ACADEMIC REGULATIONS AND COURSE STRUCTURE

(R19 Regulations)

COMPUTER SCIENCE & ENGINEERING

FOR

B.Tech., FOUR YEAR DEGREE COURSE

(Applicable for the batches admitted from 2019-20)

VVIT

VASIREDDY VENKATADRI
INSTITUTE OF TECHNOLOGY

VASIREDDY VENKATADRI INSTITUTE OF TECHNOLOGY

NAMBUR, PEDA KAKANI MANDAL, GUNTUR-522508
An Autonomous Institution, Approved by AICTE,

All Courses Accredited by NBA & NAAC with 'A' Grade, Permanently Affiliated to JNTUK University

ACADEMIC REGULATIONS (R19) FOR B. TECH. (REGULAR)

Applicable for the students of B. Tech. (Regular) from the Academic Year 2019-20 onwards

The B.Tech Degree of Jawaharlal Nehru Technological University Kakinada, Kakinada shall be conferred on candidates who are admitted to the programme and who fulfill all the requirements for the award of the Degree.

VISION

To impart quality education through exploration and experimentation and generate socially-conscious engineers, embedding ethics and values, for the advancement in science and technology.

MISSION

- To educate students with a practical approach to dovetail them to industry-needs.
- To govern the institution with a proactive and professional management with passionate teaching faculty.
- To provide holistic and integrated education and achieve over all development of students by imparting scientific and technical, social and cognitive, managerial and organizational skills.
- To compete with the best and be the most preferred institution of the studious and the scholarly.
- To forge strong relationships and linkage with the industry.

OBJECTIVES

- Equip the institute with state-of-the-art infrastructure comparable to the best in the industry.
- Tap the resources of the best minds in the field as faculty and visiting faculty.
- Groom students to become global entrepreneurs and responsible citizens.
- Provide financial assistance to meritorious students.
- Requisition the services of the best HR managers to place our students in reputed industries.
- Provide conducive atmosphere to the faculty for Research & Development and ensure active participation of the students.

Department Vision

Providing quality education to enable the generation of socially conscious software engineers who can contribute to the advancement in the field of computer science and engineering.

Department Mission

1. To equip the graduates with the knowledge and skills required to enable them to be industry ready.
2. To train socially responsible, disciplined engineers who work with good leadership skills and can contribute for nation building.
3. To make our graduates proficient in cutting edge technologies through student centric teaching-learning process and empower them to contribute significantly to the software industry
4. To shape the department into a centre of academic and research excellence.

1. Admission Criteria

The eligibility criteria for admission into UG Engineering programmes are as per the norms approved by Government of Andhra Pradesh from time to time.

The sanctioned seats in each programme in the college are classified into CATEGORY-A, and CATEGORY-B at 1st year level and only CATEGORY-A at Lateral Entry 2nd year level.

The percentages of Category–A, Category-B and Lateral Entry Seats are decided from time to time by the Government of Andhra Pradesh.

- CATEGORY – A (70%): These seats are filled through Convener, EAMCET as per the norms approved by the Government of Andhra Pradesh.
- CATEGORY – B (30%): These seats are filled by the College as per the norms approved by the Government of Andhra Pradesh.
- Lateral Entry: Lateral entry candidates shall be admitted into the Third semester directly as per the norms approved by the Convener, ECET, and Government of Andhra Pradesh.

2. Award of B. Tech. Degree

A student will be declared eligible for the award of B. Tech. Degree if he fulfils the following academic regulations:

- A student after securing admission shall complete the B.Tech programme in a minimum of four academic years (8 Semesters), and a maximum period of eight academic years starting from the date of commencement of first year first semester, failing which student shall forfeit seat in B.Tech Course. Each student shall secure 160 credits (with CGPA>=4) required for the completion of the under graduate programme and award of B.Tech Degree.

3. Courses of Study

The following courses of study are offered at present as specializations for the B. Tech. Courses

S. No	Branch	Branch Code	Intake
1	Civil Engineering	01	120
2	Electrical and Electronics Engineering	02	180
3	Mechanical Engineering	03	180
4	Electronics and Communication Engineering	04	180
5	Computer Science and Engineering	05	240
6	Information Technology	12	180

4. Distribution and Weightage of Marks

i) The performance of a student in each semester shall be evaluated subject wise with a maximum of 100 marks for theory subject and 75 marks for practical subject. The Mini project work shall be evaluated for 50 marks and the Major Project work shall be evaluated for 150 Marks.

ii) For theory subjects the distribution shall be 40 marks for Internal Evaluation and 60 marks for the Semester End Examinations.

iii) For theory subjects, during the semester there shall be two internal Mid Examinations. The weightage of internal marks for 40 consists of Descriptive Test – 15 Marks, Assignment Test- 10 Marks (Open book system with questions in accordance with BLOOMS taxonomy), and Objective Test -10 Marks and Subject Seminar 5 marks.

- The Descriptive Test is for 90 minutes duration conducted for 30 marks and will be scaled down to 15 Marks. Each Descriptive test question paper shall contain 3 questions, one question from each unit and all questions need to be answered. All the questions should be prepared in accordance with BLOOMS Taxonomy.
- The Assignment Test conducted for 20 Marks and will be scaled down to 10 Marks. The test is open book system and the duration of the exam is 60 minutes. The assignment question paper contains 3 questions given by the subject teacher concerned and all questions should be answered. Students can bring a maximum of three printed text books related to that subject. (Soft copies of the text books will not be allowed.) The assignments have to provide broadened exposure to the course. The questions shall include problem solving approach, problem analysis & design, implementation, case studies etc.
- The objective examination is for 20 minutes duration. (Conducted with 20 multiple choice question with a weightage of ½ Mark each)
- For the subject seminar, marks of each student shall be evaluated based on the presentation on any topic of his/her choice in the subject duly approved by the faculty member concerned.
- Internal Marks shall be calculated with 70% weightage for better of the two Mid Exams and 30% weightage for other.

iv) The Semester end examination shall be conducted for 3 hours duration. The question paper shall be given in the following pattern:

The question paper contains one question from each unit with internal choice. Each question carries 12 marks. Each course shall consist of five units of syllabus. The questions shall be framed in line with the Course Outcomes defined and cognitive levels.

v) For practical subjects there shall be continuous internal evaluation during the semester for 25 marks and 50 Marks for Semester end examination. The internal 25 marks shall be awarded

as follows: day to day work - 05 marks, Record-05 marks and the remaining 15 marks are to be awarded by conducting an internal laboratory test of 3 hours duration.

The semester end examination for laboratory courses shall be conducted for three hour duration at the end of semester for 50 marks as follows: Procedure - 10 marks, Experiment/Program execution – 15 Marks, Results-10 Marks and Viva-voice -15 Marks. For laboratory course in English 30 marks for written exam which includes listening comprehension and 20 marks for viva which includes JAM and Group Discussion.

vi) For the subject having design and / or drawing, (such as Engineering Graphics, Engineering Drawing, Machine Drawing) and estimation, the distribution shall be 40 marks for internal evaluation (20 marks for day –to– day work, and 20 marks for internal tests) and 60 marks for end examination. There shall be two internal tests in a Semester and the Marks for 20 can be calculated with 70% weightage for better of the two performances and 30% weightage for other and these are to be added to the marks obtained in day-to-day work.

vii) For Engineering Project on Community services / Mini Project, there shall be continuous evaluation during the semester for 20 marks and semester end evaluation for 30 marks. The distribution of continuous evaluation marks is as follows: Day to Day Assessment- 05 Marks and average of two reviews of 15 Marks each.

The distribution of semester end examination marks for Engineering Project on Community services/Mini Project is as follows: Report -10 Marks and Presentation and Viva Voce – 20 Marks.

vii) For Major Project, there shall be continuous evaluation during the semester for 50 marks and semester end evaluation for 100 marks

The distribution of continuous evaluation marks is as follows: Day-to-day Assessment- 30 Marks and average of at least two reviews of 20 Marks each. The Departmental review committee consists of HoD, two senior Faculty and supervisor concerned.

The semester end examination for Major Project work shall be conducted at the end of VIII Semester. It is evaluated by the Committee consisting of an external examiner, Head of the Department, Senior Faculty and Supervisor of the Project.

viii) Laboratory marks and the internal marks awarded by the faculty are final. However, any grievance regarding marks will be addressed by the result committee if necessary. The recommendations of the committee are final and binding.

ix) MOOCS Courses: All students are eligible to register and complete MOOCS courses relevant to their professional electives listed by the respective departments in the curriculum. However, if any student fails to complete a MOOCS course or the course is not offered by the agency concerned, that student is eligible to attend the examination following the same syllabus and pattern of examination in the VIII semester.

The MOOCS grades awarded to the student by the agency are converted to the course grades based on the percentage of marks obtained. The duration for course registered under MOOCS should range between 8 to 12 Weeks.

x) A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to Industrial Oriented Mini Project/Summer Internship/practical training, if the student secures not less than 40% of marks (i.e., 40 out of 100 allotted marks) in each of them. The student is deemed to have failed, if he/she (i) does not submit a report on Industrial Oriented Mini Project/Summer Internship, or does not make a presentation of the same before the evaluation committee as per schedule, or (ii) does not present the seminar as

required, or (iii) secures less than 40% of marks in Industrial Oriented Mini Project/Summer Internship and project seminar evaluations.

A student may reappear once for each of the above evaluations, when they are scheduled again; if the student fails in such 'one reappearance' evaluation also, the student has to reappear for the same in the next subsequent semester, as and when it is scheduled.

5. Attendance Requirements

- Students shall put in a minimum average attendance of 75% in the semester.
- Condonation of shortage in attendance may be recommended by the respective Head of the Department on genuine medical grounds, provided the student puts in at least 65% attendance and the Principal is satisfied with the genuineness of the reasons and the conduct of the student.
- Students, having more than 65% and less than 75% of attendance, shall have to pay requisite fee towards condonation.
- Students whose shortage of attendance is not condoned in any semester are not eligible to take their end examinations of that semester. They get detained and their registration for that semester shall stand canceled. They will not be promoted to the next semester. They may rejoin in that semester in which the student is detained by getting approval from the principal.
- If any candidate fulfills the attendance requirement in the present semester, he shall not be eligible to readmit into the same class.

6. Minimum Academic Requirements

The following academic requirements have to be satisfied in addition to the attendance requirements mentioned in item No.5

- A student is deemed to have satisfied the minimum academic requirements if he has earned the credits allotted to each theory/practical design/drawing subject/project and secures not less than 35% of marks in the end semester exam, and minimum 40% of marks in the sum total of the internal marks and end semester examination marks.
- A student shall be promoted from first year to second year if he fulfills the minimum attendance requirement.
- A student will be promoted from II year to III year if he fulfills the academic requirement of 40% of the credits up to II B.Tech II semester from all the examinations, whether or not the candidate takes the examinations and secure prescribed minimum attendance in II Year II Semester.
- A student shall be promoted from III year to IV year if he fulfills the academic requirements of 40% of the credits up to III-year II semester from all the examinations, whether or not the candidate takes the examinations and secure prescribed minimum attendance in III Year II Semester.
- A student shall register and put-up minimum attendance in all 160 credits and earn all 160 credits.
- Break in Study: Student, who discontinues the studies for whatever may be the reason, can get readmission into appropriate semester of B. Tech programme after break in study, with the prior permission of the principal and following the transitory regulations applicable to each batch in which he/she joins. A student may utilize this break in study (Maximum of Two years for Regular Students and Maximum of One Year for Lateral Entry Students) only once in the entire period of B. Tech program.

7. Course Pattern

- The entire course of study is for four academic years, all the years are on semester pattern and the medium of instruction is English.
- A student who eligible to appear for the end semester examination in a subject, but absent from it or has failed in the end semester examination, may write the exam in that subject when conducted next.
- When a student is detained for lack of credits/shortage of attendance, he may be readmitted into the same semester in which he has been detained. However, the academic regulations under which he was first admitted shall continue to be applicable to him.

8. CGPA

The grade points and letter grade will be awarded to each course based on students' performance as per the grading system shown in the following Table.

Range of Marks (Theory)	Range of Marks (Lab)	Letter Grade	Level	Grade Points
≥ 90	≥ 67	**O**	Outstanding	10
≥80 to <90	≥60 to <67	**S**	Excellent	9
≥70 to <80	≥52 to <60	**A**	Very Good	8
≥60 to <70	≥45 to <52	**B**	Good	7
≥50 to <60	≥37 to <45	**C**	Fair	6
≥40 to <50	≥30 to <37	**D**	Satisfactory	5
<40	<30	**F**	Fail	0
ABSENT	ABSENT	**AB**	Absent	0

- **Computation of Semester Grade Point Average (SGPA)**

The performance of each student at the end of each semester is indicated in terms of Semester Grade Point Average (SGPA) calculated as shown in below equation (1).

$$\textbf{SGPA (Si)} = \sum \textbf{(Ci X Gi)} / \sum \textbf{Ci} \quad \text{---------- (1)}$$

Where Ci is the number of credits of the i^{th} course and Gi is the grade point scored by the student in the i^{th} course.

- **Computation of Cumulative Grade Point Average (CGPA)**

The Cumulative Performance of each student at the end of each semester is indicated in terms of CGPA and it is calculated as shown in equation (2).

$$\textbf{CGPA} = \sum (\textbf{C}_i \textbf{ X } \textbf{S}_i) / \sum \textbf{C}_i \quad \text{---------- (2)}$$

Where Si is the SGPA of the i^{th} semester and C_i is the total number of credits in that semester.

- The SGPA and CGPA shall be rounded off to 2 decimal points and reported in the transcripts.
- The approximate equivalence of marks to a given CGPA is calculated by using the formula:

Percentage Equivalence of CGPA = [CGPA – 0.5] x 10

9. Award of Class

The criterion for the award of division, after successful completion of the program isas shown in the following table.

Class Awarded	CGPA to be secured	From the CGPA secured from 160 credits
First Class with distinction*	≥7.75	
First Class	≥6.5 - <7.75	

Second Class	≥5.5 - <6.5	
Pass Class	≥4 - <5.5	
Fail	<4	

* Awarded only if all the credit courses prescribed are cleared within four years for regular candidates and three years for lateral entry candidates
* The students who are approved for break in study for entrepreneurship/start-ups will also be considered for award of first class with distinction
* For the purpose of awarding First, Second and Pass Class, CGPA obtained in the examinations appeared within the maximum period allowed for the completion of the program shall be considered.

10. Minimum Days of Instructions

Each semester consists of a minimum of 90 instruction days excluding examination days.

11. Transfer of Branch

There shall be no branch transfer after the completion of the first year admission process.

12. Withholding of results

If the student has not paid any dues to the college or if any case of indiscipline is pending against him/her, the result of the student will be withheld. His/her degree will be withheld in such cases.

13. Transitory Regulations

A candidate who is detained or discontinued a semester, on re-admission, he shall be required to pass all the courses in the curriculum prescribed for such batch of students in which the student joins subsequently. Also, the academic regulations be applicable to him/her which are in force at the time of his/her admission. However, exemption will be given to those candidates who have already passed in such courses in the earlier semester(s) and additional courses are to be studied as approved by the Board of Studies and ratified by the Academic Council.

14. Amendments to Regulations

Revisions of Regulations, Curriculum and Syllabi

The college may from time-to-time revise, amend or change the Regulations, Curriculum, Syllabus and Scheme of examinations through the Board of Studies with the approval of Academic Council and Governing Body of the college.

15. Transferred Students

The students seeking transfer to VVIT from various Universities/ Institutions have to obtain the credits of any equivalent subjects as prescribed by the Academic Council. Only the internal marks obtained in the previous institution will be considered for the evaluation of failed subjects.

*****

ACADEMIC REGULATIONS (R19) FOR B. Tech.

(LATERAL ENTRY SCHEME)

Applicable for the students admitted into II-year B. Tech. from the Academic Year 2020-21 onwards

1. Award of B. Tech. Degree

A student will be declared eligible for the award of B. Tech. Degree if he fulfils the following academic regulations:

- A student shall be declared eligible for the award of the B. Tech Degree, if he pursues a course of study in not less than three academic years and not more than six academic years.
- The candidate shall register for 121 credits and secure all the 121 credits.

2. The attendance regulations of B. Tech. (Regular) shall be applicable to B.Tech Lateral Entry Students.

3. Promotion Rule

- A student shall be promoted from second year to third year if he fulfills the minimum attendance requirement.
- A student shall be promoted from III year to IV year if he fulfills the academic requirements of 40% of the credits up to III-year II semester from all the examinations, whether or not the candidate takes the examinations and secures prescribed minimum attendance in III-year II semester.

4.Award of Class

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

Class Awarded	**CGPA to be secured**	
First Class with distinction	≥7.75	**From the CGPA secured from 121 credits from II Year to IV Year**
First Class	≥6.5 - <7.75	
Second Class	≥5.5 - <6.5	
Pass Class	≥4 - <5.5	
Fail	<4	

5. All the other regulations as applicable to B. Tech. 4-year degree course (Regular) will hold good for B. Tech Lateral Entry Scheme.

MALPRACTICE RULES

DISCIPLINARY ACTION FOR IMPROPER CONDUCT IN EXAMINATIONS

S. No.	Nature of Malpractices/Improper conduct	Punishment
1. (a)	Possesses or keeps accessible in examination hall, any paper, note book, programmable calculators, Cell phones, pager, palm computers or any other form of material concerned with or related to the subject 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 candidate which can be used as an aid in the subject of the examination)	Expulsion from the examination hall and cancellation of the performance in that subject only.
(b)	Gives assistance or guidance or receives it from any other candidate orally or by any other body language methods or communicates through cell phones with any candidate 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 subject only of all the candidates involved. In case of an outsider, he will be handed over to the police and a case is registered against him.
2.	Has copied in the examination hall from any paper, book, programmable calculators, palm computers or any other form of material relevant to the subject of the examination (theory or practical) in which the candidate is appearing.	Expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted to appear for the remaining examinations of the subjects of that Semester/year. The Hall Ticket of the candidate is to be cancelled and sent to the University.
3.	Impersonates any other candidate in connection with the examination.	The candidate who has impersonated shall be expelled from examination hall. The candidate is also debarred and forfeits the seat. The performance of the original candidate who has been impersonated, shall be cancelled in all the subjects of the examination (including practical and project work) already appeared and shall not be allowed to appear for examinations of the remaining subjects of that semester/year. The candidate is also debarred for two

		consecutive semesters from class work and all University examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat. If the imposter is an outsider, he will be handed over to the police and a case is registered against him.
4.	Smuggles in the Answer book or additional sheet or takes out or arranges to send out the question paper during the examination or answer book or additional sheet, during or after the examination.	Expulsion from the examination hall and cancellation of performance in that subject and all the other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred for two consecutive semesters from class work and all University examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat.
5.	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 subject.
6.	Refuses to obey the orders of the Chief Superintendent /Assistant – 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	In case of students of the college, they shall be expelled from examination halls and cancellation of their performance in that subject and all other subjects the candidate(s) has (have) already appeared and shall not be permitted to appear for the remaining examinations of the subjects of that semester/year. The candidates also are debarred and forfeit their seats. In case of outsiders, they will be handed over to the police and a police case is registered against them.

	the officer on duty amounts to use of unfair means or misconduct or has the tendency to disrupt the orderly conduct of the examination.	
7.	Leaves the exam hall taking away answer script or intentionally tears of the script or any part thereof inside or outside the examination hall.	Expulsion from the examination hall and cancellation of performance in that subject and all the other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred for two consecutive semesters from class work and all University examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat.
8.	Possess any lethal weapon or firearm in the examination hall.	Expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred and forfeits the seat.
9.	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 clause 6 to 8.	Student of the colleges expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects 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 will be registered against them.
10.	Comes in a drunken condition to the examination hall.	Expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year.
11.	Copying detected on the basis of	Cancellation of the performance in that

	internal evidence, such as, during valuation or during special scrutiny.	subject and all other subjects the candidate has appeared including practical examinations and project work of that semester/year examinations.
12.	If any malpractice is detected which is not covered in the above clauses 1 to 11 shall be reported to the Controller o Examinations for further action to award suitable punishment.	

Prohibition of ragging in educational institutions Act 26 of 1997

Salient Features

- Ragging within or outside any educational institution is prohibited.
- Ragging means doing an act which causes or is likely to cause Insult or Annoyance of Fear or Apprehension or Threat or Intimidation or outrage of modesty or Injury to a student

	Imprisonment upto		Fine Upto
Teasing, Embarrassing and Humiliation	6 Months	+	Rs. 1,000/-
Assaulting or Using Criminal force or Criminal intimidation	1 Year	+	Rs. 2,000/-
Wrongfully restraining or confining or causing hurt	2 Years	+	Rs. 5,000/-
Causing grievous hurt, kidnapping or Abducts or rape or committing unnatural offence	5 Years	+	Rs. 10,000/-
Causing death or abetting suicide	10 Months	+	Rs. 50,000/-

In case any emergency call Toll Free No. 1800 425 1288

LET US MAKE VVIT A RAGGING FREE CAMPUS

ABSOLUTELY NO TO RAGGING

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

In case any emergency call Toll Free No. 1800 425 1288

LET US MAKE VVIT A RAGGING FREE CAMPUS

COURSE STRUCTURE

I YEAR I SEMESTER						
S. No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	HS	Communicative English	3	0	0	3
2	BS	Applied Chemistry	3	0	0	3
3	BS	Mathematics – I	3	0	0	3
4	ES	Programming for Problem Solving using C	3	0	0	3
5	HS	Communicative English Lab-I	0	0	3	1.5
6	BS	Applied Chemistry Lab	0	0	3	1.5
7	ES	Problem solving using C Lab	0	0	3	1.5
8	ES	Engineering Workshop	0	0	3	1.5
9	MC	Environmental Studies	3	0	0	0
		Total				18

I YEAR II SEMESTER						
S. No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	BS	Mathematics – II	3	0	0	3
2	BS	Mathematics – III	3	0	0	3
3	BS	Applied Physics	3	0	0	3
4	ES	Python Programming	3	0	0	3
5	ES	Basic Electrical & Electronics Engineering	3	0	0	3
6	ES	Engineering graphics and design	1	0	3	2.5
7	ES	Python Programming Lab	0	0	3	1.5
8	BS	Applied Physics Lab	0	0	3	1.5
9	HS	Communicative English Lab-II	0	0	3	1.5
10	MC	Constitution of India	3	0	0	0
		Total				22

II YEAR I SEMESTER						
S. No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	CS	Mathematical Foundations of Computer Science	3	0	0	3
2	CS	Data Structures	3	0	0	3
3	ES	Digital Logic & Circuit Design	3	0	0	3
4	CS	Java Programming	2	1	0	3
5	BS	Probability & Statistics	2	1	0	3
6	CS	Data Structures Lab	0	0	3	1.5
7	CS	Java Programming Lab	0	0	3	1.5
8	MC	Essence of Indian Traditional Knowledge	3	0	0	0
9	MC	Employability Skills – I	3	0	0	0
		Total				18

II YEAR II SEMESTER						
S. No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	CS	Software Engineering	3	0	0	3
2	CS	Advanced Data Structures	3	0	0	3
3	CS	Operating Systems	3	0	0	3
4	CS	Database Management Systems	3	0	0	3
5	CS	Computer Organization	3	0	0	3
6	CS	Database Management Systems Lab	0	0	3	1.5
7	CS	Operating Systems Lab	0	0	3	1.5
8	MC	Professional Ethics & Human Values	3	0	0	0
9	PR	Socially Relevant Project	0	0	2	1
		Total				19

III YEAR I SEMESTER						
S.No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	CS	Formal Languages and Automata Theory	3	0	0	3
2	CS	Design & Analysis of Algorithms	3	0	0	3
3	HS	Managerial Economics & Financial Analysis	3	0	0	3
4	CS	Unix and Shell Programming	3	0	0	3
5	CS	Advanced Java & Web Technologies	3	0	0	3
6	CS	Design & Analysis of Algorithms Lab	0	0	3	1.5
7	CS	Unix and Shell Programming Lab	0	0	3	1.5
8	CS	Advanced Java & Web Technologies Lab	0	0	3	1.5
9	MC	Employability Skills – II	3	0	0	0
		Total				19.5

III YEAR II SEMESTER						
S.No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	CS	Data Warehousing & Data Mining	3	0	0	3
2	CS	Computer Networks	3	0	0	3
3	CS	Artificial Intelligence	3	0	0	3
4	CS	Compiler Design	3	0	0	3
5	PE	Professional Elective – I	3	0	0	3
6	PE	Professional Elective – II	3	0	0	3
7	CS	Computer Networks Lab	0	0	3	1.5
8	CS	Artificial Intelligence Lab	0	0	3	1.5
9	PR	Mini Project	0	0	5	2.5
		Total				23.5

IV YEAR I SEMESTER						
S. No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	CS	Cryptography & Network Security	3	0	0	3
2	CS	Machine Learning	3	0	0	3
3	CS	UML & Design Patterns	3	0	0	3
4	OE	Open Elective – I	3	0	0	3
5	PE	Professional Elective – III	3	0	0	3
6	CS	Machine Learning Lab	0	0	3	1.5
7	CS	Mobile Application Development Lab	0	0	3	1.5
8	MC	Intellectual Property Rights & Patents	3	0	0	0
9	PR	Project – I	0	0	6	3
		Total				21

IV YEAR II SEMESTER						
S. No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	HS	Management & Organizational Behavior	3	0	0	3
2	OE	Open Elective – II	3	0	0	3
3	PE	Professional Elective – IV	3	0	0	3
4	PE	Professional Elective – V	3	0	0	3
5	PR	Project – II	0	0	14	7
		Total				19

PROFESSIONAL ELECTIVES

PE – 1	Computer Graphics	NO – SQL Databases	Full Stack Development	Software Project Management
PE – 2	** Can be contemporary Online Certification Courses which are conducted under standard technical bodies or higher learning institutions such as NPTEL, UDACITY, MOOCS by JNTUK etc.,			
PE – 3	Multimedia & Animation	Big Data Analytics	Cloud Computing	Software Architecture & Design Patterns
PE – 4	Image Processing	Deep Learning	Cyber Security & Forensics	Software Testing Methodologies
PE – 5	Computer Vision	Data Science	Block chain Technologies	Devops

OPEN ELECTIVES OFFERED BY OTHER DEPARTMENTS:

Open Elective- I	Open Elective- II
Number theory and cryptanalysis	Statistics with R
Internet of Things	Fuzzy Sets, Logic and Systems
Supply Chain Management	Entrepreneurship
MATLAB for Engineering Applications	Optimization Techniques
Operations Management	Environmental Pollution and Control
Green Buildings	Remote Sensing and GIS Applications

I Year – I SEMESTER

L	T	P	C
3	0	0	3

COMMUNICATIVE ENGLISH
(Common to All Branches)

Course Objectives

1. Adopt activity-based teaching-learning methods to ensure that learners would be engaged in use of language both in the classroom and laboratory sessions.
2. Facilitate effective listening skills for better comprehension of academic lectures and English spoken by native speakers
3. Focus on appropriate reading strategies for comprehension of various academic texts and authentic materials
4. Help improve speaking skills through participation in activities such as role plays, discussions and structured talks/oral presentations
5. Impart effective strategies for good writing and demonstrate the same in summarizing, writing well organized essays, record and report useful information
6. Provide knowledge of grammatical structures and vocabulary and encourage their appropriate use in speech and writing

