for additional courses in Minor specialization groups offered by a department other than their parent department.
b. The BOS concerned shall identify as many tracks as possible in the areas of emerging technologies and industrial relevance / demand. For example, the Minor tracks can be the fundamental courses in CSE, ECE, EEE, CE, ME etc or industry tracks such as Artificial Intelligence (AI), Machine Learning (ML), Data Science (DS), Robotics, Electric vehicles, VLSI etc.
c. The list of disciplines / branches eligible to opt for a particular industry relevant minor specialization shall be clearly mentioned by the respective BOS.
d. There shall be no limit on the number of programs offered under Minor. The Institution can offer Minor programs in emerging technologies based on expertise in the respective departments or can explore the possibility of collaborating with the relevant industries/agencies in offering the program.
e. The concerned BOS shall decide on the minimum enrolments for offering Minor program by the department. If a minimum enrolments criterion is not met, then the students may be permitted to register for the equivalent MOOC courses as approved by the concerned Head of the department in consultation with BOS.
f. A student shall be permitted to register for Minor program at the beginning of 4th semester provided that the student must have acquired a minimum of **7.0 CGPA** up to the end of 3rd semester without any backlogs. A CGPA of 7.0 must be maintained in the subsequent semesters without any backlog to keep the Minor registration active.
g. A student must earn an additional 16 credits in the specified area to be eligible for the award of B. Tech degree with Minor. This is in addition to the credits essential for obtaining the Undergraduate degree in Major discipline (i.e. 160 credits for regular students and 121 credits for Lateral Entry students). The concerned BOS shall finalize the modalities to earn the above credits.
h. For securing the above additional 16 credits, the students must register and complete three courses of 4 credits each offered by the department concerned. These 3 courses must contain a laboratory component also (i.e. Embedded course having three lecture hours and two practical hours). The balance of 4 credits may be secured through two MOOCs courses of 2 credits each or an embedded course

offered by the department.

i. Courses that are used to fulfil the student's primary Major may not be double counted towards the Minor. Courses with content substantially equivalent to courses in the student's primary Major may not be counted towards the Minor.
j. The student registered for Minor shall pass in all subjects that constitute the requirement for the Minor program. No class / division (i.e., second class, first class, distinction, etc.) shall be awarded for Minor degree programme
k. If a student drops (or terminated) from the Minor program, they cannot convert the earned credits into free or core electives; they will remain extra.
l. In case a student fails to meet the CGPA requirement for B.Tech degree as per clause 15.f or drops (or terminated) from the Minor program, he/she will be dropped from the list of students eligible for Minors degree and they will receive B.Tech degree only. However, such students will receive a separate grade sheet mentioning the additional courses completed by them.
m. The Minor will be mentioned in the Major degree certificate only. No additional degree certificate will be given for Minor degree.
n. Transfer of credits from Minor to regular B. Tech degree and vice-versa shall not be permitted
o. Minor must be completed simultaneously with a Major degree program. A student cannot earn the Minor degree after he / she has already earned bachelor's degree.
p. The documents for registration of Minor courses are available from the departments and college website.

4.2 Guidelines for offering a Honors in a Discipline:

The objective of introducing B.Tech (Honors) is to facilitate the students to choose additional 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.

Honors is introduced in the curriculum of all B. Tech. programs offering a Major degree and is applicable to all B. Tech (Regular and Lateral Entry) students admitted in Engineering & Technology. Students are eligible to opt for Honors program offered by the same Department / Discipline.

a. Students who are desirous of pursuing special interest / advanced areas of their discipline of Engineering may opt for additional courses as part of Honors programs offered by the parent department.
b. The BOS concerned shall identify as many tracks as possible in the areas of emerging technologies and industrial relevance / demand.
c. A student shall be permitted to register for Honors program at the beginning of 4th semester provided that the student must have acquired a minimum of **7.5 CGPA** up to the end of 3rd semester without any backlogs. A CGPA of 7.5 must be maintained in the subsequent semesters without any backlog to keep the Honors registration active.
d. A student must earn additional 16 credits for award of B.Tech. (Honors) degree from the same branch / department / discipline registered for Major degree. This is in addition to the credits essential for obtaining the Undergraduate degree in Major discipline (i.e., 160 credits for regular students and 121 credits for Lateral Entry

students). The concerned BOS shall finalize the modalities to earn the above credits.

e. For securing the above additional 16 credits, the students must register and complete three courses of 4 credits each offered by the department concerned. These 3 courses must contain a laboratory component also (i.e. Embedded course having three lecture hours and two practical hours). The balance of 4 credits may be secured through two MOOCs courses of 2 credits each or an embedded course offered by the department.

f. Courses that are used to fulfil the student's primary Major may not be counted towards the Honors.

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

h. If a student drops or is terminated from the Honors program, the additional credits earned so far cannot be converted into open or core electives; they will remain extra.

i. In case a student fails to meet the CGPA requirement for B.Tech degree as per clause 16.c or drops (or terminated) from the Honors program, he/she will be dropped from the list of students eligible for degree with Honors and they will receive B.Tech degree only. However, such students will receive a separate grade sheet mentioning the additional courses completed by them

j. The Honors will be mentioned in the Major degree certificate only as Bachelor of Technology (Honors). No additional degree certificate will be given for Honors.

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

l. Honors is to be completed simultaneously with a Major degree. A student cannot earn the Honors after he / she has already earned bachelor's degree

m. The documents for registration of Honors are available from the departments and college website.

4.3 Course Evaluation Process:

The performance of the students in each semester shall be assessed course wise. All assessments will be done on an absolute mark basis. However, for the purpose of reporting the performance of a candidate, letter grades and grade points will be awarded.

The performance of a student in each course is assessed with term examinations during the semester called Continuous Internal Evaluation (CIE) and a Semester End Examination (SEE) conducted at the end of the semester. For each course, there shall be a comprehensive SEE of three hours duration at the end of each semester.

4.3.1 Weightage for Course Evaluation:

The distribution of marks between CIE and SEE will be as follows:

Nature of the Course	CIE	SEE
Embedded Courses (Theory + Practical)	50	50

4.3.2 CIE in Embedded Course:

In each embedded course 25 marks is alloted for theory part and 25 marks for practical part. Theory part is evaluated in terms of two term examinations and practical part is evaluated in terms of day-to-day work and one term examination.

a. **Theory Part – 25 marks:** For theory part the term examination is conducted in the regular mode according to a schedule given for Honors / Minor courses which will be common for a particular semester of study. The weightage for Term Examinations and the calculation of marks for CIE in theory part is given in the following Table.

Term Exams in Theory Part (Max. 25 marks)
15 marks from the best performed term exam + 10 marks from the other term exam

b. **Practical part – 25 marks:** The practical part comprises of 10 marks for day-to-day laboratory work, 5 marks for record submission and 10 marks for a practical examination at the end of the practical course work. In any semester, a minimum of 90% of prescribed number of experiments / exercises specified in the syllabi for laboratory course shall be completed by the students. They shall complete these experiments / exercises in all respects and get the record certified by the internal lab teacher concerned and the Head of the Department.

4.3.3 SEE in Embedded Course:

For each course, SEE shall be conducted for theory part only. A comprehensive SEE will be conducted of three hours duration at the end of each semester course for 50 marks.

4.3.4 MOOCs Course:

If a student opts for two MOOC's, the courses must be of a minimum of 12 weeks in duration and shall contain proctored examinations. As per MOOCs guidelines the Internal Department Committee comprising Head of Department and two senior faculty members shall evaluate the certificate / grades / marks awarded for a course by external agencies and convert the same to equivalent marks / grades.

4.4 Pass Criteria in Embedded Course:

a. In CIE
A minimum of 25 (50%) marks are to be secured exclusively in the CIE with a minimum of 65% attendance in that course to be declared as qualified (Q) in that course and be eligible to appear for the SEE of that course. If a student fails to obtain 25 marks in CIE or a minimum of 65% attendance in that course, then the student will be regarded as not qualified (NQ) and such a student will be discontinued from the Honors / Minor degree program.

b. In SEE
A minimum of 20 (40%) marks of theory part is to be secured exclusively in the SEE of each course. A student eligible to appear for the SEE in a course but is absent or failed in the examination will be discontinued from the Honors / Minor degree program.

List of Courses for Honors in…….

1. The student can opt for any three courses from the following table.
2. The fourth course is optional; the student may either take two MOOC courses of 2 credits each or choose one course from the following table.
3. Concerned BOS can add or delete the subjects as per their decision.
4. Prerequisites if any, to be defined by the BOS for each course.
5. The list of MOOC courses will be specified every year at the beginning of the odd and even semesters by the Department MOOC Committee.

List of Honors Courses:

S.No	Code	Course Title	L	T	P	C
1	A	Advanced Data Structures & Algorithms	3	0	2	4
2	B	Advanced Database Management Systems	3	0	2	4
3	C	Realtime Operating Systems	3	0	2	4
4	D	Quantum Computing	3	0	2	4

Note: The course will be specified in the “Scheme of Instruction & Examination” pertaining to that semester. The course code format is 24XXHYZ

XX – Department code

H – Honors

Y – Semester

Z – Code in the above table

5 Courses of study:

At present the following B.Tech programs of study are offered.

S. No.	Title of the UG Program	Abbreviation
1.	Civil Engineering	CE
2.	Computer Science & Engineering	CS
3.	Computer Science & Engineering (Cyber Security)	CB
4.	Computer Science & Engineering (Data Science)	DS
5.	Computer Science & Engineering (Artificial Intelligence & Machine Learning)	CM
6.	Electronics & Communication Engineering	EC
7.	Electrical & Electronics Engineering	EE
8.	Information Technology	IT
9.	Mechanical Engineering	ME

6 Credits:

6.1 **Credit:** A unit by which the course work is measured. It determines the number of hours (60 minutes) 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.

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

6.3 **Choice Based Credit System (CBCS):** The CBCS provides a choice for students to select courses (Professional, Job Oriented & Open Electives) from the prescribed set of courses.

6.4 Each course in a semester is assigned certain number of credits based on the following

1 Hr. Lecture (L) per week	1 Credit
1 Hr. Tutorial (T) per week	1 Credit

1 Hr. Practical (P) per week	0.5 Credit
Internship of 4 – 6 weeks	2 Credits
Project Work of 16 weeks	12 Credits

7 Course Structure:

Every course of the B.Tech program will be placed in one of the 7 categories with suggested credits as listed below.

S. No.	Category	AICTE Recommended Credits (%)	Breakup of Credits (Total 160)
1.	Humanities and Social Sciences including Management (HM)	5 – 8 %	8 – 13
2.	Basic Science Courses (BS)	12 – 16 %	19 – 26
3.	Engineering Science Courses (ES)	10 – 18 %	16 – 29
4.	Professional Core Courses (PC)	30 – 36 %	48 – 58
5.	Electives – Professional Electives (PE); Job Oriented Electives (JOE); Open Electives (OE); Skill Enhancement Courses (SEC)	19 – 25 %	31 - 40
6.	Internships & Project Work (PR)	8 – 11 %	16
7.	Mandatory Courses (MC)	-	Non-credit

8 Course Evaluation Process:

The performance of the students in each semester shall be assessed course wise. All assessments will be done on an absolute mark basis. However, for the purpose of reporting the performance of a candidate, letter grades and grade points will be awarded.

The performance of a student in each course is assessed with alternate assessment methods, term examinations on a continuous basis during the semester called Continuous Internal Evaluation (CIE) and a Semester End Examination (SEE) conducted at the end of the semester. For each course, there shall be a comprehensive SEE of three hours duration at the end of each semester, except Mandatory courses.

The performance of a student in Internships, NSS/NCC/Scouts & Guides/Community Service and Health & Wellness/Yoga/Sports will be evaluated after completion of the course at the end of that semester.

8.1 Weightage for Course Evaluation:

The distribution of marks between CIE and SEE to be conducted at the end of the semester will be as follows:

Nature of the Course	CIE	SEE
Theory Courses	40	60
Practical Courses	40	60
Mandatory Courses	40	-
NSS/NCC/Scouts & Guides/Community Service and Health & Wellness/Yoga/Sports	-	100
Summer Internship	-	100
Project Work	40	60

8.2 CIE in Theory/Mandatory Courses:

In each Semester there shall be two Term Examinations and ***Alternate Assessment Tools (AAT)*** like Home Assignment, Class Test, Problem Solving, Group Discussion, Quiz, Seminar and Field Study in every theory course. The AAT with detailed modality of evaluation for each course shall be finalized by the teacher concerned before beginning of the course. It will be reviewed and approved by the Department Committee.

The Term Examination is conducted in the regular mode according to a schedule which will be common for a particular semester of study. The maximum weightage for Term Examinations, AATs and the calculation of marks for CIE in a theory course is given in the following Table.

Term Exams (Max. 20 marks*)	AAT (Max. 15 marks**)	Attendance (Max. 5 marks)
75% of marks obtained in the best performed term exam + 25% of marks obtained in the other term exam	Continuous assessment by teacher as per the predetermined course delivery & assessment plan. (Minimum two & maximum four assessments). AAT marks shall be considered based on average of all tests conducted.	Attendance secured & marks awarded will be as under: ≥75% and <80% - 2 marks ≥80% and <85% - 3 marks ≥85% and <90% - 4 marks ≥90% - 5 marks

*Term Examination will be conducted for 30 marks and reduced to 20 marks.

**Each AAT will be conducted for 10 marks and the average performance shall be scaled up to 15 marks.

8.3 CIE in Laboratory Courses:

The CIE for 40 marks of a laboratory course comprises of 15 marks for day-to-day laboratory work, 5 marks for record submission, 5 marks for attendance and 15 marks for a laboratory examination at the end of the laboratory course work. In any semester, a minimum of 90% of prescribed number of experiments / exercises specified in the syllabi for laboratory course shall be completed by the students. They shall complete these experiments / exercises in all respects and get the record certified by the internal lab teacher concerned and the Head of the Department.

8.4 CIE in Project Work:

The CIE is for 40 marks which consist of 20 marks for reviews at the end of each month as per the process document in the form of seminars / presentations, 5 marks for attendance and 15 marks for the evaluation of project report submitted at the end of the semester.

8.5 Pass criteria for CIE:

A minimum of 20 (50%) marks are to be secured exclusively in the CIE with a minimum of 65% attendance in that course to be declared as qualified (Q) in that course and be eligible to appear for the SEE of that course. If a student fails to obtain 20 marks in CIE or a minimum of 65% attendance in that course, then the student will be regarded as not qualified (NQ) and such a student can register for the course repetition as per the guidelines mentioned in clause 13 to qualify in that course. After securing 20 marks in course repetition, the student can appear for the SEE of that course as a supplementary candidate.

8.6 SEE in Theory Course, Laboratory Course and Project Work:

a. For each theory course, there shall be a comprehensive SEE of three hours duration at the end of each Semester for 60 marks.
b. For each laboratory course, the SEE shall be conducted by one internal and one external examiner appointed by the Principal and the duration of the exam shall be for three hours. The SEE is for 60 marks which include 15 marks for write up, 35 marks for lab experiment / exercise and 10 marks for Viva-voce.
c. Project Work shall be evaluated in the form of a Viva-Voce and demonstration of the thesis work for 60 marks. Viva-voce Examination in project work shall be conducted by one internal examiner appointed by the HOD and one external examiner to be appointed by the Principal.

8.7 Evaluation of Internships:

Summer Internship at the end of IV & VI semesters carried out in industry / organization are to be evaluated in V & VII semesters respectively after the submission of certificate provided by the organization and a concise report submitted by the student to the department committee. The internship will be evaluated by the department committee for a total of 100 marks with 50 marks for the report and 50 marks based on seminars / presentation given to the department committee by the student.

8.8 Evaluation of NSS/NCC/Scouts & Guides/Community Service and Health & Wellness/Yoga/Sports:

The above courses will be evaluated by the department committee for a total of 100 marks with 50 marks for the activities pursued by the student during that semester and 50 marks based on seminars / presentation given to the department committee by the student.

8.9 Pass Criteria for SEE:

a. Theory/Laboratory Courses and Project Work

A minimum of 21 (35%) marks are to be secured exclusively in the SEE of the above courses for the award of the grade and securing the credits for that course.

A student eligible to appear for the SEE in a course but is absent or has failed the examination may appear for SEE of that course in the next supplementary examination when offered.

b. Internship, NSS/ NCC/ Scouts & Guides/ Community Service and Health & Wellness/ Yoga/ Sports

A minimum of 40 (40%) marks are to be secured exclusively in the evaluation of the above courses for the award of the grade and securing the credits for that course.

A student eligible to appear for the evaluation in the above courses but is absent or has failed in the examination may appear for evaluation of that course in the next supplementary examination when offered.

9 Choice Based Courses:

Students can select a course from a prescribed set of courses offered by the department in the following categories.

a. **Professional Elective Courses:** There shall be five Professional Elective Courses from V Semester to VII. For each elective course there shall be a choice such that the student can choose a course from the list of courses offered by the department for that elective.

b. **Job Oriented Elective Courses:** There shall be three Job Oriented Elective Courses in all programs from V to VII semester. For each elective course there shall be a choice such that the student can choose a course from the list of courses offered by the department for that elective.

c. **Open Elective Courses:** One Open Elective Course in VII semester will be offered by various departments. A student can choose and register for an open elective course which is offered by other departments only and he / she has not studied the same course in any form during the Program.

d. **Massive Open Online Courses (MOOCs):** A Student must pursue and complete one course compulsorily through MOOCs from approved organizations for awarding the degree. A student can pursue MOOCs courses from Professional Elective / Job Oriented Elective / Open Elective Courses only. The student must

inform and take prior permission / approval from the Internal Department Committee. The courses must be of a minimum of 12 weeks in duration and shall contain proctored examinations. The student must acquire a certificate for the concerned course from the agency to earn the credits for that course. For further details and guidelines, the students can visit the college website.

The Internal Department Committee comprising Head of Department and two senior faculty members shall evaluate the certificate / grades / marks awarded for a course by external agencies and convert the same to equivalent marks / grades.

10 Induction Program:

There shall be a mandatory induction program for three weeks before the commencement of the first semester with no credits.

11 There shall be credit programs like NSS/NCC/Scouts & Guides/Community Service and Health & Wellness/Yoga/Sports. Also Design Thinking & Innovation and Tinkering lab are made compulsory credit courses for all branches.

12 Make-up Test:

a. A student can appear for a Make-up Test for a maximum of two theory courses of a semester to improve marks in the Continuous Internal Evaluation (CIE).

b. A student is eligible for the Make-up test which is conducted after the second Mid Term examination and before SEE examination if the student satisfies the following conditions.

i) Unable to secure 50% internal marks (CIE) and has more than or equal to 65% attendance in a particular theory course (After finalizing the internal marks).

ii) Attendance in Remedial classes is more than or equal to 65% (if Remedial classes are conducted) or secured greater than 50% marks in the I Mid Term Examination and AAT-1 together.

iii) Attended 50% of CIE tests (at least one AAT & one Mid Term Examinations).

The make-up test will be conducted for 40 marks (8 questions of 1 mark each, 2 questions of 16 marks each) in Mid Examination format covering the entire syllabus and the marks obtained in this test are final. However, the maximum marks awarded will be 20 only.

The students must apply to the principal through the respective HOD by paying prescribed fees.

The documents for registration of the Make-up test are available from the departments and college website.

13 Course Repetition:

The students not qualified to write SEE in a course may register for the repeater courses through Course Repetition. The students must apply to the principal through the respective HOD by paying prescribed fees.

A student can take up a maximum of two theory courses and one laboratory course in a semester immediately after the semester end examinations of that semester. The students who are not taking regular semester courses may additionally register for one more theory course.

The documents for registration of course and monitoring the candidates registered for course repetition are available from the departments and college website.

14 Minimum Academic Requirements for Promotion:

a. Semester Promotion

A student is eligible to register for SEE if he/she satisfies the following conditions. However, the student can appear only for the SEE of those courses in which the student is qualified (Q).

i) Attendance Requirements

A student shall be eligible to register for SEE, if he / she acquires a minimum of 75% of attendance in aggregate of all the courses in a semester.

Condonation of shortage of attendance in aggregate up to 10% (65% and above and below 75%) in each semester may be granted on genuine medical grounds with a doctor certificate and duly approved by the principal.

A shortage of attendance below 65% in aggregate shall in NO case be condoned. Students whose shortage of attendance is not condoned in any semester are not eligible to take their SEE of that semester and will be considered as detained in that semester.

If a student does not satisfy the attendance requirements of the present semester, he / she will not be promoted to the next semester (considered as detained in the present semester). They may seek readmission for that semester when offered next.

A stipulated fee shall be payable towards condonation of shortage of attendance to the college.

ii) Qualification in CIE

A student must qualify in a minimum of three courses in each semester (as per Clause 8.2.4) in CIE to register for the SEE of that semester.

If a student does not satisfy the above conditions, he / she will not be promoted to the next semester (considered as detained in that semester). They may seek readmission for the detained semester when offered next.

b. Promotion / Detention Conditions based on the minimum credits to be secured by the student:

A student shall be promoted from I to II, III to IV, V to VI and VII to VIII semesters if he / she fulfills the academic requirement as specified in 14.a). For other semesters i.e. II to III (1st year to 2nd year), IV to V (2nd year to 3rd year) and VI to VII (3rd year to 4th year) semesters, the following criteria is to fulfilled in addition to 14.a) clause.

i) II semester to III semester (1st year to 2nd year)

A student shall be promoted from II semester to III semester only if he / she fulfills the academic requirement of securing 25% of the credits in the courses that have been studied up to I Semester.

ii) IV semester to V semester (2nd year to 3rd year)

A student shall be promoted from IV semester to V semester only if he/she fulfills the academic requirement of securing 40% of the credits in the courses that have been studied up to III Semester.

iii) VI semester to VII semester (3rd year to 4th year)

A student shall be promoted from VI semester to VII semester only if he/she fulfills the academic requirements of securing 40% of the credits in the courses that have been studied up to V semester.

If a student is not promoted or detained for want of credits in a particular semester as per clause 14.b) above, the student may secure the required credits through supplementary examinations and only after securing the required credits he / she shall be permitted to join in the III or V or VII Semester as the case may be.

c. With–holding of Results

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

15 Summer Internships:

Students shall undergo two summer internships each for a minimum of four weeks duration at the end of second and third years of the program for 2 credits each. The organization in which the student wishes to carry out Internship needs to be approved by Internal Department Committee comprising Head of Department and two senior faculty members. The student shall submit a report along with an internship certificate from the organization. The evaluation of the first and second summer internships shall be conducted at the end of the V Semester & VII semester respectively.

Completion of the internship is mandatory, if any student fails to complete internship, he/she will not be eligible for the award of degree. In such a case, the student shall repeat the internship in the subsequent summer. The student pursuing two summer internships in the same summer is not permitted.

Community Service Project focussing on specific local issues shall be an alternative to the four weeks of summer Internship. The Community Service Project shall be for 4 weeks in duration which includes preliminary survey for 1 week, community awareness

programs for one week, community immersion program in consonance with Government agencies for 1 week and a community exit report (a detailed report) for 1 week.

16 A student shall register and put-up minimum attendance in all 160 credits and earn all the 160 credits. In the case of lateral entry students, the number of credits is 121.

17 Students who fail to earn 160 credits as indicated in the course structure within eight academic years from the year of their admission shall forfeit their seat in B.Tech. Program, and their admission shall be cancelled. However, for the students availing the Gap year facility, this period shall be extended by corresponding gap year duration availed.

Lateral entry students who fail to earn 121 credits as indicated in the course structure within six academic years from the year of their admission shall forfeit their seat in B.Tech. Course and their admission shall be cancelled. However, for the students availing gap year facility, this period shall be extended by corresponding gap year duration availed.

18 Securing Credits and award of Grade Points:

Grading

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

Structure of Grading of Academic Performance

Range in which the % of marks in the course fall	Grade	Grade Points Assigned
≥ 90	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 (Below Average)	5
< 40	F (Fail)	0
Absent	Ab (Absent)	0

A student obtaining Grade F shall be considered failed and will be required to reappear for that course when the next supplementary SEE is offered. Same is the case with a student who obtains 'Ab' in the SEE.

Since there are no credits for Mandatory /Audit courses, only 'Pass' or 'Fail' shall be mentioned for such courses.

19 Semester Grade Point Average (SGPA) and Cumulative Grade Point Average (CGPA):

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

$$\text{SGPA} = \frac{\sum_{i=1}^{n} C_i \times GP_i}{\sum_{i=1}^{n} C_i}$$

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

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

$$\text{CGPA} = \frac{\sum_{j=1}^{m} SGPA_j \times TC_j}{\sum_{j=1}^{m} TC_j}$$

where "$SGPA_j$" is the SGPA of the j^{th} semester and TC_j is the total number of credits in that semester.

c. Both SGPA and CGPA shall be truncated to 2 decimal points and reported in the transcripts.