Course Outcomes

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

CO1. identify the context, topic, and pieces of specific information from social or transactional dialogues spoken by native speakers of English (L3)
CO2. formulate sentences using proper grammatical structures and correct word forms (L3)
CO3. speak clearly on a specific topic using suitable discourse markers in informal discussions (L3)
CO4. write summaries based on global comprehension of reading/listening texts (L3)
CO5. produce a coherent paragraph interpreting a figure/graph/chart/table (L4)
CO6. take notes while listening to a talk/lecture to answer questions (L3)

Syllabus Blueprint

Contents	Learning Outcomes	Bloom's Level	No of Hrs
Unit-1 **Listening:** Identifying the topic, the context and specific pieces of information by listening to short audio texts and answering a series of questions. **Speaking:** Asking and answering general questions on familiar topics such as home, family, work, studies and interests; introducing oneself and others. **Reading:** Skimming to get the main idea of a text; scanning to look for	1. Identify the context, topic, and pieces of specific information from social or transactional dialogues spoken by native speakers of English 2. ask &answer general questions on familiar topics 3. employ suitable strategies for skimming &scanning to get the general idea of a text and	L3 L2 L3	

specific pieces of information. **Reading for Writing:** Beginnings and endings of paragraphs - introducing the topic, summarizing the main idea and/or providing a transition to the next paragraph. **Grammar and Vocabulary:** Content words and function words; word forms: verbs, nouns, adjectives and adverbs; nouns: countables and uncountables; singular and plural; basic sentence structures; simple question form - wh-questions; word order in sentences.	specific information 4. recognize paragraph structure with beginnings/endings 5. form sentences using proper grammatical structures and correct word forms	L3 L3	10
Unit-2 **Listening:** Answering a series of questions about main idea and supporting ideas after listening to audio texts. **Speaking:** Discussion in pairs/ small groups on specific topics followed by short structured talks. **Reading:** Identifying sequence of ideas; recognizing verbal techniques that help to link the ideas in a paragraph together. **Writing:** Paragraph writing (specific topics) using suitable cohesive devices; mechanics of writing - punctuation, capital letters. **Grammar and Vocabulary:** Cohesive devices - linkers, sign posts and transition signals; use of articles and zero article; prepositions.	1. comprehend short talks on general topics 2. speak clearly on a specific topic using suitable discourse markers in informal discussions 3. understand the use of cohesive devices for better reading comprehension 4. write well-structured paragraphs on specific topics 5. make necessary grammatical corrections in short texts	L2 L3 L2 L3 L3	10
Unit-3 **Listening:** Listening for global comprehension and summarizing what is listened to. **Speaking:** Discussing specific topics in pairs or small groups and reporting what is discussed	1. summarize the content with clarity &precision from short talks 2. report what is discussed in informal discussions 3. infer meanings of unfamiliar words using contextual clues	L3 L3 L3	10

Reading: Reading a text in detail by making basic inferences - recognizing and interpreting specific context clues; strategies to use text clues for comprehension. **Writing:** Summarizing - identifying main idea/s and rephrasing what is read; avoiding redundancies and repetitions. **Grammar and Vocabulary:** Verbs - tenses; subject-verb agreement; direct and indirect speech, reporting verbs for academic purposes.	4. write summaries based on global comprehension of reading/ listening texts 5. use correct tense forms, appropriate structures and a range of reporting verbs in speech and writing	L3 L3	
Unit-4 **Listening:** Making predictions while listening to conversations/ transactional dialogues without video; listening with video. **Speaking:** Role plays for practice of conversational English in academic contexts (formal and informal) - asking for and giving information/directions. **Reading:** Studying the use of graphic elements in texts to convey information, reveal trends/patterns/relationships, communicate processes or display complicated data. **Writing:** Information transfer; describe, compare, contrast, identify significance/trends based on information provided in figures/charts/graphs/tables. **Grammar and Vocabulary:** Quantifying expressions - adjectives and adverbs; comparing and contrasting; degrees of comparison; use of antonyms	1. infer &predict about content of spoken discourse 2. engage in formal/informal conversationsunderstanding verbal &non-verbal features of communication 3. interpret graphic elements used in academic texts 4. produce a coherent paragraph interpreting a figure / graph / chart / table 5. use language appropriate for description and interpretation of graphical elements	L4 L3 L2 L4 L4	10
Unit-5 **Listening:** Identifying key terms, understanding concepts and answering a series of relevant	1. take notes while listening to a talk/lecture to answer questions 2. make formal oral presentations using effective	L3	

questions that test comprehension. **Speaking:** Formal oral presentations on topics from academic contexts - without the use of PPT slides. **Reading:** Reading for comprehension. **Writing:** Writing structured essays on specific topics using suitable claims and evidences **Grammar and Vocabulary:** Editing short texts – identifying and correcting common errors in grammar and usage (articles, prepositions, tenses, subject verb agreement)	strategies 3. produce a well-organized essay with adequate details 4. edit short texts by correcting common errors	L3 L3 L4	10

Detailed Syllabus

Unit 1 A Proposal to Girdle the Earth (Excerpt) by Nellie Bly
Theme: Exploration
1. "How to Fashion Your Own Brand of Success" by Howard Whitman
2. "How to Recognize Your Failure Symptoms" by Dorothea Brande
Listening
- identifying thetopic, the context and specific pieces of information

Speaking
- introducing oneself and others

Reading
- skimming for main ideas
- scanning for specific pieces of information

Writing/ Reading for Writing
- paragraphs, beginnings, introducing the topic, key words, main idea

Grammar and Vocabulary
- content words and function words
- word forms: verbs, nouns, adjectives and adverbs
- nouns: countable and uncountable; singular and plural forms
- basic sentence structures; simple question form: why-questions; word order in sentences

Learning Outcomes
- Understand social or transactional dialogues spoken by native and non-native speakers of English and identify the context, topic, and pieces of specific information.
- Ask and answer general questions on familiar topics and introduce oneself/others
- employ suitable strategies for skimming and scanning to get the general idea of a text and locate specific information
- Recognize paragraph structure and be able to match headings/main ideas with paragraphs
- Form sentences using proper grammatical structures and correct word forms

Unit 2 An excerpt from The District School As It Was by One Who Went to It by Warren Burton
Theme: On Campus
3. **"How to Conquer the Ten Most Common Causes of Failure" by Lois Binstock**
4. **"How to Develop Your Strength to Seize Opportunities" by Maxwell Maltz**

Listening

- answering a series of questions about main idea and supporting ideas after listening to audio texts

Speaking

- discussion in pairs/ small groups on specific topics; preparing and delivering short structured talks using suitable cohesive devices

Reading

- identifying sequence of ideas
- recognizing verbal techniques that help link the ideas in a paragraph

Writing/ Reading for Writing

- paragraph writing (specific topics) using suitable cohesive devices; using key words/phrases and organizing points in a coherent manner
- mechanics of writing: punctuation, capital letters

Grammar and Vocabulary

- cohesive devices-linkers, sign posts and transition signals
- use of articles and zero articles
- prepositions

Learning Outcomes

- comprehend short talks on general topics
- participate in informal discussions and speak clearly on a specific topic using suitable discourse markers
- understand the use of cohesive devices for better reading comprehension
- write well-structured paragraphs on specific topics using suitable cohesive devices
- identify basic errors of grammar/usage and make necessary corrections in short texts

Unit 3 The Future of Work?

Theme: Working Together

5. **"How to Make the Most of Your Abilities" by Kenneth Hildebrand**
6. **"How to Raise Your Self-Esteem and Develop Self-Confidence" by James W. Newman**

Listening

- listening for global comprehension
- summarizing what is listened to

Speaking

- discussing specific topics in pairs/ small groups
- reporting what is discussed

Reading

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

Writing/ Reading for Writing

- summarizing-identifying main idea/s
- rephrasing what is read
- avoiding redundancies and repetitions

Grammar and Vocabulary

- Verbs-tenses; subject-verb agreement; direct and indirect speech, reporting verbs for academic purposes

Learning Outcomes

- comprehend short talks and summarize the content with clarity and precision
- participate in informal discussions and report what discussed
- infer meanings of unfamiliar words using contextual clues
- write summaries based on global comprehension of reading/listening texts
- use correct tense forms, appropriate structure and a range of reporting verbs in speech and writing.

Unit 4 H.G Wells and the Uncertainties of Progress by Peter J. Bowler
Theme: Fabric of Change

7. **"How to Win Your War Against Negative Feelings" by Dr Maxwell Maltz**
8. **"How to Find the Courage to Take Risks" by Drs Tom Rust and Randy Reed**

Listening

- making predictions while listening to conversations/transnational dialogues without video
- listening with video

Speaking

- role plays for practice of conversational English in social and academic contexts (formal & informal)
- asking for and giving information/directions/instructions/suggestions

Reading

- understand and interpret graphic elements used in texts (convey information, reveal trends/patterns/relationships, communicate processes or display data)

Writing/ Reading for Writing

- information transfer
- describe, compare, contrast, identify significance/trends based on information provided in figures/charts/graphs/tables

Grammar and Vocabulary

- quantifying expressions-adjectives and adverbs
- comparing and contrasting
- degrees of comparison
- use of antonyms

Learning Outcomes

- make inferences and predictions while listening to spoken discourse
- understand verbal and non-verbal features of communication and hold formal / informal conversations
- interpret graphic elements used in academic texts
- produce a coherent paragraph interpreting a figure/graph/chart/table
- use language appropriate for description and interpretation of graphical elements

Unit 5 Leaves from the Mental Portfolio of a Eurasian by Sui Sin Far
Theme: Tools for Life

9. **"How to Become a Self-Motivator" by Charles T Jones**
10. **"How to Eliminate Your Bad Habits"by Og Mandino**

Listening

- identifying the key terms
- understanding concepts
- answering a series of relevant questions that test comprehension

Speaking

- formal oral presentations on topics from academic contexts-without the use of PPT slides

Reading

- reading for comprehension

Writing/ Reading for Writing

- writing structured essays on specific topics using suitable claims and evidences

Grammar and Vocabulary

- reinforcing learning: articles, prepositions, tenses, subject-verb agreement

Learning Outcomes

- take notes while listening to a talk/lecture and make use of them to answer questions
- make formal oral presentations using effective strategies
- comprehend, discuss and respond to academic texts oral and in writing
- produce a well-organized essay with adequate support and detail
- edit short texts by correcting common errors

CO-PO MAPPING

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1									2	3		1
CO2									2	3		1
CO3									2	3		1
CO4									2	3		1
CO5									2	3		1

*****

I Year – I SEMESTERI

L	T	P	C
3	0	2	3

APPLIED CHEMISTRY

Course Objectives:

- Significance and use of plastics in household appliances and composites (FRP) in aerospace and automotive industries.
- Outline the basics for the construction of electro-chemical cells, batteries and fuel cells.
- Understand the mechanism of corrosion and how it can be prevented.
- Importance of advanced materials and their engineering applications.
- Differentiate and discuss the materials used in major industries like steel industry, metallurgical industries, construction industries, electrical equipment and manufacturing industries. Lubrication is also summarized.
- Essentiality of fuel technology.
- Need of water purification and importance of various water purification methods.

Course Outcomes:

Knowledge of basic concepts of chemistry for engineering students will help them as professional engineers later in design and material selection as well as utilizing the available resources. At the end of the course, the students will be able to

1. Explain the preparation, properties and applications of thermoplastics, thermosettings, elastomers and conducting polymers.
2. Know the importance of various materials and their uses in the construction of batteries and fuel cells.
3. To acquire the knowledge of nano-materials, refractories, lubricants and cement.
4. Assess the quality of various fuels.
5. Understand the importance of water and its usage in various industries.

UNIT-I

Polymer Technology (14 hrs)

Polymerisation*:* Introduction-Methods of polymerisation-(emulsion and suspension)-Physical and mechanical properties.

Plastics*:* Compounding-Fabrication (compression, injection, blown film, extrusion)-Preparation, properties and applications of PVC, ploycarbonates and Bakelite-Mention some examples of plastic materials used in electronic gadgets, recycling of e-plastic waste.

Elastomers*:* Natural rubber-Drawbacks-Vulcanization-Preparation-Properties and applications of synthetic rubbers (Buna S, thiokol and polyurethanes)

Composite Materials: Fiber reinforced plastics-CFRP and GFRP.

Conducting polymers: Polyacetylene, doped conducting polymers- p-type and n-type doping.

Bio degradable polymers: Biopolymers and biomedical polymers.

UNIT-II

Electrochemical Cells and Corrosion (12 hrs)

Single electrode potential-Electrochemical series and uses of series-Standard hydrogen electrode, calomel electrode, concentration cell, construction of glass electrode, Batteries: Dry cell, Ni-Cd cells, Ni-Metal hydride cells, Li-ion battery, Zinc air cells, Fuel cells-H_2 –O_2, CH_3OH-O_2, phosphoric acid, molten carbonate.

Corrosion*:* Definition-theories of corrosion (chemical and electrochemical)-galvanic corrosion, differential aeration corrosion, stress corrosion, water-line corrosion- passivity of metals-galvanic series-factors influencing rate of corrosion-corrosion control: (proper

designing, cathodic protection)-protective coatings: cathodic and anodic coatings, electroplating, electroless plating (nickel), paints (constituents and its functions).

UNIT-III

Chemistry of Materials **(12 hrs)**

Nano materials: Introduction, sol-gel method, characterization by BET, SEM and TEM methods, applications of graphene- carbon nanotubes and fullerenes: Types, preparation of carbon nanomaterials by carbon-arc, laser ablation method, and applications.

Refractories: Definition, classification, properties (refractoriness, refractoriness under load, porosity and thermal spalling), failure of refractories.

Lubricants: Definition, mechanism of lubricants and properties (definition and importance).

Cement: Constituents, manufacturing, parameters to characterize the Clinker formation: lime saturation factor (LSF), silica ratio (SR), and alumina ratio (AR). Chemistry of setting and hardening, deterioration of cement.

UNIT-IV

Fuels **(12 hrs)**

Introduction-calorific value - HCV and LCV – problems using Dulong's formula – proximate and ultimate analysis of coal sample – significance of these analysis – problems – petroleum (refining – cracking) – synthetic petrol (Fischer-Tropsch & Bergius) – petrol knocking, diesel knocking – octane and cetane rating – anti-knocking agents – introduction to alternative fuels (bio-diesel, ethanol, methanol, natural gas, LPG, CNG) – Flue gas analysis by Orsat apparatus – rocket fuels.

UNIT-V

Water Technology **(12 hrs)**

Hardness of water – determination of hardness by complexometric method – boiler troubles (priming and foaming, scale formation, boiler corrosion, caustic embrittlement) – internal treatments – softening of hard water (zeolite process and ion exchange process) – treatment of industrial waste water – potable water and its specifications – steps involved in purification of water – chlorination, break point chlorination – reverse osmosis and electro dialysis.

Text Books

- *Engineering Chemistry* by Jain & Jain; Dhanpat Rai Publicating Co., Latest Edition
- *Engineering Chemistry* by Shikha Agarwal; Cambridge University Press, 2019 Edition
- *Engineering Chemistry* by Prasanth Rath, B. Ramadevi, Ch. Venkata Ramana Reddy, Subendu Chakravarthy; Cengage Publications, 2019 Edition.

Reference Books

1. *A text book of Engineering Chemistry* by S.S. Dara, S. S. Umare; S. Chand & Co., Ltd., Latest Edition.
2. *Engineering Chemistry* by Shashi Chawla; Dhanpat Rai Publicating Co., Latest Edition.

I Year – I SEMESTER

L	T	P	C
3	0	0	3

MATHEMATICS – I (Calculus)
(Common to ALL branches)

Course Objectives:

1. This course will illuminate the students in the concepts of calculus.
2. To enlighten the learners in the concept of differential equations and multi-variable calculus.
3. To equip the students with standard concepts and tools at an intermediate to advanced level mathematics to develop the confidence and ability among the students to handle various real-world problems and their applications.

Unit-1: Differential equations of first order and first degree:

Linear differential equations-Bernoulli's equations - Exact equations and equations reducible to exact form.

Applications: Newton's Law of cooling – Law of natural growth and decay – Orthogonal trajectories – Electrical circuits.

Unit-2: Linear differential equations of higher order:

Non-homogeneous equations of higher order with constant coefficients – with non-homogeneous term of the type e^{ax},sinax,cosax,polynomials in x^n, $e^{ax}V(x)$and $x^nV(x)$ - Method of Variation of Parameters.

Applications: LCR circuit – Simple harmonic motion

Unit-3: Mean value theorems:

Mean value theorems (without proofs): Rolle's Theorem – Lagrange's mean value theorem – Cauchy's mean value theorem – Taylor's and Maclaurin's theorems with remainders.

Unit-4: Partial differentiation:

Introduction – Homogeneous function – Euler's theorem - Total derivative – Chain rule – Jacobian – Functional dependence – Taylor's and Mc Laurent's series expansion of functions of two variables.

Applications: Maxima and Minima of functions of two variables without constraints and Lagrange's method (with constraints).

Unit-5: Multiple integrals:

Double integrals (Cartesian and Polar) – Change of order of integration – Change of variables (Cartesian to Polar) –Triple integrals.

Applications: Areas by double integrals and Volumes by triple integrals.

TEXT BOOKS:

1. **B.S. Grewal,** Higher Engineering Mathematics, 44th Edition, Khanna Publishers.
2. **B.V. Ramana,** Higher Engineering Mathematics, 2007 Edition, Tata Mc. Graw Hill Education.

REFERENCE BOOKS:

1. **H. K. Das,** Advanced Engineering Mathematics, 22nd Edition, S. Chand & Company Ltd.
2. **Erwin Kreyszig**, Advanced Engineering Mathematics, 10th Edition, Wiley-India.

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

- Solve the differential equations related to various engineering fields.
- Utilize mean value theorems to real life problems.
- Familiarize with functions of several variables which is useful in optimization.
- Apply double integration techniques in evaluating areas bounded by region.
- Learn important tools of calculus in higher dimensions. Students will become familiar with 2-dimensional and 3 – dimensional coordinate systems.

Micro-Syllabus of MATHEMATICS – I (Calculus)

Unit-1: Differential equations of first order and first degree:

Linear differential equations-Bernoulli's equations - Exact equations and equations reducible to exact form.

Applications: Newton's Law of cooling – Law of natural growth and decay – Orthogonal trajectories – Electrical circuits.

Unit	Module	Micro content
1a. & 2a. Differential equations of first order and first degree	Linear differential equations	Solution of Linear differential equations in $'y'$
		Solution of Linear differential equations in $'x'$
		Initial value problem
	Non-Linear differential equations	Bernoulli's equations
		Equations reducible to Linear differential equations
	Exact differential equations	Solution of Exact differential equations
	Non-Exact differential equations	Equations reducible to Exact equations
		Integrating factor found by inspection
		Integrating factor of a Homogeneous equation
		Integrating factor for an equation of the type $f_1(xy)\,ydx + f_2(xy)\,xdy = 0$
		Integrating factor, if $\frac{\frac{\partial M}{\partial y} - \frac{\partial N}{\partial x}}{N}$ be a function of $'x'$
		Integrating factor, if $\frac{\frac{\partial N}{\partial x} - \frac{\partial M}{\partial y}}{M}$ be a function of $'y'$
1b. & 2b. Applications	Application of differential equations of first order and first degree	Newton's Law of cooling
		Law of natural growth and decay
		Orthogonal trajectories
		Electrical circuits

Unit-2: Linear differential equations of higher order:

Non-homogeneous equations of higher order with constant coefficients – with non-homogeneous term of the type e^{ax},sinax,cosax,polynomials in x^n, $e^{ax}V(x)$and $x^nV(x)$ - Method of Variation of Parameters.

Applications: LCR circuit – Simple harmonic motion

Unit	Module	Micro content
3a. & 4a. Linear differential equations of higher order	Homogeneous equations of higher order with constant coefficients	Finding the Complementary function
	Non-homogeneous equations of higher order with constant coefficients	Particular integral of the type $'e^{ax}'$
		Particular integral of the type $'\sin ax' \ (or) \ 'cos\, ax'$
		Particular integral of the type x^n
		Particular integral of the type $'e^{ax}\ V(x)'$
		Particular integral of the type $'x^n v(x)'$
3b. & 4b. Applications	Applications of Non-homogeneous equations of higher order with constant coefficients	Method of variation of parameters
		LCR circuit
		Basic problems on simple harmonic motion

Unit-3: Mean value theorems:

Mean value theorems (without proofs): Rolle's theorem – Lagrange's mean value theorem – Cauchy's mean value theorem – Taylor's and Maclaurin's theorems with remainders.

Unit	Module	Micro content
5a. & 6a. Mean value theorems	Mean value theorems	Rolle's theorem
		Lagrange's mean value theorem
5b. & 6b. Mean value theorems	Mean value theorems	Cauchy's mean value theorem
		Taylor's expansions of $f(x)$
		Maclaurin's expansions of $f(x)$

Unit-4: Partial differentiation:

Introduction – Homogeneous function – Euler's theorem - Total derivative – Chain rule – Jacobians – Functional dependence – Taylor's and Mc Laurent's series expansion of functions of two variables.

Applications: Maxima and Minima of functions of two variables without constraints and Lagrange's method (with constraints).

Unit	Module	Micro content

7a. & 8a. Partial differentiation	**Partial Differentiation**	Euler's theorem
		Total derivative
		Chain rule
		Jacobians
7b. & 8b. Applications	**Applications of Partial Differentiation**	Taylor's and Mc Laurent's series expansion of functions of two variables
		Maxima and Minima of functions of two variables
		Lagrange's method of undetermined multipliers

Unit-5: Multiple integrals:

Double integrals (Cartesian and Polar) – Change of order of integration – Change of variables (Cartesian to Polar) –Triple integrals.

Applications: Areas by double integrals and Volumes by triple integrals.

Unit	**Module**	**Micro content**
9a. & 10a. Multiple integrals	Evaluation of Double Integrals	Double integrals
		Change of order of integration
		Double integrals in Polar co-ordinates
		Change of variables
9b. & 10b. Applications	Evaluation of Triple Integrals	Triple integrals
	Applications of Multiple Integrals	Areas by double integrals
		Volumes by triple integrals

CO – PO MAPPING

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2										1
CO2	3	2										1
CO3	3	2										1
CO4	3	2										1
CO5	3	2										1

I Year – I SEMESTER

L	T	P	C
3	1	0	3

PROGRAMMING FOR PROBLEM SOLVING USING C

(Common to All Branches)

Course Objectives:

1. To familiarize to notion of an algorithm, editing and executing programs in Linux.
2. To Understanding branching, iteration.
3. To represent Data using arrays.
4. To use Modular programming and recursive solution formulation.
5. To familiarize pointers and dynamic memory allocation.
6. To handle data through files

UNIT-I: Introduction to C **10 hrs**

Introduction to Computers: hardware, Memory hierarchy, Types of Computers, Types of Software – Operating Systems, Translators, Device drivers and packages. Algorithms and its characteristics, Program development steps. Structure of a C program, Features of C, The main () Function, Standard I/O functions.

Programming Style - Indentation, Comments, Identifiers, Data Types**,** Operators, Precedence and Associativity. Variables and Declarations, Format Modifiers, Escape Sequences, Types of Statements

Casting - Implicit Type Conversions, Explicit Type Conversions, Mathematical Library Functions

UNIT-II: Control Flow & Modules **12 hrs**

Selection: if-else Statement, nested if, examples, Multi-way selection: switch, else-if, examples. **Repetition**: Basic Loop Structures, Pre-test and Post-test Loops, Counter-Controlled and Condition-Controlled Loops, for, while and do while.

Branching: break & continue.

Modular Programming: Function and Parameter Declarations, Returning a Value, Types of parameters. Parameter – scalar data as argument.

Recursion: Definition, Base condition for recursion, Mathematical Recursion, Recursion versus Iteration.

UNIT-III Arrays & Strings **12 hrs**

Arrays: Introduction to Arrays, Input and Output of Array Values, Array Initialization, Arrays as Function Arguments, Two-Dimensional Arrays, Larger Dimensional Arrays- Matrices, 1D & 2D arrays as arguments.

Strings: String Fundamentals, String Input and Output, String Processing, Library Functions, Strings as arguments.

Unit – IV Pointers & Structures **12 hrs**

Pointers: Concept of a Pointer, Initialization of Pointer variables, Pointers as function arguments, Passing by address, Dangling memory, Pointer Arithmetic, Character pointers, Pointers to Pointers, Array of pointers & Pointer to array, Dynamic memory management functions, Command line Arguments.

Structures: Derived types, Structure's declaration, Initialization of structures, accessing structures, nested structures, arrays of structures, structures and functions, pointers to structures, self-referential structures, unions, typedef, enum, bit-fields.

UNIT-V: Files **10 hrs**

Storage classes – auto, static, extern, register. Pre-processor statements

Data Files: Declaring, Opening, and Closing File Streams, File handling functions, Reading from and Writing to Text Files, File copy, merge, Writing and reading records, Random File Access.

Text Books:

1. ANSI C Programming, E Balaguruswamy, Mc-GrawHill, 5th Edition
2. ANSI C Programming, Gary J. Bronson, Cengage Learning.
3. Programming in C, ReemaThareja, OXFORD Publications

Reference Books:

1. C Programming-A Problem Solving Approach, Forouzan, Gilberg, Cengage.
2. Let us C, YashwantKanetkar, BPB Publications
3. Mastering in C, KR Venu Gopal, TMH

Course Outcomes: After completing this course, Students will be able to-

CO 1: Understand algorithms and basic terminology of C

CO 2: Solve problems using control structures and modular approach

CO 3: Make use of 1D and 2D arrays along with strings for linear data handling

CO 4: Determine the use of pointers and structures

CO 5: Implement various operations on data files.

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO11	PO 12	PSO1	PSO2
CO1	1	2	3	2	1	-	-	-	3	3	1	2	1	2
CO2	2	3	3	2	-	-	-	-	1	1	2	2	2	2
CO3	3	3	3	2	-	-	-	-	2	1	2	2	2	3
CO4	2	2	2	2	-	-	-	-	2	1	2	2	2	2
CO5	2	2	2	2	-	-	-	-	2	1	2	2	1	2

Micro-Syllabus of Programming for Problem Solving Using C

UNIT I:

Introduction to Computers: Hardware, Memory hierarchy, Types of Computers, Types of Software – Operating Systems, Translators, Device drivers and packages. Algorithms and its characteristics, Program development steps. Structure of a C program, Features of C, The main () Function, Standard I/O functions.

Programming Style - Indentation, Comments, Identifiers, Data Types**,** Operators, Precedence and Associativity. Variables and Declarations, Format Modifiers, Escape Sequences, Types of Statements

Casting - Implicit Type Conversions, Explicit Type Conversions, Mathematical Library Functions.

Unit	Module	Micro content
Introduction to C	Introduction to Computers	Components of Computer: Hardware & Software
		Algorithm and its characteristics
		Program development steps
		Structure of a C Program
		Features of C
		The main () function and standard I/O functions
	Programming Style	Indentation, Comments, Identifiers, Data Types
		Operators, Precedence and Associativity. Variables and Declarations
		Format Modifiers, Escape Sequences
		Types of Statements
	Casting	Implicit Type Conversions
		Explicit Type Conversions
		Mathematical Library Functions

UNIT II:

Selection: if-else Statement, nested if, examples, Multi-way selection: switch, else-if, examples. **Repetition**: Basic Loop Structures, Pre-test and Post-test Loops, Counter-Controlled and Condition-Controlled Loops, for, while and do while.

Branching: break & continue.

Modular Programming: Function and Parameter Declarations, Returning a Value, Types of parameters. Parameter – scalar data as argument.

Recursion: Definition, Base condition for recursion, Mathematical Recursion, Recursion versus Iteration.

Unit	Module	Micro content
	Selection Statements	if else, nested if examples
		Multi Way Selection: switch, else if examples
	Iterative	Counter Controlled Loops

Control Flow & Modular Programming	Statements	Logic Controlled Loops
	Unconditional Branching	Break & Continue
	Modular Programming	Function and Parameter Declarations
		Returning a Value
		Types of parameters. Parameter – scalar data as argument.
	Recursion	Definition, Base condition for recursion
		Mathematical Recursion
		Recursion versus Iteration

UNIT III:

Arrays: Introduction to Arrays, Input and Output of Array Values, Array Initialization, Arrays as Function Arguments, Two-Dimensional Arrays, Larger Dimensional Arrays- Matrices, 1D & 2D arrays as arguments.

Strings: String Fundamentals, String Input and Output, String Processing, Library Functions, Strings as arguments.

Unit	**Module**	**Micro content**
Arrays & Strings	Arrays	Introduction to Arrays, Input and Output of Array Values, Array Initialization
		Arraysas Function Arguments
		Two-Dimensional Arrays, Larger Dimensional Arrays
		Matrices, 1D & 2D arrays as arguments
	Strings	String Fundamentals, String Input and Output
		String Processing, Library Functions
		Strings as arguments

UNIT IV:

Pointers: Concept of a Pointer, Initialization of Pointer variables, Pointers as function arguments, Passing by address, Dangling memory, Pointer Arithmetic, Character pointers, Pointers to Pointers, Array of pointers & Pointer to array, Dynamic memory management functions, Command line Arguments.

Structures: Derived types, Structures declaration, Initialization of structures, accessing structures, nested structures, arrays of structures, structures and functions, pointers to structures, self-referential structures, unions, typedef, enum, bit-fields.

Unit	**Module**	**Micro content**
Pointers and Structures	Pointers	Concept of a Pointer, Initialization of Pointer variables
		Pointers as function arguments, Passing by address
		Dangling memory, Pointer Arithmetic, Character pointers
		Pointers to Pointers
		Dynamic Memory Allocation

		Pointer to Arrays and Array of Pointers
	Command line Arguments	Command line Arguments
	Structures	Derived types, Structures declaration, Initialization of structures
		Accessing structures, nested structures, arrays of structures
		structures and functions, pointers to structures, self-referential structures
		Unions, typedef, enum, bit-fields.

UNIT V:
Storage classes – auto, static, extern, register. Pre-processor statements
Data Files: Declaring, Opening, and Closing File Streams, File handling functions, Reading from and Writing to Text Files, File copy, merge, Writing and reading records, Random File Access.

Unit	**Module**	**Micro content**
Storage Classes and Files	Storage Classes	auto, static, extern and register
	Pre-processor Statements	Pre-processor Statements
	Data Files	Declaring, Opening, and Closing File Streams
		File handling functions, Reading from and Writing to Text Files
		File copy, merge, Writing and reading records
		Random File Access

I Year – I SEMESTER

L	T	P	C
0	0	3	1.5

COMMUNICATIVE ENGLISH LAB I
(Common to All branches)

Course Objectives

The main objective of the course is to adopt activity-based teaching-learning methods to ensure that learners would be engaged in use of language both in the classroom and laboratory sessions and appear confidently for competitive examinations for career development.

The specific objectives of the course are to

1. Facilitate effective listening skills for better comprehension of academic lectures and English spoken by native and non-native speakers
2. Focus on appropriate reading strategies for comprehension of various academic texts and authentic materials like newspapers, magazines, periodicals, journals, etc.
3. Help improve speaking skills through participation in activities such as role plays, discussions and structured talks/oral presentations
4. Impart effective strategies for good writing and demonstrate the same in summarizing, writing well organized essays, record and report useful information
5. Provide knowledge of grammatical structures and vocabulary and encourage their appropriate use in speech and writing

Course Outcomes

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

CO1. identify the context, topic, and pieces of specific information from social or transactional dialogues spoken by native speakers of English and speak clearly on a specific topic using suitable discourse markers in informal discussions (L3)

CO2. take notes while listening to a talk/lecture; to answer questions in English; formulate sentences using proper grammatical structures and correct word forms; and use language effectively in competitive examinations (L3)

CO3. write summaries based on global comprehension of reading/listening texts; produce a coherent write-up interpreting a figure/graph/chart/table; and use English as a successful medium of communication. (L3)

Detailed Syllabus

CALL based activity. English course books selected for classroom teaching will be used for practice in the computer-based language labs. However, a brief introduction to the English Phonetics will be given to the students. Activities that encourage individual learning of the students based on the suggested texts and web resources will be used in the practical sessions.

Introduction to Sound System of English

Articulation - Airstream mechanism, Manners of Articulation, Places of Articulation, English phonetic symbols.

Accent - Syllabification, word stress and accent, stress rules and stress shift, exceptions to rules.

Intonation - Stress and accent in connected speech. Types and functions of Intonation in English.

Pair work, Role play, conversational practice and Individual speaking activities based on following essays from ***University of Success.***

1. "How to Fashion Your Own Brand of Success" by Howard Whitman
2. "How to Recognize Your Failure Symptoms" by Dorthea Brand
3. "How to Conquer the Ten Most Common Causes of Failure" by Lois Binstock
4. "How to Develop Your Strength to Seize Opportunities" by Maxwell Maltz
5. "How to Make the Most of Your Abilities" by Kenneth Hildebrand
6. "How to Raise Your Self-Esteem and Develop Self-Confidence" by James W. Newman
7. "How to Win Your War Against Negative Feelings" by Dr Maxwell Maltz
8. "How to Find the Courage to Take Risks" by Tom Rust and Randy Reed
9. "How to Become a Self-Motivator" by Charles T Jones
10. "How to Eliminate Your Bad Habits" by Og Mandino

Text Books

1. English All Round: Communication Skills for Undergraduate Learners-Volume 1, Orient Black Swan, 2019 (to be released)
2. University of Success by Og Mandino, Jaico, 2015.

Reference Books

1. Bailey, Stephen. Academic writing: A handbook for international students. Routledge, 2014.
2. Chase, Becky Tarver. Pathways: Listening, Speaking and Critical Thinking. Heinley ELT; 2nd Edition, 2018.
3. Skilful Level 2 Reading & Writing Student's Book Pack (B1) Macmillan Educational.
4. Hewings, Martin. Cambridge Academic English (B2). CUP, 2012.

AICTE Recommended Books

1. Meenakshi Raman and Sangeeta Sharma. Technical Communication. Oxford University Press, 2018.
2. Pushplata and Sanjay Kumar. Communication Skills, Oxford University Press, 2018
3. Kulbushan Kumar. Effective Communication Skills. Khanna Publishing House, Delhi

Sample Web Resources

Grammar / Listening / Writing 1-language.com http://www.5minuteenglish.com/ https://www.englishpractice.com/ Grammar/Vocabulary English Language Learning Online http://www.bbc.co.uk/learningenglish/ http://www.better-english.com/ http://www.nonstopenglish.com/ https://www.vocabulary.com/ BBC Vocabulary Games Free Rice Vocabulary Game	Reading: https://www.usingenglish.com/comprehension/ https://www.englishclub.com/reading/short stories.htm https://www.english-online.at/Listening https://learningenglish.voanews.com/z/3613 http://www.englishmedialab.com/listening.html Speaking https://www.talkenglish.com/ BBC Learning English – Pronunciation tips Merriam-Webster – Perfect pronunciation Exercises
All Skills https://www.englishclub.com/ http://www.world-english.org/ http://learnenglish.britishcouncil.org/	

CO-PO MAPPING

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1									2	3		1
CO2									2	3		1
CO3									2	3		1

I Year – I SEMESTER

L	T	P	C
0	0	3	1.5

APPLIED CHEMISTRY LAB

Introduction to chemistry laboratory – Molarity, Normality, Primary, Secondary standard solutions, Volumetric titrations quantitative analysis.

Course objectives:

Students will be able to:

- estimate and assess the water quality parameters like hardness, p^H, conductance and turbidity
- Practice instrumental titrimetric analysis
- Estimate some important drugs like Vitamin-C
- Learn the preparation of some important polymers like Bakelite, Nylon-6,6 etc...
- Perform spectroscopic estimations by using UV- Visible spectrophotometry

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

1. Estimate the amount of metal ions present in different solutions (L5)
2. Analyze the quality parameters of water (L4)
3. Determine the strength of different solutions by using different instrumentation techniques (L5)

1. Determination of HCl using standard Na_2CO_3 solution
2. Determination of alkalinity of a sample containing Na_2CO_3 and NaOH
3. Determination of Mn (VII) using standard oxalic acid solution
4. Determination of ferrous iron using standard $K_2Cr_2O_7$ solution
5. Determination of Copper (II) using standard EDTA solution
6. Determination of temporary and permanent hardness of water using standard EDTA solution
7. Determination of Iron (III) by colorimetric method
8. Determination of the concentration of acetic acid using sodium hydroxide (pH-metric method)
9. Determination of concentration of strong acid vs strong base (by conductometric method)
10. Determination of strong acid vs strong base (by potentiometric method)
11. Determination of Mg^{+2} present in an antacid
12. Determination of $CaCO_3$ present in an egg shell
13. Estimation of vitamin- C
14. Determination of phosphate content in soft drinks
15. Adsorption of acetic acid by charcoal
16. Preparation of Nylon-6, 6 and Bakelite (demonstration only)

Note: Choice of any 10 experiments from the above.

Reference Books:

A Text Book of Quantitative Analysis, Arthur J. Vogel.

I Year – I SEMESTER

L	T	P	C
0	0	3	1.5

PROBLEM SOLVING USING C LAB

(Common to All Branches)

Course Objectives:

1. Apply the principles of C language in problem solving.
2. To design flowcharts, algorithms and knowing how to debug programs.
3. To design & develop of C programs using arrays, strings pointers & functions.
4. To review the file operations, pre-processor commands.

Exercise - 1 Control Flow - I

a) Write a C Program to Find Whether the Given Year is a Leap Year or not.

b) Write a C Program to find second biggest of three numbers (Assume that all the numbers are unique).

Exercise – 2 Control Flow - II

b) Write a C Program to Find Whether the Given Number is

i) Prime Number

ii) Armstrong Number

Exercise – 3 Control Flow - III

a) Write a C program to print Floyd Triangle

b) Write a C Program to print Pascal Triangle

c) Write a C program to display a Pyramid

Exercise – 4 Arrays - Demonstration of arrays

a) Search-Linear.

b) Sorting-Bubble

c) Operations on Matrix. - Add, Subtract, Multiply

Exercise – 5 Strings

a) Implementation of string manipulation operations **with** library function: Copy, length, compare

b) Implementation of string manipulation operations **without** library function: copy, length, compare

Exercise – 6 Functions

a) Write a C Program demonstrating of parameter passing in Functions and returning values.

b) Write a C Program illustrating Fibonacci, Factorial with Recursion without Recursion

Exercise – 7 Functions - Continued

Write a C Program to compute the values of sin x and cos x and ex values using Series expansion. (Use factorial function)

Exercise - 8 Arrays, Strings and Pointers

a) Write a C Program to find min and max of an array of elements using pointers

b) Write a C Program to concatenate one string to another using pointer.

Exercise – 9 Dynamic Memory Allocations

Write a C program to represent 1D and 2D arrays using malloc () function.

Exercises - 10 Structures

a) Write a C Program to Store Information of a Movie Using Structure

b) Write a C Program to sort a set of student records in ascending order.

c) Write a C Program to Add, subtract & multiply Two Complex Numbers.

Exercise -11 Files

a) Write a C programming code to open a file and to print it contents on screen.

b) Write a C program to copy the content of one file to another.

C) Write a C program merges two files and stores their contents in another file

Course Outcomes: By the end of the Lab, the student able to

4. **Comprehend** the various concepts of a C language
5. **Develop** algorithms and flowcharts
6. **Design** and development of C problem solving skills.
7. **Acquire** modular programming skills.

Correlation of Course Outcomes with Program Outcomes and Program Specific Outcomes

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

****

I Year – I SEMESTER

L	T	P	C
0	0	3	1.5

ENGINEERING WORK SHOP
(Common to CE, CSE & IT)

Course Objective: To familiarize students with wood working, sheet metal operations, fitting and electrical house wiring skills

Wood Working: Familiarity with different types of woods and tools used in wood working and make following joints

a) Half – Lap joint
b) Dovetail joint
c) Bridle joint

Sheet Metal Working: Familiarity with different types of tools used in sheet metal working, Developments of following sheet metal job from GI sheets

a) Tapered tray b) Conical funnel c) Elbow pipe d) Brazing

Fitting: Familiarity with different types of tools used in fitting and do the following fitting exercises

a) V-fit b) Dovetail fit c) square fit d) Semi-circular e) Two Wheeler tyre puncture and change of two wheeler tyre

Electrical Wiring: Familiarities with different types of basic electrical circuits and make the following connections

a) Parallel and series b) Two-way switch c) Godown lighting d) Tube light

e) Three phase motor f) Soldering of wires

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

1. Apply wood working skills in real world applications. (L3)
2. Build different parts with metal sheets in real world applications. (L3)
3. Apply fitting operations in various applications. (L3)
4. Apply different types of basic electric circuit connections. (L3)
5. Demonstrate soldering and brazing. (L2)

CO-PO MATRIX:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	2	-	-	-	-	-	-	-	-	-	1
CO2	2	2	-	-	-	-	-	-	-	-	-	1
CO3	2	2	-	-	-	-	-	-	-	-	-	1
CO4	2	2	-	-	-	-	-	-	-	-	-	1
CO5	2	2	-	-	-	-	-	-	-	-	-	1

CO-PSO Matrix:

	PSO1	PSO2
CO1	2	2
CO2	2	2
CO3	2	2
CO4	2	2
CO5	2	2

I Year – I SEMESTER

L	T	P	C
3	0	0	0

ENVIRONMENTAL SCIENCE
(Common to CE, CSE & IT)

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

UNIT – I: MULTIDISCIPLINARY NATURE OF ENVIRONMENTAL STUDIES
Definition, Scope and Importance – Need for Public Awareness.

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

LEARNING OUTCOMES

Students will be able to

1. Articulate the basic structure, functions, and processes of key social systems affecting the environment.
2. Explain how water resources should be used.
3. Articulate basic understanding of effects of modern agriculture on environment.
4. Explain how various paradigms or world views and their implicit and explicit assumptions and values shape the viewer's perception of environmental problems and solutions.

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

a. Forest ecosystem.
b. Grassland ecosystem
c. Desert ecosystem
d. Aquatic ecosystems (ponds, streams, lakes, rivers, oceans, estuaries)

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

LEARNING OUTCOMES

Students will be able to

1. Get a clear picture of structure and functions of ecosystems.
2. Explain why renewable and non-renewable energy resources are important.
3. Get awareness about land degradation, soil erosion & desertification.
4. Gain a rigorous foundation in various scientific disciplines as they apply to environmental science, such as ecology, evolutionary biology, hydrology, and human behaviour.

UNIT – III: Environmental Pollution and Solid Waste Management

ENVIRONMENTAL POLLUTION: Definition, Cause, effects and control measures of :

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

SOLID WASTE MANAGEMENT: Causes, effects and control measures of urban and industrial wastes – Role of an individual in prevention of pollution – Pollution case studies – Disaster management: floods, earthquake, cyclone and landslides.

LEARNING OUTCOMES UNIT-3

Students will be able to

1. Demonstrate knowledge and understanding of theories in the field of Biodiversity and Systematics in the broad sense.
2. Conduct basic conservation biology research.
3. Explainendangered and endemic species of India.
4. Identify the threats to biodiversity.

UNIT – IV: Social Issues and the Environment

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

LEARNING OUTCOMES:

Students will be able to

1. Understand Cause, effects and control measures of air pollution.
2. Understand soil, noise & water pollution.
3. explain the enforcement of Environmental legislation
4. Understand solid waste management.

UNIT – V: Human Population and the Environment

HUMAN POPULATION AND THE ENVIRONMENT: Population growth, variation among nations. Population explosion – Family Welfare Programmed. – Environment and human health

– Human Rights – Value Education – HIV/AIDS – Women and Child Welfare – Role of information Technology in Environment and human health – Case studies.

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

LEARNING OUTCOMES

Students will have

1. Knowledge about watershed management and environmental ethics.
2. Explain the reasons for global warming
3. Explain principles and impact of disasters on environment.
4. Explain disaster management cycle in India.

TEXT BOOKS:

1. Text book of Environmental Studies for Undergraduate Courses by Erach Bharucha for University Grants Commission, Universities Press.
2. Environmental Studies by Palaniswamy – Pearson education
3. Environmental Studies by Dr.S.Azeem Unnisa, Academic Publishing Company

REFERENCES:

1. Textbook of Environmental Science by Deeksha Dave and E.Sai Baba Reddy, Cengage Publications.
2. Text book of Environmental Sciences and Technology by M.Anji Reddy, BS Publication.
3. Comprehensive Environmental studies by J.P.Sharma, Laxmi publications.
4. Environmental sciences and engineering – J. Glynn Henry and Gary W. Heinke – Prentice hall of India Private limited.
5. A Text Book of Environmental Studies by G.R.Chatwal, Himalaya Publishing House
6. Introduction to Environmental engineering and science by Gilbert M. Masters and Wendell P. Ela - Prentice hall of India Private limited.

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

COURSE OUTCOMES

CO1	Able to U**nderstand** The concepts of the ecosystem
CO2	Able to U**nderstand** The natural resources and their importance
CO3	Able to learn The biodiversity of India and the threats to biodiversity ,and **Apply** conservation practices
CO4	Able to learn Various attributes of the pollution and their impacts
CO5	Able to U**nderstand** Social issues both rural and urban environment
CO6	Able to U**nderstand** About environmental Impact assessment and **Evaluate** the stages involved in EIA

*****

I Year – II SEMESTER

L	T	P	C
3	0	0	3

MATHEMATICS-II
(Common to All)

Course Objectives:

- To elucidate the different numerical methods to solve nonlinear algebraic equations
- To disseminate the use of different numerical techniques for carrying out numerical integration
- To equip the students with standard concepts and tools at an intermediate to advanced level mathematics to develop the confidence and ability among the students to handle various real-world problems and their applications

UNIT-1: Iterative methods: (10 hrs)

Introduction–Bisection method–Method of false position–Iteration method–Newton-Raphson method (one variable)–Jacobi and Gauss-Seidel methods for solving system of equations.

UNIT-2: Interpolation: (12 hrs)

Introduction–Errors in polynomial interpolation–Finite differences–Forward differences–Backward differences–Central differences –Relations between operators–Newton's forward and backward formulae for interpolation–Gauss's forward and backward formulae for

Interpolation – Interpolation with unequal intervals–Lagrange's interpolation formula–Newton's divide difference formula.

UNIT-3: Numerical integration and solution of ordinary difference equations: (10 hrs)

Trapezoidal rule–Simpson's $1/3^{rd}$ and $3/8^{th}$ rule–Solution of ordinary differential equations by Taylor's series–Picard's method of successive approximations–Euler's method–Modified Euler's method–Runge-Kutta method (second and fourth order).

UNIT–4: Laplace Transforms: (14 hrs)

Laplace transforms of standard functions – Shifting theorems – Transforms of derivatives and integrals – Unit step function – Dirac's delta function –Periodic function - Inverse Laplace transforms – Convolution theorem (without proof)

Applications: Evaluation of integrals using Laplace transforms - Solving ordinary differential equations (Initial value problems) using Laplace transforms.

UNIT 5: Fourier series and Fourier Transforms: (14 hrs)

Fourier series: Introduction – Periodic functions – Fourier series of periodic function – Dirichlet's conditions – Even and odd functions – Change of interval – Half-range sine and cosine series.

Fourier Transforms: Fourier integral theorem (without proof) - Fourier sine and cosine integrals – Sine and cosine transforms – Properties – Inverse transforms – Finite Fourier transforms.

Text Books:

1. **B.S. Grewal,** Higher Engineering Mathematics, 44^{th} Edition, Khanna Publishers.

Reference Books:

3. **B.V. Ramana,** Higher Engineering Mathematics, 2007 Edition, Tata Mc. Graw Hill Education.
4. **H.K.Das,** Advanced Engineering Mathematics, 22nd Edition, S. Chand & Company Ltd.
5. **Erwin Kreyszig**, Advanced Engineering Mathematics, 10th Edition, Wiley-India.

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

- Evaluate approximate in the roots of polynomial and transcendental equations by different algorithms (EVALUATE)
- Solve system of linear algebraic equations using Gauss Jacobi, Gauss Seidel and apply Newton's forward and backward interpolation and Lagrange's formulae for equal and unequal intervals (SOLVE , APPLY,FIND)
- Apply different algorithms for approximating the solutions of ordinary differential equations to its analytical computations and also by Laplace the transforms for solving differential equations (SOLVE , APPLY,FIND)
- Find or compute the Fourier series of periodic signals (SOLVE ,APPLY, FIND, ANALYSE)
- Know and be able to apply integral expressions for the forwards and inverse Fourier transform to range of non-periodic waveforms (SOLVE , APPLY, FIND)

Micro-Syllabus of MATHEMATICS-II

UNIT-1: Iterative methods: Introduction–Bisection method–Method of false position–Iteration method–Newton-Raphson method (one variable)–Jacobi and Gauss-Seidel methods for solving system of equations.

Unit	Module	Micro content
1a. Solving given polynomial	Numerical solution of algebraic and transcendental polynomials	Bisection method
		Method of false position
		Iteration method
		Newton-Raphson's method
1b Solving linear system	Solving linear system	Jacobi's method
		Gauss-seidel method

UNIT-2 : Interpolation: Introduction–Errors in polynomial interpolation–Finite differences–Forward differences–Backward differences–Central differences –Relations between operators–Newton's forward and backward formulae for interpolation–Gauss's forward and backward formulae for
Interpolation – Interpolation with unequal intervals–Lagrange's interpolation formula–Newton's divide difference formula.

Unit	Module	Micro content
2a. Equal-Spaced difference tables	Finite difference tables	Forward, backward & central difference tables
		Errors in polynomials
	Finding functional values for given data	Newton's forward and backward difference interpolation formula
		Gauss forward and backward difference interpolation formula

2b. Unequal spaced data & relation between various operators	Unequal spaced data & relation between various operators	Lagrange's interpolation formula
		Relation between various operators (Shift, forward, backward, central, average & differential operators)
UNIT-3: Numerical integration and solution of ordinary difference equations: Trapezoidal rule–Simpson's $1/3^{rd}$ and $3/8^{th}$ rule–Solution of ordinary differential equations by Taylor's series–Picard's method of successive approximations–Euler's method–Modified Euler's method–Runge-Kutta method (second and fourth order).		
Unit	**Module**	**Micro content**
3a. Numerical integration	Numerical Integration	Trapezoidal rule
		Simpson's $1/3^{rd}$ rule
		Simpson's $3/8^{th}$
3b. Numerical solution of ordinary differential equations for single variable	Numerical solution of ordinary differential equations for single variable	Taylors series method
		Picard's method
		Euler's method
		Modified Euler's method
UNIT – 4: Laplace Transforms: Laplace transforms of standard functions – Shifting theorems – Transforms of derivatives and integrals – Unit step function – Dirac's delta function –Periodic function - Inverse Laplace transforms – Convolution theorem (without proof) Applications: Evaluation of integrals using Laplace transforms - Solving ordinary differential equations (Initial value problems) using Laplace transforms.		
Unit	**Module**	**Micro content**
4a Laplace Transforms	Laplace transforms and theorem	Shifting theorems
		Derivatives and integrals
		Multiplication and division
4b. Inverse Laplace transforms and Applications	Periodic functions &Inverse Laplace Transforms	Periodic functions
		Dirac delta functions
		Evaluation integrals using Laplace Transforms
		Solving differential equations using Laplace transforms
UNIT 5: Fourier series and Fourier Transforms: Fourier series: Introduction – Periodic functions – Fourier series of periodic function – Dirichlet's conditions – Even and odd functions – Change of interval – Half-range sine and cosine series. Fourier Transforms: Fourier integral theorem (without proof) - Fourier sine and cosine integrals – Sine and cosine transforms – Properties – Inverse transforms – Finite Fourier transforms.		
Unit	**Module**	**Micro content**
5a. Fourier Series	Fourier Series	Periodic functions
		Dirichlet's conditions

		Even and odd function's
		Change of interval
		Half range sine and cosine series
5b. Fourier Transforms	Fourier Transforms	Fourier Sine and Cosine integral
		Properties of Fourier Transforms
		Fourier and Inverse Fourier Transforms
		Fourier cosine and Inverse Fourier cosine Transforms
		Fourier sine and Inverse Fourier sine Transforms
		Finite Fourier Transforms
		Inverse Finite Fourier Transforms

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2										1
CO2	3	2										1
CO3	3	2										1
CO4	3	2										1
CO5	3	2										1

****

I Year – II SEMESTER

L	T	P	C
3	0	0	3

MATHEMATICS – III
(Common to ALL branches)

Course Objectives:

1. To instruct the concept of Matrices in solving linear algebraic equations
2. To familiarize the techniques in partial differential equations
3. To furnish the learners with basic concepts and techniques at plus two level to lead them into advanced level by handling various real-world applications

UNIT-I:Solving system of linear equations, Eigen values and Eigen Vectors (12 hrs)

Rank of a matrix by Echelon form and normal form–solving system of homogeneous and non-homogeneous linear equations–Gauss elimination, Gauss Jordan for solving system of equations- Eigen values and Eigen vectors and their properties

UNIT-II: Cayley-Hamilton theorem and quadratic forms: (12 hrs)

Cayley-Hamilton theorem (without proof)–Finding inverse and power of a matrix by Cayley-Hamilton theorem–Reduction to Diagonal form–Quadratic forms and nature of the quadratic forms–Reduction of quadratic form to canonical forms by orthogonal transformation.

Application: Free vibration of two mass systems.

UNIT – III: Vector Differentiation: (10 hrs)

Scalar and Vector point functions-Vector Differential operator- Gradient – Directional derivatives– Divergence – Curl – Laplacian second order operator- Vector identities- Scalar Potential.

UNIT– IV: Vector Integration: (12 hrs)

Line integral – Work done – Circulation- Surface integral- Volume integral Vector integral theorems (without proof): Greens theorem in a plane- Stokes theorem- Gauss Divergence theorem.

UNIT– V: Solutions of Partial differential Equations (14 hrs)

Formation of partial differential equations by elimination of arbitrary constants and arbitrary functions – Solutions of first order linear (Lagrange) equation and nonlinear (standard types) equations.

Second order PDE: Solutions of linear partial differential equations with constant coefficients RHS term of the type $e^{ax+by}, \sin(ax+by), \cos(ax+by), x^m y^n$.

Text Books:

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

Reference Books:

1. **B.V. Ramana,** Higher Engineering Mathematics, 2007 Edition, Tata Mc. Graw Hill Education.
2. **H.K.Das,** Advanced Engineering Mathematics, 22nd Edition, S. Chand & Company Ltd.
3. **Erwin Kreyszig**, Advanced Engineering Mathematics, 10th Edition, Wiley-India.