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

e. Grade Point: It is a numerical weightage allotted to each letter grade on a 10-point scale.

f. Letter Grade: It is an index of the performance of students in a said course. Grades are denoted by letters S, A, B, C, D, E and F.

20 Award of Class:

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

Class Awarded	CGPA Secured
First Class with Distinction	≥ 7.5
First Class	≥ 6.5 < 7.5
Second Class	≥ 5.5 < 6.5

Pass Class	≥ 5.0 < 5.5

21 Gap Year:

Gap year concept for Student Entrepreneur shall be 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 entrepreneurship program / to establish startups. This period may be extended to two years at the most and these two years would not be counted as the maximum time for graduation.

An evaluation committee shall be constituted by the College to evaluate the proposal submitted by the student and the committee shall decide whether to permit the student(s) to avail themselves of the Gap Year.

After rejoining the student can pursue the remaining period of study under transitory regulations (if the regulation changes).

22 Transitory Regulations:

Discontinued or detained candidates (as per clause 14.b) are eligible for readmission as and when the semester is offered and after fulfillment of academic regulations. Candidates who have been detained as per clause 14.a) are eligible for readmission into the unfinished semester from the date of commencement of class work with the same or equivalent subjects as and when subjects are offered.

Candidates who were permitted with Gap Year shall be eligible for rejoining into the succeeding year of their B.Tech from the date of commencement of class work.

The readmitted students must follow the regulations in which he/she is admitted and residual courses if any must be completed based on the equivalent courses for each semester specified by the BOS considering the previous and readmitted regulations.

23 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 15 credits (5 courses, approximately 10% for the total credits of the program) through MOOCs platform.

a. The Institution shall offer credit mobility for MOOCs and give the equivalent credit weightage to the students for the credits earned through online learning courses.
b. Student registration for the MOOCs shall be only through the respective department of the institution, it is mandatory for the student to share necessary information and take prior approval from the department.
c. Credit transfer policy will be applicable to the Professional Elective Courses, Job Oriented Elective Courses, Open Elective Courses & Management Courses only.
d. The concerned department shall identify the courses permitted for credit transfer.
e. The department shall notify the list of the online learning courses at the beginning of semester eligible for credit transfer.

f. The department shall designate a faculty member as a Mentor for each course to guide the students from registration till completion of the course.
g. The department shall ensure no overlap of MOOC exams with that of the university examination schedule. In case of delay in results, the Institution will re-issue the marks sheet for such students.
h. Credits transfer will be considered only after successful completion of the course and submitting a certificate issued by the competent authority along with the percentage of marks and/or grades.
i. The institution shall submit the following to the examination section:
    i. List of students who have passed MOOC courses in the current semester along with the certificate of completion.
    ii. Undertaking form filled by the students for credit transfer.
j. The Institution shall resolve any issues that may arise in the implementation of this policy from time to time and shall review its credit transfer policy in the light of periodic changes brought by UGC, SWAYAM, NPTEL and state government.

24 Academic Bank of Credits (ABC):

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

a. Provide option of mobility for learners across the universities of their choice.
b. Provide option to gain the credits through MOOCs from approved digital platforms.
c. Facilitate award of Certificate / Diploma / Degree (B.Sc) in line with the accumulated credits in ABC
d. Execute Multiple Entry and Exit system with credit count and credit transfer.

25 Exit Policy:

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

a. **UG Certificate** (in Field of study / discipline) - Programme duration: First year (first two semesters) of the undergraduate programme, 39 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.

b. **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.

c. **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.

26 Student Transfers

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

27 Punishments for Malpractice cases – Guidelines:

a) If any student caught under malpractice during the CIE examinations, the entire cycle of examinations will be cancelled and awarded zero marks for all the courses during that cycle. For example, if any student is caught while doing malpractice in an AAT, the AAT marks of all the courses in that cycle will be cancelled. Similar punishment will be considered for mid-term examinations also.

b) For Semester End Examinations, the examinations committee may take the following guidelines into consideration while dealing with the suspected cases of malpractice reported by the invigilators / squad members etc. The punishment may be more severe or less severe depending on the merits of the individual cases.

S. No.	Nature of Malpractice/Improper conduct	Punishment
1.	Possesses or keeps accessible in examination hall, any paper, note book, programmable calculators, Cellphones, pager, palm computers or any other form of material concerned with or related to the course of the examination (theory or practical) in which he is appearing but has not made use of (material shall include any marks on the body of the student which can be used as an aid in the course of the examination).	Expulsion from the examination hall and cancellation of the performance in that course only.
2.	Uses objectionable, abusive or offensive language in the answer paper or in letters to the examiners or writes to the examiner requesting him to award pass marks.	Cancellation of the performance in that course.
3.	Copying detected on the basis of internal evidence, such as, during valuation or during special scrutiny.	Cancellation of the performance in that course and all other courses the candidate has appeared including practical examinations and project work of that semester/year examinations.
4.	Gives assistance or guidance or receives it from any other student orally or by any other body language methods or communicates through cell phones with any other student or persons in or outside the exam hall in respect of any matter.	Expulsion from the examination hall and cancellation of the performance in that course only of all the students involved. In case of an outsider, he will be handed over to the police and a case shall be registered against him.

5.	Has copied in the examination hall from any paper, book, programmable calculators, palm computers or any other form of material relevant to the course of the examination (theory or practical) in which the student is appearing.	Expulsion from the examination hall and cancellation of the performance in that course and all other courses including practical examinations and project work of that semester/year.
6.	Comes in a drunken condition to the examination hall.	Expulsion from the examination hall and cancellation of the performance in that course and all other courses including practical examinations and project work of that semester/year.
7.	Smuggles in the Answer book or takes out or arranges to send out the question paper during the examination or answer book during or after the examination	Expulsion from the examination hall and cancellation of performance in that course and all the other courses including practical examinations and project work of that semester/year. The student is also debarred for two consecutive semesters from class work and all examinations. The continuation of the course by the student is subject to the academic regulations in connection with forfeit of seat.
8.	Refuses to obey the orders of the Chief Superintendent / any officer on duty or misbehaves or creates disturbance of any kind in and around the examination hall or organizes a walk out or instigates others to walk out, or threatens the officer-in charge or any person on duty in or outside the examination hall of any injury to his person or to any of his relations whether by words, either spoken or written or by signs or by visible representation, assaults the officer-in-charge, or any person on duty in or outside the examination hall or any of his relations, or indulges in any other act of misconduct or mischief which result in damage to or destruction of property in the examination hall or any part of the College campus or engages in any other act which in the opinion of the officer on duty amounts to use of unfair means or	In case of students of the college, they shall be expelled from examination halls and cancellation of their performance in that course and all other courses of that semester/year. The students also are debarred and forfeit their seats. In case of outsiders, they will be handed over to the police and a police case shall be registered against them.

	misconduct or has the tendency to disrupt the orderly conduct of the examination.	
9.	Leaves the exam hall taking away answer script or intentionally tears up the script or any part there of inside or outside the examination hall.	Expulsion from the examination hall and cancellation of performance in that course and all the other courses including practical examinations and project work of that semester/year. The candidate is also debarred for two consecutive semesters from classwork and all end examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat.
10.	Possesses any lethal weapon or firearm in the examination hall.	Expulsion from the examination hall and cancellation of the performance in that course and all other courses including practical examinations and project work of that semester/year. The student is also debarred and forfeits the seat.
11.	If student of the college, who is not a candidate for the particular examination or any person not connected with the college indulges in any malpractice or improper conduct mentioned in S.No. 7 to S.No. 9.	For Student of the college: Expulsion from the examination hall and cancellation of the performance in that course and all other courses including practical examinations and project work of that semester/year. The candidate is also debarred and forfeits the seat. Person(s) who do not belong to the College will be handed over to police and, a police case shall be registered against them.
12.	Impersonates any other student in connection with the examination	The student who has impersonated shall be expelled from examination hall. The student is debarred from writing the remaining exams, and rusticated from the college for one academic year during which period the student will not be permitted to write any exam. If the imposter is an outsider, he will be

		handed over to the police and a case shall be registered against him. The performance of the original student who has been impersonated, shall be cancelled in all the courses of the examination including practical's and project work of that semester/ year. The student is rusticated from the college for two consecutive years during which period the student will not be permitted to write any exam. The continuation of the course by the student is subject to the academic regulations in connection with forfeiture of seat.
13.	If any malpractice is detected which is not covered in the above S.No. 1 to S.No. 12 items, it shall be reported to the college academic council for further action and award suitable punishment.	
14.	Malpractice cases identified during sessional examinations will be reported to the examination committee nominated by Academic council to award suitable punishment.	

28 ADDITIONAL ACADEMIC REGULATIONS:

a) Any attempt to impress upon the teachers, examiners, faculty and staff of Examinations, bribing for either marks or attendance will be treated as malpractice.
b) When a component of Continuous Internal Evaluation (CIE) or Semester End Examination (SEE) is cancelled as a penalty, he/she is awarded zero marks in that component.

29 AMENDMENTS TO REGULATIONS:

The Academic Council of Bapatla Engineering College (Autonomous) reserves the right to revise, amend, change or nullify the Regulations, Schemes of Examinations and / or Syllabi, Academic schedules, Examination schedules, Examination pattern, Moderation to students, Special opportunity to complete degree beyond stipulated time and any other matter pertained that meets to the needs of the students, society and industry without any notice and the decision is final.

DEPARTMENT VISION

To produce Computer Science Engineers with Global Standards who can handle the challenges in developing software services and products needed by the society and industry with their innovations and services.

DEPARTMENT MISSION

- To impart high quality education with effective teaching and learning process.
- To provide an environment where the students can handle research problems confidently.
- To prepare the students with latest technologies with fidelity towards industry.
- To inculcate professional ethics and human values in handling the engineering challenges.

PROGRAM EDUCATIONAL OBJECTIVES

PEO1: Choose diverse professional careers in software industry, research, academia, engineering, and administrative services.

PEO2: Apply the principles of basic sciences, mathematics and computer science to solve real world problems using digital computing systems.

PEO3: Analyze, design, implement and evaluate robust, scalable and cost-effective computer-based systems and processes in the industry with sustained self learning.

PEO4: Be aware of professional and ethical practices in the context of social impacts of computing.

Transitory Regulations – R23/R20 to R24 - Equivalence Courses

R24 1-1 SEM		R23		SEM
24CS101	Linear Algebra and Ordinary Differential Equations	20CS101/MA01	Linear Algebra and Ordinary Differential Equations	1.1
24CS102	Semiconductor Physics and Nano materials	20CS202/PH03	Semiconductor Physics	1.2
24CS103	Communicative English	20CS103/EL01	Communicative English	1.1
24CS104	Introduction to Programming	20CS204/CS01	Programming for Problem Solving	1.2
24CSL101	Engineering Graphics Lab	**NIL**		
24CSL102	Semiconductor Physics Lab	20CSL201/PHL02	Semiconductor Physics Lab	1.2
24CSL103	English Communication skills Lab	20CSL103/ELL01	English Communication skills Lab	1.1
24CSL104	Introduction to Programming Lab	20CSL203/CSL01	Programming for Problem Solving Lab	1.2
24CSL105	IT Workshop	20CSL101/CSL03	Computer Fundamentals Lab	1.1

R24 1-1 SEM		R20		SEM
24CS101	Linear Algebra and Ordinary Differential Equations	20CS101/MA01	Linear Algebra and Ordinary Differential Equations	1.1
24CS102	Semiconductor Physics and Nano materials	20CS202/PH03	Semiconductor Physics	1.2
24CS103	Communicative English	20CS103/EL01	Communicative English	1.1
24CS104	Introduction to Programming	20CS204/CS01	Programming for Problem Solving	1.2
24CSL101	Engineering Graphics Lab	20CSL101/MEL01	Engineering Graphics	1.1
24CSL102	Semiconductor Physics Lab	20CSL201/PHL02	Semiconductor Physics Lab	1.2
24CSL103	English Communication skills Lab	20CSL103/ELL01	English Communication skills Lab	1.1

24CSL104	Introduction to Programming Lab	20CSL203/CSL01	Programming for Problem Solving Lab	1.2
24CSL105	IT Workshop	**NIL**		

R24 1-2 SEM		**R20**		**SEM**
24CS201	Numerical methods & Advanced Calculus	20CS201/MA02	Numerical methods& Advanced Calculus	1.2
24CS202	Engineering Chemistry	20CS102/CY01	Engineering Chemistry	1.1
24CS203	Basic Electrical & Electronics Engineering	20CS203/EE01	Basic Electrical & Electronics Engineering	1.2
24CS204	Programming for Problem Solving	**NIL**		1.2
24CS205	Discrete Mathematics	20CS206/CC02	Discrete Mathematics	1.2
24CSL201	Engineering Mechanics Lab	**NIL**		
24CSL202	Engineering Chemistry Lab	20CSL102/CYL01	Chemistry Lab	1.1
24CSL203	Basic Electrical & Electronics Engineering Lab	20CSL202/EEL01	Basic Electrical & Electronics Engineering Lab	1.2
24CSL204	Programming for Problem Solving Lab	**NIL**		

R24 2-1 SEM		**R20**		**SEM**
24CS301	Probability & Statistics	20CS301/MA03	Probability & Statistics	2.1
24CS302	Digital Logic Design	20CS205/CC01	Digital Logic Design	1.2
24CS303	Data Structures	20CS302/CC03	Data Structures	2.1
24CS304	Object Oriented Programming	20CS303/CC04 20CS304/CC05	Object Oriented Programming Operating System	2.1
24CS305	Computer Networks	20CS502/CC15	Computer Networks	3.1
24CSL301/ SEC1	Python Programming	20CSL401/SOC2	Python Programming	2.2
24CSL302	Data Structures Lab	20CSL302/CC07	Data Structures Lab	2.1
24CSL303	Object Oriented Programming Lab	20CSL303/CC08	Object Oriented Programming Lab	2.1
24CSL304	Health and Wellness, Yoga and Sports	**NIL**		

24CS306/ MC02	Constitution of India	20CS606/MC03	Indian Constitution	3.2

R24 2-2 SEM		R20		SEM
24CS401	Computer Organization	20CS305/CC06	Computer Organization	2.1
24CS402	Automate Theory & Formal Languages	20CS501/CC14	Automate Theory & Formal Languages	3.1
24CS403	Web Technologies	20CS402/CC09	Web Technologies	2.2
24CS404	Database Management Systems	20CS403/CC10	Database Management System	2.2
24CS405	Design and Analysis of Algorithms	20CS404/CC11	Design and Analysis of Algorithms	2.2
24CSL401/ SEC2	Competitive Prog. Skills	**NIL**		
24CSL402	Design Thinking & Innovation	**NIL**		
24CSL403	Web Technologies Lab	20CSL402/CC12	Web Technologies Lab	2.2
24CSL404	RDBMS Lab	20CSL403/CC13	RDBMS Lab	2.2
24CSL405	NSS / NCC / Scouts & Guides / Community Service	**NIL**		
24CS406/ MC01	Environmental Science	20CS104/MC01	Environmental Studies	1.1

List of Residual Courses **to be completed by students** of R23/R20 Regulation who migrate into R24 Regulation

R20 Regulation (Completed)	R24 Regulation (Joining in)	Code	Subject Name
1-1 SEM (R23)	1-2 SEM	24CSL101	Engineering Graphics Lab
1-1 SEM	1-2 SEM	24CS104	Introduction to Programming
		24CSL104	Introduction to Programming Lab
		24CSL105	IT Workshop
1-2 SEM	2-1 SEM	24CS204	Programming for Problem Solving
		24CSL201	Engineering Mechanics Lab
		24CSL204	Programming for Problem Solving Lab
2-1 SEM	2-2 SEM	24CS305	Computer Networks
		24CSL301/ SEC1	Python Programming
		24CSL304	Health and Wellness, Yoga and Sports
		24CS306/ MC02	Constitution of India
2-2 SEM	3-1 SEM	24CS402	Automate Theory & Formal Languages
		24CSL401/ SEC2	Competitive Prog. Skills
		24CSL402	Design Thinking & Innovation
		24CSL405	NSS / NCC / Scouts & Guides / Community Service

Course Structure Summary

S. No.	Category	BEC Breakup of Credits
1	Humanities & Social Science including Management Courses	9
2	Basic Science courses	20
3	Engineering Science courses including workshop, drawing, basics of electrical/mechanical/computer etc.	24
4	Professional core courses	51.5
5	Professional Elective courses relevant to chosen specialization/branch	15
6	Open subjects – Electives from other technical and /or emerging subjects	16.5
7	Project work, seminar, and internship in industry or elsewhere	16
8	Mandatory Courses [Professional Ethics & Human Values, Indian Constitution, Essence of Indian Knowledge Tradition]	(non-credit)
9	Skill Oriented Subjects	8
Total		**160**

Semester Wise Credits Summary

Semester	Credits	With Honor Credits
Semester-I	18	18
Semester-II	21	21
Semester-III	20.5	20.5
Semester-IV	22.5	26.5
Semester-V	22	26
Semester-VI	21.5	25.5
Semester-VII	22.5	26.5
Semester-VIII	12	12
Total	**160**	**176**

SCHEME OF INSTRUCTION & EXAMINATION (Semester System)
For
Computer Science & Engineering
First Year B.Tech. (SEMESTER – I) W.E.F. A.Y. 2024-25 (R24)

Course Code	Category	Course Title	Scheme of Instruction (Hours per week)				Scheme of Examination (Maximum marks)			No. of Credits
			L	T	P	Total	CIE	SEE	Total	
24CS101	BS	Linear Algebra and Ordinary Differential Equations	2	1	0	3	40	60	100	3
24CS102	BS	Semiconductor Physics and Nano materials	3	0	0	3	40	60	100	3
24CS103	HM	Communicative English	2	0	0	2	40	60	100	2
24CS104	ES	Introduction to Programming	3	0	0	3	40	60	100	3
24CSL101	ES	Engineering Graphics Lab	1	0	3	4	40	60	100	2.5
24CSL102	BS	Semiconductor Physics Lab	0	0	2	2	40	60	100	1
24CSL103	HM	English Communication skills Lab	0	0	2	2	40	60	100	1
24CSL104	ES	Introduction to Programming Lab	0	0	3	3	40	60	100	1.5
24CSL105	ES	IT Workshop	0	0	2	2	40	60	100	1
Induction Program		**First Three Weeks** (Physical activity, Creative Arts, Universal Human Values, Literary, Proficiency Modules, Lectures by Eminent People, Familiarization to Dept./Branch & Innovations)								
TOTAL			**11**	**1**	**12**	**24**	**360**	**540**	**900**	**18**

L: Lecture T: Tutorial P: Practical
CIE: Continuous Internal Evaluation SEE: Semester End Examination

SCHEME OF INSTRUCTION & EXAMINATION (Semester System)
For
Computer Science & Engineering
First Year B.Tech. (SEMESTER – II) W.E.F. A.Y. 2024-25 (R24)

Course Code	Category	Course Title	Scheme of Instruction (Hours per week)				Scheme of Examination (Maximum marks)			No. of Credits
			L	T	P	Total	CIE	SEE	Total	
24CS201	BS	Numerical methods & Advanced Calculus	2	1	0	3	40	60	100	3
24CS202	BS	Engineering Chemistry	3	0	0	3	40	60	100	3
24CS203	ES	Basic Electrical & Electronics Engineering	3	0	0	3	40	60	100	3
24CS204	ES	Programming for Problem Solving	3	0	0	3	40	60	100	3
24CS205	BS	Discrete Mathematics	3	0	0	3	40	60	100	3
24CSL201	ES	Engineering Mechanics and Surveying Lab	1	0	2	3	40	60	100	2
24CSL202	BS	Engineering Chemistry Lab	0	0	2	2	40	60	100	1
24CSL203	ES	Basic Electrical & Electronics Engineering Lab	0	0	3	3	40	60	100	1.5
24CSL204	ES	Programming for Problem Solving Lab	0	0	3	3	40	60	100	1.5
TOTAL			**15**	**1**	**10**	**26**	**360**	**540**	**900**	**21**

SCHEME OF INSTRUCTION & EXAMINATION (Semester System)
For
Computer Science & Engineering
Second Year B.Tech. (SEMESTER – III) W.E.F. A.Y. 2024-25 (R24)

Course Code	Category	Course Title	Scheme of Instruction (Hours per week)				Scheme of Examination (Maximum marks)			No. of Credits
			L	T	P	Total	CIE	SEE	Total	
24CS301	BS	Probability & Statistics	2	1	0	3	40	60	100	3
24CS302	ES	Digital Logic Design	3	0	0	3	40	60	100	3
24CS303	PC	Data Structures	2	1	0	3	40	60	100	3
24CS304	PC	Object Oriented Programming	2	1	0	3	40	60	100	3
24CS305	PC	Computer Networks	3	0	0	3	40	60	100	3
24CSL301/ SEC1	SEC	Python Programming	1	0	2	3	40	60	100	2
24CSL302	PC	Data Structures Lab	0	0	3	3	40	60	100	1.5
24CSL303	PC	Object Oriented Programming Lab	0	0	3	3	40	60	100	1.5
24CSL304	HM	Health and Wellness, Yoga and Sports	0	0	1	1	100	-	100	0.5
24CS306/ MC02	MC	Constitution of India	2	0	0	2	40	-	40	0
TOTAL			**15**	**3**	**9**	**27**	**460**	**480**	**940**	**20.5**

SCHEME OF INSTRUCTION & EXAMINATION (Semester System)
For
Computer Science & Engineering
Second Year B.Tech. (SEMESTER – IV) W.E.F. A.Y. 2024-25 (R24)

Course Code	Category	Course Title	Scheme of Instruction (Hours per week)				Scheme of Examination (Maximum marks)			No. of Credits
			L	T	P	Total	CIE	SEE	Total	
24CS401	PC	Computer Organization	3	0	0	3	40	60	100	3
24CS402	PC	Automate Theory & Formal Languages	2	1	0	3	40	60	100	3
24CS403	PC	Web Technologies	3	0	0	3	40	60	100	3
24CS404	PC	Database Management Systems	3	0	0	3	40	60	100	3
24CS405	PC	Design and Analysis of Algorithms	2	1	0	3	40	60	100	3
24CSL401/ SEC2	SEC	Competitive Programming Skills	1	0	2	3	40	60	100	2
24CSL402	ES	Design Thinking & Innovation	1	0	2	3	40	60	100	2
24CSL403	PC	Web Technologies Lab	0	0	3	3	40	60	100	1.5
24CSL404	PC	RDBMS Lab	0	0	3	3	40	60	100	1.5
24CSL405	HM	NSS / NCC / Scouts & Guides / Community Service	0	0	1	1	100	-	100	0.5
24CS406/ MC01	MC	Environmental Science	2	0	0	2	40	-	40	0
TOTAL			**17**	**2**	**11**	**30**	**500**	**540**	**1040**	**22.5**
24CSH4_/ 24CSM4_	Honor Course - 1		**3**	**0**	**2**	**5**	**50**	**50**	**100**	**4**

SCHEME OF INSTRUCTION & EXAMINATION (Semester System)
For
Computer Science & Engineering
Third Year B.Tech. (SEMESTER – V) W.E.F. A.Y. 2024-25 (R24)

Course Code	Category	Course Title	Scheme of Instruction (Hours per week)				Scheme of Examination (Maximum marks)			No. of Credits
			L	T	P	Total	CIE	SEE	Total	
24CS501	PC	Compiler Design	3	0	0	3	40	60	100	3
24CS502	PC	Operating Systems	3	0	0	3	40	60	100	3
24CS503	PC	Cryptography & Network Security	3	0	0	3	40	60	100	3
24CS504/ PE1_	PE	Professional Elective - I	3	0	0	3	40	60	100	3
24CS505 JO1_	JOE	Job Oriented Elective - I	3	0	0	3	40	60	100	3
24CSL501/ SEC3	SEC	Linux essentials	1	0	2	3	40	60	100	2
24CSL502	PC	Cryptography & Network Security Lab	0	0	3	3	40	60	100	1.5
24CSL503	JOE	Job Oriented Elective - I Lab	0	0	3	3	40	60	100	1.5
24CSL504/ INT1	PR	Summer Internship - I*	0	0	0	0	-	100	100	2
24CS506/ MC04	MC	T & P Skills	2	0	0	2	40	-	40	0
TOTAL			**18**	**0**	**8**	**26**	**360**	**580**	**940**	**22**
24CSH5_/ 24CSM5_	Honor Course - 2		**3**	**0**	**2**	**5**	**50**	**50**	**100**	**4**

Job Oriented Elective - I

1A	Enterprise Programming
	Enterprise Programming Lab
1B	Full Stack Development
	Full Stack Development Lab
1C	Middleware Technologies
	Middleware Technologies Lab

Professional Elective – I

1A	Artificial Intelligence
1B	Advanced Computer Architecture
1C	Parallel Algorithms

* Summer Internship - I (PR01) need to be completed after 4th semester and it is evaluated by the end of 5th semester.