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

- develop the use of matrix algebra techniques that is needed by engineers for practical applications (L6)
- solve system of linear algebraic equations using Gauss elimination, Gauss Jordan (L3)
- to interpret the physical meaning of different operators such as gradient, curl and divergence (L5)
- estimate the work done against a field, circulation and flux using vector calculus (L5)
- identify the solution methods for partial differential equation that model physical processes (L3)

Micro-Syllabus of MATHEMATICS – III

UNIT-I: Solving system of linear equations, Eigen values and Eigen Vectors

Rank of a matrix by Echelon form and normal form–solving system of homogeneous and non-homogeneous linear equations–Gauss elimination, Gauss Jordan for solving system of equations- Eigen values and Eigen vectors and their properties

Unit	Module	Micro content
1a. Solving system of linear equations	Rank of the given matrix	Find rank of the given matrix by reducing into Echelon form.
		Find rank of the given matrix by reducing into Normal form.(Canonical form)
	System of linear equations	Solve the system of homogeneous linear equations.
		Solve the system of Non- homogeneous linear equations.
		Solve the given system of linear equations using Gauss Elimination method.
		Solve the given system of linear equations using Gauss Jordan method.
1b. Applications	Eigen values and Eigen vectors	Find eigen values and Eigen vectors of given matrix.
	Properties of Eigen values and Eigen vectors	If λ is an eigen value of Matrix A then find eigen values of A^m or A^{-1} or $B = A^2+k_1A+K_2I$ or ….
		The eigen vectors corresponding to distinct eigen values of real symmetric matrix are orthogonal.

UNIT-II: Cayley-Hamilton theorem and quadratic forms:

Cayley-Hamilton theorem (without proof)–Finding inverse and power of a matrix by Cayley-Hamilton theorem–Reduction to Diagonal form–Quadratic forms and nature of the quadratic forms–Reduction of quadratic form to canonical forms by orthogonal transformation.

Unit	Module	Micro content
	Cayley-Hamilton theorem	Verify Cayley-Hamilton theorem for given matrix A and hence find A^{-1} or A^4.
	Quadratic Forms	Reduce the given matrix into diagonal form.
		Reduce the quadratic form into canonical form using orthogonal transformation method.

UNIT – III: Vector Differentiation:

Scalar and Vector point functions-Vector Differential operator- Gradient – Directional derivatives Divergence – Curl – Laplacian second order operator- Vector identities- Scalar Potential.

Unit	Module	Micro content
3a. Vector Differential operator	Divergent, Curl and Gradient	Find Gradient of given scalar function.
		Find Unit normal vector at given point on given surface.
		Find divergent or Curl of given vector function.
3b. Vector identities	Vector identities	Find Scalar potential function.
		Problems on Laplacian second order operator.
		Prove the given vector identity.

UNIT– IV: Vector Integration:
Line integral – Work done – Circulation- Surface integral- Volume integral Vector integral theorems (without proof): Greens theorem in a plane- Stokes theorem- Gauss Divergence theorem.

Unit	Module	Micro content
4a. Vector integration	Line integration, surface integration & volume integration	Evaluate given line integration along the given curve.
		Find work done by force in moving a particle from A to B along curve C.
		Find surface integral of vector function.
		Find volume integral of vector function.
4b. Vector integration theorems	Green's theorem, Stoke's theorem and Gauss Divergence theorem.	Verify Green's theorem.
		Evaluate using stoke's theorem.
		Evaluate using Divergence theorem.

UNIT– V: Solutions of Partial differential Equations: Formation of partial differential equations by elimination of arbitrary constants and arbitrary functions – Solutions of first order linear (Lagrange) equation and nonlinear (standard types) equations.
Second order PDE: Solutions of linear partial differential equations with constant coefficients – RHS term of the type e^{ax+by},$\sin(ax+by)$,$\cos(ax+by)$,$x^m y^n$.

Unit	Module	Micro content
5a. First order PDE	Formation of PDE	Form PDE by eliminating arbitrary constants.
		Form PDE by eliminating arbitrary functions.
	Solve First order PDE	Solve first order linear PDE.
		Solve first order nonlinear PDE.
5b. Higher order PDE	Solve Second order PDE.	Solve Second order linear PDE with constant coefficients with RHS terms e^{ax+by},$\sin(ax+by)$,$\cos(ax+by)$,$x^m y^n$.

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2										1
CO2	3	2										1
CO3	3	2										1
CO4	3	2										1
CO5	3	2										1

****

I Year – II SEMESTER

L	T	P	C
3	0	0	3

APPLIED PHYSICS

Course Objectives:

Applied Physics curriculum which is re-oriented to the needs of Circuital branches of graduate engineering courses offered by Vasireddy Venkatadri Institute of Technology, which serves as a transit to understand the branch specific advanced topics. The course is designed to:

CO1. Impart Knowledge of Physical Opticsphenomena like Interference and Diffraction required to design instruments with higher resolution.

CO2. Understand the physics of Semiconductors and their working mechanism for their utility in electronic devices.

CO3. Impart the knowledge of materials with characteristic utility in appliances.

Course Outcomes:

The students will be able to

1. Understand the principles such as interference and diffraction to design and enhance the resolving power of various optical instruments.
2. Learn the basic concepts of LASER light Sources and Apply them to holography
3. Study the magnetic and dielectric materials to enhance the utility aspects of materials.
4. Learn the fundamental concepts of Quantum behavior of matter.
5. Identify the type of semiconductors using Hall Effect

Unit - I

Wave Optics (10 hrs)

Interference: Principle of Superposition – Interference of light – Conditions for sustained Interference – Interference in thin films (reflected geometry) - Newton's Rings (reflected geometry).

Diffraction: Fraunhofer Diffraction:- Diffraction due to single slit (quantitative), double slit (qualitative), N –slits (qualitative) and circular aperture (qualitative) – Intensity distribution curves - Diffraction grating – Grating spectrum – missing order– resolving power – Rayleigh's criterion – Resolving powers of Microscope (qualitative), Telescope (qualitative) and grating (qualitative).

Unit Outcomes

The students will be able to

- Explain the need of coherent sources and the conditions for sustained interference.
- Analyze the differences between interference and diffraction with applications.
- Illustrate the resolving power of various optical instruments.

Unit - II

LASERs and Holography (8 hrs)

LASERs: Interaction of radiation with matter – Spontaneous and Stimulated emission of radiation – population inversion – Einstein's coefficients & Relation between them and their significance - Pumping Mechanisms - Ruby laser – Helium-Neon laser – Applications.

Holography: Introduction – principle – differences between photography and holography – construction and reconstruction of hologram – applications of holograms

Unit Outcomes

The students will be able to

- Understand the basic concepts of LASER light Sources

- Study different types of laser systems
- Apply the concept of Lasers and interference to construct Holograms

Unit - III

Magnetism and Dielectrics **(12 hrs)**

Magnetism: Introduction - Magnetic dipole moment - Magnetization-Magnetic susceptibility and permeability- Origin of permanent magnetic moment - Bohr magneton-Classification of magnetic materials: Dia, para & Ferro – Domain concept of Ferromagnetism - Hysteresis – soft and hard magnetic materials – applications of Ferromagnetic material.

Dielectrics: Introduction- Dielectric polarization – Dielectric polarizability, Susceptibility and Dielectricconstant- Types of polarizations: Electronic (Quantitative) and Ionic (Quantitative), Orientation Polarizations (Qualitative) - Lorentz Internal field-Claussius –Mossotti's equation- Frequency dependence of polarization - Applications of dielectrics.

Unit Outcomes

The students will be able to

- Classify the magnetic materials based on susceptibility and their temperature dependence.
- Explain the applications of dielectric and magnetic materials.
- Apply the concept of magnetism to magnetic devices.
- Explain the concept of polarization in dielectric materials.
- Summarize various types of polarization of dielectrics.
- Interpret Lorentz field and Claussius- Mossotti's relation in dielectrics.

Unit - IV

Quantum **(8 hrs)**

Introduction– matter waves – de Broglie's hypothesis – Davisson - Germer experiment – G.P.Thomson experiment – Heisenberg's Uncertainty Principle– Schrödinger time independent and time dependent wave equations – physical significance of Schrödinger wave function – Particle in a potential box (determination of energy).

Unit Outcomes:

The students will be able to

- Explain the fundamental concepts of quantum mechanics.
- Analyze the physical significance of wave function.
- Apply Schrödinger's wave equation for energy values of a free particle

Unit - V

Semiconductor Physics **(10 hrs)**

Origin of energy bands (qualitative) -Classificationofsolidsbasedonenergybands– Intrinsicsemiconductors-densityof charge carriers –Electrical conductivity-Fermi level – extrinsic semiconductors – P–type & N-type –Density of charge carriers- Dependence of Fermi-energy on carrier concentration and temperature- Hall effect – Hall coefficient- Applications of Hall effect- Drift and Diffusion currents - Einstein's equation.

Unit Outcomes:

The students will be able to

- Classify the energy bands of semiconductors.
- Outline the properties of n-type and p-type semiconductors.
- Identify the type of semiconductor using Hall Effect.

Text Books

1. *Engineering Physics* by B. K. Pandey, S. Chaturvedi,Cengage Publications, 2012.

2. *A Text book of Engineering Physics* by M.N. Avadhanulu, P.G.Kshirsagar, S.Chand, 2017.
3. *Engineering Physics* by D.K.Bhattacharya and Poonam Tandon, Oxford press, 2015.
4. *Engineering Physics* by R.K. Gaur and S.L. Gupta, Dhanpat Rai publishers, 2012.

Reference Books

1. *Engineering Physics* by M.R.Srinivasan, New Age international publishers, 2009.
2. *Optics* by Ajoy Ghatak, 6th Edition McGraw Hill Education, 2017.
3. *Solid State Physics* by A.J.Dekker, McMillan Publishers, 2011.

***

I Year – II SEMESTER

L	T	P	C
3	0	0	3

PYTHON PROGRAMMING

Course Objectives:

1. Introduction to Scripting Language
2. Use various data handling mechanisms
3. Exposure to various problems solving approaches of computer science

Course Outcomes:

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

- Understand the need and the Jargon of Python language
- Experiment with various Data structures in interpreted Language.
- Build modules and packages for real software needs.
- Implement object oriented principles in Python
- Identify solutions using GUI and testing mechanisms.

UNIT – I

Introduction **(8 hrs)**

History of Python, Need of Python Programming, differences between C and Python, Applications Basics of Python Programming Using the REPL(Shell), Running Python Scripts, Variables, Assignment, Keywords, Input-Output, Indentation.

Types, Operators and Expressions: Types - Integers, Strings, Booleans; Operators- Arithmetic Operators, Comparison (Relational) Operators, Assignment Operators, Logical Operators, Bitwise Operators, Membership Operators, Identity Operators, Expressions.

UNIT – II

Flow Control & Data Structures **(14 hrs)**

Control Flow - order of evaluations Control Flow- if, if-elseif, for, while, break, continue, pass

Data Structures- Lists - Operations, Slicing, Methods; Tuples, Sets, Dictionaries, Sequences. Comprehensions.

UNIT – III

Modules & Packages **(10 hrs)**

Functions - Defining Functions, Calling Functions, Passing Arguments, Keyword Arguments, Default Arguments, Variable-length arguments, Anonymous Functions, Fruitful Functions (Function Returning Values), Scope of the Variables in a Function - Global and Local Variables.

Modules: Creating modules, import statement, from. Import statement, name spacing.

Python packages, Introduction to PIP, Installing Packages via PIP, Using Python Packages

UNIT – IV

OOPs **(12 hrs)**

Object Oriented Programming in Python: Definition, advantages of OOPs, OOPs principles, Classes, 'self-variable', Methods, Constructor Method, Inheritance, Overriding Methods, and Data hiding.

Error and Exceptions: Difference between an error and Exception, Handling Exception, try except block, Raising Exceptions, User Defined Exceptions

UNIT – V

STL **(8 hrs)**

Brief Tour of the Standard Library - Operating System Interface - String Pattern Matching, Mathematics, Internet Access, Dates and Times, Data Compression, Multithreading, GUI Programming, Turtle Graphics

Testing: Why testing is required ?, Basic concepts of testing, Unit testing in Python, Writing Test cases, Running Tests.

Text Books

(2) *Python Programming: A Modern Approach*, Vamsi Kurama, Pearson
(3) *Learning Python*, Mark Lutz, Orielly

Reference Books

5. *Think Python*, Allen Downey, Green Tea Press
6. *Core Python Programming*, W.Chun, Pearson.
7. *Introduction to Python*, Kenneth A. Lambert, Cengage.

***

I Year – II SEMESTER

L	T	P	C
3	0	0	3

BASIC ELECTRICAL & ELECTRONICS ENGINEERING

Course Objectives:

- To introduce basics of electric circuits and to teach DC and AC electrical circuit analysis.
- To explain the working principles DC machines and speed control of various DC motors.
- To explain the working principles of transformers and AC machines and its applications.
- To introduce the basics of semiconductor physics and operation and applications of Diodes.
- To introduce the basics of transistors and explain the transistor configurations and OPAMPs

Course Outcomes:

- Apply concepts of KVL/KCL in solving DC circuits. (L3)
- Choose correct machine for a specific application. (L5)
- Illustrate working principles of DC and AC Machines. (L3)
- Describe working principles of diodes and transistors. (L2)
- Understand the applications of diodes and transistors. (L2)

Unit 1 - DC & AC Circuits

DC Circuits: Electrical circuit elements (R - L and C) – Kirchhoff's laws - Voltage and Current division rules-series, parallel circuits and star-delta and delta-star transformations- [Elementary treatment only]

AC Circuits: Representation of sinusoidal waveforms - Peak and RMS values - phasor representation - real power - reactive power - apparent power - power factor [Elementary treatment only]

Unit 2 - DC Machines

DC Generator: Construction-Principle and operation of DC Generator - EMF equation - Types– Applications [Elementary treatment only]

DC Motor: Principle and operation of DC Motor – types-Torque equation - Speed control of DC Motor-Brake test- Swinburne's test-Applications. [Elementary treatment only]

Unit 3 - AC Machines

Single Phase Transformer: Construction, Principle and operation of Single Phase Transformer –EMF Equation-Losses-Efficiency [Elementary treatment only]

Three Phase Induction Motor: Construction- Principle and operation of three phase Induction Motor-concept of slip-Running Torque, Starting Torque, Maximum Torque relationships- Applications. [Elementary treatment only]

Single Phase Induction Motor: Construction -Principle and operation of single phase Induction Motor-Applications.[Elementary treatment only]

Unit 4 - Semiconductor Devices

Semiconductor Physics, PN Junction Diode & Zener Diode-characteristics- Applications: Rectifiers, Voltage multipliers, Clippers and Clampers. [Elementary treatment only]

Unit 5 - Bipolar Junction Transistors

Construction and working of bipolar junction transistor, CB, CE and CC Configurations and characteristics, basic concepts of amplifiers, Operational amplifiers [Elementary treatment only]

Text Books

1. D. P. Kothari and I. J. Nagrath.*Basic Electrical Engineering.* Tata McGraw Hill, 2010.
2. R. L. Boylestad and Louis Nashelsky.*Electronic Devices and Circuits.* 9th edition, PEI/PHI 2006.

References

1. L. S. Bobrow. *Fundamentals of Electrical Engineering*. Oxford University Press, 2011.
2. E. Hughes.*Electrical and Electronics Technology*. Pearson, 2010.

***

ENGINEERING GRAPHICS AND DESIGN

Course Objectives:

1. Expose the students to use Drafting packages for generating Engineering curves and conventions Followed in
2. Preparation of engineering drawings.
3. Make the students to understand the concepts of orthographic projections of Lines & Plane Surfaces
4. To understand the concepts of orthographic projections of Regular Solids.
5. Develop the ability of understanding sectional views & Development of Solid Surfaces.
6. Enable them to use computer aided drafting packages for Conversion of Isometric view to Orthographic Projection and vice versa.

Course Outcomes:

After successful completion of the course, the students are able to

CO-1: Prepare engineering drawings as per BIS conventions. (L2)

CO-2: Produce computer generated of orthographic projections of Lines and Plane Surfaces using CAD software. (L2)

CO-3: Use the knowledge of orthographic projections of Solids to represent engineering information / concepts and present the same in the form of drawings. (L2)

CO-4: Use the knowledge of sectional views and Development of Solid Surfaces in Real time Applications. (L3)

CO-5: Develop isometric drawings of simple objects reading the orthographic projections of those objects. (L3)

Unit 1 - Introduction to AutoCAD

Basic commands, Customization, ISO and ANSI standards for coordinate dimensioning, Annotations, layering, 2D drawings of various mechanical components, 2D drawings of various electrical and electronic circuits. Creation of engineering models- floor plans that include: windows, doors, and fixtures such as WC, bath, sink, shower, etc. Applying colour coding according to building drawing practice; (Experiments should be Planned According to respective Core Branch Applications)

Unit 2 - Theory of Projection (On Grid paper)

Principles of Orthographic Projections-Convention: Projections of Points, Projections of Lines inclined to both planes, Projections of planes inclined to one Plane & Projections of planes inclined to both Planes

Unit 3 - Projections of Regular Solids (Auto CAD)

Projections of Solids –with the axis perpendicular to one of the principal planes, with the axis Inclined to one of the principal planes, Projections of Solids –with the axis Inclined to Both the principal planes

Unit 4 - Development of Surfaces & Sectional Orthographic Views (AutoCAD)

Development of surfaces of Right Regular Solids – Prism, Pyramid, Cylinder and, Cone. Draw the sectional orthographic views of geometrical solids

Unit 5

Isometric Projections (On Isometric Grid paper)

Conversion of isometric views to orthographic views, drawing of isometric views – simple Solids, Conversion of orthographic views to isometric views of simple Drawings

TEXT BOOKS

1. *Engineering Drawing* by N.D. Bhat, Chariot Publications
2. *Engineering Graphicswith Autocad* by Kulkarni D.M , PHI Publishers
3. *Engineering Drawing + AutoCad*by K. Venugopal, V. Prabhu Raja, New Age
4. *Engineering Drawing* by Agarwal & Agarwal, Tata McGraw Hill Publishers

REFERENCE BOOKS

1. *Engineering Drawing* by K.L.Narayana& P. Kannaiah, Scitech Publishers
2. *Engineering Graphics for Degree* by K.C. John, PHI Publishers
3. *Engineering Graphics* by P. I. Varghese, McGrawHill Publishers
4. *AutoCAD 2018 Training Guide* (English, ISBN 9789386551870, 938655187X)RUPA.

***

I Year – II SEMESTER

L	T	P	C
0	0	3	1.5

PYTHON PROGRAMMING LAB

Course Objectives:

- Experiment with scripting language
- Evaluate expression evaluation, control statements
- Use Data structures
- Model Functions, Modules and packages
- Outline OOP through Python and Exception Handling
- Select required Python Standard Library and Testing

Course Outcomes

- Comprehend how software easily to be build right out of the box – L1.
- Demonstrates the use of an interpreted language for problem solving through control statements including loops and conditionals.-L2
- Practice with data structures for quick programming solutions L3
- Demonstrates software building for real needs by breaking out code into reusable functions and modules.-L2
- Comprehend the software reliability through exception handling. L1
- Use of python standard library for problem solving and Identifies the necessity of testing software. L3

Exercise 1 – Basics

1. Running instructions in Interactive interpreter and a Python Script
2. Write a program to purposefully raise Indentation Error and Correct it

Exercise 2 – Operations

Write a program to compute distance between two points taking input from the user (Pythagorean Theorem)

Write a program add.py that takes 2 numbers as command line arguments and prints its sum.

Exercise – 3 Control Flow

Write a Program for checking whether the given number is a even number or not.

Using a for loop, write a program that prints out the decimal equivalents of ½, 1/3, ¼, . . . , 1/10.

Write a program using for loop that loops over a sequence. What is sequence?

Write a program using a while loop that asks the user for a number, and prints a countdown from that number to zero.

Exercise 4 – Control Flow – Continued

E) Find the sum of all the primes below two million.

Each new term in the Fibonacci sequence is generated by adding the previous two terms. By starting with 1 and 2, the first 10 terms will be:1, 2, 3, 5, 8, 13, 21, 34, 55, 89, ...

- By considering the terms in the Fibonacci sequence whose values do not exceed four million, find the sum of the even-valued terms.

Exercise – 5 – DS

- Write a program to count the numbers of characters in the string and store them in a dictionary data structure

- Write a program to use split and join methods in the string and trace a birthday with a dictionary data structure.

Exercise – 6 DS – Continued

1. Write a program combine lists that combines these lists into a dictionary.
2. Write a program to count frequency of characters in a given file. Can you use character frequency to tell whether the given file is a Python program file, C program file or a text file?

Exercise – 7 Files

1. Write a program to print each line of a file in reverse order.
2. Write a program to compute the number of characters, words and lines in a file.

Exercise – 8 Functions

1. Write a function ball collides that takes two balls as parameters and computes if they are colliding. Your function should return a Boolean representing whether or not the balls are colliding.
   Hint: Represent a ball on a plane as a tuple of (x, y, r), r being the radius
   If (distance between two balls centers) <= (sum of their radii) then (they are colliding)
2. Find mean, median, mode for the given set of numbers in a list.

Exercise – 9 Functions – Continued

- Write a function nearly equal to test whether two strings are nearly equal. Two strings A and B are nearly equal when a can be generated by a single mutation on A.
- Write a function dups to find all duplicates in the list.
- Write a function unique to find all the unique elements of a list.

Exercise – 10 – Functions – Problem Solving

1. Write a function cumulative product to compute cumulative product of a list of numbers.
2. Write a function reverse to reverse a list. Without using the reverse function.
3. Write function to compute GCD, LCM of two numbers. Each function shouldn't exceed one line.

Exercise 11 – Multi-D Lists

1. Write a program that defines a matrix and prints
2. Write a program to perform addition of two square matrices
3. Write a program to perform multiplication of two square matrices

Exercise – 12 – Modules

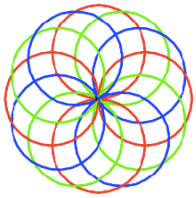
- Install packages requests, flask and explore them. Using (pip)
- Write a script that imports requests and fetch content from the page. Eg. (Wiki)
- Write a simple script that serves a simple HTTPResponse and a simple HTML Page

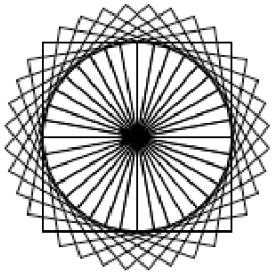
Exercise – 13 OOP

- Class variables and instance variable and illustration of the self variable
- Robot
- ATM Machine

Exercise – 14 GUI, Graphics

1. Write a GUI for an Expression Calculator using tk
2. Write a program to implement the following figures using turtle





Exercise – 15 – Testing

- Write a test-case to check the function even numbers which return True on passing a list of all even numbers.
- Write a test-case to check the function reverse string which returns the reversed string.

Exercise – 16 – Advanced

- Build any one classical data structure.
- Write a program to solve knapsack problem.

Text Books

1. *Python programming: a modern approach*, vamsi kurama, pearson
2. *Learning python*, mark lutz, orielly

***

I Year – II SEMESTER

L	T	P	C
0	0	3	1.5

APPLIED PHYSICS LAB

Any Ten of the Listed Fifteen Experiments

Course Objectives:

The Applied Physics Lab is designed to:

- Understand the concepts of interference and diffraction and their applications.
- Apply the concept of LASER in the determination of wavelength.
- Recognize the importance of energy gap in the study of conductivity and Hall Effect.
- Illustrate the magnetic and dielectric materials applications.
- Apply the principles of semiconductors in various electronic devices.

Course Outcomes:

The students will be able to:

- Operate optical instruments like microscope and spectrometer (L2)
- Determine thickness of a paper with the concept of interference (L2)
- Estimate the wavelength of different colors using diffraction grating and resolvingpower(L2)
- Plot the intensity of the magnetic field of circular coil carrying current with distance(L3)
- Determine magnetic susceptibility of the material and its losses by B-H curve (L3)
- Determine the resistivity of the given semiconductor using four probe method (L3)
- Identify the type of semiconductor i.e., n-type or p-type using hall effect (L3)
- Calculate the band gap of a given semiconductor (L3)

LIST OF EXPERIMENTS

1. Determination of wavelength of a source-Diffraction Grating-Normal incidence.
2. Newton's rings – Radius of Curvature of Plano - Convex Lens.
3. Determination of thickness of a spacer using wedge film and parallel interference fringes.
4. Magnetic field along the axis of a current carrying coil – Stewart and Gee's apparatus.
5. Energy Band gap of a Semiconductor p - n junction.
6. Characteristics of Thermistor – Temperature Coefficients

7. Determination of dielectric constant by charging and discharging method

8. Variation of dielectric constant with temperature

9. Study the variation of B versus H by magnetizing the magnetic material (B-H curve).

10. LASER - Determination of wavelength by plane diffraction grating

11. Determination of resistivity of semiconductor by Four probe method.

12. Determine the radius of gyration using compound pendulum

13. Rigidity modulus of material by wire-dynamic method (torsional pendulum)

14. Dispersive power of diffraction grating.

15. Determination of Hall voltage and Hall coefficients of a given semiconductor using Hall effect.

Text Books

1. *Engineering Physics* by B. K. Pandey, S. Chaturvedi,Cengage Publications, 2012.
2. *A Text book of Engineering Physics* by M.N. Avadhanulu, P.G.Kshirsagar, S.Chand, 2017.
3. *Engineering Physics* by D.K.Bhattacharya and Poonam Tandon, Oxford press, 2015.
4. *Engineering Physics* by R.K. Gaur and S.L. Gupta, Dhanpat Rai publishers, 2012

***

I Year – II SEMESTER

L	T	P	C
0	0	3	1.5

COMMUNICATIVE ENGLISH LAB-2

Course Objectives:

1. Adopt activity based teaching-learning methods to ensure that learners would be engaged in use of language both in the classroom and laboratory sessions.
2. Facilitate active listening to enable inferential learning through expert lectures and talks
3. Impart critical reading strategies for comprehension of complex texts
4. Provide training and opportunities to develop fluency in English through participation in formal group discussions and presentations using audio-visual aids
5. Demonstrate good writing skills for effective paraphrasing, argumentative essays and formal correspondence
6. Encourage use of a wide range of grammatical structures and vocabulary in speech and writing

Course Outcomes:

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

CO-1: Prioritize information from reading texts after selecting relevant and useful points (L3)
CO-2: Paraphrase short academic texts using suitable strategies and conventions (L3)
CO-3: Make formal structured presentations on academic topics using PPT slides with relevant graphical elements (L3)
CO-4: Participate in group discussions using appropriate conventions and language strategies (L3)
CO-5: prepare a CV with a cover letter to seek internship/ job (L2)
collaborate with a partner to make presentations and Project Reports (L2)

Detailed Syllabus

CALL based activity. English course books selected for classroom teaching will be used for practice in the computer based language labs. Watching and listening to Video clips.
Listening Activity: Selected speeches of eminent personalities, audio texts, dialogues and discussions
Speaking: JAM, Oral Presentations, Group Discussions
Writing: Different types of reports
Project: Power point presentation of 5 min on a specific topic
Pair work, Role play, conversational practice and Individual speaking activities based on following essays from *University of Success.*

d) "How to Get Yourself Organized" by Michael LeBeouf
e) "How to Turn Your Desires into Gold" by Napoleon Hill
f) "How to Look Like a Winner How to Increase Your Value" by Og Mandino

g) “How to Swap a Losing Strategy” by Auren Uris and Jack Tarrant
h) “How to Bounce Back From Failure” by Og Mandino
i) “How to Prevent Your Success From Turning into Ashes” by Allan Fromme
j) “How to Have a Happy Life” by Louis Binstock
k) “How to Keep the Flame of Success Shining Brightly” by Howard Whitman

Any ten Supplementary Language Activities from *UN Global Goals* document

CO4. **“**Developing children’s understanding of the Global Goals” by Carol Read
CO5. **“**End poverty in all its forms everywhere” by Sylwia Zabor-Zakowska
CO6. **“**End hunger, achieve food security and improved nutrition and promote sustainable agriculture” by Linda Ruas
CO7. **“**Ensure healthy lives and promote well-being for all at all ages” by Carmen Flores
CO8. **“**Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all” by Daniel Xerri
CO9. “Achieve gender equality and empower all women and girls” by Jemma Prior and Tessa Woodward
CO10. “Ensure availability and sustainable management of water and sanitation for all” by Wei Keong Too
CO11. **“**Ensure access to affordable, reliable, sustainable and modern energy for all” by Phil Wade
CO12. **“**Promote sustained, inclusive and sustainable economic growth, full and productive employment and decent work for all” by Nik Peachey
CO13. **“**Build resilient infrastructure, promote inclusive and sustainable industrialisation and foster innovation” by Malu Sciamarelli
CO14. **“**Reduce inequality within and among countries” by Alan Maley
CO15. **“**Make cities and human settlements inclusive, safe, resilient and sustainable” by David Brennan
CO16. **“**Ensure sustainable consumption and production patterns” by Laszlo Katona and Nora Tartsay
CO17. “Take urgent action to combat climate change and its impacts” by Maria Theologidou
CO18. “Conserve and sustainably use the oceans, seas and marine resources for sustainable development” by Jill Hadfield and Charlie Hadfield
CO19. **“**Protect, restore and promote sustainable use of terrestrial ecosystems, sustainably manage forests, combat desertification, and halt and reverse land degradation and halt biodiversity loss” by Chrysa Papalazarou
CO20. **“**Promote peaceful and inclusive societies for sustainable development, provide access to justice for all and build effective, accountable and inclusive institutions at all levels” by Rebeca Duriga
CO21. **“**Strengthen the means of implementation and revitalise the global partnership for sustainable development” by Jennifer Verschoor and Anna Maria Menezes
CO22. “Content and the Sustainable Development Goals: going beyond language learning” by Adrian Tennant
CO23. “Using extensive reading creatively to raise awareness of issues of equality and justice” by Sue Leather
CO24. “Storytelling for a better world” by David Heathfield
CO25. “Using the Sustainable Development Goals in the EAP classroom” by Averil Bolster and Peter Levrai

Text Books

1. Alan Maley and Nik Peachy. *Integrating global issues in the creative English Classroom: With reference to the United Nations Sustainable Development Goals.* British Council Teaching English, 2018 (Public Domain UN Document)
2. University of Success by Og Mandino, Jaico, 2015 (Reprint).

Reference Books

a) Bailey, Stephen. *Academic writing: A handbook for international students*. Routledge, 2014.
b) Chase, Becky Tarver. *Pathways: Listening, Speaking and Critical Thinking.* Heinley ELT; 2nd Edition, 2018.
c) Skillful Level 2 Reading & Writing Student's Book Pack (B1) Macmillan Educational.
d) Hewings, Martin. *Cambridge Academic English (B2)*. CUP, 2012.
e) Chaturvedi, P. D. and Chaturvedi Mukesh. *The Art and Science of Business Communication: Skills, Concepts, Cases and Applications.* 4Ed. Pearson, 2017.

Sample Web Resources

Grammar / Listening / Writing	**Reading**
1-language.com http://www.5minuteenglish.com/ https://www.englishpractice.com/ **Grammar/Vocabulary** English Language Learning Online http://www.bbc.co.uk/learningenglish/ http://www.better-english.com/ http://www.nonstopenglish.com/ https://www.vocabulary.com/ BBC Vocabulary Games Free Rice Vocabulary Game	https://www.usingenglish.com/comprehension/ https://www.englishclub.com/reading/short-stories.htm https://www.english-online.at/ Listening https://learningenglish.voanews.com/z/3613 http://www.englishmedialab.com/listening.html **Speaking** https://www.talkenglish.com/ BBC Learning English – Pronunciation tips Merriam-Webster – Perfect pronunciation Exercises
All Skills https://www.englishclub.com/ http://www.world-english.org/ http://learnenglish.britishcouncil.org/	

***

I Year – II SEMESTER

L	T	P	C
3	0	0	0

CONSTITUTION OF INDIA

Course Objectives:

- To Enable the student to understand the importance of constitution
- To understand the structure of executive, legislature and judiciary
- To understand philosophy of fundamental rights and duties
- To understand the autonomous nature of constitutional bodies like Supreme Court and high court controller and auditor general of India and election commission of India.
- To understand the central and state relation financial and administrative.

UNIT-I

Introduction to Indian Constitution: Constitution' meaning of the term, Indian Constitution - Sources and constitutional history, Features - Citizenship, Preamble, Fundamental Rights and Duties, Directive Principles of State Policy.

LEARNING OUTCOMES:

After completion of this unit student will

- Understand the concept of Indian constitution
- Apply the knowledge on directive principle of state policy
- Analyse the History, features of Indian constitution
- Evaluate Preamble Fundamental Rights and Duties

UNIT-II

Union Government and its Administration Structure of the Indian Union: Federalism, Centre-State relationship, President: Role, power and position, PM and Council of ministers, Cabinet and Central Secretariat, Lok Sabha, Rajya Sabha, The Supreme Court and High Court: Powers and Functions;

LEARNING OUTCOMES: - After completion of this unit student will

- Understand the structure of Indian government
- Differentiate between the state and central government
- Explain the role of President and Prime Minister
- Know the Structure of supreme court and High court

UNIT-III

State Government and its Administration Governor - Role and Position - CM and Council of ministers, State Secretariat: Organisation, Structure and Functions

LEARNING OUTCOMES: - After completion of this unit student will

- Understand the structure of state government

- Analyse the role Governor and Chief Minister
- Explain the role of state Secretariat
- Differentiate between structure and functions of state secretariat

UNIT-IV

Local Administration - District's Administration Head - Role and Importance, Municipalities - Mayor and role of Elected Representative - CEO of Municipal Corporation Panchayats: Functions PRI: Zila Panchayat, Elected officials and their roles, CEO Zila Panchayat: Block level Organizational Hierarchy - (Different departments), Village level - Role of Elected and Appointed officials - Importance of grass root democracy

LEARNING OUTCOMES: - After completion of this unit student will

- Understand the local Administration
- Compare and contrast district administration role and importance
- Analyse the role of Myer and elected representatives of Municipalities
- Evaluate Zilla panchayat block level organisation

UNIT-V

Election Commission: Election Commission- Role of Chief Election Commissioner and Election Commissioner ate State Election Commission: Functions of Commissions for the welfare of SC/ST/OBC and women

LEARNING OUTCOMES: - After completion of this unit student will

- Know the role of Election Commission apply knowledge
- Contrast and compare the role of Chief Election commissioner and Commissioner ate
- Analyse role of state election commission
- Evaluate various commissions of viz SC/ST/OBC and women

REFERENCES:

6. Durga Das Basu, Introduction to the Constitution of India, Prentice – Hall of India Pvt. Ltd.. New Delhi
7. Subash Kashyap, Indian Constitution, National Book Trust
8. J.A. Siwach, Dynamics of Indian Government & Politics
9. D.C. Gupta, Indian Government and Politics
10. H.M.Sreevai, Constitutional Law of India, 4th edition in 3 volumes (Universal Law Publication)
11. J.C. Johari, Indian Government and Politics Hans
12. J. Raj Indian Government and Politics
13. M.V. Pylee, Indian Constitution Durga Das Basu, Human Rights in Constitutional Law, Prentice – Hall of India Pvt. Ltd.. New Delhi
14. Noorani, A.G., (South Asia Human Rights Documentation Centre), Challenges to Civil Right), Challenges to Civil Rights Guarantees in India, Oxford University Press 2012

E-RESOURCES:

6. nptel.ac.in/courses/109104074/8
7. nptel.ac.in/courses/109104045/
8. nptel.ac.in/courses/101104065/
9. www.hss.iitb.ac.in/en/lecture-details

10. www.iitb.ac.in/en/event/2nd-lecture-institute-lecture-series-indian-constitution

Course Outcomes:

CO-1	Know the sources, features and principles of Indian Constitution.
CO-2	Learn about Union Government, State government and its administration.
CO-3	Get acquainted with Local administration and Pachayati Raj.
CO-4	Be aware of basic concepts and developments of Human Rights.
CO-5	Gain knowledge on roles and functioning of Election Commission

CO-PO Matrix:

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO1	3	-	3			3		2	3	-	3	2
CO2	2	-	2			2		2	2	-	3	2
CO3	3	-	3			2		2	2	-	3	3
CO4	2	-	3			2		2	2	-	3	3
CO5	3	-	1			3		3	3	-	3	2

****

II Year – I SEMESTER

L	T	P	C
3	0	0	3

MATHEMATICAL FOUNDATIONS FOR COMPUTER SCIENCE

Course Objectives:

- To introduce concepts of mathematical logic.
- To introduce concepts and perform operations with sets, relations and functions.
- To solve counting problems by applying elementary counting techniques.
- To introduce algebraic structures, generating functions and recurrence relations.
- To use graph theory for solving problems.

Syllabus

Unit – I: **(12 hrs)**

Mathematical Logic: Propositional Calculus: Statements and Notations, Connectives, Well Formed Formulas, Truth Tables, Tautologies, Equivalence of Formulas, Duality Law, Tautological Implications, Normal Forms, Theory of Inference for Statement Calculus, Consistency of Premises, and Indirect Method of Proof.

Predicate Calculus: Predicative Logic, Statement Functions, Variables and Quantifiers, Free and Bound Variables, Inference Theory for Predicate Calculus.

Unit – II: Set Theory &Relations: (12 hrs)

***Set Theory*:** Introduction, Operations on Binary Sets, Principle of Inclusion and Exclusion.

Relations: Properties of Binary Relations, Relation Matrix and Digraph, Operations on Relations, Partition and Covering, Transitive Closure, Equivalence, Compatibility and Partial Ordering Relations, Hasse Diagrams, *Functions:* Bijective Functions, Composition of Functions, Inverse Functions, Permutation Functions, Recursive Functions, Lattice and its Properties.

UNIT- III: Algebraic Structures and Number Theory: (12hrs)

Algebraic Structures: Algebraic Systems, Examples, General Properties, Semi Groups and Monoids, Homomorphism of Semi Groups and Monoids, Group, Subgroup, Abelian Group, Homomorphism, Isomorphism.

Number Theory: Properties of Integers, Division Theorem, The Greatest Common Divisor, Euclidean Algorithm, and Least Common Multiple, Testing for Prime Numbers, The Fundamental Theorem of Arithmetic, Modular Arithmetic (Fermat's Theorem and Euler's Theorem)

Unit – IV: Combinatorics & Recurrence Relations (12 hrs)

Combinatorics: Binomial and Multinomial Coefficients, Binomial and Multinomial Theorems, Pigeonhole Principle and its Application.
Recurrence Relations: Solving Recurrence Relations by Substitution and Generating Functions, Method of Characteristic Roots, Solving non homogeneous Recurrence Relations.

Unit – V: Graph Theory (12 hrs)

Graph Theory: Basic Concepts of Graphs, Sub graphs, Matrix Representation of Graphs: Adjacency Matrices, Incidence Matrices, Isomorphic Graphs, Paths and Circuits, Eulerian and Hamiltonian Graphs, Multi graphs, Planar Graphs, Euler's Formula, Graph Colouring, Chromatic Number, Spanning Trees, Algorithms for Spanning Trees (Problems Only and Theorems without Proofs).

Text Books:

1. Discrete Mathematical Structures with Applications to Computer Science, J. P. Tremblay and P. Manohar, Tata McGraw Hill.
2. Elements of Discrete Mathematics-A Computer Oriented Approach, C. L. Liu and D. P.Mohapatra, 3rdEdition, Tata McGraw Hill.
3. Discrete Mathematics and its Applications with Combinatorics and Graph Theory, K. H.Rosen, 7th Edition, Tata McGraw Hill.

Reference Books:

1. Discrete Mathematics for Computer Scientists and Mathematicians, J. L. Mott, A. Kandel,T.P. Baker, 2nd Edition, Prentice Hall of India.
2. Discrete Mathematical Structures, BernandKolman, Robert C. Busby, Sharon CutlerRoss, PHI.
3. Discrete Mathematics, S. K. Chakraborthy and B.K. Sarkar, Oxford, 2011

E-Resources:

1. https://nptel.ac.in/courses/106/103/106103205/
2. https://nptel.ac.in/courses/106/106/106106183/

Course Outcomes:

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

CO-1: Apply mathematical logic to solve problems.(L3)
CO-2: Understand sets, relations and discrete structures
CO-3: Apply number theory to perform modulo arithmetic and computer arithmetic. (L3)
CO-4: Solve problems on recurrence relations and counting principles.(L3)
CO-5: Analyseand solve real world problems using graphs and trees. (L5)

CO-PO-PSO Mapping:

CO/ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	P011	PO12
CO1	2	2	2	-	-	-	-	-	-	-	-	-

CO2	3	2	-	-	-	-	-	-	-	-	-	-
CO3	2	3	2	-	-	-	-	-	-	-	-	-
CO4	3	2	2	-	-	-	-	-	-	-	-	-
CO5	3	2	2	-	-	-	-	-	-	-	-	-

CO-PSO Mapping Matrix:

	PSO-1	**PSO-2**
CO1	2	-
CO2	1	-
CO3	2	-
CO4	2	-
CO5	2	-

II Year – I SEMESTER

L	T	P	C
3	0	0	3

DATA STRUCTURES

Course Objectives:

- To make students learn the basic concepts of Data Structures and Algorithms.
- To solve problems using data structures such as linear lists, stacks, queues.
- To explore advanced data structures such as balanced search trees.
- To be familiar with Graphs and their applications.
- To analyse various sorting techniques.

Syllabus

UNIT-I: Linear Lists (12 hrs)

Introduction to Data Structures, Definition, Need & Types of Data Structures

Algorithms: Introduction, Time complexity and Space complexity, Performance and Analysis

Linear lists (Arrays) – Introduction, Operations, Searching.

Sorting - Insertion Sort, Quick Sort, Merge Sort and Radix Sort.

UNIT-II: Stack & Queue (10 hrs)

Stacks: Introduction, Operations, implementation, Applications.

Queues: Introduction, Operations, implementation, Applications, Circular Queue

Unit – III: Linked Lists (10 hrs)

Single Linked List: Introduction, Representation, Operations, Applications.

Circular Lists: Introduction, Representation, Operations.
Double linked lists – Representation, operations.

UNIT-IV: TREES (8 hrs)

Trees: Introduction, Terminology, Representation of Trees

Binary Trees: Properties, Representations, Traversals, Types of Trees

Binary Search Trees: Definition, Operations.

UNIT-V: GRAPHS (12 hrs)

Graphs: Introduction, Definition, Representation, Degree of vertex, Types of graphs, Elementary Graph Operations, Graph Traversals – Depth First Search, Breadth First Search, Spanning trees-Prim's algorithm, Krushkal's algorithm.

Micro-Syllabus of Data Structures

Unit-1: Introduction to Data Structures, Definition, Need & Types of Data Structures
Algorithms: Introduction, Time complexity and Space complexity, Performance and Analysis, Asymptotic Notations - Big Oh(O), Small Oh(o) and Theta Notation (θ), necessary examples.
Linear lists (Arrays) – Introduction, operations: insert, delete, min, max, and Searching-Binary Search, Fibonacci Search.
Sorting - Insertion Sort, Quick Sort, Merge Sort and Radix Sort.

Unit	Module	Micro content
1a. Introduction to Data Structures and Algorithms	Introduction to Data Structures	Introduction to Data Structure
		Types of Data Structures
		Need of Data Structures
	Introduction to Algorithms	Introduction
		Time Complexity
		Space Complexity
		Performance and Analysis
	Asymptotic Notations	Big oh(O) notation
		Theta Notation
		Omega Notation
1b. Linear lists (Arrays)	Introduction to Linear Lists	Introduction to Arrays
		Operations on Arrays
		Searching-binary search and Fibonacci Search
1c.Sorting	Introduction to Sorting	Insertion Sort
		Quick Sort
		Merge Sort
		Radix Sort

UNIT-2
Stack & Queue

Stacks: Introduction, Operations – push, pop, underflow, overflow, peek and implementation, Applications – Infix to Postfix Conversion, Postfix evaluation.

Queues: Introduction, Operations – enqueue, dequeue, underflow, overflow and implementation, Applications – Circular Queue (operations), FIFO, Hot Potato Problem Simulation.

Unit	Module	Micro content
2.a Stack and	Stack	Introduction
		Operations-push, pop, underflow, overflow and peak

queue		Stack Implementation
		Applications- Decimal to binary.
		Infix to pre and postfix conversion, prefix to postfix conversion.
		Post fix Evaluation
	Queue	Introduction
		Operations- en-queue, dequeue, overflow,underflow
		Implementation
		Applications – Round robin Algorithm
		Circular queue
		Hot potato problem Simulation

Unit-3: Linked Lists

Single Linked List: Introduction, Differences between arrays & linked lists. Representation, Operations – insert, delete, concat, count and search, Applications – Polynomial representation, addition, multiplication.

Circular Lists: Introduction, Representation and implementation.

Doubly Linked list: Representation, Operations – insert, delete and search.

Unit	**Module**	**Micro content**
3.a. Linked Lists	Single Linked Lists	Introduction to Linked Lists
		Differences Between Arrays and Linked Lists
		Operations on Linked Lists
		Implementation
		Polynomial Representation
		Addition
		Multiplication
		Linked List Using Stack
		Linked List Using Queue
		Sparse matrix representation.
	Double Linked List	Introduction
		Differences Between Single Linked list and Double Linked List
		Operations
		Implementation
	Circular Linked List	Introduction
		Comparison of Circular and non circular Linked Lists
		Operations and Implementation
		Advantages and Disadvantages

Unit-4: TREES

Trees: Introduction, Terminology, Representation of Trees

Binary Trees: Properties, Representations, Traversal – Inorder Traversal, Preorder Traversal, Postorder Traversal (Recursive and Non Recursive) Types of trees **–** complete binary tree, Full binary tree, Thread Binary Trees, Expression Tree.

Binary Search Trees: Definition, Operations – insertion, deletion and findmin, findmax, count, leaf and Searching.

Unit	Module	Micro content
4a. Trees	Tree Terminology	Introduction to Trees
		Representation and Terminologies
4b. Binary Trees	Binary Trees	Introduction
		Tree Representation and Properties
		Conversion of General to binary tree, Construction of a binary tree from the tree traversals.
		Tree Traversal Recursive and non-Recursive approaches
		Types of Trees-Complete Binary Tree, Full Binary Tree, Thread Binary Tree
		Expression Trees
	Binary Search Trees	Introduction and Definition
		Operations on Binary Search Trees – insert, delete, height, count, counting leaf nodes, search.
		Advantages over Binary Trees
		Binary Search Tree Implementation

Unit-5: GRAPHS

Graphs: Introduction to graphs, Definition, Types of graphs, Degree of vertex

Representation - Adjacency matrix &Adjacency list

Elementary Graph Operations – Add Vertex, Add Edge, Delete Vertex, Delete Edge, Find Vertex and Find Edge.

Graph Traversals **–** Depth First Search, Breadth First Search**.**

Spanning trees-Prim's algorithm, Kruskal's algorithm.

Unit	Module	Micro content
5. a. Graphs	**Introduction and Representation**	Introduction
		Types of Graphs
		Graph Operations
		Memory representation-Matrix Representation and Linked list Representation
		Graph implementation
5.b. Graph Traversal	**Traversal Techniques and Minimum Spanning Trees**	Depth First Search
		Breadth First Search
		Prim's & Kruskal's Algorithm

Text books:

1. Data structures, Algorithms and Applications in C, S. Sahni, University Press (India) Pvt. Ltd, 2nd edition, Universities Press, Pvt. Ltd.

2. Data structures and Algorithm Analysis in C, Mark Allen Weiss, Pearson Education. Ltd, 2nd Edition.

3. Data Structures, Schaum's Outline, Seymour Lipschutz, Kindle Edition

Reference Books:

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

2. Classical Data Structures, Second Edition, Debasis Samanta, PHI

E-resources:

Data Structures Visualizations : https://www.cs.usfca.edu/~galles/visualization/Algorithms.html

Code Archery Youtube Channel:

https://www.youtube.com/playlist?list=PLrKBFf87Cy9CNZpzi3poq8BFWc0h4f0vL

Course Outcomes:

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

CO1: Implement various operations on linear lists.

CO2: Apply data structure strategies like stacks and queues for exploring complex data structures.

CO3: Identify performance and trade-offs of static and dynamic data structures.

CO4: Incorporate data structures into the applications such as binary trees, binary search trees.

CO5: Identify appropriate data structure algorithms for graphs.

CO-PO-PSO Mapping Matrix:

	PO1	**PO 2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSPO1**	**PSPO 2**
CO1	2	2	1	-	-	-	-	-	-	-	-	-	1	1
CO2	1	2	2	-	-	-	-	-	-	-	-	-	2	1
CO3	1	-	2	2	-	-	-	-	-	-	-	-	2	1
CO4	2	-	2	1	-	-	-	-	-	-	-	-	1	1
CO5	-	2	1	2	-	-	-	-	-	-	-	-	1	1

******

II Year – I SEMESTER

L	T	P	C
3	0	0	3

DIGITAL CIRCUITS & LOGIC DESIGN

Course Objectives:

1. To understand common forms of number representation in digital electronic circuits and to be able to convert between different representations.
2. To learn basic techniques for the design of digital circuits and fundamental concepts used in the design of digital systems and simplify logic expressions using basic theorems, K-map and Tabular.
3. Explain the concept of Combinational logic design and Realize logic expressions using MUX, and Decoder.
4. Illustrate the concept of sequential logic design; analyse the operation of flip-flop and conversion from one flip to another.
5. To impart to student the concepts of registers and counters of computer system.

Syllabus

Unit-1: Number Systems and Boolean algebra **13 Hours**

Number systems: Introduction to different number system and their conversions, Complement of number system and subtraction using complement method, Floating-Point Representation, Weighted and Non-weighted codes and its Properties.
Boolean Algebra: Fundamental Postulates of Boolean Algebra, Basic Theorems and Properties, Switching Functions, Boolean Functions of Canonical and Standard Forms, Simplification of Boolean Equations

Unit-2: Switching Functions and Minimization Methods **11 Hours**
Digital Logic Gates, NAND/NOR Realizations, K-Map Method, Prime Implicants, Don't Care Combinations, Minimal SOP and POS Forms, Tabular Method, Prime Implicants Chart, Simplification Rules. Error Detecting & Correcting Codes

Unit-3: Combinational Logic Design **14 Hours**

Design of Combinational Logic Circuits of Adders, Subtractor, Carry look ahead adder, BCD adder, Multiplexer, De-Multiplexer, MUX Realization of Switching Functions, Encoder, Priority encoders Decoder, Implementation of Higher-Order Device Using Lower Order devices, Error detection and correction codes.

Unit-4: Sequential Logic Design **10 Hours**

Sequential Circuits, Latches and Flip-Flops Flip Flops: SR, JK, JK Master Slave, D and T Type, Excitation Tables, Timing and Triggering Consideration, Conversion from one type of Flip-Flop to another. Designing of Clocked Sequential Circuits

Unit-5: Registers and Counters **12 Hours**

Shift Registers, Data Transmission in Shift Registers, Operation of Shift Registers, Shift Register Configuration, Universal Shift Registers, Applications of Shift Registers (any FF), Operation of Asynchronous and Synchronous Counters., Design and Operation of Ring and Twisted Ring Counter, modulo-N counters, Serial binary adder. Mealy and Moore Models of Finite State Machines

Micro-Syllabus

Unit-1: Number Systems and Boolean Algebra **13 Hours**

Number systems: Introduction to different number system and their conversions, Complement of number system and subtraction using complement method, Floating-Point Representation, Weighted and Non-weighted codes and its Properties.

Boolean Algebra: Fundamental Postulates of Boolean Algebra, Basic Theorems and Properties, Switching Functions, Boolean Functions of Canonical and Standard Forms, Simplification of Boolean Equations.

Unit	Module	Micro content	No of hrs
1a.Number systems and codes	Introduction to different number system and their conversions	Introduction to number system	3
		Binary numbers	
		Number base Conversion	
		Octal and Hexadecimal Numbers	
	Complement of number system and subtraction using complement method	r-1's Compliments	3
		r's Compliments	
		Compliment Arithmetic	
		Signed numbers	
	Floating-Point Representation	IEEE 754 Standard]][32-bit single precision, 64-bit double precision	1

	Weighted and Non-weighted codes and its Properties	BCD Code, 2421, Excess-3, 84-2-1, Gray Code, ASCII Character Code	2
1b.Boolean Algebra	Introduction to Boolean algebra and Boolean theorems	Postulates of a mathematical system and Axiomatic Systems, Algebra Basic Theorems and Properties	2
		Boolean Functions of Canonical and Standard Forms	2
		Minimization of Boolean Expressions by using theorems	

Unit-2: Switching Functions and Minimization Methods 11 Hours
Digital Logic Gates, NAND/NOR Realizations, K-Map Method, Prime Implicants, Don't Care Combinations, Minimal SOP and POS Forms, Tabular Method, Prime Implicants Chart, Simplification Rules. Error Detecting & Correcting Codes.

Unit	**Module**	**Micro content**	**No of hrs**
2a. Switching Functions	Digital Logic Gates	Basic Logic Gates and XOR Gates	1
2a. Minimization Methods		NAND/NOR Realizations of all logic logic gates	1
	The Minimization of Switching Function using Karnaugh Map	Introduction to minterms, maxterms, SOP and POS	6
		Implementation of Boolean functions (SOP & POS) with NAND & NOR gates only	
		Introduction to 2 - 5 variable K-Map with Implicants, prime Implicants, and Essential Prime Implicants	
		POS minimization with K-Map	
		K-Maps with don't care terms	
	The Minimization of Switching Function using Tabular method	Introduction to Tabular (Q-M) method with examples	2
		Q-M method with don't care terms	
		Prime Implicants Chart, Simplification	1

		Rules	

Unit-3: Combinational Logic Design **14 Hours**

Design of Combinational Logic Circuits of Adders, Subtractor, Carry look ahead adder, BCD adder, Multiplexer, De-Multiplexer, MUX Realization of Switching Functions, Encoder, Priority encoders Decoder, Implementation of Higher-Order Device Using Lower Order devices, Error detection and correction codes.

Unit	Module	Micro content	No of hrs
3. Combinational Logic Design	Designing of Half/Full Adder /Subtractor and Carry look ahead adder, BCD adder	Designing of Half Adder and Subtractor	2
		Full Adder and Subtractor	
		full adder by HA	
		Realization of above circuits with NAND & NOR	
		Carry look ahead adder	1
		Designing of comparator and BCD adder	2
	Multiplexers, Demultiplexers, Decoders, Encoders and Code Converters	Multiplexers, Demultiplexers	1
		Decoders, Encoders, Priority encodes	1
		Function realization using Multiplexers and Decoders	3
		Code Converters	1
	Implementation of Higher-Order Device Using Lower Order devices	Multiplexers, Demultiplexers, Decoders, Encoders and Code Converters	2
	Error detection and correction codes	Parity Bit, Hamming Code	1

Unit-4: Sequential Logic Design **10 Hours**

Sequential Circuits, Latches and Flip-Flops Flip Flops: SR, JK, JK Master Slave, D and T Type, Excitation Tables, Timing and Triggering Consideration, Conversion from one type of Flip-Flop to another. Designing of Clocked Sequential Circuits

Unit	Module	Micro content	No of hrs

4. Sequential Logic Design	Analysis of Sequential Circuits	Introduction to Sequential Circuits and difference between Combinational and Sequential circuits	1
		SR latch by NAND / NOR gates and introduction of flip flop	1
	Storage elements: Flip Flops	Design various flip flops like SR, JK, D, Master Slave FF & T with truth tables, logic diagrams, and timing relationships	3
		Excitation table and characteristic equations of various flip flops and Conversion of one flips flop to other flip flop	3
	Designing of Sequential Circuits	Analysis of Clocked Sequential Circuits	1
		Designing of Clocked Sequential Circuits	2

Unit-5: Registers and Counters **12 Hours**

Shift Registers, Data Transmission in Shift Registers, Operation of Shift Registers, Shift Register Configuration, Universal Shift Registers, Applications of Shift Registers (any FF), Operation of Asynchronous and Synchronous Counters., Design and Operation of Ring and Twisted Ring Counter, modulo-N counters, Serial binary adder. Mealy and Moore Models of Finite State Machines

Unit	**Module**	**Micro content**	**No of hrs**
5. Registers and Counters	Registers	Introduction of registers and Design of shift registers	1
		Design of bidirectional shift registers, Universal SR and applications,	1
		To Produce Time Delay,To Simplify Combinational Logic, To Convert Serial Data to Parallel Data by any FF	2
	Counters	Designing basic Asynchronous Counter	1
		Designing basic Synchronous Counter	1
		Other counters: modulo-n counters, Ring and twisted ring counters, Johnson Counter,	2
		Design of sequence generators and	2

		detectors	
		Mealy and Moore Models of Finite State Machines with examples	2

Text Book:

1. Digital Design by Mano, PHI
2. Switching Theory and Logic Design by Hill and Peterson Mc-Graw Hill TMH edition
3. Switching Theory and Logic Design by A. Anand Kumar, PHI.