SCHEME OF INSTRUCTION & EXAMINATION (Semester System)
For
Computer Science & Engineering
Third Year B.Tech. (SEMESTER – VI) W.E.F. A.Y. 2024-25 (R24)

Course Code	Category	Course Title	Scheme of Instruction (Hours per week)				Scheme of Examination (Maximum marks)			No. of Credits
			L	T	P	Total	CIE	SEE	Total	
24CS601	PC	Software Engineering	3	0	0	3	40	60	100	3
24CS602	PC	Machine Learning	3	0	0	3	40	60	100	3
24CS603/ PE2_	PE	Professional Elective - II	3	0	0	3	40	60	100	3
24CS604/ PE3_	PE	Professional Elective - III	3	0	0	3	40	60	100	3
24CS605	JOE	Job Oriented Elective - II	3	0	0	3	40	60	100	3
24CSL601	HM	Soft Skills	1	0	2	3	40	60	100	2
24CSL602	PC	Software Engineering Lab	0	0	3	3	40	60	100	1.5
24CSL603	PC	Machine Learning Lab	0	0	3	3	40	60	100	1.5
24CSL604	JOE	Job Oriented Elective - II Lab	0	0	3	3	40	60	100	1.5
24CS606/ MC03	MC	Technical paper writing & IPR	2	0	0	2	40	-	40	0
TOTAL			**18**	**0**	**11**	**29**	**400**	**540**	**940**	**21.5**
24CSH6_/ 24CSM6_		Honor Course - 3	**3**	**0**	**2**	**5**	**50**	**50**	**100**	**4**

Job Oriented Elective - II	
2A	Cyber Security
	Cyber Security Lab
2B	Cloud Programming
	Cloud Programming Lab
2C	Mobile Application Development
	Mobile Application Development Lab

Professional Elective – II	
2A	Wireless Networks
2B	Design patterns & Frame works
2C	Software Defined Networks

Professional Elective – III	
3A	Block Chain Technology
3B	Distributed Systems
3C	Digital Image Processing

SCHEME OF INSTRUCTION & EXAMINATION (Semester System)
For
Computer Science & Engineering
Fourth Year B.Tech. (SEMESTER – VII) W.E.F. A.Y. 2024-25 (R24)

Course Code	Category	Course Title	Scheme of Instruction (Hours per week)				Scheme of Examination (Maximum marks)			No. of Credits
			L	T	P	Total	CIE	SEE	Total	
24CS701	HM	Industrial Management & Entrepreneurship Development	3	0	0	3	40	60	100	3
24CS702/ PE4_	PE	Professional Elective – IV	3	0	0	3	40	60	100	3
24CS703 PE5_	PE	Professional Elective – V	3	0	0	3	40	60	100	3
24CS704	OE	Open Elective	3	0	0	3	40	60	100	3
24CS705/ JO_	JOE	Job Oriented Elective – III	3	0	0	3	40	60	100	3
24CSL701/ SEC4	SEC	DevOps	1	0	2	3	40	60	100	2
24CSL702	PC	Programming with Gen AI	1	0	2	3	40	60	100	2
24CSL703	JOE	Job Oriented Elective – III Lab	0	0	3	3	40	60	100	1.5
24CSL704/ INT2	PR	Summer Internship – II*	0	0	0	0	-	100	100	2
TOTAL			**17**	**0**	**7**	**24**	**320**	**580**	**900**	**22.5**
24CSH7_/ 24CSM7_		Honor Course - 4 (MOOCs – 1 & MOOCs – 2)	**0**	**0**	**0**	**0**	**50**	**50**	**100**	**4**

Job Oriented Elective – III	
3A	Generative AI Technologies
	Generative AI Technologies Lab
3B	Data Handling & Visualization
	Data Handling & Visualization Lab
3C	Deep Learning
	Deep Learning Lab

Professional Elective – IV	
4A	Software Testing Methodologies
4B	Digital Forensics
4C	Semantic Web Technologies

Professional Elective – V	
5A	Natural Language Processing
5B	Social Network Analysis
5C	Prompt Engineering

* Summer Internship – II (PR02) need to be completed after 6th semester and it is evaluated by the end of 7th semester.

SCHEME OF INSTRUCTION & EXAMINATION (Semester System)
For
Computer Science & Engineering
Fourth Year B.Tech. (SEMESTER – VIII) W.E.F. A.Y. 2024-25 (R24)

Course Code	Category	Course Title	Scheme of Instruction (Hours per week)				Scheme of Examination (Maximum marks)			No. of Credits
			L	T	P	Total	CIE	SEE	Total	
24CSL801	PR	Project Work	0	0	24	24	40	60	100	12
Total			**0**	**0**	**24**	**24**	**40**	**60**	**100**	**12**

SCHEME OF INSTRUCTION & EXAMINATION (Semester System)
For
Computer Science & Engineering
Honor Course List W.E.F. A.Y. 2024-25 (R24)

Course Code	Course Title	Scheme of Instruction (Hours per week)				Scheme of Examination (Maximum marks)			No. of Credits
		L	T	P	Total	CIE	SEE	Total	
24CSH4A	Advanced Data Structures & Algorithms	3	0	2	5	50	50	100	4
24CSH5A	Advanced Database Management Systems	3	0	2	5	50	50	100	4
24CSH6A	Realtime Operating Systems	3	0	2	5	50	50	100	4
24CSH7A	Quantum Computing	3	0	2	5	50	50	100	4

LINEAR ALGEBRA AND ORDINATRY DIFFERENTIAL EQUATIONS
I B. Tech. – I Semester (Code: 24CS101)

Lectures	2	Tutorial	1	Practical	0	Credits	3
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: None.

Course Objectives:

- Solve a system of linear homogeneous and non-homogeneous equations, finding the inverse of a given square matrix and also its Eigen values and Eigen vectors
- Identify the type of a given differential equation and select and apply the appropriate analytical technique for finding the solution of first order ordinary differential equations.
- Create and analyze mathematical models using higher order differential equations to solve application problems that arise in engineering.
- Verify mean value theorems and expand functions of a single variable using Taylor's and Maclaurin's series.

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

CO1	Find the eigen values and eigen vectors of a given matrix and its inverse.
CO2	Apply the appropriate analytical technique to find the solution of a first order ordinary differential equation.
CO3	Solve higher order linear differential equations with constant coefficients arise in engineering applications.
CO4	Learn the applications of mean value theorems and Taylor's theorem.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3	3	2	-	-	-	-	-	-	-	2	-	-	-
CO2	3	3	3	-	-	-	-	-	-	-	2	-	-	-
CO3	3	3	3	-	-	-	-	-	-	-	2	-	-	-
CO4	2	2	2	-	-	-	-	-	-	-	2	-	-	-

UNIT-I

Linear Algebra: Rank of a Matrix; Elementary transformations of a matrix; Gauss-Jordan method of finding the inverse; Normal form of a matrix, Consistency of linear System of equations: Rouche's theorem, System of linear Non-homogeneous equations, System of linear homogeneous equations; vectors; Eigen values; properties of Eigen values(without proofs); Cayley-Hamilton theorem (without proof).
[Sections: 2.7.1; 2.7.2; 2.7.6; 2.7.7; 2.10.1; 2.10.2; 2.10.3; 2.12; 2.13; 2.14; 2.15.]

UNIT-II

Differential Equations of first order: Definitions; Formation of a Differential equation; Solution of a Differential equation; Equations of the first order and first degree; variables separable; Linear Equations; Bernoulli's equation; Exact Differential equations; Equations

reducible to Exact equations: I.F found by inspection, I.F of a Homogeneous equation, In the equation M dx+ N dy = 0, $\frac{\frac{\partial M}{\partial y}-\frac{\partial N}{\partial x}}{N}$ is a function of x and $\frac{\frac{\partial N}{\partial x}-\frac{\partial M}{\partial y}}{M}$ is a function of y.

Applications of a first order Differential equations: Newton's law of cooling; Rate of decay of Radio-active materials.
[Sections: 11.1; 11.3; 11.4.1; 11.5; 11.6; 11.9; 11.10; 11.11; 11.12.1; 11.12.2; 11.12.4; 12.6; 12.8]

UNIT-III

Linear Differential Equations: Definitions; Theorem; Operator D; Rules for finding the complementary function; Inverse operator; Rules for finding the Particular Integral; Working procedure to solve the equation; Method of Variation of Parameters; Applications of Linear Differential Equations: Introduction, Oscillatory Electrical Circuits.
[Sections: 13.1; 13.2; 13.3; 13.4; 13.5; 13.6; 13.7;13.8.1;14.1;14.5].

UNIT-IV

Differential Calculus:
Mean Value Theorems: Rolle's theorem, Lagrange's mean value theorem with their geometrical interpretation. Cauchy's mean value theorem. Taylor's and Maclaurin theorems with remainders (without proof), Maclaurin's series, Expansion by use of known series, Taylor's series.
[4.3.1; 4.3.2; 4.3.3; 4.3.4; 4.4.1; 4.4.2; 4.4.3]

Text Books :	B.S.Grewal, "Higher Engineering Mathematics", 44thedition, Khanna publishers, 2017.
References :	1. Erwin Kreyszig, "Advanced Engineering Mathematics", 9th edition, John Wiley & Sons. 2. N.P.Bali and M.Goyal, "A Text book of Engineering Mathematics" Laxmi Publications, 2010.

SEMICONDUCTOR PHYSICS AND NANO MATERIALS
I B. Tech. – I Semester (Code: 24CS102)

Lectures	3	Tutorial	0	Practical	0	Credits	3
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: None.

Course Objectives:

- Build the interest on the concept of quantum mechanics and its importance to solve and evaluate the properties of materials (conductors and semiconductors)
- Provides various types of properties of semiconductor materials and their importance in various device fabrications
- Educate the student on various opto-electronic devices and their applications.
- Provide information about the principles of processing, manufacturing and characterization of nano materials, nanostructures and their applications

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

CO1	Understand concepts of quantum mechanics and its applications to explain material properties
CO2	Know the concept of Fermi level and various semiconductor junctions.
CO3	Familiar with working principles of various opto-electronic devices and their applications.
CO4	Understand importance of nano-materials and their characteristic properties.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3	3	-	-	-	-	-	-	-	-	-	2	3	3
CO2	3	3	2	-	-	-	-	-	-	-	-	2	3	3
CO3	3	3	2	-	-	-	-	-	-	-	-	2	3	3
CO4	3	3	3	-	-	-	-	-	-	-	-	2	3	3

UNIT-I

QUANTUM MECHANICS AND APPLICATIONS:
Schrodinger time independent wave equation, Applications: Particle in one dimensional box, Quantum Tunneling, Scanning Tunneling Microscope, Somerfield free electron theory: conductivity of metals and concept of Fermi level, Failure of quantum free electron theory (Qualitative), Band theory of solids (Kronig –Penny model), E-K diagrams, Effective mass, Concept of hole, Types of Electronic materials: Metals, Semiconductors and Insulators.

UNIT-II

SEMICONDUCTORS AND PROPERTIES: Introduction to semiconductors, intrinsic and extrinsic semiconductors, Direct and Indirect band gap semiconductors. Density of states, carrier concentration equations, Fermi level and temperature dependence, Drift and Diffusion currents, Continuity equation, P-N junction diode (V-I characteristics).

UNIT-III

OPTO-ELECTRONIC DEVICES AND DISPLAY DEVICES: Principle and working of

LED, Semiconducting laser (Laser diode), Photo detectors: Photo diode, PIN & APD Diode, Applications of Photo detectors, Photo voltaic effect, Solar cell, Efficiency of solar cell and applications, Types of liquid crystals, Liquid crystal display(LCD), Opto electric effect(Kerr effect), Magneto optic effect (Faraday Effect)

UNIT-IV

NANO MATERIALS: Introduction to nano technology, quantum confinement, surface to volume ratio, properties of nano materials, synthesis of nano-materials: CVD, sol-gel methods, laser ablation.

Carbon nano tubes: types, properties, applications. Characterization of nano materials: XRD, SEM, applications of nano materials.

Text Books :	1. A textbook of engineering physics by Avadhanulu and Kshirsagar S.Chand & Co. (2013) 2. Applied physics by Dr.*P.SrinivasaRao*. Dr.K.*Muralidhar*
References :	1. Text book on Nanoscience and Nanotechnology (2013): B.S. Murty, P. Shankar, Baldev Raj, B.B. Rath and J. Murday, Springer Science & Business Media. 2. Opto electronics by T. Wilson, J.F.Hawkes, PHI

COMMUNICATIVE ENGLISH
I B. Tech. – I Semester (Code: 24CS103)

Lectures	2	Tutorial	0	Practical	0	Credits	2
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: None.

Course Objectives:

- Enhance the vocabulary competency of the students
- Enable the students to demonstrate proficiency in the use of written English, including proper spelling, grammar, and punctuation
- Enhance theoretical and conceptual understanding of the elements of grammar
- Understand and apply the conventions of academic writing in English

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

CO1	Understand how to build academic vocabulary to enrich their writing skills
CO2	Produce accurate grammatical sentences
CO3	Analyse the content of the text in writing
CO4	Produce coherent and unified paragraphs with adequate support and detail

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	-	-	-	-	-	-	2	2	3	2	2	-	2	-
CO2	-	-	-	-	-	-	2	2	3	2	2	-	2	-
CO3	-	-	-	-	-	-	2	2	3	2	2	-	2	-
CO4	-	-	-	-	-	-	2	2	3	2	2	-	2	-

UNIT-I

1.1 **Vocabulary Development:** Word formation-Formation of Nouns, Verbs & Adjectives from Root words-Suffixes and Prefixes
1.2 **Essential Grammar:** Prepositions, Conjunctions, Articles
1.3 **Basic Writing Skills:** Punctuation in writing
1.4 **Writing Practices:** Mind Mapping, Paragraph writing (structure-Descriptive, Narrative, Expository & Persuasive)

UNIT-II

2.1 **Vocabulary Development:** Synonyms and Antonyms
2.2 **Essential Grammar:** Concord, Common Errors: Practice
2.3 **Basic Writing Skills**: Coherence in Writing: Jumbled Sentences
2.4 **Writing Practices**: Letter writing

UNIT-III

3.1 **Vocabulary Development:** One word Substitutes
3.2 **Essential Grammar:** Tenses, Modal Verbs, Voices
3.3 **Basic Writing Skills:** Using Phrases and clauses
3.4 **Writing Practices:** Note Making

UNIT-IV	
4.1 **Vocabulary Development**: Words often confused 4.2 **Essential Grammar**: Reported speech, Common Errors: Practice 4.3 **Basic Writing Skills**: Sentence structures (Simple, Complex & Compound) 4.4 **Writing Practices**: Paraphrasing & Summarizing, Essay Writing	
Text Books :	1. Communication Skills, Sanjay Kumar & Pushpa Latha. Oxford University Press: 2011. 2. Practical English Usage, Michael Swan. Oxford University Press: 1995. 3. Remedial English Grammar, F.T.Wood. Macmillan: 2007. 4. Study Writing, Liz Hamplyons & Ben Heasley. Cambridge University Press:2006

INTRODUCTION TO PROGRAMMING
I B. Tech. – I Semester (Code: 24CS104)

Lectures	3	Tutorial	0	Practical	0	Credits	3
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: None.

Course Objectives:

- Understand basic concepts of C Programming such as: C-tokens, Operators, Input/output, and Arithmetic rules.
- Develop problem-solving skills to translate 'English' described problems into programs written using C language.
- Use conditional branching, looping and Arrays.
- Understand the concepts of functions and recursion in C.

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

CO1	Choose the right data representation formats based on the requirements of the problem and solve the mathematical problems using operators.
CO2	Solve problems which contain multiple decisions using else…if.
CO3	Work with lists and matrices using arrays.
CO4	Solve real time complex problems by decomposition using user defined functions.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3	2	-	1	-	1	-	-	-	-	-	-	3	2
CO2	-	1	3	2	1	1	-	-	-	-	-	-	2	1
CO3	-	1	2	3	-	1	-	-	-	-	-	-	2	2
CO4	2	1	1	2	-	1	-	-	-	-	-	-	2	1

UNIT-I

Introduction: Computers, Classification of Computers, Software Life Cycle, Algorithms, Flowcharts, Structured Programming, Compilers, Linker, Preprocessor, Standard Input and Output Devices, Popular Features of C.

Variables and Expressions: Introduction, Character Set, Identifiers and Keywords, Variables, Displaying Variables, Reading Variables, Characters and Character Strings, Qualifiers, typedef Statement, Promotion and Typecasting, Value-initialized Variables, Constants, const Qualifier, Operators and Expressions, Operator Precedence and Associativity, Programming Examples.

UNIT-II

Basic Input-Output: Introduction, Single Character Input-Output, String Input and Output, General Output, General Input, Types of Characters in Format Strings, scanf Width Specifier, Search Sets, Assignment Suppression Character, Format Specifiers for scanf, Input Fields for scanf, When scanf Stops Scanning, Programming Examples.

Control Structures: Introduction, if Statement, if-else Statement, Multi-way Decisions, Compound Statements, Loops - for Loop, while Loop, do-while Loop, break Statement, switch Statement, continue Statement, goto Statement, Programming Examples.
UNIT-III
Arrays and Strings: Introduction, How Arrays are Useful? Multidimensional Arrays, Strings, Arrays of Strings, Functions in string.h, Programming Examples. **Scope and Extent:** Introduction, Scope, Extent.
UNIT-IV
Functions: Introduction, Function main, Where are Functions Useful? Functions Accepting More than One Parameter, User Defined and Library Functions, Concepts Associated with Functions (Review), Function Parameters, Return Values, Recursion, Comparison of Iteration and Recursion, Variable Length Argument Lists, Programming Examples.

Text Books :	1. Mastering C by K R Venugopal and Sudeep R Prasad McGraw -Hill Edition.
References :	1. Programming in ANSI C by E. Balaguruswamy, Fifth Edition. 2. Kernighan BW and Dennis Ritchie M, "C programming language", 2nd ed, Prentice Hall. 3. Yashavant P. Kanetkar, "Let us C", BPB Publications. 4. Ashok N.Kamthane, "Programming in C", PEARSON 2nd Edition.

ENGINEERING GRAPHICS
I B. Tech. – I Semester (Code: 24CSL101)

Lectures	1	Tutorial	0	Practical	3	Credits	2.5
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: None.

Course Objectives:

- Clear picture about the importance of engineering graphics in the field of engineering
- The drawing skills and impart students to follow Bureau of Indian Standards
- Give an idea about Geometric constructions and orthographic projections
- Imagination skills about orientation of points, surfaces and solids
- Basic drafting skills of AutoCAD

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

CO1	Draw projections of points and projections of lines using Auto CAD
CO2	Plot projections of surfaces like circle, pentagon, hexagon and rhombus
CO3	Plot the Projections of solids like Prisms and pyramids
CO4	Convert the Isometric views into Orthographic views for simple objects.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	1	2	1	-	-	-	-	-	-	-	-	1	1	2
CO2	3	2	1	-	-	-	-	-	-	-	-	2	3	2
CO3	1	2	3	-	-	-	-	-	-	-	-	1	3	2
CO4	1	2	1	-	-	-	-	-	-	-	-	1	2	3

UNIT-I

INTRODUCTION: Introduction to Engineering drawing, geometrical constructions.

INTRODUCTION TO AUTOCAD: Advantages of AutoCAD over manual drafting, Basics of sheet selection, Draw tools, Modify tools, Dimensioning.

METHOD OF PROJECTIONS: Principles of projection, First angle and third angle projections, projections of points, projections of straight lines inclined to one plane only.

UNIT-II

PROJECTIONS OF PLANES: Projections of plane figures: circle, triangle, pentagon, hexagon and rhombus.

UNIT-III

PROJECTIONS OF SOLIDS: Projections of solids like square, pentagonal, hexagonal prisms and pyramids, axis inclined to one plane only.

UNIT-IV

ORTHOGRAPHIC PROJECTIONS: Conversion of pictorial views into Orthographic views. (Treatment is limited to simple castings only).	
Text Books :	1. Engineering Drawing with AutoCAD by Dhananjay M. Kulkarni, Revised Edition (PHI publication), 2018. 2. Engineering Drawing by N.D. Bhatt & V.M. Panchal, 43rd Edition, (Charotar Publishing House, Anand). (First angle projection) 2014.
References :	Engineering Drawing by Dhananjay A Jolhe, Revised Edition, Tata McGraw hill publishers,2019.

SEMICONDUCTOR PHYSICS LAB
I B.Tech – I Semester (Code: 24CSL102)

Lectures	0	Tutorial	0	Practical	2	Credits	1
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: None.

Course Objectives:

- Basic experiments such as Magnetic Field Measurements, Hall Effect and LCR resonance give the knowledge to apply them in magnetic applications and circuits design.
- The measurements relating to various physical parameters of materials make the student to understand their utility, design and fabrication of several devices.
- The experiments like CRO, Solar Cell, Photocell provides the thorough understanding of Opto Electronic devices useful in Engineering and Industrial applications.
- Utilization of the principles of light such as interference and diffraction to measure wavelength and radius of curvature of Lenses.

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

CO1	Acknowledge the important aspects of earth magnetic field, realize the use of Maxwell's equations in various magnetic applications
CO2	Realization of material properties and parameters
CO3	Get hands-on experience in various opto-electronic devices like CRO, Solar Cell, Photo Cell and their applications
CO4	Apply the phenomenon of interference and LASER principles to find radius of curvature and wavelength respectively by various methods

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3	3	2	2	-	-	-	2	-	-	-	-	-	-
CO2	3	3	2	2	-	-	-	2	2	-	-	-	-	-
CO3	3	3	2	2	2	-	-	2	-	-	-	-	-	-
CO4	3	2	-	-	-	-	-	-	-	-	2	-	-	-

LIST OF EXPERIMENTS

1. Determination of acceleration due to gravity at a place using compound pendulum.
2. To study the variation of intensity of magnetic field along the axis of a circular coil using Stewart-Gee's apparatus.
3. To draw the characteristic curves of P-N Junction diode.
4. Determination of radius of curvature of a Plano convex lens by forming Newton's rings.
5. Determination of wavelengths of mercury spectrum using grating normal incidence method.

6. To draw the characteristic curves of Zener diode.
7. To draw the resonant characteristic curves of L.C.R. series circuit and calculate the Resonant frequency.
8. To draw the characteristic curves of a photocell and calculate the maximum velocity of electron.
9. Verify the laws of transverse vibration of stretched string using Sonometer.
10. Determination of rigidity modulus of the given material of the wire using Torsional pendulum.
11. To draw the load characteristic curves of a solar cell.
12. Determination of Hall coefficient of a semiconductor.
13. Determination of voltage and frequency of an A.C. signal using C.R.O.
14. Determination of Forbidden energy gap of Si &Ge.
15. Determination of wavelength of laser source using Diode laser.
16. To draw the characteristic curves of Photo diode.
17. To draw the Diode valve characteristics.

Any three experiments are virtual

Note: A minimum of **ten (10 no.)** experiments to be done and recorded

ENGLISH COMMUNICATION SKILLS LAB
I B.Tech – I Semester (Code: 24CSL103)

Lectures	0	Tutorial	0	Practical	2	Credits	1
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: None.

Course Objectives:

- Comprehend the importance, barriers and strategies of listening skills in English.
- Illustrate and impart practice Phonemic symbols, stress and intonation.
- Practice oral skills and receive feedback on learners' performance.
- Practice language in various contexts through pair work, role plays, group work and dialogue conversations

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

CO1	Better understand the nuances of English language through audio- visual experience and group activities
CO2	Develop neutralization of accent for intelligibility
CO3	Build confidence to enhance their speaking skills
CO4	Use effective vocabulary both in formal and informal situations

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	-	-	-	-	-	-	-	2	3	-	2	-	-	-
CO2	-	-	-	-	-	-	-	2	3	-	2	-	-	-
CO3	-	-	-	-	-	-	-	2	3	-	2	-	-	-
CO4	-	-	-	-	-	-	-	2	3	-	2	-	-	-

LIST OF EXPERIMENTS

Unit-I

1.1 Introduction to Communication Skills- Importance-Process-Types
1.2 Barriers to Communication & Strategies for effective Communication
1.3 Listening Skills; Importance – Purpose- Process- Types
1.4 Barriers to Listening & Strategies for Effective Listening

Unit-II

2.1 Phonetics; Introduction to Consonant, Vowel and Diphthong sounds
2.2 Syllable & Stress
2.3 Rhythm & Intonation

Unit-III

3.1 Interpersonal Communication in English
3.2 Conversational Practice in English

Unit-IV

4.1 JAM Session

4.2 Debates	
Text Books :	1. Communication Skills, Sanjay Kumar and Pushpa Lata. Oxford University Press. 2011 2. Better English Pronunciation, J.D. O' Connor. Cambridge University Press:1984 3. New Interchange (4rth Edition), Jack C Richards. Cambridge University Press:2015 4. English Conversation Practice, Grant Taylor. McGraw Hill:2001
References :	1. iTell Orell Digital Lab 2. Buzzers for conversations, New Interchange series

INTRODUCTION TO PROGRAMMING LAB
I B.Tech – I Semester (Code: 24CSL104)

Lectures	0	Tutorial	0	Practical	3	Credits	1.5
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: None.

Course Objectives:

- Understand basic concepts of C Programming such as: C-tokens, Operators, Input/output, Arithmetic rules.
- Develop problem-solving skills to translate "English" described problems into Programs written using C language.
- Use Conditional Branching, Looping, and Functions.
- Understand basic concepts of C Programming such as: C-tokens, Operators, Input/output, Arithmetic rules.

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

CO1	Choose the right data representation formats based on the requirements of the problem and solve the mathematical problems using operators.
CO2	Solve problems which contain multiple decisions using else…if.
CO3	Work with lists and matrices using arrays.
CO4	Solve real time complex problems by decomposition using user defined functions.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3	2	2	-	-	-	-	-	-	-	-	-	3	2
CO2	2	3	2	-	-	-	-	-	-	-	-	-	2	1
CO3	2	2	1	-	-	-	-	-	-	-	-	-	2	2
CO4	2	1	2	-	-	-	-	-	-	-	-	-	2	1

LIST OF EXPERIMENTS

1. a) Write a C program to demonstrate the use of printf statement to print values of variables of different data types.
   b) Write a C program to demonstrate the use of printf and scanf statements to read and print values of variables of different data types.
2. Write a C program to perform addition, subtraction, multiplication, division and modulo division on two integer numbers.
3. Write a C program for electricity bill tacking different categories of users, different slabs in each category. (using nested if else statement).
4. Write a C program to evaluate the following using loops
   a)1+x2/2!+x4/4!+...upto 5 terms
   b) x+x3/3!+x5/5!+...upto 5 terms

5. Write a C program to check whether the given number is
   a) Prime or not
   b) Perfect or abundant or deficient
6. Write a C program to print the following patterns

   i)
   ```
   1
   12
   123
   1234
   ```

   ii)
   ```
   1
   1 2 1
   1 2 3 2 1
   1 2 3 4 3 2 1
   ```
7. Write a C program to find the mean, mode, median, and variance of list of values by using one dimensional array.
8. Write a menu driven program to read a list of numbers and perform the following operations
   a) Print the list
   b) Delete duplicates from the list
   c) Reverse the list
9. Write a program to read a list of numbers and search for given number using binary search algorithm and if found display its index otherwise display the message "element not found in the list" using functions.
10. Write a menu driven program to read two matrices and compute their sum and product using functions.
11. Write a menu driven program to read list of student names and perform the following operations using functions.
    a) To print list of names
    b) To sort them in ascending order
    c) To print the list after sorting
12. Write a c program that consists of recursive functions to find
    a) Factorial of a given number
    b) Solve towers of Hanoi with three towers (A,B,C) with three towers initially on tower A.

IT WORKSHOP

I B.Tech – I Semester (Code: 24CSL105)

Lectures	0	Tutorial	0	Practical	2	Credits	1
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: None.

Course Objectives:

- Introduce the internal parts of a computer, peripherals, and I/O ports.
- Demonstrate configuring the system with Windows Operating System and other Application Software's.
- Introduce Office tools such as Word processors, Excel and Presentation tools.
- Demonstrate AI tools such as ChatGPT, Dialogflow.

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

CO1	Explore computer system peripherals and components of a mother board.
CO2	Evaluate computer system architectures.
CO3	Troubleshoot a computer system.
CO4	Prepare a PPT, Certificate and Calculate GPA.
CO5	Generate a report using Mail merge.
CO6	Implement AI Solutions in their respective engineering branches.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	-	-	-	-	-	-	-	-	-	-	-	3	-	-
CO2	-	-	-	-	-	-	-	-	-	-	-	3	-	-
CO3	-	-	-	-	-	-	-	-	-	-	-	3	-	-
CO4	-	3	-	-	-	-	-	-	-	-	-	3	1	-
CO5	-	2	-	-	-	-	-	-	-	-	-	3	1	-
CO6	-	-	3	1	3	-	-	-	-	-	-	3	2	-

LIST OF EXPERIMENTS

1. Explore Peripherals of a Computer, Components of a motherboard and its functions.
2. Install and Uninstall System and Application software on a Computer.
3. Disassemble and Assemble the PC.
4. Troubleshoot a computer.
5. Prepare the following using MS office:
   i) PPT using MS-Power Point.
   ii) Design a Project Certificate and Newsletter using MS-Word
6. Implement the following using Excel:
   i) Create an Excel Work sheet for the six subjects and calculate Total, Average, Grade and Rank.

ii) Merge the contents of two excel sheets using VLOOKUP and sort them. 7. Generating reports using Mail Merge. 8. Prepare a report using Latex or equivalent (FOSS) tool word as word Processors. 9. Prompt Engineering in Chat GPT. 10. Develop a simple AI Chatbot.

References :	1. "IT Essentials PC Hardware and Software Companion Guide", by David Anfinson and Ken Quamme, Third edition, CISCO Press, Pearson Education, 2008, ISBN: 978-1-58713-199-8. 2. "LaTeX Companion" by Frank MittelBach, Ulrike Fischer, Third Edition, Addison-Wesley Professional, 2023. ISBN: 978-0138166489. 3. "ChatGPT: Comprehensive Study On Generative AI Tool" by Midhun Moorthi C, Dr. K. Vimala Devi, Dr. V. Manjula, Tareek Pattewar, First Edition, AG Publishing House,2023, ISBN: 978-81-19338-79-5

NUMERICAL METHODS AND ADVANCED CALCULUS
I B. Tech. – II Semester (Code: 24CS201)

Lectures	2	Tutorial	1	Practical	0	Credits	3
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: None.

Course Objectives:

- Solve algebraic, transcendental and system of linear equations with the help of numerical methods.
- Apply the techniques of numerical integration whenever and wherever routine methods are not applicable and solve the first order ordinary differential equations numerically with the given initial condition using different methods.
- Evaluate double and triple integrals and apply them to find areas and volumes.
- Evaluate the line, surface and volume integrals and learn their inter-relations and applications.

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

CO1	Solve non-linear equations and system of linear equations with the help of Numerical techniques.
CO2	Solve the first order ordinary differential equations numerically with the given initial condition.
CO3	Find the area and volume of plane and three dimensional figures using multiple integrals.
CO4	Apply vector integral theorems to obtain the solutions of engineering problems involving circulation, flux, and divergence in vector fields.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3	3	2	-	-	-	-	-	-	-	2	-	-	-
CO2	3	3	2	-	-	-	-	-	-	-	2	-	-	-
CO3	3	3	2	-	-	-	-	-	-	-	2	-	-	-
CO4	3	3	2	-	-	-	-	-	-	-	2	-	-	-

UNIT-I

Numerical Solution of Equations: Introduction; Solution of algebraic and transcendental equations: Bisection method, Method of false position, Newton-Raphson method; Useful deductions from the Newton-Raphson formula; Solution of linear simultaneous equations; Direct methods of solution: Gauss-Jordan method, Factorization method; Iterative methods of solution: Jacobi's iteration method, Gauss-Seidel iteration method.

[Sections: 28.1; 28.2; 28.3; 28.5; 28.6.2, 28.6.3; 28.7.1; 28.7.2].

UNIT-II

Finite differences and Interpolation: Finite differences: Forward differences, Backward differences; Newton's interpolation formulae: Newton's forward interpolation formula;

Newton's backward interpolation formula; Interpolation with unequal intervals; Lagrange's interpolation formula; Divided differences; Newton's divided difference formula; Numerical integration; Trapezoidal rule; Simpson's one-third rule; Simpson's three-eighth rule; **Numerical solution of ODE's:** Introduction; Euler's method; Runge-Kutta method. [Sections: 29.1.1; 29.1.2; 29.6; 29.9; 29.10; 29.11; 29.12; 30.4; 30.6; 30.7; 30.8; 32.1; 32.4; 32.7].
UNIT-III
Multiple Integrals: Double integrals; Change of order of integration; Double integrals in polar coordinates; Area enclosed by plane curves; Triple integrals; Volumes of solids: Volume as Triple integral, Change of variables: For triple integrals. [Sections: 7.1; 7.2; 7.3; 7.4; 7.5; 7.6.2; 7.7.2].
UNIT-IV
Vector calculus and its Applications: Scalar and vector point functions; Del applied to scalar point functions-Gradient: Definition, Directional derivative; Del applied to vector point functions: Divergence, Curl; Line integral; Surfaces: Surface integral, Flux across a surface; Green's theorem in the plane (without proof); Stokes theorem (without proof); Gauss Divergence theorem (without proof). [Sections: 8.4; 8.5.1; 8.5.3; 8.6; 8.11; 8.12.2; 8.12.3; 8.13; 8.14; 8.16]

Text Books :	B.S.Grewal, "Higher Engineering Mathematics", 44th edition, Khanna publishers, 2017.
References :	1. Erwin Kreyszig, "Advanced Engineering Mathematics", 9th edition, John Wiley & Sons. 2. N.P.Bali and M.Goyal, "A Text book of Engineering Mathematics" Laxmi Publications, 2010.

ENGINEERING CHEMISTRY
I B. Tech. – II Semester (Code: 24CS202)

Lectures	3	Tutorial	0	Practical	0	Credits	3
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: None.

Course Objectives:

- Familiarize importance of usage of various polymers and fuels in household & industry
- Outline the basics for the construction of electrochemical cells, batteries and fuel cells. Understand the mechanism of corrosion and how it can be prevented.
- Impart the concept of soft and hard waters, softening methods of hard water and various instrumental methods of analysis of samples.
- Outline the basics of some advanced concepts like computational chemistry, nanomaterials and liquid crystals.

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

CO1	Explain the preparation, properties, and applications of plastics, elastomers and biodegradable polymers also to explain calorific value, characteristics and applications of conventional and alternative fuels.
CO2	Apply the knowledge of electrochemistry for understanding the working of electrodes and electrochemical energy systems, as well as corrosion theories and protection methods.
CO3	Analyse the methods to produce soft water for industrial use and potable water by economical means and study the principles of different analytical techniques and their applications.
CO4	Demonstrate the knowledge of computational chemistry, and applications of advanced materials in engineering.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3	3	2	-	-	2	-	-	-	-	2	-	-	-
CO2	3	3	3	-	-	-	-	-	-	-	2	-	-	-
CO3	3	3	3	2	-	2	-	-	-	-	3	-	-	-
CO4	3	3	2	3	3	-	-	-	-	-	2	-	-	-

UNIT-I

POLYMERS AND FUEL CHEMISTRY: Introduction to polymers, functionality of monomers. Thermoplastics and Thermo-setting plastics- Preparation, properties and applications of PVC and Bakelite.
Biodegradable polymers- Preparation, properties and applications of PHB and PHBV
Elastomers-Preparation, properties and applications of Buna S and Buna N
Fuels-Types of fuels, calorific value of fuels-determination by Bomb calorimeter, Liquid Fuels-refining of petroleum, Knocking, Octane and Cetane number, Flue gas analysis by Orsat's

apparatus, Introduction to alternative fuels-methanol, ethanol and bio fuel-bio diesel (preparation and applications).

UNIT-II

ELECTROCHEMICAL CELLS AND CORROSION: Single electrode potential, Reference electrodes- construction and working of standard hydrogen electrode and calomel electrode; Batteries (Li ion battery and zinc air cells), fuel cells (H2-O2, and molten carbonate). Electrochemical sensors-potentiometric sensors and amperometric sensors with examples.
Corrosion-Definition, theories of corrosion (chemical and electrochemical), Types of corrosion-galvanic corrosion, differential aeration corrosion, stress corrosion, factors influencing rate of corrosion, corrosion control (cathodic protection), Protective coatings-electroplating (Gold) and electroless plating (nickel).

UNIT-III

WATER TECHNOLOGY: Soft and hard water, Estimation of hardness of water by EDTA Method-numerical problems, Boiler troubles-Priming, foaming, scale and sludge, Caustic embrittlement, Specifications for drinking water- World health organization (WHO) standards, Industrial water treatment- Ion-exchange process, desalination of brackish water by reverse osmosis (RO) and electro dialysis.
INSTRUMENTAL METHODS OF ANALYSIS: Electromagnetic spectrum-UV (Principle, instrumentation, and applications), FT-IR (Principle, instrumentation, and applications), magnetic resonance imaging and CT scan (procedure and applications).

UNIT-IV

Organic reactions and synthesis of a drug molecule
Computational chemistry: Introduction to computational chemistry, and docking studies
Semiconductors-Introduction, basic concept, Types-Intrinsic & Extrinsic Semiconductors, applications.
Nano materials: Introduction, classification of nano materials, engineering applications, properties and applications of Carbon nano tubes and Graphenes nanoparticles.
Liquid crystals: Introduction, liquid crystalline displays (LCD)-applications. Polymers for light emitting diodes (LEDs)-Introduction, classification of polymer LEDs, Organic LEDs-their commercial uses.

Text Books :	1. P.C. Jain and Monica Jain, "Engineering Chemistry" Dhanpat Rai Pub, Co., New Delhi 17th edition (2017). 2. Seshi Chawla, "Engineering Chemistry" DhanpatRai Pub, Co LTD, New Delhi 13 th edition, 2013. 3. S.S. Dara, "A Textbook of Engineering Chemistry", S.Chand & Co, (2010).
References :	1. Engineering Chemistry by K. Maheswaramma, Pearson publishers 2015. 2. Textbook of Polymer Science, Fred W. Billmayer Jr, 3rd Edition 3. B. S. Murthy, P. Shankar and others, "Textbook of Nanoscience and Nanotechnology", University press (latest edition) 4. CNR Rao and JM Honig (Eds) "Preparation and characterization of materials" Academic press, New York (latest edition)

BASIC ELECTRICAL AND ELECTRONICS ENGINEERING
I B. Tech. – II Semester (Code: 24CS203)

Lectures	3	Tutorial	0	Practical	0	Credits	3
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: Basics of Mathematics, Physics and Chemistry

Course Objectives:

- Develop a comprehensive understanding of basic electrical circuit principles, including the analysis of DC and AC circuits using laws and theorems.
- Analyze the construction, working principles, and applications of various electrical machines like DC machines, transformers, and AC machines.
- Comprehend the fundamental properties and applications of semiconductor materials and devices, focusing on diodes and their practical uses.
- Acquire proficiency in the operation and applications of transistors and operational amplifiers in electronic circuits, including their use in amplification and switching.

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

CO1	Solve and analyze basic DC and AC electrical circuits using fundamental laws and theorems.
CO2	Explain the construction, operation, and applications of different types of electrical machines.
CO3	Acquire an understanding of semiconductor materials and devices and their practical applications in electronic circuits.
CO4	Illustrate and analyze basic electronic circuits involving transistors (BJTs and MOSFETs) and operational amplifiers.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3	3	3	2	-	-	-	-	-	-	2	3	-	-
CO2	3	3	2	2	-	-	-	-	-	-	2	3	-	-
CO3	3	3	2	1	-	-	-	-	-	-	2	3	-	-
CO4	3	3	3	1	-	-	-	-	-	-	2	3	-	-

UNIT-I

Basic Definitions: Electric charge, Current, Voltage, Power, and Energy, Ohm's Law

DC Circuits: Series and Parallel Circuits their Characteristics and analysis, Kirchhoff's Voltage Law (KVL) and Kirchhoff's Current Law (KCL), Simple problems on circuit analysis. Thevenin's, Norton's and Superposition theorem.

AC Fundamentals: Alternating Current (AC) and Voltage, Peak, RMS, and Average values of AC, form and peak factors, Phasors and phase relationships in AC circuits. AC excitation to R, L, C and their combination (series and parallel), Impedance and Power triangle.

3-Phase circuits: Star and Delta connections, relation between line and phase values of currents and voltages with derivations.

UNIT-II

DC Machines:

Construction and working of DC generator and its emf equation. Types of DC generators. Open circuit characteristic of DC shunt generator. Construction and working of DC motor and its torque equation, load test on DC shunt motor. Applications of DC machines.

Transformers (AC non-rotating):

Construction and working of Transformer, Types of transformers, Transformer emf equation, losses and efficiency, Regulation, load test on transformer, Applications.

AC Machines (AC rotating):

Induction Motors: Construction and working of single-phase induction motor. Capacitors start single phase induction motor.

Synchronous Machines: Construction and working of Alternator, Construction and working of Synchronous motor and their applications.

UNIT-III

Diodes and Applications: Difference between Conductors, Insulators, and Semiconductors-Intrinsic and Extrinsic Semiconductors-N-type and P-type semiconductors.

PN Junction Diode: Construction, working, and VI characteristics.

Zener Diode: Construction, working, and VI characteristics, Its application as voltage regulator.

Rectifiers: Half-wave and Full-wave rectifiers, working principles, and efficiency

UNIT-IV

Transistors:

Bipolar Junction Transistor (BJT): Construction, operation, and characteristics in Common Emitter, Common Base, and Common Collector configurations. MOSFET Construction, working principle, characteristics, and applications as amplifier and as switch.

Operational Amplifiers (Op-Amps):

Characteristics and parameters, Inverting and Non-inverting Amplifiers, Summing amplifier, Difference amplifier, Integrator, Differentiator with their Circuit diagrams, formulas, and applications.

Text Books :	1. S.K. Bhattacharya, "Basic Electrical and Electronics Engineering", Pearson Publications, Second Edition. 2. V.K. Mehta, Rohit Mehta, "Basic Electrical Engineering", S.Chand Publishers, Sixth edition.

	3. V.K. Mehta, Rohit Mehta, "Principles Of Electronics" S.Chand Publishers, 11th edition.
References :	1. Nagsarkar T K and Sukhija M S, "Basics of Electrical and Electronics Engineering", Oxford press University Press, 3rd edition. 2. T. Thyagarajan, K.P. Sendur Chelvi, T.R. Rangaswamy, "Electrical, Electronics and Computer Engineering" by, New Age International (P) Ltd., Publishers. 3. PV Prasad, S.Sivanagaraju "Electrical Engineering Concepts and Applications" Cenage Publishers, 1st Edition.

PROGRAMMING FOR PROBLEM SOLVING
I B. Tech. – II Semester (Code: 24CS204)

Lectures	3	Tutorial	0	Practical	0	Credits	3
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: Introduction to Programming (24CS104)

Course Objectives:

- Understand and apply pointers in C for memory management and manipulation, including arrays, strings, functions, and structures.
- Learn file handling, dynamic memory allocation, and object-oriented programming concepts, including classes, inheritance, and polymorphism.
- Learn to define and use classes and objects, including access control, member functions, and overloading in C++.
- Implement operator overloading and various forms of inheritance in C++.

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

CO1	Effectively use pointers for advanced data handling, perform pointer arithmetic, and manipulate structures and unions in C.
CO2	Manage files, allocate memory dynamically, and apply object-oriented principles like inheritance and polymorphism in programming.
CO3	Create and manage classes and objects, implement constructors and destructors, and use function overloading and inline functions effectively.
CO4	Overload operators and use different inheritance types, including abstract, multilevel, multiple, hierarchical, and hybrid inheritance.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3	2	-	1	-	1	-	-	-	-	-	-	3	2
CO2	-	1	3	2	1	1	-	-	-	-	-	-	2	1
CO3	-	1	2	3	-	1	-	-	-	-	-	-	2	2
CO4	2	1	1	2	-	1	-	-	-	-	-	-	2	1

UNIT-I

Pointers: Introduction, Definition and Uses of Pointers, Address Operator &, Pointer Variables, Dereferencing Pointers, void Pointers, Pointer Arithmetic, Pointers to Pointers, Pointers and Arrays, Passing Arrays to Functions, Pointers and Functions, Accessing Arrays Inside Functions, Pointers and Two-dimensional Arrays, Array of Pointers, Pointer Constants, Pointers and Strings, Pointers in Standard String Library Functions, Two-dimensional Array of Characters, Array of Pointers to Strings, More String Library functions, Pointers to Functions.

Structures and Unions: Introduction, Declaring and Using Structures, Structure Initialization, Structure within Structure, Operations on Structures, Array of Structures, Array within Structure, Creating User Defined Data Types, Pointers to Structures, Pointers within Structures,

Structures and Functions, Unions, Differences between Structures and Unions, Operations on A union, Scope of A union, Bit Fields in Structures.

UNIT-II

Files: Introduction, End of File, File-handling Functions, File Types, Unbuffered and Buffered Files, Error Handling.

Dynamic Memory Allocation: Introduction, Library Functions for Dynamic Memory Allocation.

Object-Oriented Paradigm: Why new Programming Paradigms?, Evolution of Programming Paradigms, Objects, Classes, Inheritance, Polymorphism, Streams based I/O, Scope Resolution Operator, Variable Definition at the Point of Use, Type Conversion.

UNIT-III

Classes and Objects: Introduction, Class Specification, Class Objects, Accessing Class Members, Defining Member Functions, Outside member functions as inline, Accessing member functions within the Class, Data hiding, Passing objects as arguments, Returning objects from functions, Friend functions and Friend classes, Static data and member functions, Constructors, Destructors.

Overloading: Introduction, Parameters Passing by Reference, Inline Functions, Function Overloading.

UNIT-IV

Operator Overloading: Introduction, Overloadable Operators, Unary Operator Overloading, operator Keyword, Operator return values, Binary Operator Overloading, Arithmetic Operator Overloading, Concatenation of strings, Comparison Operators, Arithmetic Assignment Operators, Overloading with friend functions, Assignment operator Overloading.

Inheritance: Introduction, Derived class declaration, forms of inheritance, inheritance and member accessibility, Constructors in derived classes, Destructors in derived classes, abstract classes, multilevel inheritance, multiple inheritance, hierarchical inheritance, hybrid inheritance, object composition-delegation.

Text Books :	1. "Programming with C" by Byron Gottfried, Second Edition, Schaum's Outline Series, McGraw-Hill. 2. "Programming with C++" by John R. Hubbard, Second Edition, Schaum's Outline Series, McGraw-Hill.
References :	1. Nagsarkar T K and Sukhija M S, "Basics of Electrical and Electronics "Programming in ANSIC" by E. Balaguruswamy, Fifth Edition, McGraw Hill Education India. 2. "Let us C" by Yashavant P.Kanetkar, 14th Edition, BPB Publications. 3. Kernighan BW and Dennis Ritchie M, "C programming language", 2nd edition, Prentice Hall.

DISCRETE MATHEMATICS
I B. Tech. – II Semester (Code: 24CS205)

Lectures	3	Tutorial	0	Practical	0	Credits	3
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: None.

Course Objectives:

- Understand operations on discrete structures such as sets, functions, and relations. Formulate short proofs using methods of proof of an implication. Verify the correctness of an argument using propositional logic and truth tables. Construct mathematical arguments using logical connectives and quantifiers.
- Verify the correctness of an argument using rules of inference for quantified propositions. Apply algorithms and use definitions to solve problems to prove statements in elementary number theory. Understand counting and indirect counting techniques and combinatory in the context of discrete probability.
- Understand sequences, generating functions, and recurrence relations. Understand and compute coefficients for generating functions. Understand and solve homogeneous recurrence relations.
- Understand and solve Inhomogeneous recurrence relations.
- Understand the properties of binary relations, partial orderings and lattices. Construct graphs and adjacency matrices for binary relations.

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

CO1	Understand the basic principles of sets, relations, functions and inference rules for validating arguments.
CO2	Prove that the given statement is valid by using mathematical induction and utilize a variety of counting strategies to solve computational problems.
CO3	Discuss different methods for solving different types of recurrence relations.
CO4	Understand various operations and representations of a binary relation.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3	3	-	-	-	-	-	-	-	-	-	-	3	-
CO2	3	3	-	-	-	-	-	-	-	-	-	-	3	-
CO3	3	3	-	-	-	-	-	-	-	-	-	-	3	-
CO4	3	3	-	-	-	-	-	-	-	-	-	-	3	-

UNIT-I

Foundations: Sets, Relations and Functions, Fundamentals of Logic, Logical Inferences, Methods of Proof of an implication, First order Logic & Other methods of proof.