REFERENCE BOOKS:

1. Modern Digital Electronics by RP Jain, TMH
2. Fundamentals of Logic Design by Charles H. Roth Jr, Jaico Publishers

Course Outcomes:

CO1: Distinguish the analog and digital systems, apply positional notations, number systems and computer codes in digital systems. **(Remember, Understand, and Apply)**

CO2: To understand the Boolean algebra theorems, simplify and design logic circuits. **(Understand, Apply, Analyse and valuate)**

CO3: Implemented combinational logic circuit design and modular combinational circuits using encoders, decoders, multiplexers and demultiplexers. (**Apply, Analyse, valuate, and create)**

CO4: To understand the basic elements of sequential logic circuits. **(Understand, Apply, Analyse)**

CO5: Able to design and analyse the registers and counters. **(Understand, Apply, Analyze and create)**

CO-PO mapping Table with justification

Mapping	PO1	PO2	PO3	PO10
CO1	3	2	2	1
CO2	3	2	2	1
CO3	3	2	2	1
CO4	3	2	2	1
CO5	3	2	2	1

****

II Year – I SEMESTER

L	T	P	C
3	0	0	3

JAVA PROGRAMMING

Course Objectives:

- To understand object oriented programming concepts, and apply them in solving problems.
- To make the students to learn the principles of inheritance and polymorphism; and to demonstrate how they relate to the design of abstract classes; to introduce the implementation of packages and interfaces.
- To make the students to learn the concepts of exception handling and multithreading.
- To impart the knowledge on collection framework.
- To make the students to develop GUI applications network based applications.

Syllabus

Unit – I: Introduction to OOPS Concepts, Classes and Strings (12 hrs)

Introduction to Object Oriented Programming, Java buzzwords, Java Programming Basics, Sample programs, Data types and operators, Control statements.

Classes: Classes, Objects, Methods, Constructors, this and static keywords, Method and Constructor Overloading, Access modifiers, arrays-One Dimensional and multi-dimensional arrays, Searching, Sorting. **Strings**-Exploring the String class, String buffer class, Command-line arguments.

Unit – II: Inheritance, Interfaces, Packages And Exception Handling (14 hrs)

Inheritance: Need of inheritance, types, super keyword, abstract classes, interfaces, compile time and runtime polymorphism, Packages.
Exception Handling: Concepts of Exception handling, Built-in exceptions, creating own exception sub classes, Assertions.

Unit – III: Multi-Threading and I/O Streams(14 hrs)

Multithreading : Concepts of Multithreading, differences between process and thread, thread life cycle, Thread class, Runnable interface, creating multiple threads, Synchronization, thread priorities, inter thread communication, daemon threads, thread groups.

Stream based I/O (java.io) – The Stream classes-Byte streams and Character streams, Reading console Input and Writing Console Output, File class, Reading and writing Files, Random access file operations, Object Serialization, exploring java.nio .

Unit – IV: Collection Frame Work Classes (12 hrs)

The Collections Framework (java.util)- Collections overview, Collection Interfaces, The Collection classes- Array List, Linked List, Hash Set, Tree Set, Priority Queue, Array Deque. Accessing a Collection via an Iterator, Using an Iterator, The For-Each alternative, Map Interfaces and Classes, Comparators, Collection algorithms, Arrays, The Legacy Classes and Interfaces- Dictionary, Hashtable, Properties, Stack, Vector.

Unit – V: GUI Programming and Networking (12 hrs)

GUI Programming with Swing: Introduction, limitations of AWT, Various swing components & hierarchy. **Event Handling**- event delegation model, sources of event, Event Listeners, adapter classes, inner classes.

Introduction to Networking: Basics of Networking, Networking classes and Interfaces, Networking with URLs, exploring java.net package.

Text Books:

1. Java the Complete Reference, Herbert Schildt, MC GRAW HILL Education, 9th Edition, 2016

Reference Books:

1. "Java – How to Program", Paul Deitel, Harvey Deitel,PHI.
2. "Core Java", NageswarRao, WileyPublishers.
3. "Thinking in Java", Bruce Eckel, PearsonEducation
4. "A Programmers Guide to Java SCJP", Third Edition, Mughal, Rasmussen,Pearson.

e- Resources:

Programming in Java: https://nptel.ac.in/courses/106/105/106105191/

Course Outcomes:

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

CO-1: Summarize object-oriented programming concepts for problem solving.
CO-2: Build class hierarchy and packages for real world problems.
CO-3: Develop thread safe Java programs with appropriate Exception handling.
CO-4: Implement various data structures using java collections.
CO-5: Design GUI and network based applications using swings and multithreading.

CO-PO Mapping Matrix:

CO/ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	P011	PO12
CO1	-	-	2	-	-	-	-	-	2	-	-	-
CO2	-	-	2	-	-	-	-	-	2	-	-	-
CO3	-	-	2	2	-	-	-	-	2	-	-	-
CO4	-	-	-	2	-	-	-	-	2	-	-	-
CO5	-	-	2	-	-	-	-	-	2	-	-	

CO-PSO Mapping Matrix:

	PSO-1	PSO-2
CO1	-	-
CO2	-	2
CO3	-	2
CO4	-	2
CO5	-	-

******

II Year – I SEMESTER

L	T	P	C
3	0	0	3

PROBOBILITY & STATISTICS

Course objectives: The student should be able to

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

Syllabus

Unit- I Descriptive statistics and methods for data science: 10 hrs

Data science-Statistics Introduction-Population vs Sample-Collection of data-primary and secondary data-Types of variable: dependent and independent Categorical and Continuous variables-Data visualization-Measures of Central tendency-Measures of Variability (spread or variance)-Skewness Kurtosis.

Unit-II Correlation and Curve fitting: 10 hrs

Correlation-correlation coefficient-Rank correlation-Regression coefficient and properties-regression lines-Multiple regression-Method of least squares-Straight line-parabola-Exponential-Power curves.

Unit-III Probability and Distributions: 12hrs

Probability-Conditional probability and Baye's theorem-Random variables-Discrete and Continuous random variables-Distribution function-Mathematical Expectation and Variance-Binomial, Poisson, Uniform and Normal distributions.

Unit -IV Sampling Theory: 10 hrs

Introduction–Population and samples-Sampling distribution of Means and Variance (definition only)-Central limit theorem (without proof)-Point and Interval estimations, Good estimator, Unbiased estimator, Efficiency estimator-Maximum error of estimate.

Unit – V Test of Hypothesis: 14 hrs

Introduction–Hypothesis-Null and Alternative Hypothesis-Type I and Type II errors-Level of significance-One tail and two-tail tests-Tests concerning one mean, two means, and proportions using Z test, Tests concerning one mean, two means using t test, also chi-square and F tests use for small samples.

Micro-Syllabus of Probability and Statistics

(Common to CSE and IT)

UNIT-I:Descriptive statistics and methods for data science: 10 hrs

Data science-Statistics Introduction-Population vs Sample-Collection of data-primary and secondary data-Types of variable: dependent and independent Categorical and Continuous variables-Data visualization-Measures of Central tendency-Measures of Variability (spread or variance)-Skewness Kurtosis.

Unit	Module	Micro content	No of hrs
1a. Descriptive Statistics	Introduction-Population vs Sample	Collection of data-primary and secondary data	3
		Population	
		Sample	
	Types of variable	dependent and	2
		independent	
		Categorical	
		Continuous variables	
	Data visualization	Data visualization	1
1b.methods for data science	Measures of Central tendency and Measures of Variability	Measures of Central tendency	2
		Measures of Variability	2
		Skewness Kurtosis.	

UNIT-II: Correlation and Curve fitting: 10 hrs

Correlation-correlation coefficient-Rank correlation-Regression coefficient and properties-regression lines-Multiple regression-Method of least squares-Straight line-parabola-Exponential-Power curves.

Unit	Module	Micro content	No of hrs
2.Correla	Correlation	correlation coefficient	4

tion and Curve fitting		Rank correlation	
	Regression	Regression coefficient	4
		properties	
		regression lines	
		Multiple regression	
	Method of least squares	Straight line	4
		Parabola.	
		Exponential curves	
		Power curves.	

UNIT-III: Probability and Distributions: 12hrs

Probability-Conditional probability and Baye's theorem-Random variables-Discrete and Continuous random variables-Distribution function-Mathematical Expectation and Variance-Binomial, Poisson, Uniform and Normal distributions.

Unit	Module	Micro content	No of hrs
3. Probability and Distributions	Probability	Conditional probability	2
		Baye's theorem	
	Random variables	DiscreteRandom variables	1
		ContinuousRandom variables	1
		Distribution function	1
		Mathematical Expectation and variance	1
	Distributions	Binomialdistribution.	4
		Poissondistribution	
		Uniform distribution	
		Normaldistribution	

UNIT-IV: Sampling Theory: 10 hrs

Introduction–Population and samples-Sampling distribution of Means and Variance (definition only)-Central limit theorem (without proof)-Point and Interval estimations, Good estimator, Unbiased estimator, Efficiency estimator-Maximum error of estimate.

Unit	Module	Micro content	No of hrs
4.Sampling Theory	Introduction	Populationsamples	1
		Central limit theorem (without proof	
	Sampling distributions	Sampling distribution of Means	4
		Sampling distribution of Variance	
	Estimation	Point estimations	5
		Interval estimation	
		Good estimator	
		Unbiased estimator	
		Efficiency estimator	
		Maximum error of estimate.	

UNIT-V: Test of Hypothesis: 14hrs

Introduction–Hypothesis-Null and Alternative Hypothesis-Type I and Type II errors-Level of significance-One tail and two-tail tests-Tests concerning one mean, two means, and proportions using Z test, Tests concerning one mean, two means using t test, also chi-square and F tests use for small samples.

Unit	Module	Micro content	No of hrs
5. Test of Hypothesis	Hypothesis	Null Hypothesis	**2**
		Alternative Hypothesis	
		Type I and Type II errors	
		Level of significance	
		One tail and two-tail tests	
	Test for large samples	Tests concerning one mean using Z test	**6**
		Tests concerning one two means using Z test.	
		Tests concerning proportions using Z test	
	Tests for small samples	Tests concerning one mean, two means using t test	**6**

		chi-square test	
		F test	

Text books:

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

Reference books:

1. Shron L. Myers, Keying Ye, Ronald E Walpole, Probability and Statistics Engineers and the Scientists, 8th Edition, Pearson 2007.
2. Jay I. Devore, Probability and Statistics for Engineering and the Sciences, 8th Edition, Cengage.
3. Sheldon M. Ross, Introduction to probability and statistics Engineers and the Scientists, 4th Edition, Academic Foundation, 2011.
4. Johannes Ledolter and Robert V. Hogg, Applied statistics for Engineers and Physical Scientists, 3rd Edition, Pearson, 2010.
5. T. K. V. Iyenger, Probability and Statistics, S. Chand & Company Ltd, 2015.

e- Resources & other digital material

1. https://www.youtube.com/watch v=COI0BUmNHT8&list=PLyqSpQzTE6M_JcleDbrVyPnE0PixKs2JE (For Probability and Statistics)
2. https://www.youtube.com/watch?v=VVYLpmKRfQ8&list=PL6C92B335BD4238AB (For Probability and Statistics)
3. https://www.mathsisfun.com/data/standard-normal-distribution-table.html (Information about Normal distribution)
4. https://www.statisticshowto.com/tables/t-distribution-table/ (Information about T- distribution)
5. Statistical Tables to be allowed in examinations:

1. Normal distribution table
2. T- distribution table

Course Outcomes:

CO1: Classify the concepts of data science and its importance (L4) or (L2) (Understand, Analyse)

CO2: Understand the concepts of probability and their applications, & apply discrete and continuous probability distributions (L3)(Understand, Apply)

CO3: Interpret the association of characteristics and through correlation and regression tools (L4) Analyse

CO4: Design the components of a classical hypothesis test (L6)(Understand, Design, create)

CO5: Infer the statistical inferential methods based on small and large sampling tests (L4) (Understand, Analyse)

CO-PO mapping Matrix

Contribution of Course Outcomes towards achievement of Program Outcomes & Strength of correlations **(High: 3, Medium: 2,Low: 1)**

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2
CO1	2	2	-	-	-	-	-	-	-	-	-	-	-	-
CO2	2	3	-	-	-	-	-	-	-	-	-	-	-	-
CO3	2	2	-	-	-	-	-	-	-	-	-	-	-	-
CO4	2	2	-	-	-	-	-	-	-	-	-	-	-	-
CO5	2	3	-	-	-	-	-	-	-	-	-	-	-	-

****

II Year – I SEMESTER

L	T	P	C
-	0	3	1.5

DATA STRUCTURES LAB

Learning Objectives:

The objective of this laboratory is to teach students various data structures and to explain them algorithms for performing various operations on these data structures. This lab complements the Algorithms and Data Structures course. Students will gain practical knowledge by writing and executing programs in C using various data structures such as arrays, linked lists, stacks, queues, trees, graphs, and search trees.

SEARCHING AND SORTING (2 Exercises) **[CO – 1]**

Write a C program to Implement the following searching techniques using linear list(arrays)
- Binary Search
- Fibonacci Search

Write a C program to implement the following sorting techniques using arrays
- Selection sort
- Insertion sort
- Quick Sort
- Merge Sort
- Radix Sort

STACK & QUEUE (2 Exercises) **[CO – 1]**

Write a C program to
- Implement stack using arrays.
- Convert infix expression to postfix expression
- Evaluation of postfix expression.

Write a C program to implement
- Queue using arrays
- Round Robin Algorithm.
- Simulation : Hot Potato

LINKED LISTS (3 Exercises) **[CO – 1]**

Write a C program to implement Singly Linked List.

Write a C program to implement Circular Linked List.
Write a C program to implement Doubly Linked List.

Implement C code for polynomial representation, addition, subtraction & multiplication.

TREES (5 Exercises) **[CO – 2]**

Write a C program to implement Binary trees.
Write a C program to implement tree traversal techniques (Both Recursive and Non Recursive).
Write a C program to implement Binary Search trees.
Write a C program to implement Complete Binary Search tree.
Write a C program to implement Huffman Coding.

GRAPHS (2 Exercises) **[CO – 3]**

Write a C program to implement graphs.
Write a C program to implement graphs traversal techniques (both recursive and non-recursive)
Breadth First Search
Depth First Search

ADDITIONAL EXERCISES:

The below list of problem statements can be solved in either www.hackerrank.com or www.hackerearth.com, and must submit the solution

SEARCHING AND SORTING (Any 2 additional problems from below list of 6 problems)

Sherlock and Numbers / Ice cream Parlour (Binary Search)
The Exam / The Missing Numbers (Fibonacci Search)
Monk and Nice Strings / Insertion Sort (Insertion Sort)
K- Palindrome / Quick Sort (Quick Sort)
Pebbles Game (Merge Sort)
Monk and Sorting Algorithm (Radix Sort)

STACK & QUEUE (Any 2 additional problems from below list of 4 problems)

Stack Operations / Maximum Elements (Stack Operations)
Balanced Brackets / Balanced Brackets (Stack)
Robin Robin, Round Robin (Queue)
Double Ended Queue (Queue)

LINKED LIST (Any 2 additional problems from below list of 4 problems)

1. Insert At Begin, Insert At End, Insert At Position, Delete a Node (Linked List Operations)
2. Remove Friends (Single Linked List)
3. Cycle Detection (Circular Linked List)
4. Reversing a Double Linked List (Double Linked List)

TREES (Any 2 additional problems from below list of 3 problems)

Mirror Image, Nodes in a Tree (Binary Tree)

Level Order traversal. (Binary Tree Traversal)
Monk Watching Fight, Distinct Count (Binary Search Tree)

GRAPHS (Any 2 additional problems from below list of 3 problems)

1. Build a graph, Monk at Graph Factory (Graph representation)
2. Monk and the islands, Zeta and Thanos (Breadth First Search Tree Traversal)
3. Words and Trees, Water Supply (Depth First Search Tree Traversal)

Course Outcomes: Upon completion of this laboratory, the student will be able to

- Identify appropriate list for solving general data structure problems. (L3)
- Incorporate data structures into the applications such as binary trees, binary search trees (L3)
- Choose appropriate algorithm for solving graph related problems (L3).

****

II Year – I SEMESTER

L	T	P	C
-	0	3	1.5

JAVA PROGRAMMING LAB

Course Objectives:

To write programs using abstract classes.
To write programs for solving real world problems using java collection frame work.
To write multithreaded programs.
To design GUI application using swing controls.
To introduce java compiler and eclipse platform
To impart hands on experience with java programming.

Note:

Mandatory to follow test driven development with Eclipse IDE empowered JUnit testing framework and code coverage plugin.
The list suggests the minimum program set. Hence, the concerned staff is requested to add more problems to the list as needed.

List of Experiments

1. Create a class called Invoice that a hardware store might use to represent an invoice for an item sold at the store. An Invoice should include four pieces of information as instance variables-a part number (type String),a part description(type String),a quantity of the item being purchased (type int) and a price per item (double). Your class should have a constructor that initializes the four instance variables. Provide a set and a get method for each instance variable. In addition, provide a method named getInvoiceAmount() that calculates the invoice amount (i.e., multiplies the quantity by the price per item), then returns the amount as a double value. If the quantity is not positive, it should be set to 0. If the price per item is not positive, it should be set to 0.0. Write a test application named InvoiceTest that demonstrates class Invoice's capabilities. [CO1]

2. Develop a Java application to generate Electricity bill. Create a class with the following members: Consumer no., consumer name, previous month reading, current month reading, and type of EB connection (i.e. domestic or commercial). Compute the bill amount using the following tariff. [CO1]
If the type of the EB connection is domestic, calculate the amount to be paid as follows:

1. First 100 units - Rs. 1 perunit
2. 101-200units - Rs. 2.50 perunit

3. 201 -500 units - Rs. 4 perunit
4. >501 units - Rs. 6 perunit

If the type of the EB connection is commercial, calculate the amount to be paid as follows:

5. First 100 units - Rs. 2 perunit
6. 101-200units - Rs. 4.50 perunit
7. 201 -500 units - Rs. 6 perunit
8. >501 units - Rs. 7 perunit

3. Create class SavingsAccount. Use a static variable annualInterestRate to store the annual interest rate for all account holders. Each object of the class contains a private instance variable savingsBalance indicating the amount the saver currently has ondeposit. Provide method calculateMonthlyInterest to calculate the monthly interest by multiplying the savingsBalance by annualInterestRate divided by 12 this interest should be added to savingsBalance. Provide a static method modifyInterestRate that sets the annualInterestRate to a new value.Write a program to test class SavingsAccount. Instantiate two savingsAccount objects, saver1 and saver2, with balances of $2000.00 and $3000.00, respectively. Set annualInterestRate to 4%, then calculate the monthly interest and print the new balances for both savers. Then set the annualInterestRate to 5%, calculate the next month's interest and print the new balances for both savers. [CO1]

4. Create a class called Book to represent a book. A Book should include four pieces of information as instance variables-a book name, an ISBN number, an author name and a publisher. Your class should have a constructor that initializes the four instance variables. Provide a mutator method and accessor method (query method) for each instance variable. In addition, provide a method named getBookInfo that returns the description of the book as a String (the description should include all the information about the book). You should use this keyword in member methods and constructor. Write a test application named BookTest to create an array of object for 30 elements for class Book to demonstrate the class Book's capabilities. [CO1].

5. Write a JAVA program to search for an element in a given list of elements using binary search mechanism. [CO1]

6. Write a Java program that implements Merge sort algorithm for sorting and also shows the number of interchanges occurred for the given set of integers. [CO1]

7. Write a java program to make rolling a pair of dice 10,000 times and counts the number of times doubles of are rolled for each different pair of doubles. Hint: Math.random() [CO1].

8. Develop a java application to validate user information using regular expressions. [CO1].

9. Develop a java application with Employee class with Emp_name, Emp_id, Address, Mail_id, Mobile_no as members. Inherit the classes, Programmer, Assistant Professor, Associate Professor and Professor from employee class. Add Basic Pay (BP) as the member of all the inherited classes with 97% of BP as DA, 10 % of BP as HRA, 12% of BP as PF, 0.1% ofBP for staff club fund. Generate pay slips for the employees with their gross and net salary. [CO1]

10. Write a Java Program to create an abstract class named Shape that contains two integers and an empty method named print Area(). Provide three classes named Rectangle, Triangle and

Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method print Area () that prints the area of the givenshape.[CO2]

11.Developajavaapplicationtoimplementcurrencyconverter(DollartoINR,EUROtoINR,YentoINR and vice versa), distance converter (meter to KM, miles to KM and vice versa) , timeconverter (hours to minutes, seconds and vice versa) using packages. [CO1]

12. Write a Java Program to Handle Arithmetic Exceptions and InputMisMatchExceptions. [CO1]

13. Write a multi-threaded Java program to print all numbers below 100,000 that are both prime and Fibonacci number (some examples are 2, 3, 5, 13, etc.). Design a thread that generates prime numbers below 100,000 and writes them into a pipe. Design another thread that generates Fibonacci numbers and writes them to another pipe. The main thread should read both the pipes to identify numbers common to both. [CO3].

14. Write a java program that implements a multi-threaded application that has three threads. First thread generates a random integer every 1 second and if the value is even, second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of thenumber. [CO3].

15. Write a Java program that correctly implements the producer – consumer problem using the concept of inter-thread communication. [CO3].

16. Write a Java program that reads a file name from the user, displays information about whether the file exists, whether the file is readable, or writable, the type of file and the length of the file inbytes. [CO1].

17. Develop Rational number class in Java. Use JavaDoc comments for documentation. Your implementation should use efficient representation for a rational number, i.e. (500 / 1000) should be represented as (½). [CO1].

18. To write a Java Program to design an interface for Stack ADT and implement Stack ADT using both Array and Linked List. [CO1].

19. To Implement basic operations such as 'car', 'cdr', and 'cons' using Lisp-like list in Java. If L is a list [3, 0, 2, 5], L.car() returns 3, while L.cdr() returns [0,2,5] [CO1].

20. Write a Java program to build a Calculator in Swings/ [CO4]

21. Write a Java program to implement JMenu to draw all basic shapes using Graphics. [CO4]

22.Write a Java program to implement JTable and JTree. [CO4]

23. Write a Java program to implement JTabbedPane. [CO4]

24. Write a Java Program that implements a simple client/server application. The client sends data to a server. The server receives the data, uses it to produce a result and then sends the result back to the client. The client displays the result on the console. For ex: The data sent from the client is the radius of a circle and the result produced by the server is the area of the circle. [CO3]

25. Develop multi-threaded echo server and a corresponding GUI client. [CO3,4]

Course Outcomes: at the end of the lab, the student will be able to

CO1: Develop programs for solving real world problems using java collection frame work.
CO2: Develop and apply multithreaded programs in network applications.
CO3: Develop GUI programs using swing controls in Java.

CO-PO mapping Table with justification

Mapping	P01	P02	P03	P04	P05	P06	P07	P08	P09	P010	P011	P012	PS01	PSO2
C01	2	2	2		2				2				2	2
C02	2	2	2		2				2				2	2
C03	2	2	2		2				2				2	2

****

II Year – I SEMESTER

L	T	P	C
3	0	0	0

ESSENTIALS OF INDIAN TRADITIONAL KNOWLEDGE

Objectives:

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

The course aim of the importing basic principle of third process reasoning and inference sustainability is at the course of Indian traditional knowledge system

To understand the legal framework and traditional knowledge and biological diversity act 2002 and geographical indication act 2003.

The courses focus on traditional knowledge and intellectual property mechanism of traditional knowledge and protection.

To know the student traditional knowledge in different sector.

Syllabus

Unit-I:

Introduction to traditional knowledge: Define traditional knowledge, nature and characteristics, scope and importance, kinds of traditional knowledge, the physical and social contexts in which traditional knowledge develop, the historical impact of social change on traditional knowledge systems. Indigenous Knowledge (IK), characteristics, traditional knowledge vis-à-vis indigenous knowledge, traditional knowledge Vs western knowledge traditional knowledge vis-à-vis formal knowledge

Learning Outcomes:

At the end of the unit the student will able to:

- Understand the traditional knowledge.
- Contrast and compare characteristics importance kinds of traditional knowledge.
- Analyze physical and social contexts of traditional knowledge.
- Evaluate social change on traditional knowledge.

Unit-II:

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

Learning Outcomes:

At the end of the unit the student will able to:

- Know the need of protecting traditional knowledge.
- Apply significance of TK protection.
- Analyse the value of TK in global economy.
- Evaluate role of government

Unit-III:

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

Learning Outcomes:

At the end of the unit the student will able to:

- Understand legal framework of TK.
- Contrast and compare the ST and other traditional forest dwellers
- Analyse plant variant protections
- Evaluate farmers right act

Unit-IV:

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

Learning Outcomes:

At the end of the unit the student will able to:

- Understand TK and IPR
- Apply systems of TK protection.
- Analyse legal concepts for the protection of TK.
- Evaluate strategies to increase the protection of TK.

Unit-V:

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

Learning Outcomes:

At the end of the unit the student will able to:

- Know TK in different sectors.
- Apply TK in engineering.
- Analyse TK in various sectors.
- Evaluate food security and protection of TK in the country.

Reference Books:

1. Traditional Knowledge System in India, by Amit Jha, 2009.
2. Traditional Knowledge System and Technology in India by Basanta Kumar Mohanta and Vipin Kumar Singh, Pratibha Prakashan 2012.
3. Traditional Knowledge System in India by Amit Jha Atlantic publishers, 2002
4. "Knowledge Traditions and Practices of India" Kapil Kapoor, Michel Danino

E-Resources:

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

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

- Understand the concept of Traditional knowledge and its importance
- Know the need and importance of protecting traditional knowledge
- Know the various enactments related to the protection of traditionalknowledge.

- understand the concepts of Intellectual property to protect the traditional knowledge

****

II Year – I SEMESTER

L	T	P	C
3	0	0	0

EMPLOYABILITY SKILLS

Components

1. Verbal Ability
2. Quantitative Ability
3. Reasoning Ability
4. Soft Skills

Verbal Ability

Most of the recruitment tests and test like GRE, TOEFL, IELTS etc require the students to posses good language skills. The Verbal Ability sessions are to enhance the competence of the students in Vocabulary, Grammar, Reading Comprehension and Writing so that they can face answer verbal ability questions confidently.