UNIT-II

Rules of Inference for Quantified propositions, Mathematical Induction.
Elementary Combinatorics: Basics of Counting, Combinations and Permutations,

Enumeration of Combinations and Permutations, Enumerating Combinations and Permutations with repetitions, Enumerating Permutation with Constrained repetitions.
UNIT-III
Recurrence relations: Generating functions of sequences, Calculating Coefficients of Generating Functions **Recurrence Relations:** Solving recurrence relations by Substitution and generating functions, The methods of characteristic roots.
UNIT-IV
Recurrence Relations: solutions of Inhomogeneous recurrence relations. **Relations:** Special properties of binary relations, Operations on relation. Ordering relations, Lattice, Paths and Closures, Directed Graphs and Adjacency Matrices.

Text Books :	Toe L.Mott, Abraham Kandel &Theodore P.Baker, "Discrete Mathematics for Computer Scientists & Mathematicians", PHI 2nd edition, 2012.
References :	1. C.L. Liu, "Elements of Discrete Mathematics", McGraw-Hill Education, 2nd edition. 2. Rosen, "Discrete Mathematics", McGraw-Hill Education, 8th edition.

ENGINEERING MECHANICS AND SURVEYING LAB
I B.Tech – II Semester (Code: 24CSL201)

Lectures	1	Tutorial	0	Practical	2	Credits	2
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: Basics of Mathematics & Physics

Course Objectives:

- Equip students with the fundamental principles and techniques necessary to solve problems related to forces and supports in engineering mechanics.
- Provide students with a thorough understanding of frictional forces and their applications in engineering mechanics.
- Provide students with the essential skills and knowledge to analyze truss structures.
- Provide students with a deep understanding of rotational dynamics and the principles governing the motion of rigid bodies.
- Provide students with the fundamental skills and techniques required for various surveying methods in civil engineering.

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

CO1	Utilize the Parallelogram Law, Triangle Law, and Polygon Law to determine the resultant of concurrent forces. Apply Varignon's Principle to find the magnitude and position of the resultant force in a system. Calculate support reactions for beams subjected to transverse loads using principles of equilibrium. Calculate the geometric center (centroid) of various lamina shapes through integration and composite area methods.
CO2	Determine the coefficient of static friction between a block and a rough surface under horizontal force. Calculate the angle of inclination at which a block just starts to slide down an inclined plane. Determine the axial forces in truss members using the method of joints.
CO3	Experimentally determine and verify the angular acceleration of a rolling disc on an inclined plane. Calculate the moment of inertia of a flywheel through experimental procedures.
CO4	Perform a cross-staff survey to determine the area of a plot. Determine the elevation difference between two points and the height of the ceiling of a building using leveling techniques. Determine the horizontal distance between inaccessible points and the height of an object using a theodolite.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs			
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**	**4**
CO1	3	3	-	2	-	-	-	-	1	-	2	3	3	3	3
CO2	3	3	-	2	-	-	-	-	1	-	2	3	3	3	3
CO3	3	3	-	2	-	-	-	-	1	-	2	3	3	3	3
CO4	3	3	-	2	-	-	-	-	1	-	2	3	3	3	3

LIST OF EXPERIMENTS

FORCE SYSTEM

Force – characteristics of force – system of forces – moment of a force - laws of forces – supports and their reactions, Centroid – determination of centroid for plane figures. (2)

List of experiments

1. Determination of the magnitude of the resultant force using a) Parallelogram law and b) Triangle law c) Polygon law
2. Determination of the magnitude of the resultant force using Varignon's principle.
3. Determination of the support reactions for a beam subjected to transverse loads.
4. Determination of the geometric center of different lamina.

FRICTION

Friction – laws of friction – coefficient of friction – angle of repose. (2)

List of experiments

5. Determination of the coefficient of static friction between the block and rough surface when the block is subjected to horizontal force.
6. Determination of the angle of inclination at which a block just starts to slide down an inclined plane.

ANALYSIS OF TRUSS

Truss – Method of analysis. (2)

List of experiments

7. Determination of the axial forces in the truss members.

MASS MOMENT OF INERTRIA & ROTATION OF A RIGID BODY ABOUT A FIXED AXIS

Area moment of inertia – mass moment of inertia – Relation between mass and area moment of inertia, Kinematics of rotation – Equation of motion for a rigid body rotating about a fixed axis – D'Alembert's principle. (3)

List of experiments

8. Verification of angular acceleration of a rolling disc on an inclined plane.
9. Determination of the moment of inertia of flywheel.

SURVEYING

Surveying – principles of surveying – chain surveying – theodolite surveying – leveling. (5)

List of experiments

10. Determination of the area of a plot using cross - staff survey.
11. Determination of the elevation difference between two points using leveling and height of ceiling of a building.
12. Determination of the horizontal distance between inaccessible points using theodolite.
13. Determination of the height of an object using theodolite.

ENGINEERING CHEMISTRY LAB
I B.Tech – II Semester (Code: 24CSL202)

Lectures	0	Tutorial	0	Practical	2	Credits	1
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: None.

Course Objectives:

- Familiarize students with practical chemical analysis techniques for determining key water quality parameters.
- Provide hands-on experience in performing volumetric and instrumental titrations to understand their chemical principles and applications.
- Develop proficiency in using laboratory equipment, following safety protocols, and accurately conducting experiments.
- Teach students the synthesis of common organic compounds and their characterization techniques.

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

CO1	Determine water quality parameters such as alkalinity and hardness.
CO2	Conduct volumetric titrations to estimate the concentration of chemical substances.
CO3	Apply instrumental methods such as pH metry and conductometry for titration experiments and colorimetry for verification of Beers law.
CO4	Synthesize and characterize common organic compounds like soap, resins, and aspirin.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3	3	2	3	-	3	-	-	-	-	2	-	-	-
CO2	3	3	2	2	-	3	-	-	-	-	2	-	-	-
CO3	3	3	2	3	3	-	-	-	-	-	2	-	-	-
CO4	2	-	-	3	-	2	-	-	-	-	-	-	-	-

LIST OF EXPERIMENTS

1. Determination of Alkalinity of Tap water.
2. Determination of Total Hardness of ground water sample by EDTA method
3. Estimation of Mohr's salt by Permanganometry.
4. Estimation of Active Chlorine Content in Bleaching Powder
5. pH metric titration between strong acid and strong base.
6. Conductometric Titrations between Strong acid and strong base.
7. Verification of Beers Law using potassium permanganate by colorimetry.
8. Preparation of Soap.
9. Preparation of Urea-formaldehyde resin
10. Preparation of Aspirin.

Text Books :	1. Practical Engineering Chemistry by K.Mukkanti, Etal, B.S. Publicaitons, Hyderabad, 2009. 2. Inorganic quantitative analysis, Vogel, 5th edition, Longman group Ltd. London, 1979.
References :	1. Text Book of engineering chemistry by R.n. Goyal and Harrmendra Goel. 2. A text book on experiments and calculations- Engineering Chemistry. S.S. Dara. 3. Instrumental methods of chemical analysis, Chatwal, Anand, Himalaya Publications.

BASIC ELECTRICAL AND ELECTRONICS ENGINEERING LAB
I B.Tech – II Semester (Code: 24CSL203)

Lectures	0	Tutorial	0	Practical	3	Credits	1.5
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: None.

Course Objectives:

- Enable students to verify and apply basic electrical laws and theorems
- Equip students with practical skills to conduct experiments on electrical machines
- Provide students with hands-on experience in analyzing the V-I characteristics of semiconductor devices and their applications
- Develop students' abilities to analyze and characterize the operations transistor, MOSFET and Op-Amps.

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

CO1	Verify and apply basic network theorems through practical experimentation, thereby gaining a deep understanding of circuit analysis techniques.
CO2	Demonstrate the ability to perform experiments on electrical machines and interpret their operational characteristics and performance metrics.
CO3	Analyze the V-I characteristics of diodes and understand their roles in rectifier circuits and enhancing their understanding of semiconductor behavior in practical applications.
CO4	Characterize the performance of BJTs, MOSFETs and designing and analyzing operational amplifier circuits to prepare them for advanced electronic circuit design.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3	3	2	3	-	-	-	3	2	-	1	3	-	-
CO2	3	3	2	3	-	-	-	3	2	-	1	3	-	-
CO3	3	3	2	3	-	-	-	3	2	-	1	3	-	-
CO4	3	3	2	3	-	-	-	3	2	-	1	3	-	-

LIST OF EXPERIMENTS

1. Verification of KCL and KVL
2. Verification of Superposition theorem
3. Verification of Thevenin's theorem
4. Verification of Norton's theorem
5. Parameters of choke coil
6. Open Circuit Characteristics of DC Shunt Generator
7. Load Test on DC Shunt Motor
8. Load Test on Transformer

9. V-I characteristics of PN junction Diode
10. V-I characteristics of Zener Diode
11. Half-Wave Rectifier
12. Full-Wave Rectifier
13. Characteristics of BJT in Common Emitter Configuration
14. Characteristics of MOSFET
15. Summing and Difference Amplifiers using Op-Amps

Note: Minimum 10 experiments should be carried.

PROGRAMMING FOR PROBLEM SOLVING LAB
I B.Tech – II Semester (Code: 24CSL204)

Lectures	0	Tutorial	0	Practical	3	Credits	1.5
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: Introduction to Programming Lab (24CSL104)

Course Objectives:

- Understand and apply pointers in C for memory management and manipulation, including arrays, strings, functions, and structures.
- Learn file handling, dynamic memory allocation, and object-oriented programming concepts, including classes, inheritance, and polymorphism.
- Learn to define and use classes and objects, including access control, member functions, and overloading in C++.
- Implement operator overloading and various forms of inheritance in C++.

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

CO1	Effectively use pointers for advanced data handling, perform pointer arithmetic, and manipulate structures and unions in C.
CO2	Manage files, allocate memory dynamically, and apply object-oriented principles like inheritance and polymorphism in programming.
CO3	Create and manage classes and objects, implement constructors and destructors, and use function overloading and inline functions effectively.
CO4	Overload operators and use different inheritance types, including abstract, multilevel, multiple, hierarchical, and hybrid inheritance.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3	2	2	-	-	-	-	-	-	-	-	-	3	2
CO2	2	3	2	-	-	-	-	-	-	-	-	-	2	1
CO3	2	2	1	-	-	-	-	-	-	-	-	-	2	2
CO4	2	1	2	-	-	-	-	-	-	-	-	-	2	1

LIST OF EXPERIMENTS

1. Write C programs for the following using pointers and functions:
   a) Declare the array of 20 integer numbers. Find and display the average of only even numbers by using pointers.
   b) Write a program to reverse the string using pointers.
   c) Calculate the addition of two 4 X 4 matrices using pointers. Take the input from user.

2. A Bookshop maintains the inventory of books that are being sold at the shop. The list includes details such as author, title, price, publisher and stock position. Whenever a customer wants a book the sales person inputs the title and the author, and the system searches the list and displays whether it is available or not. If it is not, an appropriate message is displayed, if it is,

then the system displays the book details and request for the number of copies required, if the requested copies are available the total cost of the requested copies is displayed otherwise the message "required copies not in stock" is displayed. Write a program for the above in structures with suitable functions.

3. Write a program in C to merge two files and write them to another file.

4. Write a C program to read a data file of students' records with fields (Regno, Name, M1, M2, M3, M4, M5) and write the successful students data (percentage > 40%) to a data file.

5. Write a C++ program with a class to represent a bank account. Include the following members:
   Data Members – 1. Name of the depositor 2. Account Number 3. Type of account 4. Balance amount in the account.
   Member function – 1. To assign initial values 2. To deposit an amount 3. To withdraw an amount after checking the balance 4. To display name and Balance.

6. Create a class called 'TIME' that has
   - three integer data members for hours, minutes and seconds
   - constructor to initialize the object to zero
   - constructor to initialize the object to some constant value
   - member function to add two TIME objects
   - member function to display time in HH:MM:SS format
   Write a main function to create two TIME objects, add them and display the result in HH:MM:SS format.

7. Write a C++ program to illustrate static member functions.

8. Write a C++ program to implement function overloading in order to compute power(m,n) where i) m is double and n is int ii) m and n are int. Use a default value of 2 for n to make the function to calculate squares when this argument is omitted.

9. Create a 'STRING' class which overloads '==' operator to compare two STRING objects using a) friend function b) without using friend function.

10. Write a C++ program to overload '++' and '--'operators to convert a string to uppercase and lowercase respectively.

11. Write a C++ program on Composition.

12. Write a C++ program on Inheritance.

PROBABILITY AND STATISTICS
II B. Tech. – III Semester (Code: 24CS301)

Lectures	2	Tutorial	1	Practical	0	Credits	3
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: None

Course Objectives:

- Employ discrete and continuous probability distributions to analyze and solve real world problems in Engineering fields.
- Estimate the point and interval estimators of the mean, variance and proportion for the given Sample data and apply Z-test, t-test to various real-life problems
- Apply various sample tests like F-test and $\chi 2$ -test for decision making regarding the population based on sample data.
- Compute the level of correlation, the best fit curve to the given data by the method of least squares and also perform ANOVA arising in the field of engineering.

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

CO1	Apply discrete and continuous probability distributions to various problems arising in Engineering applications.
CO2	Perform Test of Hypothesis for a population parameter for single sample.
CO3	Perform Test of Hypothesis for population parameters for multiple samples.
CO4	Interpret the results of correlation, regression and one way ANOVA for the given data.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3	3	-	-	-	-	-	-	-	-	2	-	-	-
CO2	3	3	2	-	-	-	-	-	-	-	2	-	-	-
CO3	3	3	2	-	-	-	-	-	-	-	2	-	-	-
CO4	3	3	3	-	-	-	-	-	-	-	2	-	-	-

UNIT-I

Descriptive measures, Random variables and Probability distributions:
Arithmetic mean, median and mode, Random variables, Binomial distribution, The mean and variance of a probability distribution, Poisson approximation to the Binomial distribution, Continuous Random Variables, Normal Distribution, Normal Approximation to the Binomial Distribution, Uniform Distribution, Weibull distribution.
(Sections 2.5, 4.1, 4.2, 4.4, 4.6, 5.1, 5.2, 5.3, 5.5, 5.9.)

UNIT-II

Sampling distributions and Inferences concerning one mean:
Populations and Samples, The sampling distribution of the mean (σ known), The sampling distribution of the mean (σ unknown), The sampling distribution of the variance, Point

estimation, Interval estimation, Tests of Hypotheses, Null Hypothesis and Tests of hypotheses, Hypothesis concerning one mean. (Sections 6.1, 6.2, 6.3, 6.4, 7.1, 7.2, 7.4, 7.5, 7.6)
UNIT-III
Comparing two treatments and Inferences concerning variances: Comparisons-Two independent large samples, Comparisons-Two independent small samples, matched pairs comparisons, the estimation of variances, Hypotheses concerning one variance, Hypotheses concerning two variances. (Sections 8.2, 8.3, 8.4, 9.1, 9.2, 9.3)
UNIT-IV
Inferences concerning proportions, Regression Analysis and Analysis of variance: Estimation of proportions, Hypotheses concerning one proportion, Hypotheses concerning two proportions, the method of least squares, curvilinear regression, multiple regression, correlation, some general principles, Completely Randomized Designs (One way ANOVA). (10.1, 10.2, 10.3, 11.1, 11.3, 11.4, 11.6, 12.1, 12.2)

Text Books :	Miller & Freund's "Probability and Statistics for Engineers", Richard A. Johnson, 8th Edition, PHI, 2011.
References :	1. R.E Walpole, R.H. Myers & S.L. Myers „Probability & Statistics for Engineers and Scientists", 6th Edition, PHI. 2. Murray R Spiegel, John J. Schiller, R. Alu Srinivas Probability & Statistics", Schaum's outline series.

DIGITAL LOGIC DESIGN

II B. Tech. – III Semester (Code: 24CS302)

Lectures	3	Tutorial	0	Practical	0	Credits	3
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: Basic Computer Knowledge.

Course Objectives:

- Apply various simplification methods to simplify the Boolean expressions.
- Design and analysis of combinational circuits.
- Design and analysis of sequential circuits.
- Design various counters, registers and PLD's.

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

CO1	Understand different number systems, binary codes and conversion between number system. Understand and apply Boolean algebra and K-maps to simplify Boolean functions
CO2	Understand and apply tabulation method to simplify the Boolean functions. Understand, analyze and design various combinational circuits.
CO3	Know the fundamentals of various flip-flops, analyze and design sequential circuits.
CO4	Understand design of various registers and counters. Design of various PLD's for Boolean functions.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3	3	3	-	-	-	-	-	-	-	-	3	-	-
CO2	3	3	3	-	-	-	-	-	-	-	-	3	-	-
CO3	3	3	3	-	-	-	-	-	-	-	-	3	-	-
CO4	3	3	3	-	-	-	-	-	-	-	-	3	-	-

UNIT-I

DIGITAL SYSTEMS AND BINARY NUMBERS: Digital System, Binary Numbers, Number base Conversions, Octal and Hexadecimal Numbers, Complements of Numbers, Signed Binary Numbers, Binary Codes, Binary Storage and Registers, Binary Logic, Error Detection and Correction: 7 bit Hamming Code.

BOOLEAN ALGEBRA & LOGIC GATES: Introduction, Basic definitions, Axiomatic definition of Boolean algebra, Basic theorems and properties of Boolean algebra, Boolean functions, Canonical and Standard Forms, Other Logic Operations, Digital logic gates.

GATE –LEVEL MINIMIZATION: Introduction, The map method, Four-variable K-Map, Product-of-Sums Simplification, Don't –Care Conditions.

UNIT-II

MINIMIZATION: The Tabulation method, Determination of prime implicants, Selection of prime-implicants.

COMBINATIONAL LOGIC: Introduction, Combinational Circuits, Analysis Procedure, Design Procedure, Binary Adders - Subtractors, Decimal Adder, Magnitude Comparator, Decoders, Encoders, Multiplexers.	
UNIT-III	
SYNCHRONOUS SEQUENTIAL LOGIC: Introduction, Sequential Circuits, Storage Elements - Latches, Storage Elements -Flip Flops, Analysis of Clocked Sequential Circuits: State Equations, State Table, State Diagram, Flip Flop Input Equations, Analysis with D, JK and T Flip Flops; State reduction and Assignment, Design Procedure.	
UNIT-IV	
REGISTERS and COUNTERS: Registers, Ripple Counters, Synchronous Counters. **MEMORY and PROGRAMMABLE LOGIC:** Introduction, Random Access Memory: Read and Write Operations, Types of Memories; Read Only Memory, Programmable Logic Devices: PROM, PLA, PAL.	
Text Books :	1. M. Morris Mano, Michael D. Ciletti, "Digital Design", 5th Edition, PrenticeHall, 2013. ISBN-10: 0-13-277420-8. 2. A. Anand Kumar, "fundamentals of digital circuits", 4th Edition, PHI.
References :	1. John F. Wakerly, "Digital Design: Principles and Practices", 4th Edition, Pearson, 2006. 2. Brian Holdsworth, Clive Woods, "Digital Logic Design", 4th Edition, Elsevier Publisher, 2002. 3. Donald E Givone, "digital principles and design", TMT.

DATA STRUCTURES

II B. Tech. – III Semester (Code: 24CS303)

Lectures	2	Tutorial	1	Practical	0	Credits	3
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: Programming for Problem Solving (24CS204)

Course Objectives:

- Understand the role of Data structures in structuring and analysis procedure of an algorithm.
- Learn the concept of Stack, Queue and various Sorting techniques.
- Understand the concept of Binary Tree, Binary Search Tree, AVL tree, Heap Data Structures
- Learn the concept of Hashing and Graph traversals

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

CO1	Analyze the concepts of algorithm evolution and compute their time & space complexities. To elaborate various lists along with their operations.
CO2	Solve various real time problems using stack and queue data structures. Develop algorithms and programs for various sorting techniques.
CO3	Analyze the concepts of trees, binary trees, AVL trees and priority queues.
CO4	Analyze various hashing techniques and Graph traversals

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3	3	3	-	-	-	-	-	-	-	-	3	3	-
CO2	3	3	3	-	-	-	-	-	-	-	-	3	3	-
CO3	3	3	3	-	-	-	-	-	-	-	-	3	3	-
CO4	3	3	3	-	-	-	-	-	-	-	-	3	3	-

UNIT-I

Introduction: Algorithm, Importance of Algorithm Analysis, Complexity of an Algorithm, Asymptotic Analysis and Notations. Data Structure, Classification of Data Structures.
Lists: Abstract Data Types, The List ADT, Singly Linked List ADT, Doubly Linked List ADT, Circular Linked List ADT, Polynomial ADT: addition, multiplication operations.

UNIT-II

Stacks and Queues: The Stack ADT and its applications such as Infix to Postfix expression conversions, Evaluation of Postfix expressions. The Queue ADT, Queue Application-Radix sort.
Basic Sorting Techniques: Bubble sort, Selection sort, Insertion sort, Shell sort

UNIT-III

Trees: Preliminaries, Binary Trees, The Search Tree ADT, Binary Search Trees, Implementations, AVL Trees-Single Rotations, Double rotations, Implementations.
Priority Queues (Heaps): Model, Simple implementations, Binary Heap, Heap Sort.

UNIT-IV

Hashing: General Idea, Hash Function, Separate Chaining, Open Addressing. **Graphs:** Preliminaries, Representation of graphs, Graph traversals (DFS, BFS)	
Text Books :	Mark Allen Weiss, "Data Structures and Algorithm Analysis in C", Pearson Education, 2013, Second Edition, ISBN- 978-81-7758-358-8.
References :	1. Y.Langsam, M.J.Augeustein and A.M.Tenenbaum, "Data Structures Using C", Pearson Education Asia, 2006, Second Edition, ISBN- 81-203-1177-9. 2. Richard F.Gilberg, Behrouz A. Forouzan, "Data Structures – A Pseudocode Approach with C", Thomson Brooks / COLE, 1998, Second Edition, ISBN-978-0-534-39080-8 3. Aho, J.E. Hopcroft and J.D. Ullman, "Data Structures and Algorithms", Pearson Education Asia, 1983, 1st edition, ISBN- 978-0201000238.

OBJECT ORIENTED PROGRAMMING
II B. Tech. – III Semester (Code: 24CS304)

Lectures	2	Tutorial	1	Practical	0	Credits	3
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: Introduction to Programming (24CS104), Programming for Problem Solving(24CS204)

Course Objectives:

- Understand the advantages of Object Oriented Programming over Procedural-Oriented Programming, learn the basics of variables, operators, control statements, arrays, classes and objects.
- Understand, write and implement the following concepts: Inheritance, Exception Handling, Interfaces, Packages and Strings and StringBuffer.
- Understand and write programs on Collections, Multithreading and I/O.
- Understand and implement applications using JFX.

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

CO1	Demonstrate OOP concepts, its advantages over structured programming.
CO2	Develop and implement Inheritance, Exception Handling, Interfaces, Packages and Strings and StringBuffer.
CO3	Analyze Collections, Multithreading and I/O.
CO4	Create code using JavaFX.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3	2	3	-	-	-	-	-	-	-	-	3	3	2
CO2	3	2	3	-	-	-	-	-	-	-	-	3	3	2
CO3	3	2	3	-	-	-	-	-	-	-	-	3	3	2
CO4	3	2	3	-	2	-	-	-	-	-	-	3	3	2

UNIT-I

An Introduction to Java: History, Java Byte Code, Java Buzzwords
Data Types, Variables and Arrays
Operators
Control Statements
Introducing Classes
A Closer Look at Methods and Classes

UNIT-II

Inheritance
Exception Handling
Packages and Interfaces
Strings: String Constructors, Any 10 String class methods, StringBuffer class, Any 10 StringBuffer class methods.

UNIT-III
Type Wrappers, Autoboxing/Unboxing **Collections:** Collections Overview, Names of Collection Interfaces, Collection Classes: LinkedList, Array List **Multithreaded Programming:** Introduction, Life Cycle, Thread creation – Thread Class, Runnable Interface, Thread Priority, Synchronization, Interthread Communication. **I/O:** I/O Basics, Reading Console Input, Writing Console Output, The Print Writer class, Reading and Writing Files, Automatically Closing a File.
UNIT-IV
Introducing JavaFX GUI Programming: JavaFX Basic Concepts, A JavaFX Application Skeleton, Compiling and Running a JavaFX Program, The Application Thread, A Simple JavaFX Control: Label, Using Buttons and Events, Drawing Directly on a Canvas. **Exploring JavaFX Controls:** Using Image and ImageView, ToggleButton, RadioButton, CheckBox, ListView, ComboBox, TextField, ScrollPane, TreeView. **Introducing JavaFX Menus:** Menu Basics, An Overview of MenuBar, Menu, and MenuItem, MenuBar, Menu, MenuItem, Create a Main Menu, Add Mnemonics and Accelerators to Menu Items, Add Images to Menu Items, Use RadioMenuItem and CheckMenuItem, Create a Context Menu, Create a Toolbar, Put the Entire MenuDemo Program Together, Continuing Your Exploration of JavaFX

Text Books :	"Java The Complete Reference", 10th Edition, Herbert Schildt, TMH Publishing Company Ltd, New Delhi, 2017.
References :	1. "Big Java ", 4th Edition, Cay Horstman, John Wiley & Sons, 2009. 2. "Java How to Program (Early Objects)", H. M. Dietel and P. J. Dietel, 11th edition Pearson Education, 2018.