Quantitative Ability

Almost all competitive examinations tests the candidate for quantitative aptitude, especially recruitment test, public service examinations management courses, where they evaluate the student's thinking prowess and analytical skills. Critical analysis of problems asked in examination reveal that they are designed to correlate multiple topics and the test taker is expected to identify those link points and come out with an out-of-box unique solution. The purpose of the test is to assess the arithmetic abilities, logical, analysis, problem solving and decision making skills.

Reasoning Ability

Reasoning ability is the ability to draw connections between factors, and the ability to synthesize a message from a body of information. Reasoning ability of the aspirants for jobs or courses is tested by means of a verbal reasoning test non-verbal reasoning. Thus reasoning is a highly specialized thinking which helps an individual to explore mentally the cause and effect relationship of an event or solution of a problem by adopting some well-organized systematic steps based on previous experience combined with present observation. Most of the recruitment tests consists questions to assess the reasoning ability of the students.

Soft skills

Soft skills play an important role in identifying the right candidate for a position in a company. Effective soft skills like communication, adaptability, team working skills, work ethics etc are

some of the most important skills which play as a differentiating factor in the success of the students in their career.

****

II Year – II Semester

L	T	P	C
3	0	0	3

SOFTWARE ENGINEERING

COURSE OBJECTIVES: The student should be able to

1. To understand the software life cycle models.
2. To understand the software requirements and SRS document.
3. To understand the importance of modeling and analyzing languages.
4. To design and develop correct and robust software products.
5. To understand the quality control and how to ensure good quality software.

Unit-1:

Introduction to Software Engineering: (14Hrs)

Software, Software Classifications and Characteristics, Emergency of Software Engineering, What is Software Engineering? Software Engineering Challenges

Software Processes Process model, Elements and Characteristics of Process model, Process Classification, Phased Development Life Cycle, Software Development

Process Models: Prescriptive Process Models, Agile process models, and RUP process model

Unit-2:

Project Management & Planning: (12Hrs)

Project management essentials, Project success and failures, Project Life Cycle, Project team structure and organization, Software Configuration Management. Project planning activities, Metrics and Measurements, Project Size Estimation, Effort Estimation Techniques, Staffing and Personnel Planning, Project Scheduling and Miscellaneous Plans.

Unit-3:

Requirement Engineering: (10 Hrs.)

Software Requirements, Requirement Engineering Process, Requirement Elicitation, Requirement Analysis (Structured Analysis, Object Oriented Analysis, Data Oriented Analysis

and Prototyping Analysis), Requirements Specification, Requirement Validation, and Requirement Management.

Unit-4:

Software Design: (14 Hrs.)

Software Design Process, Characteristics of a Good Design**,** Design Principles, Modular Design (Coupling and Cohesion), Software Architecture, Design

Methodologies (Function Oriented Design and Object Oriented Design), Structured Design Methodology (SDM), Transaction Analysis and Logical Design;

Coding: Coding principles, Coding process, Code verification and documentations.

Unit-5:

Software Testing: (14Hrs)

Testing Fundamentals, Test Planning, Black Box Testing, White Box Testing, Levels of Testing, Debugging Approaches

Quality of Software: Quality Concept, Quality Factors, Verification and Validation, Quality Assurance Activities, Quality Standards: Capability Maturity Model (CMM), ISO 9000, Six Sigma.

Maintenance: Software Maintenance, Maintenance Process Models and Reengineering.

Text Books:

1. Software Engineering: Concepts and Practices- Ugrasen Suman, Cengage Learning Publications.
2. Fundamentals of Software Engineering-Rajib Mall, PHI, New Delhi.

Reference Books

1. An Integrated Approach to S/w Engineering- Pankaj Jalote, Narosa Publishing House.
2. Software Engineering- Ian Sommerville, Pearson Education, New Delhi.
3. Software Engineering Concepts-Richard E. Fairly, Tata McGraw Hill Inc. New York.

e- Resources & other digital material

https://www.javatpoint.com/software-engineering-tutorial

COURSE OUTCOMES: Upon successful completion of the course, the student will be able to

CO1: Define and develop s/w projects from requirement gathering to implementation.
CO2: Obtain knowledge about principles and practices of software engineering.
CO3: Focus on the fundamentals of modelling a software project.
CO4: Obtain knowledge about estimation and maintenance of software systems
CO5: Design test cases, schedules and perform testing for SQA

CO-PO MAPPING MATRIX:

Mapping	P01	P02	P03	P04	P05	P06	P07	P08	P09	P010	P011	P012
C01	1	2	1	-	-	-	3	-	2	3	2	1
C02	-	2	3	1	2	2	2	-	1	2	1	-
C03	-	-	2	-	3	3	1	-	2	1	3	-
C04	1	3	2	2	2	2	3	-	-	2	2	-
C05	-	-	2	-	3	2		2	2	-	2	-

CO-PSO MAPPING MATRIX:

	PSO-01	PSO-02
C01	2	1
C02	3	1
C03	3	2
C04	3	1
C05	2	1

II Year – II Semester

L	T	P	C
3	0	0	3

ADVANCED DATA STRUCTURES

Course Objectives:

1. To impart the knowledge on various hashing techniques.
2. To help the students to learn Priority Queues and its applications.
3. To demonstrate the students about the operations of Efficient Search Trees.
4. To make the student to understand various shortest path algorithms in graphs.
5. To make the students to learn the use of Digital Search Structures and pattern matching algorithms.

Syllabus

UNIT - I **12 Hours**

Hashing: Static Hashing, Hash Functions, Secure Hash Functions, Collision Resolution Techniques-, Dynamic Hashing.

UNIT - II **12 Hours**

Priority Queues (Heaps): Binary Heaps, Applications of Heap, Binomial Heaps.

UNIT - III **14 Hours**

Efficient Search Trees: Self-balancing Binary Search Trees, AVL Tree, Red-Black Tree, B-tree, B+ Tree.

UNIT - IV **10 Hours**

Graph Algorithms: Elementary Graph Operations, Graph Algorithms: Minimal Cost Spanning Tree, Shortest Path.

UNIT - V **12 Hours**

Tries: Digital Search Trees(Tries), Operations, Different types of Tries **Pattern Matching Algorithms**.

Text Books:

1. Advanced Data Structures, Reema Thareja, S. Rama Sree, Oxford University Press, 2017.

2. Fundamentals of Data Structures in C, Horowitz, Sahni, Anderson-Freed, Second Edition, 2008.

Reference Books:

1. Advanced Data Structures, Peter Brass, Cambridge University Press, 2008.
2. Data Structures and Algorithms, A. V. Aho, J. E. Hopcroft, and J. D. Ullman, Pearson, 2002.
3. Data Structures and Algorithm Analysis in C, Mark Allen Weiss, Second Edition, Pearson.

Micro Syllabus of Advanced Data Structures

UNIT I :

Hashing: Static Hashing, Hash Functions, Secure Hash Functions, Collision Resolution Techniques-, Dynamic Hashing.

Unit	Module	Micro Content
UNIT I	Hash Functions, Securing Hash Functions	Hash Function
		Division Method
		Mid Square Method
		Folding Method
		Universal Hash Function
	Collision Resolution Techniques	Linear Probing
		Quadratic Probing
		Double Hashing
		Rehashing
		Separate Chaining
	Dynamic Hashing	Introduction
		Dynamic Hashing using Directories
		Directory less Dynamic Hashing
	Additional Topics	Implementation of Hash Table

UNIT - II

Priority Queues: Binary Heaps, Applications of Heap, Binomial Heaps.

Unit	Module	Micro Content
UNIT II	Priority Queue	Introduction, Types of Priority Queues
	Binary Heap	Properties of Heaps, Representation
		Operations: Insert, Delete, ExtractMin, BuildHeap, Reheapify
		Applications of Heap : Heap Sort, Prims Algortihm
	Binomial Heap	Properties of Binomial Tree, Representation
		Operations on Binomial Heap: Insert, Extract Min, Delete, DecreaseKey
		Applications of Binomial Heap
	Additional Topics	Huffman Coding

UNIT - III

Efficient Search Trees: Self-balancing Binary Search Trees, AVL Tree, Red-Black Tree, B-tree, B+ Tree.

Unit	Module	Micro Content
UNIT III	Self-Balancing Binary Search Trees (BSTs)	Binary Search Trees recap
		Necessity of Self Balancing BST
	AVL Trees	Properties of AVL Trees, Representation
		Rotations of AVL Trees: LL, RR, LR, and RL rotations.
		Operations: Insert, Delete, search
	Red – Black Tree	Properties of Red – Black Tree, Representation
		Operations : Insertion, Deletion
	B – Tree	Properties of B – Tree, representation
		Operations on B – Tree: Insertion, Deletion, Search
	B^+ Tree	Properties of B^+ Tree, representation
		Operations on B^+ Tree: Insertion, Deletion, Search
	Additional Topic	AVL Sort

UNIT - IV

Graph Algorithms: Elementary Graph Operations, Graph Algorithms: Minimal Cost Spanning Tree, Shortest Path.

Unit	Module	Micro Content
UNIT IV	Elementary Graph Operations	Recap of Graphs
		Connected Components, Bi connected components
	Minimal Cost Spanning Tree Algorithms	Prims Algorithm (recap)
		Kruskal's Algorithm (recap)
		Sollin's (Boruvka's) Algortihm
	Shortest Path Algorithm	Dijkstra's Algorithm
		Bellman Ford Algorithm
	All Pair Shortest Path Algorithms	Transitive Closure : Warshall's Algorithm
		Shortest Path: Floyd's Algorithm
	Additional Topic	Implementation of DAG

UNIT V

Tries: Digital Search Trees(Tries), Operations, Different types of Tries

Pattern Matching Algorithms.

Unit	Module	Micro Content
UNIT V	Tries	Digital Search Trees
		Operations on Trie: Insertion, Deletion, Search

		Binary Tries
		Compressed Binary Trie
		Suffix Trie
		PATRICIA
	Pattern Matching Algorithms	Brute Force Algorithm
		Boyer Moore Algorithm
		Knuth Morris Pratt Algorithm
	Additional Topic	Rabin – Karp Algorithm

Course Outcomes:

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

CO1: Summarize various hashing techniques. **(Understand)**

CO2: Identify the trade-offs of implementation of priority queues. **(Understand)**

CO3: Implement efficient search trees. **(Understand)**

CO4: Appreciate the importance and significance of graph algorithms in building and solving real world applications. **(Analyse)**

CO5: Use algorithms for pattern matching in a text. (Apply)

CO-PO mapping Table with justification

Mapping	P01	P02	P03	P04	P05	P06	P07	P08	P09	P010	P011	P012
C01	3	2	2	1	-	-	-	-	-	-	-	2
C02	3	2	2	2	-	-	-	-	-	-	-	2
C03	3	3	3	2	-	-	-	-	-	-	-	2
C04	3	2	2	1	-	-	-	-	-	-	-	2
C05	3	2	2	1	-	-	-	-	-	-	-	2

CO-PSO MAPPING:

Mapping	PS01	PSO2
C01	3	2
C02	3	2
C03	3	2
C04	3	2
C05	3	2

****

II Year – II Semester

L	T	P	C
3	0	0	3

OPERATING SYSTEMS

Course Objectives:

1. Study the basic concepts and functions of operating system
2. Learn about Processes, Threads and Scheduling algorithms
3. Understand the principles of concurrency and Deadlocks
4. Learn various memory management schemes
5. Study I/O management and File systems

UNIT-I **12 Hours**

Introduction to Operating System Concepts: What Operating Systems do, Computer System Organization, Functions of Operating systems, Types of Operating Systems, Operating Systems Services, System calls, Types of System calls, Operating System Structures, Distributed Systems, Special purpose systems.

UNIT-II **12 Hours**

Process Management: Process concept, Process State Diagram, Process control block, Process Scheduling- Scheduling Queues, Schedulers, Scheduling Criteria, Scheduling algorithms and their evaluation, Operations on Processes, Inter-process Communication.

Threads: Overview, User and Kernel threads, Multi-threading Models.

UNIT-III **12 Hours**

Concurrency: Process Synchronization, The Critical- Section Problem, Peterson's Solution, Synchronization Hardware, Semaphores, Monitors, and Classic Problems of Synchronization.

Principles of deadlock: System Model, Deadlock Characterization, Methods for Handling Deadlocks: Deadlock Prevention, Detection and Avoidance, Recovery form Deadlock.

UNIT- IV **12 Hours**

Memory Management: Logical vs physical address space, Swapping, Contiguous Memory Allocation, Paging, Structures of the Page Table, Segmentation.

Virtual Memory Management: Virtual memory overview, Demand Paging, Page-Replacement & its algorithms, Allocation of Frames, Thrashing.

UNIT-V **12 Hours**

File system Interface: The concept of a file, Access Methods, Directory structure, files sharing, protection.

File System implementation: File system structure, Allocation methods, and Free-space management.
Mass-storage structure: overview of Mass-storage structure, Disk scheduling, Swap space management.

Text Books:

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

Reference Books:

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

e- Resources & other digital material

- https://en.wikipedia.org/wiki/Operating_system
- https://www.tutorialspoint.com/operating_system/

Micro Syllabus of Operating Systems

UNIT I : Introduction to Operating System Concepts: What Operating Systems do, Computer System Organization, Functions of Operating systems, Types of Operating Systems, Operating Systems Services, System calls, Types of System calls, Operating System Structures, Distributed Systems, Special purpose systems.		
Unit	**Module**	**Micro Content**
	What Operating Systems do	User View, System View, Defining Operating Systems.
	Computer System Organization	Computer-system operation, Storage structure, i/o structure.

UNIT I	Functions of Operating systems	Process Management, Memory Management, File Management, I/O Management, Protection, Security, Networking.
	Types of Operating Systems	Batch processing, Multiprogramming, Timesharing, Distributed, Real time, Multi user, Multi-tasking, Embedded, Mobile operating system.
	Operating Systems Services	User interface, Program execution, I/O operations, File system manipulation, Communication, Error Detection.
	System calls, Types of System calls	Process control, File management, Device management, Information maintenance, and Communication maintenance, Protection and security maintenance system calls.
	Operating System Structures	Simple Structure Approach, Layered Approach, Microkernel Approach, Modules Approach.
	Distributed Systems	About Distributed Systems.
	Special purpose systems	Real Time Embedded Systems, Multimedia Systems, And Handheld Systems.

UNIT - II

Process Management: Process concept, Process State Diagram, Process control block, Process Scheduling- Scheduling Queues, Schedulers, Scheduling Criteria, Scheduling algorithms and their evaluation, Operations on Processes, Inter-process Communication.

Threads: Overview, User and Kernel threads, Multi-threading Models.

Unit	Module	Micro Content
UNIT II	Process concept	Define process, process in memory.
	Process State Diagram	Process states, diagram of process states.
	Process control block	Process state, process number, program counter, CPU registers, CPU switch from process to process, memory management information, accounting information, I/O status information.
	Process Scheduling	Introduction to process scheduler.
	Scheduling Queues	Job queue, ready queue, device queue, queueing diagram.
	Schedulers	Importance of scheduler, long term scheduler, short term scheduler, medium term scheduler, degree of multiprogramming, i/o bound process, cpu-bound process, swapping.
	Scheduling Criteria	Throughput, Turnaround time, Waiting Time, Response time.
	Scheduling algorithms	First-Come First-Served (FCFS) Scheduling, Shortest-Job-First(SJF) Scheduling, Priority Scheduling, Round Robin(RR) Scheduling, Multiple-Level Queue Scheduling, Multilevel Feedback Queue Scheduling.
	Evaluation of Scheduling algorithms	Deterministic modelling, Queuing models, Simulations and Implementation.

	Operations on Processes	Process creation, Process termination.
	Inter-process Communication	Shared memory systems, Message passing systems.
	Threads: Overview	Definition of thread, single threaded process, multithreaded process, benefits.
	Multi-threading Models	User and Kernel threads, many-to-one model, one-to-one model, many-to-many model.

UNIT-III

Concurrency: Process Synchronization, The Critical- Section Problem, Peterson's Solution, Synchronization Hardware, Semaphores, Monitors, and Classic Problems of Synchronization.

Principles of deadlock: System Model, Deadlock Characterization, Methods for Handling Deadlocks: Deadlock Prevention, Detection and Avoidance, Recovery form Deadlock.

Unit	Module	Micro Content
UNIT III	Process Synchronization	What is synchronization, why is it required, cooperating processes, race condition.
	Critical- Section Problem	Critical section, entry section, remainder section, mutual exclusion, progress, bounded waiting.
	Peterson's Solution	Software based solution to critical section between two processes.
	Synchronization Hardware	Locking, test and set instructions, mutual exclusion implementation with test and set, compare and swap instructions, mutual exclusion implementation with compare and swap.
	Semaphores	Semaphore usage, counting and binary semaphore, semaphore implementation, deadlock and starvation.
	Monitors	Structure of monitors, monitors vs semaphores, monitor usage, implementing a monitor using semaphores, dining-philosophers solution using monitors.
	Classic Problems of Synchronization	Bounded-buffer problem, reader-writer problem, dining-philosophers problem.
	Principles of deadlock: System Model	Deadlock definition, resources, request-use-release of resources.
	Deadlock Characterization	Necessary conditions for occurrence of deadlock, Resource allocation graph.
	Deadlock Prevention	Mutual exclusion, hold and wait, no-preemption, circular wait.
	Deadlock Detection	Graph algorithm, Banker's algorithm.
	Deadlock Avoidance	Safe state, Graph algorithm, Banker's algorithm.
	Recovery form Deadlock	Process termination, resource pre-emption.

UNIT- IV

Memory Management: Logical vs physical address space, Swapping, Contiguous Memory Allocation, Paging, Structures of the Page Table, Segmentation.

Virtual Memory Management: Virtual memory overview, Demand Paging, Page-Replacement & its algorithms, Allocation of Frames, Thrashing

Unit	Module	Micro Content
	Memory Management	Base register, limit register, protection with base and limit register.

UNIT IV	Logical vs physical address space	Logical address, memory address register, physical address, dynamic relocation using relocation register.
	Swapping	Swapping of two processes using a disk as backing store, swapping on mobile systems.
	Contiguous Memory Allocation	Memory protection, memory allocation, fragmentation.
	Paging	Basic method for implementing paging, paging hardware, TLB, protection, shared pages.
	Structure of the Page Table	Hierarchical paging, hashed page tables, inverted page tables.
	Segmentation	Basic method, segmentation hardware.
	Virtual memory overview	Virtual memory, virtual address space.
	Demand Paging	Demand paging technique, basic concepts, steps in handling page fault, locality of reference.
	Page-Replacement & its algorithms	Need for page replacement, page replacement techniques: FIFO, Optimal, LRU, LRU Approximation, Counting based.
	Allocation of Frames	Minimum number of frames, allocation algorithms: equal, proportional, global vs local allocation, non-uniform memory access,
	Thrashing	Cause of thrashing, working set model.

UNIT-V

File system Interface: The concept of a file, Access Methods, Directory structure, files sharing, protection.
File System implementation: File system structure, Allocation methods, and Free-space management.
Mass-storage structure: overview of Mass-storage structure, Disk scheduling, Swap space management.

Unit	Module	Micro Content
UNIT V	File Concept	File - attributes, operations, types, structure.
	Access Methods	Sequential, Direct, other access methods.
	Directory structure	Typical file system organization, storage structure, single level directory, two-level, tree-structured, acyclic-graph, general graph directory.
	Files sharing	Multiple users, remote file system, Consistency semantics.
	Protection	Types of access, access control.
	File system structure	File systems, basic file system, layered file system, file organization module, logical file system, FCB.
	Allocation methods	Contiguous, linked, indexed, efficiency of these methods.
	Free-space management	Free-space list, bit vector, linked list, grouping, counting.
	Overview of Mass-storage structure	Magnetic disks, solid state disks.
	Disk scheduling	FCFS, SSTF, SCAN,C-SCAN, LOOK,C-LOOK.
	Swap space management	Swap-space use, location

Course Outcomes:

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

CO-1: Understand the structure and functionalities of Operating System
CO-2: Demonstrate the concept of Process, Threads and CPU Scheduling Algorithms
CO-3: Use the principles of Concurrency to solve Synchronization problems
CO-4: Demonstrate various methods for handling Deadlocks
CO-5: Infer various Memory Management Techniques
CO-6: Illustrate File System Implementation

CO-PO Mapping Matrix:

CO/ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	P011	PO12
CO1	2	3	3	-	-	-	-	-	-	-	-	-
CO2	3	3	3	1	2	-	-	-	-	-	-	-
CO3	2	2	3	-	2	-	-	-	-	-	-	-
CO4	2	2	3	-	2	-	-	-	-	-	-	-
CO5	3	3	3	-	2	-	-	-	-	-	-	-
CO6	1	1	1	-	-	-	-	-	-	-	-	-

CO-PSO Mapping Matrix:

	PSO-1	PSO-2
CO1	3	2
CO2	3	--
CO3	2	--
CO4	2	2
CO5	3	2
CO6	1	--

****

II Year – II Semester

L	T	P	C
3	0	0	3

DATABASE MANAGEMENT SYSTEMS

Course Objectives:

1. Study the basic concepts and importance of Database Management Systems
2. Learn and understand the conceptual design of database and information retrieval
3. Learn various commands and writing of queries for information retrieval
4. Understand the concepts of Database design
5. Study of internal storage and its access

Syllabus

Unit-I: Introduction (10hrs)
Introduction to Database, Applications of Database, Purpose of Database, View of Data, Data Independence, Data Models, Users of Database, DBA, Query Processor, Storage Manager, Database Architecture

Unit-II: Conceptual Design & Relational Query Languages (14 hrs)
Conceptual Design of Database using ER Model, Notations, Types of attributes, Relation, Mapping Constraints, Features of ER Diagram, Weak Entity Set, Examples of Conceptual Design

Relational Algebra: Selection, Projection, Set Operations, Rename, Cartesian-Product, Join, Outer Join, Examples

Relational Calculus: Tuple Relational Calculus and Domain Relational Calculus

Safety Expressions

Unit-III: SQL & PL/SQL (14 hrs)
SQL Commands: DDL, DML, TCL, DCL

Types of Constraints (Primary, Alternate, Not Null, Check, Foreign), Basic form of SQL query, joins, outer joins, set operations, group operations, various types of queries, PL/SQL (Cursor, Procedures, Functions, Packages, Triggers…)

Unit-IV: Database Design (10 hrs)
Database Design: Normalization, Purpose of Normalization, Functional Dependency, Closure, 1NF, 2NF, 3NF, BCNF, MVFD, 4NF, Join Dependency, 5NF

Why NoSQL? Importance of NoSQL

Unit-V: Transaction, Data Recovery & Storage Management (12 hrs)
Transaction Management: ACID Properties of Transactions, Conflict & View serializability, Lock based protocols, Time Stamp based protocol, Thomas Write Rule, Validation Based Protocol, Deadlock detection, Deadlock avoidance, Deadlock prevention: wait-die and wound-wait

Recovery Management: Types of failures, ideal storage, Log, Log records, log based recovery techniques, Shadow Paging, ARIES

File Organization & Indexing: Types of File Organizations, Primary Indexing, Secondary Indexing, Multi-level Indexing, Hash Indexing, Tree Indexing

Text Books:

1. Data base System Concepts,5/e, Silberschatz, Korth, TMH

2. Introduction to Database Systems, CJ Date, Pearson

Reference Books

1. Data base Management Systems, Raghurama Krishnan, Johannes Gehrke, and TATA McGraw Hill 3rd Edition

2. Fundamentals of Database Systems, ElmasriNavate Pearson Education

Micro Syllabus of Database Management Systems

UNIT - I : INTRODUCTION

Introduction to Database, Applications of Database, Purpose of Database, View of Data, Data Independence, Data Models, Users of Database, DBA, Query Processor, Storage Manager, Database Architecture

Unit	Module	Micro Content
UNIT I	Introduction to Database	Definitions of data, database and information
		History of data
		Importance of databases over file systems
		Applications of Database
		Purpose of Database
		View of Data
		Data Independence
		Data Models
		Users of Database
		DBA
		Query Processor
		Storage Manager
		Database Architecture

UNIT – II: Conceptual Design & Relational Query Languages

Conceptual Design of Database using ER Model, Notations, Types of attributes, Mapping Constraints, Features of ER Diagram, Weak Entity Set, Examples of Conceptual Design
Relational Algebra: Selection, Projection, Set Operations, Rename, Cartesian-Product, Join, Outer Join, Examples
Relational Calculus: Tuple Relational Calculus and Domain Relational Calculus
Safety Expressions

Unit	Module	Micro Content
UNIT II	Conceptual Design	ER Model
		Notations
		Types of attributes
		Mapping Constraints
		Features of ER Diagram
		Weak Entity Set
		Examples of Conceptual Design
	Relational Algebra	Selection
		Projection
		Set Operations
		Rename
		Cartesian-Product
		Join
		Outer Join
		Safety Expressions
	Relational Calculus	Tuple Relational Calculus
		Domain Relational Calculus
		Safety Expressions

UNIT – III: SQL & PL/SQL

SQL Commands: DDL, DML, TCL, DCL

Types of Constraints (Primary, Alternate, Not Null, Check, Foreign), Basic form of SQL query, joins, outer joins, set operations, group operations, various types of queries, PL/SQL (Cursor, Procedures, Functions, Packages, Triggers)

Unit	Module	Micro Content
UNIT III	SQL Commands	DDL
		DML
		TCL
		DCL
	Types of Constraints	Primary
		Alternate
		Not Null
		Check
		Foreign
	SQL Queries	Basic
		Joins
		Set operations
		Group operations
		Various types of queries
	PL/ SQL	Cursor
		Procedures
		Functions
		Packages
		Triggers

UNIT – IV: Database Design

Database Design: Normalization, Purpose of Normalization, Functional Dependency, Closure, 1NF, 2NF, 3NF, BCNF, MVFD, 4NF, Join Dependency, 5NF. Why NoSQL?, Importance of NoSQL

Unit	Module	Micro Content
UNIT IV	Database Design	Normalization
		Purpose of Normalization
		Functional Dependency
		Closure
		1NF
		2NF
		3NF
		BCNF
		MVFD
		4NF
		Join Dependency
		5NF
	NoSQL	Why NoSQL?
		Importance of NoSQL
		Overview of NoSQL tools

UNIT - V: Transaction, Data Recovery & Storage Management

Transaction Management: ACID Properties of Transactions, Conflict & View serializability, Lock based protocols (2PLP, Tree & Multiple Granularity), Time Stamp based protocol, Thomas Write Rule, Validation Based Protocol, Deadlock detection, Deadlock avoidance, Deadlock prevention: wait-die and wound-wait

Recovery Management: Types of failures, ideal storage, Log, Log records, log based recovery techniques, Shadow Paging, ARIES File Organization & Indexing: Types of File Organizations, Primary Indexing, Secondary Indexing, Multi-level Indexing, Hash Indexing, Tree Indexing.		
Unit	**Module**	**Micro Content**
UNIT V	Transaction Management	ACID Properties of Transactions
		Conflict & View serializability
		Lock based protocols (2PLP, Tree & Multiple Granularity)
		Time Stamp based protocol, Thomas Write Rule
		Validation Based Protocol
		Deadlock detection
		Deadlock avoidance
		Deadlock prevention: wait-die and wound-wait
	Recovery Management	Types of failures
		Ideal storage
		Log, Log records, log based recovery techniques
		Shadow Paging
		ARIES
	File Organization & Indexing	Types of File Organizations
		Primary Indexing
		Secondary Indexing
		Hash Indexing: Static and Dynamic
		Tree Indexing

Course Outcomes:

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

CO1: To **understand** the basics of database systems and applications

CO2: To **construct** logical design of database and information retrieval

CO3: To **demonstrate** relational model practically (Structured Query Language)

CO4: To **demonstrate** and relate normalization for database design

CO5: To **outline** the necessity of transaction management, recovery management, file organization & indexing

CO-PO Mapping Matrix:

Mapping	P01	P02	P03	P04	P05	P06	P07	P08	P09	P010	P011	P012
C01	2	--	--	--	--	--	3	--	--	--	--	--
C02	3	2	2	--	--	--	--	--	--	--	--	--
C03	3	2	1	--	3	--	--	--	--	--	--	--
C04	3	2	1	--	--	--	--	--	--	--	--	--
C05	2	--	--	--	--	--	--	--	--	--	--	--

CO-PSO Mapping Matrix:

Mapping	PS01	PSO2
C01	1	--
C02	--	2
C03	2	3
C04	1	3
C05	1	--

****

II Year – II Semester

L	T	P	C
3	0	0	3

COMPUTER ORGANIZATION

Course Objectives:

- To understand basic structures of computers and to understand various machine instructions.
- To understand basic structures of computers and to understand various machine instructions.
- To analyse ALU & I/O organization of a computer.
- To understand various memory systems.
- To analyse functionalities done by processing unit and also learn micro programmed control.