COMPUTER NETWORKS
II B.Tech – III Semester (Code: 24CS305)

Lectures	3	Tutorial	0	Practical	0	Credits	3
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: None

Course Objectives:

- Understand the basic concepts of data communication, layered model, protocols and OSI & TCP layers
- Understand the basic concepts of Data Link control, Network Layer Design Issues, Routing Algorithms & Congestion.
- Understand the basic concepts of Quality of service, Network Layer & Transport Layer
- Understand the basic concepts of TCP, UDP & Application Layer

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

CO1	Learn types of communications, topologies, OSI, TCP/IP protocol architectures along with error detection and correction mechanisms and also the working of data link layer
CO2	Understand and analyze data link control mechanisms, network layer design, routing algorithms, and congestion control techniques to ensure efficient and reliable network communication.
CO3	Know the transport layer issues, establishment of remote procedure calls and TCP segment header.
CO4	Learn the working of TCP and UDP and different application layer issues.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	1	2	2	-	1	-	1	-	2	3	-	1	2	1
CO2	1	-	2	-	1	1	-	1	-	-	1	1	1	2
CO3	-	-	2	1	1	-	-	-	1	1	1	1	2	1
CO4	1	2	2	2	1	-	-	-	1	1		1	2	1

UNIT-I

Data Communications & Networking Overview: A Communications Model, Data Communications, Data Communication Networking.

Protocol Architecture: The Need for Protocol Architecture, A Simple Protocol Architecture, OSI, The TCP/IP Protocol Architecture.

Digital Data Communication Techniques: Asynchronous & Synchronous Transmission, Types of Errors, Error Detection, Error Correction.

UNIT-II

DATA Link Control: Flow Control, Error Control, HDLC.

Network Layer: Network Layer Design Issues: Store-and-Forward Packet Switching, Services Provided to the Transport Layer, Implementation of Connectionless Service, Implementation of Connection-Oriented Service, Comparison of Virtual-Circuit & Datagram Subnets.

Routing Algorithms: The Optimality Principle, Shortest Path Routing, Flooding, Distance Vector Routing, Link State Routing, Hierarchical Routing.

Congestion Control Algorithms: General Principles of Congestion Control, Congestion Prevention Policies, Congestion Control in Virtual-Circuit Subnets, Congestion Control in Datagram Subnets, Load Shedding, Jitter Control.

UNIT-III

Quality of Service: Requirements, Techniques for Achieving Good Quality of Service.

The Network Layer in the Internet: The IP Protocol, IP Addresses, Internet Control Protocols.

The Transport Layer, The Transport Service: Services Provided to the Upper Layers, Transport Service Primitives.

Elements of Transport Protocols: Addressing, Connection Establishment, Connection Release.

UNIT-IV

The Internet Transport Protocol (UDP): Introduction to UDP, Remote Procedure Call, The Real-Time Transport Protocol.

The Internet Transport Protocols (TCP): Introduction to TCP, The TCP Service Model, The TCP Protocol, The TCP Segment Header, TCP Connection Establishment, TCP Connection Release, Modeling TCP Connection Management, TCP Transmission Policy, TCP Congestion Control, TCP Timer Management.

Application Layer: The Domain Name System (DNS): The DNS Name Space, Resource Records, Name Servers.

Text Books :	1. William Stallings, "Data and Computer Communications", 8th Edition, Pearson Education. 2. Tanenbaum, "Computer Networks", 5th Edition, Pearson Education, 2011
References :	1. Wayne Tomasi, "Introduction to Data Communications and Networking", PHI. 2. Behrouz A.Forouzan, "Data Communications and Networking", Fourth edition, TMH 3. God Bole, "Data Communications & Networking", TMH. 4. Kurose & Ross, "COMPUTER NETWORKS– A Top-down approach featuring the Internet", Pearson Education, AlbertoLeon, Garciak. 5. Leon Gartia, Indra Widjaja, "Communication Networks Fundamental Concepts and Key Architectures", TMH. 6. Nader F.Mir, "Computer and Communication Networks", PHI.

PYTHON PROGRAMMING
(Skill Enhancement Course - I)
II B. Tech. – III Semester (Code: 24CSL301/SEC1)

Lectures	1	Tutorial	0	Practical	2	Credits	2
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: None

Course Objectives:

- Understand and write code using the basics of Python, Statements, Expressions, Conditional Executions, and Functions.
- Write code for Iteration, Strings, File I/O.
- Write code in creating, usage of Lists, Dictionaries, and Tuples.
- Understand the concepts of Object Orientation, Databases and write code implementing them.

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

CO1	Identify the basic python constructs with a view of using them in problem solving.
CO2	Explore the usability of functions and strings in modular programming
CO3	Apply lists, dictionaries, tuples and file operations to organize the data in real world problems.
CO4	Implement the problems in terms of real world objects using object oriented and database concepts.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3	3	3	-	-	-	2	-	2	-	3	3	3	3
CO2	3	3	3	-	-	-	2	-	2	-	3	3	3	3
CO3	3	3	3	-	-	-	2	-	2	-	3	3	3	3
CO4	3	3	3	-	-	-	2	-	2	-	3	3	3	3

Topic

Python: Introduction, Basic elements, Control structures.
Functions: function calls, built-in functions, type conversion functions, random numbers, math functions, adding new functions, definitions and uses, flow of execution, parameters and arguments, fruitful functions and void functions.
Strings: string is a sequence, getting the length of a string using len, traversal through a string with a loop, string slices, strings are immutable, looping and counting, the in operator, string comparison, string methods, parsing strings, format operator.
Lists: a list is a sequence, lists are mutable, traversing, operations, slices, methods, deleting elements, functions, strings, parsing lines, objects and values, aliasing, arguments.
Dictionaries: dictionary as a set of counters, dictionaries and files, looping and dictionaries, advanced text parsing.

Tuples: tuples are immutable, comparing tuples, tuple assignment, dictionaries and tuples, multiple assignment with dictionaries, the most common words, using tuples as keys in dictionaries, sequences.
Functional Programming: Join function, comprehension (list and dict), lambda, reduce, filter and map function.
Object-Oriented Programming: Classes and Objects, Classes and Functions, Classes and Methods, Inheritance
Files I/O: persistence, opening files, text files and lines, reading files, searching through a file, letting the user choose the file name, using try except and open, writing files.

LIST OF EXPERIMENTS

Basic elements, Control structures:

1. Write a Python program to input two numbers and display their sum, difference, product, and quotient of division. (Covers variables, expressions, operators, input/output).
2. Write a Python program to check if a number is positive, negative, or zero. (Covers if, elif, else).
3. Write a Python program to find the largest of three numbers using nested if. (Covers nested conditionals).
4. Write a Python program to print the Fibonacci sequence up to n terms using a loop. (Covers loops, updating variables).
5. Write a Python program to find the factorial of a number using a while loop. (Covers while loop and loop control).

Functions:

6. Write a Python program to define a function that checks whether a number is prime. (Covers user-defined functions, return values).
7. Write a Python program using built-in functions like len(), type(), int() and random.randint() to demonstrate their usage. (Covers built-in and standard library functions).

Strings:

8. Write a Python program to count the number of vowels, consonants, digits, and special characters in a string. (Covers string traversal, conditions, in operator).
9. Write a Python program to extract the domain name from an email address. (Covers slicing, find(), string methods)

Lists:

10. Write a Python program to remove duplicates from a list and sort it. (Covers list methods and operations).
11. Write a Python program to split a list every Nth element.
    Sample Input: list1 = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n']
    Expected Output: output_list = [['a', 'd', 'g', 'j', 'm'], ['b', 'e', 'h', 'k', 'n'], ['c', 'f', 'i', 'l']]
12. Write a Python program to compute the similarity between two lists.
    Sample Input: Colors1 = ["red", "orange", "green", "blue", "white"]
    Colors2 = ["black", "yellow", "green", "blue"]
    Expected Output: Colors1-Colors2 = ['white', 'orange', 'red']

Colors2-Colors1 = ['black', 'yellow']

13. Write a Python program to find the second largest number in a list. (Covers list traversal and conditions).

Dictionaries:

14. Write a Python program to count the frequency of each word in a user-provided string using a dictionary. (Covers dictionary creation and looping).
15. Write a Python program to combine two dictionaries by adding values for common keys.
    Sample Input: dict1 = {'a': 100, 'b': 200, 'c':300}
    dict2 = {'a': 300, 'b': 200, 'd':400}
    Sample output: Combined_dict = {'a': 400, 'b': 400, 'd': 400, 'c': 300}

Tuples:

16. Write a Python program to replace last value of tuples in a list.
    Sample Input: list1 = [(10, 20, 40), (40, 50, 60), (70, 80, 90)]
    Expected Output: new_list = [(10, 20, 100), (40, 50, 100), (70, 80, 100)]
17. Write a Python program to sort a list of tuples based on the second element. (Covers tuples and sort).

Functional Programming:

18. Write a Python program to join the list elements using join function.
    Sample Input: ['Department', 'of', 'Computer', 'Science', 'and', 'Engineering']
    Expected Output1: DepartmentofComputerScienceandEngineering
    Expected Output2: Department of Computer Science and Engineering
    Expected Output3: Department-of-Computer-Science-and-Engineering
    Expected Output4: Department@-of@-Computer@-Science@-and@-Engineering
    Expected Output5: Department+of+Computer+Science+and+Engineering
19. Write a Python single line script to create a list of squares of even numbers in a range of numbers using comprehension.
    Sample Input: [1, 2, 3, 4, 5, 6]
    Expected Output: [4, 16, 36]
20. Write a Python Program to create a lambda function to add two numbers.
21. Write a Python single line script to filter odd numbers from a list using lambda and filter functions.
    Sample Input: [1, 2, 3, 4, 5, 6]
    Expected Output: [1, 3, 5]
22. Write a python single line script to convert a list of numbers to strings and join them with any special character.
    Sample Input: [1, 2, 3, 4, 5, 6]
    Expected Output1: 1+2+3+4+5+6
    Expected Output2: “1” “2” “3” “4” “5” “6”
    Expected Output3: 123456
    Expected Output4: 1-2-3-4-5-6

Expected Output5: 1^2^3^4^5^6

23. Write a Python single line script to double the list values using map and anonymous function.
    Sample Input: [1, 2, 3, 4, 5, 6]
    Expected Output: [2, 4, 6, 8, 10, 12]
24. Write a Python single line program to get the transpose of a 2D matrix using nested list comprehension.
    Sample Input: matrix = [[1, 2, 3],
    [4, 5, 6]]
    Expected Output: [[1, 4], [2, 5], [3, 6]]
25. Write a Python single line program to sort a dictionary by values using lambda function.
    Sample Input: {'S1': 88, 'Stu2': 72, 'Stu3': 91, 'S4': 90, 'S5': 79}
    Expected Output: {'Stu2': 72, 'S5': 79, 'S1': 88, 'S4': 90, 'Stu3': 91}
26. Write a Python single line script to assign grades based on marks using conditional logic in list. Where student_grade = 'A' if student_marks >= 90, student_grade = 'B' if student_marks >= 75, student_grade = 'C' if student_marks >= 50, else student_grade = 'F'.
    Sample Input: student_marks = [95, 67, 80, 45, 76]
    Sample Output: student_grades = ['A', 'C', 'B', 'F', 'B']
27. Write a Python single line script to join dictionary keys into a single special character separated string using join function.
    Sample Input: {'name': 'John', 'age': 20, 'grade': 'A'}
    Sample Output: name, age, grade
28. Write a Python single line script to sort a list of tuples using lambda on the second element of the tuple.
    Sample Input: [('apple', 3), ('banana', 1), ('cherry', 2)]
    Sample Output: Sorted Tuples: [('banana', 1), ('cherry', 2), ('apple', 3)]
29. Write a Python program to filter prime numbers using a helper function (like: is_prime) and filter with lambda.
30. Write a Python single line script to flatten a nested list of two dimension into a single dimension list using list comprehension.
    Sample Input: nested_list = [[1, 2], [3, 4], [5, 6], [7], [8, 9, 10]]
    Sample Output: flat_list = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
31. Write a python single line script for the following using map() function
    a. Given a list of numbers, return the list of squares
    Sample Input: numbers = [1, 2, 3, 4, 5]
    Sample Output: squares = [1, 4, 9, 16, 25]
    b. Convert a list of string numbers to actual integers
    Sample Input: string_numbers = ['1', '2', '3', '4', '5']
    Sample Output: numeric_numbers = [1, 2, 3, 4, 5]

c. Convert Celsius to Fahrenheit (Formula F = (C * 9/5) + 32
   Sample Input: Celsius = [0, 20, 37, 100]
   Sample Output: Farhenheit = [32.0, 68.0, 98.6, 212.0]
d. Add two lists element-wise
   Sample Input: list1 = [1, 2, 3, 4, 5, 6], list2 = [6, 7, 8, 1, 2, 9]
   Sample Output: combined_list = [7, 9, 11, 5, 7, 15]
e. Capitalize list of Names
   Sample Input: names = ['computer', 'science', 'and', 'engineering']
   Sample Output: capitalized_names = ['Computer', 'Science', 'And', 'Engineering']
f. Extract first letter of each word
   Sample Input: words = ['apple', 'banana', 'cherry']
   Sample Output: words_first_letters = ['a', 'b', 'c']
g. Define a custom function that return the factorial of a number and use map() to apply it to a list of numbers.
   Sample Input: numbers = [3, 4, 5]
   Sample Output: numbers_factorial = [6, 24, 120]

Object-Oriented Programming:

32. Write a Python program to define a class Student with attributes name and marks, and a method to display them. (Covers class, object, methods).
33. Write a Python program to demonstrate inheritance with classes Person and Employee. (Covers inheritance and method overriding).

Files I/O:

34. Write a Python program to read a text file and count the number of lines, words, and characters. (Covers file reading and parsing)
35. Write a Python program to write a list of strings into a text file. (Covers file writing).
36. Write a Python program to find the longest word in each line of given file.

Text Books :	1. A Python Book: Beginning Python, Advanced Python, and Python Exercises, Dave Kuhlman, Open Source MIT License. 2. Python for Data Analysis, Wes McKinney, O' Reilly.
References :	1. Python Data Science Handbook-Essential Tools for Working with Data Science from Scratch, Joel Grus, O'Reilly.

DATA STRUCTURES LAB
II B. Tech. – III Semester (Code: 24CSL302)

Lectures	0	Tutorial	0	Practical	3	Credits	1.5
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: Programming for Problem Solving Lab (24CSL204)

Course Objectives:

- Understand and program basic data structures like arrays and linked lists with their applications.
- Understand and Program data structures like stacks and queues with their applications. Understand and implement sorting algorithms.
- Understand and program on trees, binary trees, binary search trees, AVL trees, expression trees and their traversal methods and priority queues
- Basic knowledge of graphs representations and traversing methods. Understand the concept of hashing and their mechanisms.

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

CO1	Apply programming techniques using pointers, DMA and structures to implement SLL and DLL.
CO2	Design and implement ADTs of stack, queue and its applications and implement different sorting techniques.
CO3	Analyze and implement BST, AVL tree and priority queue.
CO4	Analyze and implement graph traversals (DFS and BFS)

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3	3	3	-	3	-	2	-	2	-	3	3	3	3
CO2	3	3	3	-	3	-	2	-	2	-	3	3	3	3
CO3	3	3	3	-	3	-	2	-	2	-	3	3	3	3
CO4	3	3	3	-	3	-	2	-	2	-	3	3	3	3

LIST OF EXPERIMENTS

1. Write a program to perform the following operations on an ArrayList.
   a). Creation, b). Insertion, c). Deletion, d). Search, e). Display
2. Write a program to perform the following operations on a Singly Linked List.
   a). Creation, b). Insertion, c). Deletion, d). Search, e). Display
3. Write a program to perform the following operations on a Circular Single Linked List.
   a). Creation, b). Insertion, c). Deletion, d). Search, e). Display
4. Write a program to perform addition and multiplication of two polynomials using a Single Linked List.
5. Write a program to implement stack operations using a linked list.
6. Write a program to convert a given infix expression to postfix notation and then evaluate the resulting postfix expression using a stack.

7. Write a program to implement queue operations using a linked list.
8. Write a program that performs Radix sort on a given set of elements using a queue.
9. Write a program to sort the array elements using the following techniques.
   a). Bubble Sort, b). Selection Sort, c). Insertion Sort, d). Shell Sort.
10. Write a program to perform Binary Search tree operations.
    a). Creation, b). Insertion, c). Deletion, d). Search, e). Traversals
11. Write a program to implement an AVL tree that interactively allows
    a). Insertion, b). Find_min, c). Find_max.
12. Write a program to sort the array elements using Heap Sort.
13. Write a program to perform the following hashing techniques.
    a). Linear Probing b). Quadratic Probing.
14. Write a program to implement the following graph traversal methods.
    a). DFS b). BFS.

Text Books :	Mark Allen Weiss, "Data Structures and Algorithm Analysis in C", Second Edition, Pearson Education
References :	1. Y.Langsam, M.J.Augeustein and A.M.Tenenbaum, "Data Structures Using C", Pearson Education Asia, 2004. 2. Richard F.Gilberg, Behrouz A. Forouzan, "Data Structures – A Pseudocode Approach with C", Thomson Brooks / COLE, 1998.

OBJECT ORIENTED PROGRAMMING LAB
II B.Tech – III Semester (Code: 20CSL303)

Lectures	0	Tutorial	0	Practical	3	Credits	1.5
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: Introduction to Programming (24CSL104), Programming for Problem Solving(24CSL204)

Course Objectives:

- Understand the advantages of Object Oriented Programming over Procedural-Oriented Programming, learn the basics of variables, operators, control statements, arrays, classes and objects.
- Understand, write and implement the following concepts: Inheritance, Exception Handling, Interfaces, Packages and Strings and StringBuffer.
- Understand and write programs on Collections, Multithreading and I/O.
- Understand and implement applications using Events and JFX.

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

CO1	Demonstrate OOP concepts, its advantages over structured programming.
CO2	Develop and implement Inheritance, polymorphism.
CO3	Analyze Exception Handling, Multithreading, I/O.
CO4	Create code for Event Handling, JavaFX.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3	2	3	-	-	-	-	-	-	-	-	3	3	2
CO2	3	2	3	-	-	-	-	-	-	-	-	3	3	2
CO3	3	2	3	-	-	-	-	-	-	-	-	3	3	2
CO4	3	2	3	-	2	-	-	-	-	-	-	3	3	2

LIST OF EXPERIMENTS

1. Write a Java program to declare, initialize and access the elements of single-dimensional Arrays and multi-dimensional Arrays. Display the array elements using loops.
2. Develop a Java program to demonstrate the use of static variables, static methods, and static blocks.
3. Create a Java program that demonstrates method overloading (Compile-time Polymorphism) and method overriding (Runtime Polymorphism) using inheritance.
4. Write a Java program to illustrate the concept of multiple inheritance through the implementation of multiple interfaces.
5. Develop a Java program that demonstrates the use of all exception handling keywords: *try, catch, throw, throw*s, and *finally*.
6. Write a Java program to create and handle one or more user-defined exceptions with meaningful messages.

7. Design a Java program to demonstrate the creation and use of packages. Include at least two classes from different packages.
8. Implement a Java program to demonstrate inter-thread communication using *wait(), notify()*, and *notifyAll()* methods.
9. Write a Java program to copy the contents of one file to another using ***FileInputStream*** and ***FileOutputStream***. Demonstrate the use of try-with-resources for automatic resource management.
10. Design a JavaFX application that enables users to draw directly on a Canvas using mouse events.
11. Create a JavaFX application to demonstrate handling action events with appropriate UI components.
12. Develop a JavaFX application to demonstrate the usage of list-based controls with event handling.
13. Design a JavaFX application that demonstrates the creation and use of menus with event handling.

Text Books :	"Java The Complete Reference", 10th Edition, Herbert Schildt, TMH Publishing Company Ltd, New Delhi, 2017.
References :	1. "Big Java ", 4th Edition, Cay Horstman, John Wiley & Sons, 2009. 2. "Java How to Program (Early Objects)", H. M. Dietel and P. J. Dietel, 11th edition Pearson Education, 2018.

HEALTH AND WELLNESS, YOGA AND SPORTS
II B. Tech. – III Semester (Code: 24CSL304)

Lectures	0	Tutorial	0	Practical	1	Credits	0.5
Continuous Internal Evaluation			100	Semester End Examination			--

Pre-Requisite: None.

Course Objectives:

- Make the students maintain their mental and physical wellness by balancing emotions in their life.
- Mainly enhances the essential traits required for the development of the personality.

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

CO1	Outline the importance of yoga and sports for Physical fitness and sound health. (L2)
CO2	Make use of various activities that help to enhance their health. (L3)
CO3	Develop Positive Personality for individual and group work. (L3)
CO4	Categorize the health-related fitness components and analyze the current personal fitness levels. (L4)

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	-	-	-	-	-	-	-	-	2	-	3	-	-	-
CO2	-	-	-	-	-	-	-	-	2	-	3	-	-	-
CO3	-	-	-	-	-	-	-	-	2	-	3	-	-	-
CO4	-	-	-	-	-	-	-	-	2	-	3	-	-	-

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	
i) Participation in one major game and one individual sport viz., Athletics, Volleyball, Basketball, Handball, Football, Badminton, Kabaddi, Kho-kho, Table Tennis, Cricket etc., Practicing general and specific warm up, aerobics. ii) Practicing cardiorespiratory fitness, treadmill, run test, 9 min walk, skipping and running.	
Text 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.
References :	1. Wiseman, John Lofty, SAS Survival Handbook: The Ultimate Guide to Surviving Anywhere Third Edition, William Morrow Paperbacks, 2014. 2. The Sports Rules Book/ Human Kinetics with Thomas Hanlon. -- 3rd ed. Human Kinetics, Inc.2014.

CONSTITUTION OF INDIA
II B.Tech – III Semester (Code: 24CS306/MC02)

Lectures	2	Tutorial	0	Practical	0	Credits	0
Continuous Internal Evaluation			40	Semester End Examination			--

Pre-Requisite: None.

Course Objectives:

- Provide basic information about fundamental law of the country.
- Educate the student about fundamental Rights and fundamental duties of citizens.
- Educate the students about Government organs, methods of functioning
- Motivate students to leave narrow selfish outlook and inculcate broad national, human outlook.

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

CO1	Understand the importance of the constitution in a Democratic Society
CO2	Understand the fundamental rights, duties of a citizen by discharging his duties to become a good citizen.
CO3	Remember about judicial supremacy and independence of judiciary and fight for his legitimate rights through court of law.
CO4	Applying the principles to participate in the democratic process of governance and in nation building activities.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs										PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	1	2	2	-	1	-	-	3	-	3	3	-	-
CO2	-	-	1	1	1	-	-	-	-	3	3	3	2
CO3	-	-	1	-	1	2	-	-	-	3	3	-	-
CO4	3	1	2	-	-	-	-	3	-	1	3	2	-

UNIT-I

1.1 Meaning of the constitutional law and constitutionalism.
1.2 Historical perceptive of the constitution of India
1.3 Salient features and characteristics of the constitution of India.
1.4 Preamble, union and its territory and citizenship.

UNIT-II

2.1. Fundamental rights principles.
2.2. Directive principles of state policy.
2.3. Fundamental duties.
2.4. The government of the union, the president, The Prime Minister, and the council of ministers, The parliament of India, The supreme court, the union judiciary

UNIT-III

3.1. The Machinery of Government in the states, The Governor, The Chief Minister and council of Ministers, The State legislature, High court, Judiciary in the states 3.2. Union territories. 3.3. The Federal System, division of powers between centre and states, legislative administration and financial relation. 3.4. Emergency Provisions, President Rule, National Emergency, Financial Emergency
UNIT-IV
4.1. Local self-Government, Panchayat Raj, Municipalities and municipal Corporation 4.2. Miscellaneous Provisions, the comptroller and Auditor general of India, The Public Service commission, Special Provisions relating to certain classes, Elections — Political parties. 4.3. Amendment of the Constitution. 4.4. Laws Relating to Women

Text Books:	1. Introduction to constitution of India, D.D.Basu, 24th Edition, Lexis Nexus, 2019. 2. The constitution of India by P.M.Bhakshi, 18th Edition, UniversaI law publishing, 2021.
Reference:	1. Constitutional Government in India - M V Pylee , Kindle Edition, Asia Publishing House, 2004. 2. Indian Government and Politics — D C Dasgupta,8th Edition,Vikas Publishing house, 2007. 3. The Oxford Hand Book of the Indian Constitution, Sujit Chowdary, Madhav Khosla Pratapabhem Mehla, oxford university press UK, 2016. 4. Laws Relating to Women, National Commission For Women, New Delhi, July 2020.

COMPUTER ORGANIZATION
II B. Tech. – IV Semester (Code: 24CS401)

Lectures	3	Tutorial	0	Practical	0	Credits	3
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: Digital Logic Design (24CS302)

Course Objectives:

- Represent the data, micro-operations, and hardware implementation of arithmetic, logic and shift unit.
- Know about the instruction codes and generation of control signals using hardwired and micro-programmed approaches.
- Learn about the different types of instructions and arithmetic operations.
- Understand the organization of the memory and I/O units.

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

CO1	Understand the basic structure of computer and analyzing the concepts of machine instructions.
CO2	Illustrate the various arithmetic operation and learn about basic processing time.
CO3	Review the basic computer instruction set and create flowcharts for the arithmetic operations.
CO4	Recognize the I/O and memory organizations.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3	-	2	-	-	-	-	-	-	-	-	3	-	-
CO2	3	-	2	-	-	-	-	-	-	-	-	3	-	-
CO3	2	-	2	-	-	-	-	-	-	-	-	3	-	-
CO4	2	-	2	-	-	-	-	-	-	-	-	3	-	-

UNIT-I

COMPUTER DATA REPRESENTATION: Introduction to Computer Organization, Data Types, Complements, Fixed-Point Representation, Floating-Point Representation.

REGISTER TRANSFER LANGUAGE AND MICROOPERATIONS: Register Transfer Language, Register Transfer, Bus and Memory Transfers, Arithmetic Micro Operations, Logic micro operations, Shift Micro Operations, Arithmetic Logic Shift Unit.

UNIT-II

BASIC COMPUTER ORGANIZATION AND DESIGN: Instruction Codes, Computer Registers, Computer Instructions, Timing and Control, Instruction Cycle, Memory-Reference Instructions, Input-Output and Interrupt, Design of Accumulator Logic.

MICRO PROGRAMMED CONTROL: Control Memory, Address Sequencing, Microprogram Example, Design of Control Unit.

UNIT-III	
CENTRAL PROCESSING UNIT: General Register Organization, Stack Organization, Instruction Formats, Addressing Modes, Data Transfer and Manipulation, Program Control, Reduced Instruction Set Computer vs Complex Instruction Set Computers. **COMPUTER ARITHMETIC:** Addition and Subtraction, Multiplication Algorithms, Division Algorithms.	
UNIT-IV	
THE MEMORY SYSTEM: Memory Hierarchy, Main Memory, Auxiliary Memory, Associative Memory, Cache Memory. **INPUT-OUTPUT ORGANIZATION**: Peripheral Devices, Input-Output Interface, Modes of Transfer, Priority Interrupt, Direct Memory Access.	
Text Books :	Computer System Architecture, M. Morris Mano, 3rd Edition, Pearson/PHI
References :	1. Computer Organization, Carl Hamacher, Zvonks Vranesic, Safea Zaky, 5th Edition, McGraw Hill. 2. Computer Organization and Architecture, William Stallings, Sixth Edition, Pearson/PHI.

AUTOMATA THEORY & FORMAL LANGUAGES
II B. Tech. IV Semester (Code: 24CS402)

Lectures	2	Tutorial	1	Practical	0	Credits	3
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: Discrete Mathematics (24CS205)

Course Objectives:

- Understand the theory of automata and formal languages. Construct finite automata, and conversion between DFA and NFA.
- Demonstrate the connection between regular expressions, languages, and finite automata.
- Demonstrate the connection between pushdown automata and context-free languages and context-free grammars.
- Construct Turing machines for a given task. Understand the undecidability problems of the Turing Machine and the Post-Correspondence Problem (PCP).

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

CO1	Comprehend automata and their uses. Create finite automata and switch between implementations that are deterministic and nondeterministic.
CO2	Transform finite automata into regular expressions and the other way around. Make a minimal DFA.
CO3	Build push-down automata for several context-free languages. Explain how PDA and context-free grammars are related.
CO4	Design Turing machines for different languages. Learn about TM and post correspondence problems that are undecidable.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3	2	2	-	-	-	-	-	-	-	-	2	2	-
CO2	2	2	2	-	-	-	-	-	-	-	-	2	2	-
CO3	3	3	3	-	-	-	-	-	-	-	-	2	2	-
CO4	3	3	3	-	-	-	-	-	-	-	-	2	2	-

UNIT-I

Automata: Why Study Automata Theory? The central concepts of automata theory - alphabets, strings, languages, and problems.

Finite Automata: An Informal picture of finite automata, Deterministic finite automata (DFA) - Definition of DFA, DFA processing strings, notations for DFA, extended transition function, the language of DFA, Non deterministic finite automata (NFA) – Definition of NFA, extended transition function, the language of NFA, equivalence of DFA and NFA.

Automata with ϵ transitions: Use of ϵ-transitions, notation for a ϵ-NFA, epsilon closures, extended transitions and languages, eliminating ϵ-transitions.

UNIT-II
Regular Expressions and Languages: Regular expressions, finite automata and regular expressions, Algebraic laws of regular expressions. **Properties of Regular Languages:** Proving languages are not regular – Pumping lemma for regular languages, Applications of the pumping lemma, Closure Properties of Regular Languages, Equivalence and minimization of automata – Minimization of DFA.
UNIT-III
(Construction-based treatment & proofs are excluded) **Context Free Grammars:** Context Free Grammars, Parse Trees, and Ambiguous Grammars. **Pushdown Automata:** Definition of the Pushdown automata, the languages of PDA, Equivalences of PDA's and CFGs. **Context-free languages:** Normal forms for context-free grammars, the pumping lemma for context-free languages.
UNIT-IV
Properties of Context-free Languages: Closure properties for context-free languages, Decision properties for CFLs. **Introduction to Turing Machines:** The Turing Machine, Programming techniques for Turing machines. **Undecidability:** A language that is not recursively enumerable, an undecidable problem that is RE, Undecidability problems about TM, Post's Correspondence problem.

Text Books :	John E. Hopcroft, Rajeev Motwani, & Jeffrey D. Ullman, "Introduction to Automata Theory, Languages and Computations", Pearson Education, 2008, Third Edition, ISBN: 978-8131720479.
References :	1. KLP Mishra & N. Chandrasekharan, "Theory of Computer Science: Automata, Languages and Computation", PHI, 2006, Third Edition, ISBN: 978-8120329683. 2. H.R. Lewis, C.H. Papadimitriou, "Elements of the Theory of Computation", Pearson Education, 2015, Second Edition, ISBN: 978-93-325-4989-0.

WEB TECHNOLOGIES
II B.Tech – IV Semester (Code: 24CS403)

Lectures	3	Tutorial	0	Practical	0	Credits	3
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: None.

Course Objectives:

- Know skills in creating standards-compliant web pages having semantic elements, Formatting, arranging text, Multimedia, Forms.
- Apply CSS3 and DHTML features in creating dynamic and interactive web pages.
- Know the basics of JavaScript Objects, DOM and ES6 features.
- Understand the React.js front-end framework basics.

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

CO1	Develop basic HTML5 web applications using standard and semantic elements, formatting, Arranging Text, Multimedia and Forms.
CO2	Apply CSS3 styling concepts and basic JavaScript Functions, arrays and Events to create dynamic and interactive web pages
CO3	Build dynamic web pages using JavaScript Objects, DOM, and ES6 features.
CO4	Develop React applications using Props, State, handling Events, Forms and React Router.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3	2	3	1	2	1	-	-	-	-	-	3	2	-
CO2	3	3	3	2	3	1	-	-	-	-	-	3	3	-
CO3	3	3	3	2	3	1	-	-	-	-	-	3	3	-
CO4	2	2	2	2	3	2	-	-	-	-	-	3	3	-

Unit-I

HTML 5: Understanding elements: ROOT, Metadata, Heading, Flow, Basic HTML data types, Character Entities.
Formatting Text: Physical style elements, Logical style elements.
Arranging Text: Preformatted Text, DIV, SPAN; Exploring the hyperlink, Structuring URL; Creating Tables.
Working with Images and Colors: Image Formats, exploring Colors; Working with Forms.

Unit-II

CSS3: Overview of Styles, Exploring the Box model, Font and Text styles. Displaying, Positioning and Floating an Element, List Styles and Table Layouts.
Dynamic HTML: Overview of JavaScript**:** JavaScript Functions, Arrays and Events.

Unit-III

JavaScript Objects: Understanding the Window Object, working with the Document Object.

DOM (Document Object Model): Understanding DOM nodes, Node interface, Document interface and Element interface. **ES6 Features:** Arrow functions, let and const, template literals, spread and rest.	
Unit-IV	
Introduction to React: React Benefits, Disadvantages, Basic React App**,** JSX, Components, Props, State, Functional Components and Class Components, Handling Events in React, Forms in React, **Project**: Menu Component. **Routing in React:** React Router Basics	
Text Books :	1. HTML5 Black Book**:** Covers CSS3, JavaScript, XML, XHTML, Ajax, PHP and Jquery, Kogent Learning Solutions Inc., Wiley India 7 edition. 2011. ISBN: 9789350040959. 2. Learning React, Alex Banks & Eve Porcello, O'Reilly, 2nd Edition, 2020.
References :	1. Harvey M.Deitel and Paul J. Deitel, "Internet & World Wide Web How to Program", 4/e, Pearson Education. 2. Jason Cranford Teague, "Visual Quick Start Guide CSS, DHTML & AJAX", 4e, Pearson Education. 3. Tom Nerino Doli smith, "Java Script & AJAX for the web", Pearson Education 2007.

DATABASE MANAGEMENT SYSTEMS
II B. Tech. – IV Semester (Code: 24CS404)

Lectures	3	Tutorial	0	Practical	0	Credits	3
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: None

Course Objectives:

- Familiarize with fundamental concepts of database and various database architectures and Design relations for Relational databases using conceptual data modeling.
- Implement formal relational operations in relational algebra and SQL.
- Identify the Indexing types and normalization process for relational databases
- Use mechanisms for the development of multi user database applications.

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

CO1	Apply knowledge of database design methodology which give a good formal foundation in relational data model and understand and apply the principles of data modeling using ER Model.
CO2	Create relational algebra expressions, relational calculus, and SQL for queries and be familiar with relational database theory
CO3	Design database schema and identify and solve the redundancy problem in database tables using normalization.
CO4	Learn about transaction processing, concurrency management, and recovery methods.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3	3	3	3	-	-	-	-	-	-	-	3	3	2
CO2	3	3	3	3	3	-	-	-	-	-	-	3	3	2
CO3	3	3	3	3	-	-	-	-	-	-	-	3	3	2
CO4	3	3	3	3	-	-	-	-	-	-	-	3	3	2

UNIT-I

Databases and Database Users: Introduction - An Example, Characteristics of the Database Approach, Actors on the Scene, Workers behind the Scene, Advantages of Using the DBMS Approach.

Database System Concepts: Data Models, Schemas and Instances, Three-Schema Architecture and Data Independence, Database Languages and Interfaces, Classifications of Database management systems.

Data Modeling Using the Entity-Relationship (ER) Model: Using High-Level Conceptual Data Models for Database Design, An Example Database Application, Entity Types, Entity Sets, Attributes, and Keys - Relationship Types, Relationship Sets, Roles, and Structural Constraints,

Weak Entity Types, Refining the ER Design for the COMPANY Database - ER Diagrams, Naming Conventions, and Design Issues
UNIT-II
The Relational Algebra and Relational Calculus: Unary Relational Operations: SELECT and PROJECT, Relational Algebra Operations from Set Theory, Binary Relational Operations: JOIN and DIVISION, The Tuple Relational Calculus, The Domain Relational Calculus. **Schema Definition, Constraints, Queries, and Views**: SQL Data Definition and Data Types, Specifying Constraints in SQL, Schema Change Statements in SQL, Basic Queries in SQL, INSERT, DELETE, and UPDATE Statements in SQL, Views (Virtual Tables) in SQL
UNIT-III
Indexing Structures for Files: Types of Single-Level Ordered Indexes, Multilevel Indexes - Dynamic Multilevel Indexes Using B+-Trees. **Functional Dependencies and Normalization for Relational Databases:** Informal Design Guidelines for Relation Schemas, Functional Dependencies, Normal Forms Based on Primary Keys - General Definitions of Second and Third Normal Forms, Boyce-Codd Normal Form. **Relational Database Design Algorithms and Further Dependencies:** Properties of Relational Decompositions -Lossless Join Decomposition and Dependency Preserving Decomposition, Multi-valued Dependencies and Fourth Normal Form, Join Dependencies and Fifth Normal Form.
UNIT-IV
Transaction & ACID: Transaction, ACID properties, States of the transaction **Transaction Problems:** Lost update and dirty read problem, unrepeatable read and phantom problem, incorrect summary problem. **Schedules:** Number of schedules, types of schedules : Serial schedule, complete schedule, recoverable schedule, cascading aborts, cascade-less schedule, strict schedule **Serializability**: Conflict Serializability, View Serializability, comparison between conflict and view serializability **Concurrency Control Protocols:** locks, 2-phase locking protocol, strict 2PL, rigorous 2PL, conservative 2PL, example on strict 2PL. **Graph & Timestamp Protocols:** Graph-based protocol, timestamp ordering protocol, examples on timestamp ordering protocol, Thomas Write Rule.

Text Books :	Fundamentals of Database Systems, Ramez Elmasri and Navathe Pearson Education, 6thedition
References :	1. Introduction to Database Systems, C.J. Date Pearson Education 2. Database Management Systems, Raghu Rama krishnan, Johannes Gehrke, TATA McGraw Hill3rdEdition 3. Database System Concepts, Silberschatz, Korth, McGraw hill,5thedition

DESIGN AND ANALYSIS OF ALGORITHMS
II B. Tech. - IV Semester (Code: 24CS405)

Lectures	2	Tutorial	1	Practical	0	Credits	3
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: Data Structures (24CS303)

Course Objectives:

- Understand about designing and the effectiveness of an algorithm, and applying the Master Theorem to find the complexity.
- Strengthen the divide and conquer paradigms and know the optimal solution finding with the greedy method.
- Acquaintance with algorithm design strategies of Dynamic programming and easily know the significant graph algorithms and their analyses.
- Get the ability to backtrack and branch with bound values and NP problems.

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

CO1	Analyze algorithms' performance using various strategies and apply the Master theorem to estimate the complexity of divide-and-conquer algorithms.
CO2	Apply the divide-and-conquer and greedy techniques to solve problems and perform complexity analysis.
CO3	Articulate on graph problems and identify the applicability of the dynamic programming paradigm for designing solutions to problems.
CO4	Utilize the backtracking and branch and bound algorithms to find every potential solution to combinatorial and optimization issues. In addition, classify the P and NP complicated problems.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3	3	3	-	-	-	-	-	-	-	3	3	3	-
CO2	3	3	3	-	-	-	-	-	-	-	3	3	3	-
CO3	3	3	3	-	-	-	-	-	-	-	3	3	3	-
CO4	3	3	3	-	-	-	-	-	-	-	3	3	3	-

UNIT-I

Introduction: Algorithm, Pseudo code for expressing algorithms, Performance Analysis- Space complexity, Time complexity, Asymptotic Notation – Big O notation, Big Omega notation, Theta notation.
Master Theorem: Introduction, Generic Form- Case1, Case2, Case3, Inadmissible equations, Application to standard algorithms.

UNIT-II

Divide and conquer: General method, applications - Quicksort, Merge sort, Stassen's matrix multiplication.

Greedy method: General method, applications-Job sequencing with deadlines, Fractional knapsack problem, Minimum cost spanning trees – Prim's Algorithm, Kruskal's Algorithm, Single source shortest paths – Dijkstra Algorithm.
UNIT-III
Dynamic Programming: General method, applications - Multi-stage graphs using Forward & Backward approach, 0/1 knapsack problem, Reliability design, The Travelling salesperson problem, Longest common subsequence algorithm. **Graph Applications**: Connected components, Bi-Connected Components, Strongly Connected Components.
UNIT-IV
Backtracking: General method, applications - The 8-queens problem, sum of subsets problem. **Branch and Bound**: General method, applications- LC Branch and Bound solution for 0/1 knapsack problem. **NP-Hard and NP-Complete problems:** Basic concepts - non-deterministic algorithms, NP-Hard and NP Complete classes, Cook's theorem.

Text Books :	E. Horowitz, S.Sahniand S. Rajasekaran, "Fundamentals of Computer Algorithms", Galgotia Publication.
References :	1. T. H. Cormen, Leiserson, Rivest, and Stein, "Introduction to Computer Algorithms" The MITPress. 2. Sara Basse, A.V. Gelder, "Computer Algorithms: Introduction to Design & Analysis, 3e ", Pearson Education.

COMPETITIVE PROGRAMMING SKILLS
(Skill Enhancement Course – 2)
II B. Tech. – IV Semester (Code: 24CSL401/SEC2)

Lectures	1	Tutorial	0	Practical	2	Credits	2
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: Programming for problem solving (24CS204), Data Structures (24CS302)

Course Objectives:

- To provide practical exposure to all algorithms and understand the importance of algorithms and their complexities.
- To implement various divide and conquer and Greedy method techniques examples.
- To implement various Dynamic Programming-based examples.
- To implement various backtracking and branch-and-bound examples.

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

CO1	Understand and implement an algorithm to solve problems by an iterative approach.
CO2	Understand and implement an algorithm to solve problems by divide and conquer, as well as the greedy algorithm approach.
CO3	Understand and analyze algorithms to solve problems using Dynamic programming.
CO4	Understand and analyze the algorithm to solve problems by backtracking and the branch and bound approach.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	2	3	3	3	3	-	-	-	-	-	-	-	-	2
CO2	2	3	3	3	3	-	-	-	-	-	-	-	-	3
CO3	2	3	3	2	2	-	-	-	-	-	-	-	-	3
CO4	2	3	3	2	2	-	-	-	-	-	-	-	-	2

LIST OF EXPERIMENTS

1. Write a program to implement Binary Search and count the number of comparisons made during the search to analyze its time complexity.
2. Write a program to implement the Merge Sort algorithm to sort a list of elements using the divide and conquer strategy.
3. Write a program to implement the Quick Sort algorithm to sort a list of elements using the divide-and-conquer strategy.
4. Write a program for matrix multiplication using Strassen's algorithm based on the divide and conquer approach.
5. Implement a program to solve the Fractional Knapsack problem using the greedy method.
6. Implement a program to find a graph's Minimum Cost Spanning Tree (MST) using Prim's algorithm based on the greedy strategy.

7. Implement a program to find a graph's Minimum Cost Spanning Tree (MST) using Kruskal's algorithm based on the greedy strategy. 8. Implement a program to find the shortest paths from a source vertex to all other vertices in a graph using Dijkstra's algorithm with the greedy method. 9. Write a program to solve the Traveling Salesman Problem using dynamic programming. 10. Write a program to solve the 0/1 Knapsack problem using dynamic programming and display the solution vector. 11. Using dynamic programming, write a program to compute the Longest Common Subsequence (LCS) of two sequences. 12. Write a program to determine the shortest path in a multi-stage graph using both forward and backward approaches based on dynamic programming. 13. Write a program to solve the 8-Queens problem using backtracking.

Text Books :	E. Horowitz, S.Sahniand S. Rajasekaran, "Fundamentals of Computer Algorithms", GalgotiaPublication.
References :	1. T. H. Cormen, Leiserson, Rivestand Stein, "Introduction of Computer Algorithm", PHI. 2. SaraBasse, A.V.Gelder, "Computer Algorithms", Addison Wesley.

DESIGN THINKING AND INNOVATION
II B. Tech. - IV Semester (24CSL402)

Lectures	1	Tutorial	0	Practical	2	Credits	2
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: None

Course Objectives:

- Provide an overview of design thinking.
- Engage students to allow them to integrate these components into their own courses.
- Nurture their skills to contribute for solving community-based problems
- Provide a framework to work in teams to solve problems

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

CO1	Describe the components of design thinking
CO2	List attributes of expert designers
CO3	Employ prototyping into their design experiences
CO4	Discuss the importance of users in the design process in proposing solutions

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3	3	2	-	-	-	-	-	-	-	2	-	-	-
CO2	3	3	3	-	-	-	-	-	-	-	2	-	-	-
CO3	3	3	3	-	-	-	-	-	-	-	2	-	-	-
CO4	2	2	2	-	-	-	-	-	-	-	2	-	-	-

UNIT-I

Introduction to Design Thinking: Characteristics of design thinking, Expert and novice design characteristics, Opportunities for student to learn skills needed for the development of expertise.

Case study for design thinking success: Creating a culture of design learning, Gathering information from users, Rapid prototyping.

UNIT-II

Users and community partners: Understanding users, identifying users, creating tools for understanding users

Requirements and specifications: Defining specifications, State of the art comparisons, Testing requirements

UNIT-III

Prototyping: Prototypes for technology, Prototypes for communication

Ideation and concept generation: Brainstorming, Concept generation, Functional decomposition.

Testing and design to prevent failures: Testing of designs, Design for Failure Modes and Effects Analysis (DFMEA), Delivery to users

UNIT-IV

Teaming concepts in design: Managing student teams, Organizing teams, Assessing teams, Mentoring and advising teams

Closure and summary: Reviewing design cycles and concepts, Putting it into action

Practical Exercises

1. Design Thinking Mind Map Exercise
2. Functional Decomposition – eg: Mechanical Pencil
3. Thirty Circle Exercise – IDEO thinking
4. Shopping Cart Design Exercise - IDEO
5. Brain storm and innovate bicycle for better urban mobility
6, Develop an Empathy Map – case study
7. The Lean Canvas Model – business model
8. Technology Readiness Assessment – to evaluate project risks
9. Prototyping Exercise – using paper/thermos coal/cardboard/recyclable material
10. Case study of any Jugaad – economic rural solution
11, Design thinking using sprint base software
12. Presentation on Idea Proposal – along with a Prototype (Domain Specific)
13. Risk Analysis – case study – DFMEA Analysis
14. Test Plan Preparation – case study
15. Interaction with Technical community- viable product delivery

Note: *Minimum of 10 Exercises have to be completed and documented.* Out of these 5 exercises can be tuned to branch specific if necessary.

Text Books :	1. IdrisMootee, "Design Thinking for Strategic Innovation", John Wiley & Sons (2013). 2. "Change by design", Tim Brown, Harper Collins, 2009 3. "Design Thinking- The Guide Book" – Facilitated by the Royal Civil service Commission, Bhutan 4. Engineering design by George E Dieter
References :	1. 101 Design Methods: A Structured Approach for Driving Innovation in Your Organization by Vijay Kumar 2. Human-Centered Design Toolkit: An Open-Source Toolkit To Inspire New Solutions in the Developing World by IDEO https://www.interaction-desiqn.ora/literature/topics/desiqn-th/nkinq https://www.interaction-desiqn.prq/literature/article/how-tq-

WEB TECHNOLOGIES LAB

II B.Tech – IV Semester (Code: 24CSL403)

Lectures	0	Tutorial	0	Practical	3	Credits	1.5
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: None.

Course Objectives:

- Know skills in creating standards-compliant web pages having semantic elements, Multimedia, Forms.
- Apply CSS3 and DHTML features in creating a dynamic and interactive web page design.
- Know the basics of JavaScript Objects, DOM and ES6 features.
- Understand the React.js front-end framework basics.

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

CO1	Develop basic HTML5 web applications using semantic elements, Multimedia and Forms.
CO2	Create Dynamic and Interactive web pages using CSS3 and DHTML.
CO3	Build dynamic web pages using JavaScript Objects, DOM, and ES6 features.
CO4	Develop React applications using Props, State and React Router.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3	2	2	-	3	-	2	-	2	-	3	3	-	-
CO2	3	3	3	-	3	-	2	-	2	-	3	3	-	-
CO3	3	3	3	-	3	-	2	-	2	-	3	3	-	-
CO4	3	3	3	-	3	-	2	-	2	-	3	3	-	-

LIST OF EXPERIMENTS

1. Design an HTML5 webpage using fundamental elements, organizing text, Links, URLs and Tables.
2. Design an HTML5 webpage using Images, Colors and Forms.
3. Demonstrate applying different CSS styling techniques:- CSS styles, Box model, Font, Text, Displaying, Positioning & Floating an element.
4. Develop a JavaScript application covering Functions and Arrays.
5. Develop a JavaScript application handling Events.
6. Create a Student Registration Form using HTML, CSS and JavaScript with features Local Storage of registration details with validation and display in tabular form.
7. Develop a JavaScript application using the Window Object and Document Object.
8. Demonstrate the Document Object Model for an HTML document.

9. Write a program to demonstrate ES6 features. 10. Set up a React App using create-react-app. Create a component-based UI for a simple profile 11. page. 12. Develop a React App to display a Counter and a dynamic list of tasks using Props and State. 13. Develop a React App using Forms and handle Events.

Text Books :	1. HTML 5 Black Book: Covers CSS3, JavaScript, XML, XHTML, Ajax, PHP and Jquery, Kogent Learning Solutions Inc., Wiley India 7 edition. 2011. ISBN: 9789350040959. 2. HTML and CSS: Design and Build Websites, Jon Duckett, Wiley, 2011. 3. Learning React, Alex Banks & Eve Porcello, O'Reilly, 2nd Edition, 2020.
References :	1. Harvey M. Deitel and Paul J. Deitel, "Internet & World Wide Web How to Program", 4/e, Pearson Education. 2. Jason Cranford Teague, "Visual Quick Start Guide CSS DHTML & AJAX", 4e, Pearson Education. 3. Tom Nerino Doli smith, "Java Script & AJAX for the web", Pearson Education2007. https://www.w3schools.com/ https://react.dev/

RDBMS LAB

II B.Tech – IV Semester (Code: 24CSL404)

Lectures	0	Tutorial	0	Practical	3	Credits	1.5
Continuous Internal Evaluation			40	Semester End Examination			60

Pre-Requisite: None.

Course Objectives:

- Analyze the students on database languages.
- Interpret the Knowledge on database design.
- Determine the knowledge on key constraints and Normalization.
- Determine the knowledge on procedures and functions.

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

CO1	Implement DDL, DML, DCL Commands using SQL.
CO2	Implement Advanced queries using SQL.
CO3	Apply key constraints to get a normalized database.
CO4	Implement procedures and functions using PL/SQL

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	3	3	3	3	3	-	2	-	2	-	3	3	3	3
CO2	3	3	3	3	3	-	2	-	2	-	3	3	3	3
CO3	3	3	3	3	3	-	2	-	2	-	3	3	3	3
CO4	3	3	3	3	3	-	2	-	2	-	3	3	3	3

LIST OF EXPERIMENTS

Experiment I: DDL, DML, DCL

1.Create a table EMPLOYEE with following schema:

(Emp_no, E_name, E_address, E_ph_no, Dept_no, Dept_name, Job_id , Salary)

a. Add a new column; HIREDATE to the existing relation.

b. Change the datatype of JOB_ID from char to varchar2.

c. Change the name of column/field Emp_no to E_no.

d. delete E_ph_no column.

2. Create DEPARTMENT table with the following structure.

Name Type Deptno Number, Deptname Varchar2(10), location Varchar2(10)

a. Add column designation to the department table.

b. Insert values into the table.

c. Update the record where deptno is 9.

d. Delete any column data from the table.

3. Create a table called CUSTOMER

Name Type Cust_name Varchar2(20), Cust_street Varchar2(20), Cust_city Varchar2(20)

a. Insert records into the

b. Add salary column to the table.

c. Alter the table column domain.

d. Drop salary column of the customer table.

e. Delete the rows of customer table whose cust_city is "hyd".

Experiment II: Simple Queries on Employee Table-selection, projection, sorting

Emp Table

EMPNO	ENAME	JOB	MGR	HIREDATE	SALARY	DNO
7839	KING	PRESIDENT		17-Nov-81	5000	10
7698	BLAKE	MANAGER	7839	1-May-81	2850	30
7782	CLARK	MANAGER	7839	9-Jun-81	2450	10
7566	JONES	MANAGER	7839	2-Apr-81	2975	20
7654	MARTIN	SALESMAN	7698	28-Sep-81	1250	30
7499	ALLEN	SALESMAN	7698	20-Feb-81	1600	30
7844	TURNER	SALESMAN	7698	20-Feb-81	1500	30
7900	JAMES	CLERK	7698	3-Dec-81	950	30
7521	WARD	SALESMAN	7698	22-Feb-81	1250	30
7902	FORD	ANALYST	7782	3-Dec-81	3000	20
7369	SMITH	CLERK	7788	17-Dec-80	800	20
7788	SCOTT	ANALYST	7782	9-Dec-82	3000	20
7876	ADAMS	CLERK	7788	12-Jan-83	1100	20
7934	MILLER	CLERK	7788	23-Jan-82	1300	10

Department table

DNO	DNAME	LOCATION
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON

1. List empno, empname and salary.
2. List the names of all MANAGERS.
3. list all clerks in deptno. 30.
4. List the employees to who manager is 7698.
5. List jobs in dept 20.
6. List employee names whose salary is between 2000 and 3000.
7. List employees in the departments 10, 20.
8. List employee names which begin with S.
9. List employee names having 'A' in their names.
10. List employees who have joined in JAN.
11. List employees who have joined in the year 81.
12. List all distinct jobs.
13. List employee names in alphabetical order.
14. List employee names alphabetically department wise.
15. List employee names alphabetically job wise.
16. List employee numbers, name sal, DA(15% OF SAL) and PF (10% of sal).
17. List employee names having an experience more than 15 years.
18. List employee names whose commission is NULL.
19. list employees who do not report to anybody.
20. List maximum sal, minimum sal, average sal.
21. List the numbers of jobs.
22. List the numbers of people and average salary in deptno 30.
23. List maximum sal and minimum sal in the designations SALESMAN and CLERK.
24. List the numbers of people and average salary of employees joined in 81, 82 and 83.
25. List jobs that are unique to deptno 20 set operations (Add more problems).
26. Display today's date and present time.
27. List employee names and their joining date in the following formats
    A. SMITH 17th DEC NINETEEEN EIGHTY
    B. SMITH SEVENTEENTH DEC NINETEEN EIGHTY
    C. SMITH Weekday of joining
    D. SMITH 17/12/80
28. List employee names and their experience in years
29. List employee names who joined in DEC and on Monday or Friday.

30. Display a given date as a string in different formats.

Experiment III: Advanced SQL Queries on Employee Table – Dynamic Relations, EXISTS, NOT EXISTS, Aggregate Functions (COUNT, SUM, AVG, MAX and MIN), Conversion Functions & String Functions

1. List employee names and their hire dates sorted in the order of their experience.
2. List managers names and their joining dates completely spelled in alphabetical order of names.
3. List employee names and their experience in years with names arranged in descending order.
4. List the employees' names having minimum of 2years of experience sorted on experience
5. List employee names with all capital letters, with all small letters and with first letter only as capital.
6. List employee names with length of the name sorted on length.
7. List employee names appending Sri to the beginning and Garu to the end.
8. List employee names and month names of joining.
9. List employee names and year of joining in words.
10. List employee's names, job and salary with 5 hyphens in between.
11. List employee names and position of first occurrence of I in their name.
12. List employee names and the string without first character and last character in their name.
13. List employees who joined between Apr 81 and Apr 82.
14. List max sal, min sal and average sal of depts. 10, 30.
15. List the designation in dept 30 but not in 20.
16. List the number of employees in each department along with dept numbers.
17. List number of employees joined year wise.
18. List number of employees job wise.
19. List max sal, min sal, average salary dept wise.
20. List max sal, min sal, average salary job wise.
21. List max sal, min sal for the jobs MANAGER and CLERK.
22. List max sal, min sal AND average salary of the depts. Having a minimum 3 employees.
23. List the number of employees in each job in each department.
24. MGR and the number of employees report to them in the sorted order.
25. List emp numbers of employees to whom a minimum of 3 people report.
26. List dept numbers having a minimum of 3 persons.
27. List names of jobs having a minimum of 3 persons in that job.
28. List names of months in which a minimum of 3 persons joined.

29. List hiredates of employees having 2 or more employees having the same hiredate.

30. List departments having minimum of 3 people having a minimum of 28 years of experience.

Experiment IV: Advanced queries - JOIN OPERATIONS

Structure of the database (Suppliers-Parts-Projects database):

S: (S#, SNAME, STATUS, CITY) PRIMARY KEY (S#)

P: (P#, PNAME, COLOR, WEIGHT, CITY) PRIMARY KEY(P#)

J: (J#, JNAME, CITY) PRIMARY KEY(J#)

SPJ: (S#, P#, J#, QTY)

PRIMARY KEY(S#, P#, J#)

FOREIGN KEY(S#) REFERENCES S
FOREIGN KEY(P#) REFERENCES P
FOREIGN KEY(J#) REFERENCES J

Tables of the above database

S TABLE:

S	SNAME	STATUS	CITY
S1	Smith	20	London
S2	Jones	10	Paris
S3	Blake	30	Paris
S4	Clark	20	London
S5	Adams	30	Athens

P TABLE:

P	PNAME	COLOR	WEIGH	CITY
P1	Nut	Red	18	London
P2	Bolt	Green	17	Paris
P3	Screw	Blue	17	Rome
P4	Screw	Red	14	London
P5	Cam	Blue	12	Paris
P6	Cog	Red	19	London

J TABLE:

J	JNAME	CITY
J1	Sorter	Paris
J2	Display	Rome
J3	OCR	Athens
J4	Console	Athens
J5	Raid	London
J6	EDC	Oslo
J7	Tape	London

SPJ TABLE:

S#	P#	J#	QTY
S1	P1	J1	200
S1	P1	J4	700
S2	P3	J1	400
S2	P3	J2	200
S2	P3	J3	200
S2	P3	J4	500
S2	P3	J5	600
S2	P3	J6	400
S2	P3	J7	800
S2	P5	J2	100
S3	P3	J1	200
S3	P4	J2	500
S4	P6	J3	300
S4	P6	J7	300
S5	P2	J2	200
S5	P2	J4	100
S5	P5	J5	500
S5	P5	J7	100
S5	P6	J2	200
S5	P1	J4	100
S5	P3	J4	200
S5	P4	J4	800

Note: Here in subsequent exercises, the term "all" is to be taken to mean" all currently represented in the Database", not "all possible".

1. Get full details of all projects.
2. Get full details of all Projects in London
3. Get supplier numbers for Suppliers who supply Project J1.
4. Get all shipments where the quantity is in the range 300 to 700.
5. Get all part-color/part-city combinations.
6. Get all Supplier-number/part-number/Project-number triples such that the indicated Supplier part and Project are collocated.
7. Get all supplier-number/part-number/project-number triples such that the indicated supplier, part and project are not all collocated.
8. Get all supplier-number/part-number/ project-number triples such that the indicated supplier, part and project are collocated.
9. Get part numbers for parts supplied by a supplier in London.
10. Get part numbers for parts supplied by a supplier in London to a project in London.
11. Get all pairs of city names such that a supplier in the first city supplies a project in the second city.
12. Get part numbers for parts supplied to any project by a supplier in the same city as that project.
13. Get project numbers for projects supplied by atleast one supplier not in the same city.
14. Get all pairs of part numbers such that some supplier supplies both the indicated parts.
15. Get the total number of projects supplied by supplier S1.
16. Get the total quantity of part P1 supplied by suppliers S1.
17. For each part being supplied to some project get the part number, the project numbers and the corresponding total quantity.
18. Get part numbers of parts supplied to some project in an average quantity of more than 320.
19. Get project names for projects supplied by supplier S1.
20. Get colors of parts supplied by supplier S1.
21. Get parts numbers for parts supplied to any project in London.
22. Get project numbers for projects using atleast one part available from supplier S1.
23. Get supplier numbers for suppliers supplying atleast one part supplied by atleast one supplier who supplies atleast one red part.
24. Get supplier numbers for suppliers with a status lower than that of supplier S1.
25. Get project numbers for projects whose city is first in the alphabetic list

Experiment V: Working with LOOPS using PL/SQL

Program Development using WHILE LOOPS, FOR LOOPS, Nested Loops using ERROR Handling.

Experiment VI: Working with Functions Using PL/SQL

Program Development using Creation of Stored Functions, Invoke Functions in SQL Statements and Write Complex Functions.

Experiment VII: Working with Stored Procedures

Programs Development using Creation of Procedures, Passing Parameters IN and OUT of PROCEDURES

Experiment VIII: Working with CURSORS Develop Programs using Features Parameters in a CURSOR, FOR UPDATE CURSOR, WHERE CURRENT of Clause and CURSOR Variables.	
Text Books :	1. Oracle PL/SQL by Example, Benjamin Rosenzweig, Elena Silvestrova, Pearson Education 3rdEd. 2. Oracle Database Logic PL/SQL Programming, Scott Urman, Tata Mc-Graw Hill. 3. SQL and PL/SQL for Oracle 10g, Black Book, Dr. P. S. Deshpande.

NSS/ NCC/SCOUTS & GUIDES/COMMUNITY SERVICE
II B. Tech. – IV Semester (Code: 24CSL405)

Lectures	0	Tutorial	0	Practical	1	Credits	0.5
Continuous Internal Evaluation			100	Semester End Examination			-

Pre-Requisite: None

Course Objectives:

- Impart discipline, character, fraternity, teamwork, social consciousness among the students.
- Engage the students in selfless service

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

CO1	Develop 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	Extend their help through leadership skills and civic responsibilities to the fellow beings and downtrodden people.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	POs											PSOs		
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**
CO1	-	-	-	-	-	-	-	-	2	-	3	-	-	-
CO2	-	-	-	-	-	-	-	-	2	-	3	-	-	-
CO3	-	-	-	-	-	-	-	-	2	-	3	-	-	-
CO4	-	-	-	-	-	-	-	-	2	-	3	-	-	-

UNIT-I

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

Activities:

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

UNIT-I

Nature & Care

Activities:

1. Best out of waste competition.
2. Poster and signs making competition to spread environmental awareness.
4. Recycling and environmental pollution article writing competition.
5. Organising Zero-waste day.
6. Digital Environmental awareness activity via various social media platforms.

7. Virtual demonstration of different eco-friendly approaches for sustainable living. 8. Write a summary on any book related to environmental issues.
UNIT-I
Community Service Activities: 1. Conducting One Day Special Camp in a village contacting village-area leaders-Survey in the village, identification of problems-helping them to solve via media-authorities-experts-etc. 2. Conducting awareness programs on Health-related issues such as General Health, Mental health, Spiritual Health, HIV/AIDS. 3. Conducting consumer Awareness. Explaining various legal provisions etc. 4. Women Empowerment Programmes-Sexual Abuse, Adolescent Health and Population Education. 5. Any other programmes in collaboration with local charities, NGOs etc.

Text Books :	Nirmalya Kumar Sinha and Surajit Majumder. A Text Book of National Service Scheme, Vol. I. Vidya Kutir Publication, 2021. ISBN 978-81-952368-8-6

ENVIRONMENTAL SCIENCE
II B. Tech. – IV Semester (Code: 24CS406/MC01)

Lectures	2	Tutorial	0	Practical	0	Credits	0
Continuous Internal Evaluation			40	Semester End Examination			--

Pre-Requisite: Chemistry, Physics, Geography and Earth Science

Course Objectives:

- Understand and learn about ecosystem and biodiversity exist in nature.
- Know about the natural resources and sustainability
- Understand different types of pollutions present in Environment
- Know the global environmental problems with case studies

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

CO1	Students develop a strong understanding of ecosystems, biodiversity and the importance of their conservation
CO2	Students gain an understanding of the protection of natural resources for environmental protection and sustainability.
CO3	Know how to manage the harmful pollutions
CO4	Create awareness among the youth on environmental concerns important in the long-term interest of the society.

Mapping of Course Outcomes with Program Outcomes & Program Specific Outcomes

	PO's											PSO's			
CO	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**1**	**2**	**3**	**4**
CO1	3	2	-	-	-	3	-	-	-	-	-	1	-	-	-
CO2	3	2	-	-	-	3	2	-	-	-	-	1	-	-	-
CO3	3	2	-	-	-	3	2	-	3	-	-	1	-	-	-
CO4	3	2	-	-	-	3	-	-	-	-	-	1	-	-	-

UNIT-I

Ecosystems: Definition, Structure and Functions of Ecosystems, Forest Ecosystem.

Biodiversity: Definition and levels of Biodiversity; Values of Biodiversity, Threats and Conservation of Biodiversity.

UNIT-II

Natural resources: Land: Land as a resource, Causes and effects of land degradation,

Water: floods and drought, Dams - benefits and problems.

Sustainability: Rain water harvesting and Watershed management. .

UNIT-III

Pollution: Definition; Causes, effects and control of air, water pollution:

Solid Waste Management - 3R approach, composting and vermin-composting.	
UNIT-IV	
Environmental issues: Global warming, Ozone layer depletion, Acid rains. **Case Studies:** Bhopal Tragedy, Mathura Refinery and TajMahal.	
Text Books :	1. "Environmental Science and Engineering" by Benny Joseph, Tata McGraw-Hill Publishing Company Limited, New Delhi. 2. "Introduction to Environmental Science", Anjaneyulu Y, B S Publications 3. "Comprehensive environmental studies"- JP Sharma, Laxmi Publications.
References :	1. "Environmental studies", R. Rajagopalan, Oxford University Press. 2. "Environmental Science", 11th Edition – Thomson Series – By Jr. G. Tyler Miller. 3. Text Book of environmental Studies – Erach Bharucha

ADVANCED DATA STRUCTURES & ALGORITHMS
II B. Tech. – IV Semester (Code: 24CSH4A)
(Honor Course – 1)

Lectures	3	Tutorial	0	Practical	2	Credits	4
Continuous Internal Evaluation			25T + 25L	Semester End Examination			50T

Pre-Requisite: Data Structures (24CS302), Design & Analysis of Algorithms (24CS405)

Course Objectives:

- To Understand the advanced concepts of efficient binary search trees and Hashing.
- To learn various priority queues, and efficient disjoint-set data structures for solving large-scale data management problems.
- To apply advanced graph algorithms to solve computational problems optimally.
- To develop and analyze the randomized algorithms, and efficient string searching techniques.

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

CO1	Construct the various efficient binary search trees and analyze the advanced hashing techniques such as cuckoo and extendible hashing.
CO2	Implement Binomial Heaps and Apply union–find operations to find cycles in a graph.
CO3	Solve complex problems using graph algorithms.
CO4	Employ randomized algorithms and implement efficient string searching methods.

UNIT-I

Efficient Binary Search Trees & B-Trees- Red-Black Trees, Splay Trees, 2-3 Trees, 2-3-4 Trees – Properties, Rotations, Insertion, Deletion**.**

Hashing – Double hashing, Rehashing, Cuckoo hashing, Extendible Hashing.

UNIT-II

Priority Queues (Heaps) – Binomial Heaps, Fibonacci Heaps, Structure of Fibonacci heaps, Mergeable-heap operations, decreasing a key and deleting a node, Bounding the maximum degree.

Disjoint Sets – Disjoint-set operations, Linked-list representation of disjoint sets, Equivalence Relations, The Dynamic Equivalence Problem, Applications.

UNIT-III

Dictionaries- Definition, Dictionary Abstract Data Type, Implementation of Dictionaries using Linked lists and Hash Tables.

Graph Algorithms – Topological Sort, Network Flow Algorithm, Bipartite Graphs.

UNIT-IV

Randomized Algorithms - Random-Number Generators, Skip Lists, Primality Testing.

String Searching Algorithms – Brute-Forse Algorithm, Rabin-Karp Algorithm, Knuth-Morris-Pratt Algorithm.

LIST OF EXPERIMENTS
1. Implement the following operations on a 2-3 Tree: 1. Insertion 2. Search 3. Traversal 2. Implement the following operations on a Red-Black Tree: 1. Insertion with recolouring and rotations 2. Search 3. Inorder traversal 3. Implement the following operations on a Splay Tree: 1. Splay (Zig, Zig-Zig, Zig-Zag) 2. Insert 3. Search 4. Delete 4. Implement the following operations on a Binomial Heap: 1. Insert 2. Merge 3. Find-min 4. Delete-min 5. Consider the telephone book database of N clients. Make use of a Hash Table implementation to quickly look up client's telephone number. 6. Implement all the functions of Dictionary ADT using hashing. 7. Program to find number of cycles in an undirected graph using Dis Joint Sets. 8. Implement Topological Sort. 9. Program to search for the pattern in a given string using Naïve String-matching Algorithm. 10. Program to search for the pattern in a given string using Robin-Karp Algorithm.

Text Books :	1) Data Structures and Algorithm Analysis by Mark Allen Weiss, Pearson Education, Fourth edition. 2) Algorithms by Robert Sedgewick, Addison - Wesley Publishing Company.
References :	1) Introduction to Algorithms, Thomas H Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, 3rd Edition, 2009, The MIT Press. 2) Advanced Data Structures, Reema Thareja, S. Rama Sree, Oxford University Press, 2018.

ADVANCED DATABASE MANAGEMENT SYSTEMS
III B. Tech. – V Semester (Code: 24CSH5A)
(Honor Course – 2)

Lectures	3	Tutorial	0	Practical	2	Credits	4
Continuous Internal Evaluation			25T + 25L	Semester End Examination			50T

Pre-Requisite: Database Management Systems (24CS404)

Course Objectives:

- To understand the fundamentals of NoSQL databases, their architecture, types, and appropriate use cases compared to traditional RDBMS systems.
- To learn installation, configuration, data models, and CRUD operations in MongoDB using Shell and basic commands.
- To apply aggregation pipelines, operators, and indexing techniques for efficient data querying and optimization.
- To develop applications using MongoDB with Python, including data import/export, sharding, and driver-level integration.

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

CO1	Compare RDBMS with NoSQL databases and identify suitable NoSQL types for different applications.
CO2	Perform CRUD operations, manage collections, and manipulate data using MongoDB.
CO3	Use MongoDB aggregation pipelines and indexing mechanisms to optimize data processing and retrieval.
CO4	Build applications using MongoDB and Python, including sharding, import/export, and driver-based operations.

UNIT-I

Introduction to NoSQL: Difference between RDBMS and NoSQL Database, Definition of NoSQL, History of NoSQL, NoSQL Storage Architecture, Types of NoSQL databases-Document Databases, Key-value databases, Column Oriented databases, Graph databases, When to use NoSQL and when not, Interfacing and Interacting with NoSQL.

UNIT-II

Introduction MongoDB: MongoDB installation, Basics of MongoDB, MongoDB shell, MongoDB datatypes, MongoDB CRUD operations: adding new documents to a collection, selecting documents, updating existing documents, removing documents from a collection.

UNIT-III

MongoDb Aggregation frameworks and MongoDb Aggregation operations: $group, $limit, $project, $sort, $match, $add fields, $count, $lookup, $out operators. MongoDb sorting, MongoDb indexing: single field indexes, sorting with indexed, compound indexed, partial indexes.

UNIT-IV

MongoDb import and export, sharding in MongoDb, MongoDb python drivers, python and MongoDb, creating application with python and MongoDb.

LIST OF EXPERIMENTS

1. Program to demonstrate the differences between RDBMS and NoSQL using sample schema and JSON documents.
   - Create a sample SQL table and equivalent NoSQL documents.
   - Compare flexibility, schema, and query model.
2. Program to classify and demonstrate NoSQL data models (Key–Value, Document, Column-family, Graph) using Python examples.
3. Install MongoDB, create a database, and insert multiple documents using MongoDB shell.
4. Write a Python (PyMongo) program to perform CRUD operations
   - insert_one()
   - find()
   - update_one()
   - delete_one()
5. Program to demonstrate MongoDB data types
   - String, Number, Date, Boolean
   - Arrays and Embedded documents
6. Implement queries with filtering, sorting, projection, and limit in MongoDB.
7. Program to implement MongoDB Aggregation Pipeline with $match, $group, $project operators.
8. Program to perform $lookup (join) between two MongoDB collections.
9. Program to create and use indexes
   - Single-field index
   - Compound index
   - Partial index
10. Program to implement $sort, $limit, $count and display performance difference with/without indexing.
11. Program to import and export data using mongoimport and mongoexport.
12. Program to connect Python with MongoDB (PyMongo) and build a mini application Example: Student management system.
13. Program to implement pagination using skip() and limit() in Python + MongoDB.

Text Books :	1. MongoDB – The Definitive Guide, 2nd edition, Oreilly. 2. Pramod J.Sadalage, Martin Fowler, ”NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence”, 1st edition, Pearson Education, 2012.
References :	1. MongoDB Cook Book, 2nd edition, Cyrus Dasadia & Amol Nayak, PACKT Publishing. 2. Dan Sullivan, “NoSQL for Mere Mortals”, 1st edition, Pearson Education, 2015.