Syllabus

Unit – I: Basic Structure of a Computer and Machine Instructions.

Introduction, History of Computer Generations, Functional unit, Basic Operational concepts, Bus structures, System Software, Performance. Number representation: Fixed Point and Floating Point representation. Instruction and Instruction Sequencing: Register Transfer Notation, Assembly Language Notation, Basic Instruction Types

Unit – II: Addressing modes and types of Instructions

Addressing Modes, Basic Input/output Operations, and role of Stacks and Queues in computer programming equation.

Component of Instructions: Logical Instructions, shift and Rotate Instructions. Type of Instructions: Arithmetic and Logic Instructions, Branch Instructions, Addressing Modes, Input/output Operations

Unit – III: **Basic building blocks for the ALU:**

Adder, Subtractor, Shifter, Multiplication and division circuits. Accessing I/O Devices, Interrupts: Interrupt Hardware, Enabling and Disabling Interrupts, Handling Multiple Devices, Direct Memory Access, Buses: Synchronous Bus, Asynchronous Bus, Interface Circuits, Standard I/O Interface: Peripheral Component Interconnect (PCI) Bus, Universal Serial Bus (USB)

Unit – IV: The Memory Systems

Basic memory circuits, Memory System Consideration, Read- Only Memory: ROM, PROM, EPROM, EEPROM, Flash Memory, Associative Memory, Cache Memories: Mapping Functions, INTERLEAVING, Secondary Storage: Magnetic Hard Disks, Optical Disks.

Unit – V: Processing unit

Fundamental Concepts: Register Transfers, Performing an Arithmetic or Logic Operation, Fetching a Word from Memory, Execution of Complete Instruction, Hardwired Control, MICRO

PROGRAMMED CONTROL: Microinstructions, Micro program Sequencing, Wide Branch Addressing Microinstructions with next –Address Field.

Text Books:

1. Computer Organization, Carl Hamacher, ZvonksVranesic, SafeaZaky, 5th Edition, McGraw Hill.
2. Computer Architecture and Organization by William Stallings, PHI Pvt. Ltd., Eastern Economy Edition, Sixth Edition, 2003

Reference Books

1. Computer Architecture and Organization, John P. Hayes, 3rd Edition, McGraw Hill.
2. Computer System Architecture by M Morris Mano, Prentice Hall of India, 2001

Micro Syllabus of Computer Organization

UNIT I: Basic Structure of a Computer and Machine Instructions.

Unit	Module	Micro Content
UNIT I	Introduction	Introduction, History of Computer Generations, Functional unit
		Basic Operational concepts, Bus structures, System Software, Performance
	Number representation	Integer - unsigned, signed (sign magnitude, 1's complement, 2's complement);
		Characters - ASCII coding, other coding schemes;
		Real numbers - fixed and floating point, IEEE754 representation
	Instruction and Instruction Sequencing	Register Transfer Notation
		Assembly Language Notation
		Basic Instruction Types

UNIT – II: Addressing modes and types of Instructions

Unit	Module	Micro Content
UNIT II	Addressing modes	Addressing Modes
		Basic Input/output Operations
		Role of Stacks and Queues in computer programming equation
	Component of Instructions	Logical Instructions
		Shift and Rotate Instructions.
	Type of Instructions	Arithmetic and Logic Instructions, Branch Instructions, Addressing Modes, Input/output Operations

UNIT – III: Basic building blocks for the ALU

Unit	Module	Micro Content
UNIT III	Basic Building blocks	Adder, Subtractor, Shifter, Multiplication and division circuits. Accessing I/O Devices
	Interrupts	Interrupt Hardware, Enabling and Disabling Interrupts,

		Handling Multiple Devices, Direct Memory access
	Buses	Synchronous Bus, Asynchronous Bus, Interface Circuits
	Standard I/O Interface	Peripheral Component Interconnect (PCI) Bus, Universal Serial Bus (USB)

UNIT - IV - The Memory Systems

Unit	Module	Micro Content
UNIT IV	Main Memory	Basic memory circuits, Memory System Consideration, Read- Only Memory: ROM, PROM, EPROM, EEPROM, Flash Memory, and Associative Memory.
	Cache Memories	Mapping Functions
		INTERLEAVING
	Secondary Storage	Magnetic Hard Disks, Optical Disks.

UNIT V - Processing unit

Unit	Module	Micro Content
UNIT V	Fundamental Concepts	Register Transfers, Performing an Arithmetic or Logic Operation, Fetching a Word from Memory
		Execution of Complete Instruction, Hardwired Control
	Micro Programmed Control	Microinstructions, Micro program Sequencing, Wide Branch Addressing Microinstructions with next –Address Field.

Course Outcomes:

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

CO-1: understand basic structures of computers and to **understand** various machine Instructions.

CO-2: learn and use the addressing modes and types of instructions.

CO-3: analyse I/O organization of a computer.

CO-4: understand various memory systems.

CO-5: Able to **analyse** functionalities done by processing unit and also learn micro programmed control.

CO-PO Mapping Matrix:

Course Outcome	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
CO1	2	-	-	-	-	-	-	-	-	-	-	-
CO2	2	-	2	-	-	-	-	-	-	-	-	-
CO3	-	2	2	-	-	-	-	-	-	-	-	-
CO4	-	2	2	-	-	-	-	-	-	-	-	-
CO5	2	2	2	1	-	-	-	-	-	-	2	2

CO-PSO Mapping Matrix:

	PSO-1	PSO-2
CO1	-	2
CO2	-	2
CO3	2	-
CO4	2	-
CO5	2	2

****

II Year – II Semester

L	T	P	C
0	0	3	1.5

ADVANCED DATA STRCTURES LAB

Course Objectives:

1. Ability to apply computational thinking to a diverse set of problems.
2. Ability to adapt to new challenges and computational environments. Proficiency in the design and implementation of algorithms.

List of experiments

Prerequisites: Solve the following problems in Hackerrank

- Time Conversion
- Balanced Brackets
- Queue using 2 Stacks
- Cycle Detection

UNIT I

1 Implement a dictionary (Division Method)
2 Implement Linear Probing on a dictionary
3 Implement Quadratic Probing on a dictionary
4 Implement Double Hashing.
5 Implement Separate Chaining.
6 **String Pairs / Anagram (Hackerrank)**

UNIT II

7 Implement Binary Heap Operations.
8 **Minimize the Sum (Hackerrank)**
9 **Implement Expression Tree.**

UNIT III

10 Implement Operations on Binary Search Tree non recursive.
11 Implement AVL Tree.

UNIT IV

12 Implement Prims Algorithm
13 Implement Krushkal's Algorithm
14 Implement Sollin's Algorithm
15 Implement Dijkstra's Algorithm
16 Implement shortest path between all pair of vertices.
17 **Implement Island Strikes. (Hackerrank)**
18 **Implement Pawn Moves. (Hackerrank)**

UNIT V

19 Implement Brute force pattern matching algorithm.

20 Implement Boyer-Moore pattern matching algorithm.

21 Implement Knuth-Morris pattern matching algorithm.

22 **Implement Counting Numeric sub sequences. (Hackerrank)**

Course Outcomes:

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

CO1: Select the most appropriate data structure and defend the selection.
CO2: Appropriately solve a variety of computational problems.
CO3: Communicate their results and describe an algorithm.

CO-PO mapping Table with justification

Mapping	**P01**	**P02**	**P03**	**P04**	**P05**	**P06**	**P07**	**P08**	**P09**	**P010**	**P011**	**P012**	**PS01**	**PSO2**
C01	2	1	2	-	-	-	-	-	-	-	-	-	3	2
C02	3	2	1	-	-	-	-	-	-	-	-	-	3	2
C03	3	2	2	-	-	-	-	-	-	-	-	-	3	2
C04	3	2	3	-	-	-	-	-	-	-	-	-	3	2
C05	3	2	2	-	-	-	-	-	-	-	-	-	3	2

******

II Year – II Semester

L	T	P	C
0	0	3	1.5

DATABASE MANAGEMENT SYSTEMS LAB

Course Objectives:

1. To familiarize the participant with the distinctions of database environments towards an information-oriented framework
2. To give a good formal foundation on the relational model of data
3. To present SQL and procedural interfaces to SQL comprehensively

List of experiments:

SQL

1. Queries for Creating, Dropping, and Altering Tables, Views, and Constraints **[CO1]**
2. Queries to facilitate acquaintance of Built-In Functions, String Functions, Numeric Functions, Date Functions and Conversion Functions. **[CO1]**
3. Queries using operators in SQL **[CO2]**
4. Queries to Retrieve and Change Data: Select, Insert, Delete, and Update **[CO2]**
5. Queries using Group By, Order By, and Having Clauses **[CO2]**
6. Queries on Controlling Data: Commit, Rollback, and Save point **[CO2]**
7. Queries to Build Report in SQL *PLUS **[CO2]**
8. Queries on Joins and Correlated Sub-Queries **[CO2]**
9. Queries on Working with Index, Sequence, Synonym, Controlling Access, and Locking Rows for Update, Creating Password and Security features **[CO2]**

PL/SQL

1. Write a PL/SQL Code using Basic Variable, Anchored Declarations, and Usage of Assignment Operation **[CO3]**
2. Write a PL/SQL Code Bind and Substitution Variables. Printing in PL/SQL **[CO3]**
3. Write a PL/SQL block using SQL and Control Structures in PL/SQL **[CO3]**
4. Write a PL/SQL Code using Cursors, Exceptions and Composite Data Types **[CO3]**
5. Write a PL/SQL Code using Procedures, Functions, and Packages FORMS **[CO4]**
6. Write a PL/SQL Code Creation of forms for any Information System such as Student Information System, Employee Information System etc. **[CO4]**
7. Demonstration of database connectivity **[CO4]**

Course Outcomes:

CO1: To create database for user (Creation of Database)
CO2: To solve various SQL queries for user defined schemas
CO3: To generalize PL/ SQL blocks
CO4: To illustrate the usage of user defined packages

CO-PO mapping Matrix:

Mapping	P01	P02	P03	P04	P05	P06	P07	P08	P09	P010	P011	P012	PS01	PSO2
C01	1	--	3	--	--	--	--	--	--	--	--	--	3	2
C02	3	2	1	1	--	--	--	--	--	--	--	--	1	3
C03	2	1	1	--	--	--	--	--	--	--	--	--	1	--
C04	2	--	--	--	--	--	--	--	--	--	--	--	1	--

****

II Year – II Semester

L	T	P	C
3	0	0	0

PROFESSIONAL ETHICS AND HUMAN VALUES

Course Objectives:

- To create an awareness on Engineering Ethics and Human Values.
- To instil Moral and Social Values and Loyalty
- To appreciate the rights of others.
- To create awareness on assessment of safety and risk

Syllabus

Unit I: HUMAN VALUES:

Morals, Values and Ethics-Integrity-Work Ethic-Service learning – Civic Virtue – Respect for others –Living Peacefully –Caring –Sharing –Honesty -Courage-Cooperation–Commitment – Empathy –Self Confidence Character –Spirituality.

LEARNING OUTCOMES:

1. Learn about morals, values & work ethics.
2. Learn to respect others and develop civic virtue.
3. Develop commitment
4. Learn how to live peacefully

Unit II: ENGINEERING ETHICS:

Senses of 'Engineering Ethics-Variety of moral issued –Types of inquiry –Moral dilemmas – Moral autonomy –Kohlberg's theory-Gilligan's theory-Consensus and controversy –Models of professional roles-Theories about right action-Self-interest -Customs and religion –Uses of Ethical theories –Valuing time –Cooperation –Commitment.

LEARNING OUTCOMES:

1. Learn about the ethical responsibilities of the engineers.
2. Create awareness about the customs and religions.
3. Learn time management
4. Learn about the different professional roles.

Unit III: ENGINEERING AS SOCIAL EXPERIMENTATION

Engineering As Social Experimentation –Framing the problem –Determining the facts –Codes of Ethics –Clarifying Concepts –Application issues –Common Ground -General Principles – Utilitarian thinking respect for persons

.

LEARNING OUTCOMES:

1. Demonstrate knowledge to become a social experimenter.
2. Provide depth knowledge on framing of the problem and determining the facts.
3. Provide depth knowledge on codes of ethics.
4. Develop utilitarian thinking

UNIT IV: ENGINEERS RESPONSIBILITY FOR SAFETY AND RISK:

Safety and risk –Assessment of safety and risk –Risk benefit analysis and reducing risk-Safety and the Engineer-Designing for the safety-Intellectual Property rights(IPR).

COURSE OUTCOMES:

1. Create awareness about safety, risk & risk benefit analysis.
2. Engineer's design practices for providing safety.
3. Provide knowledge on Intellectual Property Rights.

UNIT V: GLOBAL ISSUES

Globalization –Cross culture issues-Environmental Ethics –Computer Ethics –Computers as the instrument of Unethical behaviour –Computers as the object of Unethical acts –Autonomous Computers-Computer codes of Ethics –Weapons Development -Ethics and Research –Analysing Ethical Problems in research.

LEARNING OUTCOMES:

1**.** Develop knowledge about global issues.
2. Create awareness on computer and environmental ethics
3. Analyse ethical problems in research.
4. Give a picture on weapons development.

Text Books:

1. "Engineering Ethics includes Human Values" byM.Govindarajan, S.Natarajananad, V.S.SenthilKumar-PHI Learning Pvt. Ltd-2009

2. "Engineering Ethics" by Harris, Pritchard and Rabins, CENGAGE Learning, India Edition, 2009.

3. "Ethics in Engineering" by Mike W. Martin and Roland Schinzinger –Tata McGraw-Hill– 2003.

4. "Professional Ethics and Morals" by Prof.A.R.Aryasri, Dharanikota Suyodhana-Maruthi Publications.

5. "Professional Ethics and Human Values" by A.Alavudeen, R.Kalil Rahman and M.Jayakumaran-LaxmiPublications.

6. "Professional Ethics and Human Values" by Prof.D.R.Kiran

7. "Indian Culture, Values and Professional Ethics" by PSR Murthy-BS Publication

COURSE OUTCOMES

Students will be able to:

CO1: Identify and analyse an ethical issue in the subject matter under investigation or in a relevant field in a real-world situation or practice

CO2: Articulate what makes a particular course of action ethically defensible and assess their own ethical values and the social context of problems

CO3: Identify ethical concerns in research and intellectual contexts, including academic integrity, use and citation of sources, the objective presentation of data, and the treatment of human subjects

CO4: Demonstrate knowledge of ethical values in non-classroom activities, such as service learning, internships, and field work

CO5: Integrate, synthesize, and apply knowledge of ethical dilemmas and resolutions in academic settings, including focused and interdisciplinary research.

****

II Year – II Semester

L	T	P	C
0	0	2	1.5

SOCIALLY RELEVANT PROJECT

PREAMBLE:

VVIT conforming to the standards, procedures initiated and steered by the AICTE, NBA, NAAC and other statutory bodies, gives utmost importance to the ***Promotion of social science research***. In this regard, students are encouraged to pursue projects in socially relevant domains by taking challenging problems that when solved will increase in the sophistry of the mankind in society. The to-be-engineers-of-society are urged to conduct cutting edge projects in various fields of social sciences that have theoretical, conceptual, and methodological and policy implications which prop up the society at large. These socially relevant projects are made as mandatory practical course in the B.Tech Curriculum of every stream and a nice guidance will be given by the processors to inculcate the philanthropic culture in the engineering posterity.

DOMAIN OF SOCIAL SCIENCES:

Following are the domains in which VVIT encourages students to pursue data, requirements analysis through implantation of a model of the project.

- Environment
- Energy
- Materials
- Computing
- Telecommunications
- Defense
- Healthcare
- Agriculture and other interesting areas that are even tangentially connected to the society.

GUIDELINES:

Every student must do the socially relevant project either individually or team as per the guidelines in the **Annexure** given.

ASSESSMENT:

The Project review panel of individual departments and Institute will assess the quality of projects based on the

1. Quality of Literature survey
2. Novelty in the topic relevance to the society and specialization
3. Understanding of the topic
4. Quality of Report and Oral Presentation
5. Efficiency in implementation
6. Scalability, Portability and ability to fuse the project with other systems

A sum of 1.5 Credits will be awarded for those who successfully complete the project and even promoted to present the project in social projects expositions etc. competitions.

****

III Year I semester

L	T	P	C
3	0	0	3

Formal Languages and Automata Theory

Course Objectives:

- To learn how to design Automata's and machines
- To understand the Regular Language and Finite Automata Relation
- To learn fundamentals of Grammars and Languages
- To understand the relation between Contexts free Languages, PDA and TM
- To learn how to design Turing machines

Course Outcomes:

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

CO1: Comprehends the behaviour of finite automata and designs finite automata for problems. **(Understand)**

CO2: Discovers and demonstrates the classification of languages and design of regular grammar. **(Apply)**

CO3: Infers about context free grammars and their capabilities. **(Understand)**

CO4: Differentiates deterministic and non-deterministic machines. **(Analyse)**

CO5: Infers about computation functions with Turing machines. **(Apply)**

UNIT-I **8 Hours**

Finite Automata: Need of Automata theory, strings, Alphabet, Language, Operations, Finite state machine Definitions, finite automation model, Transition Systems, Acceptance of a String, DFA, Design of DFAs, NFA, Design of NFA, Equivalence of DFA and NFA, Conversion of NFA into DFA, Finite Automata with Є-Transitions, Equivalence between NFA with and without ε transitions, NFA to DFA conversion, minimization FA Finite Automata with output-Mealy and Moore Machines.

UNIT-II **10 Hours**

Regular Languages: Regular Expressions, Regular Sets, Identity Rules, Finite Automata and Regular Expressions, Inter Conversion, Equivalence between FA and RE, Pumping Lemma of Regular Sets, Closure Properties of Regular Sets, Grammars, Classification of Grammars, Chomsky Hierarchy Theorem, Right and Left Linear Regular Grammars, Equivalence and Inter Conversion between RG and FA.

UNIT-III **10 Hours**

Context Free Grammar: Context Free Grammar, Leftmost and Rightmost Derivations, Parse Trees, Ambiguous Grammars, Simplification of Context Free Grammars-Elimination of Useless Symbols, Є-Productions and Unit Productions, Normal Forms-Chomsky Normal Form and Greibach Normal Form, Pumping Lemma, Closure Properties.

UNIT- IV **10 Hours**

Pushdown Automata: Definition, Model, Graphical Notation, Instantaneous Description, Language Acceptance of Pushdown Automata, Design of Pushdown Automata, Deterministic and Non – Deterministic Pushdown Automata, Equivalence of Pushdown Automata and Context Free Grammars, Conversion.

UNIT-V **10 Hours**

Turning Machine & Computability Theory: Definition, Model, Representation of TMs-Instantaneous Descriptions, Transition Tables and Transition Diagrams, Language of a TM, Design of TMs, Types of TM, Universal TM, Decidable and Un-decidable Problems, Halting Problem of TMs, Post's Correspondence Problem, Modified PCP, Classes of P and NP, NP-Hard and NP-Complete Problems.

Text Books:

1. Introduction to Automata Theory, Languages and Computation, J. E. Hopcroft, R. Motwani and J. D. Ullman, 3rd Edition, Pearson, 2008.
2. Theory of Computer Science-Automata, Languages and Computation, K. L. P. Mishra and N. Chandrasekharan, 3rd Edition, PHI, 2007

Reference Books:

1. Elements of Theory of Computation, Lewis H.P. & Papadimition C.H., Pearson /PHI.
2. Formal Languages and automata Theory, K.V.N.Sunitha and N.Kalyani TMH, 2010.

Micro Syllabus of Advanced Data Structures

UNIT I: Finite Automata: Need of Automata theory, strings, Alphabet, Language, Operations, Finite state machine Definitions, finite automation model, Transition Systems, Acceptance of a String, DFA, Design of DFAs, NFA, Design of NFA, Equivalence of DFA and NFA, Conversion of NFA into DFA, Finite Automata with Є-Transitions, Equivalence between NFA with and without ε transitions, NFA to DFA conversion, minimization FA Finite Automata with output-Mealy and Moore Machines.		
UNIT-1	Basic Concepts	Need of Automata theory
		strings
		Alphabet
		Language
		Operations
	Finite Automata Introduction	Finite state machine Definitions
		finite automation model
		Transition Systems
		Acceptance of a String
	DFA & NFA	DFA
		Design of DFAs
		NFA
		Design of NFA
		Equivalence of DFA and NFA
		Conversion of NFA into DFA
		Finite Automata with Є-Transitions
	Equivalence and Inter conversions	Equivalence between NFA with and without ε transitions
		NFA to DFA conversion
		minimization FA
		Finite Automata with output-Mealy and Moore Machines.

<table>
<tr><td colspan="3">UNIT – II: Regular Languages: Regular Expressions, Regular Sets, Identity Rules, Finite Automata and Regular Expressions, Inter Conversion, Equivalence between FA and RE, Pumping Lemma of Regular Sets, Closure Properties of Regular Sets, Grammars, Classification of Grammars, Chomsky Hierarchy Theorem, Right and Left Linear Regular Grammars, Equivalence and Inter Conversion between RG and FA.</td></tr>
<tr><td rowspan="12">UNIT-2</td><td rowspan="3">Introduction</td><td>Regular Expressions</td></tr>
<tr><td>Regular Sets</td></tr>
<tr><td>Identity Rules</td></tr>
<tr><td rowspan="3">FA & RL</td><td>Finite Automata and Regular Expressions</td></tr>
<tr><td>Inter Conversion</td></tr>
<tr><td>Equivalence between FA and RE</td></tr>
<tr><td rowspan="2">Properties of RL</td><td>Pumping Lemma of Regular Sets</td></tr>
<tr><td>Closure Properties of Regular Sets</td></tr>
<tr><td rowspan="5">Regular Grammars</td><td>Grammars</td></tr>
<tr><td>Classification of Grammars</td></tr>
<tr><td>Chomsky Hierarchy Theorem</td></tr>
<tr><td>Right and Left Linear Regular Grammars</td></tr>
<tr><td>Equivalence and Inter Conversion between RG and FA.</td></tr>
<tr><td colspan="3">UNIT – III: Context Free Grammar: Context Free Grammar, Leftmost and Rightmost Derivations, Parse Trees, Ambiguous Grammars, Simplification of Context Free Grammars-Elimination of Useless Symbols, Є-Productions and Unit Productions, Normal Forms-Chomsky Normal Form and Greibach Normal Form, Pumping Lemma, Closure Properties.</td></tr>
<tr><td rowspan="11">UNIT-3</td><td rowspan="3">Introduction</td><td>Context Free Grammar</td></tr>
<tr><td>Leftmost most Derivations</td></tr>
<tr><td>Rightmost Derivations</td></tr>
<tr><td rowspan="3">Ambiguity</td><td>Ambiguity</td></tr>
<tr><td>Parse Trees</td></tr>
<tr><td>Ambiguous Grammars</td></tr>
<tr><td rowspan="3">Simplification of Context Free Grammars</td><td>Elimination of Є-Productions</td></tr>
<tr><td>Elimination of Useless Symbols</td></tr>
<tr><td>Elimination of Unit productions</td></tr>
<tr><td rowspan="2">Normal Forms</td><td>Normal Forms-Chomsky Normal Form</td></tr>
<tr><td>Greibach Normal Form</td></tr>
<tr><td rowspan="2">CFG Properties</td><td>Pumping Lemma</td></tr>
<tr><td>Closure Properties.</td></tr>
<tr><td colspan="3">UNIT - IV: Pushdown Automata: Definition, Model, Graphical Notation, Instantaneous Description, Language Acceptance of Pushdown Automata, Design of Pushdown Automata, Deterministic and Non – Deterministic Pushdown Automata, Equivalence of Pushdown Automata and Context Free Grammars, Conversion.</td></tr>
<tr><td rowspan="7">UNIT-4</td><td rowspan="5">Introduction</td><td>Definition of PDA</td></tr>
<tr><td>Model</td></tr>
<tr><td>Graphical Notation</td></tr>
<tr><td>Instantaneous Description</td></tr>
<tr><td>Language Acceptance of Pushdown Automata</td></tr>
<tr><td rowspan="2">Designing PDA</td><td>Design of Pushdown Automata</td></tr>
<tr><td>Deterministic and Non – Deterministic Pushdown Automata</td></tr>
</table>

	Relating to CFG	Equivalence of Pushdown Automata and Context Free Grammars
		Inter Conversion.
UNIT V: Turning Machine & Computability Theory: Definition, Model, Representation of TMs-Instantaneous Descriptions, Transition Tables and Transition Diagrams, Language of a TM, Design of TMs, Types of TM, Universal TM, Decidable and Un-decidable Problems, Halting Problem of TMs, Post's Correspondence Problem, Modified PCP, Classes of P and NP, NP-Hard and NP-Complete Problems.		
UNIT-5	Introduction	Definition
		Model
		Representation of TMs-Instantaneous Descriptions
		Transition Tables and Transition Diagrams
	Turing Machine Designing and types	Language of a TM
		Design of TMs
		Types of TM
		Universal TM
	Computability Theory	Decidable and Un-decidable Problems
		Halting Problem of TMs
		Post's Correspondence Problem
		Modified PCP
		Classes of P and NP
		NP-Hard and NP-Complete Problems.

III - I Semester

L	T	P	C
3	0	0	3

Design and Analysis of Algorithms

Course Objectives:

1. To provide an introduction to formalisms to understand, analyze and denote time complexities of algorithms
2. To introduce the different algorithmic approaches for problem solving through numerous example problems
3. To provide some theoretical grounding in terms of finding the lower bounds of algorithms and the NP-completeness

Course Outcomes:

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

CO1: Infer the divide-and-conquer paradigm and its context. Recite algorithms that employ this paradigm. Apply this paradigm to design algorithms for apt problems. Derive and solve recurrences describing the performance of divide-and-conquer algorithms.

CO2: Infer the greedy paradigm and its context. Recite algorithms that employ this paradigm. Apply this paradigm to design algorithms for apt problems.

CO3: Infer the dynamic-programming paradigm and its context. Recite algorithms that employ this paradigm. Apply this paradigm to design algorithms for apt problems.

CO4: Infer the backtracking paradigm and its context. Recite algorithms that employ this paradigm. Apply this paradigm to design algorithms for apt problems.

CO5: Infer the branch and bound paradigm and its context. Recite algorithms that employ this paradigm. Apply this paradigm to design algorithms for apt problems.

UNIT - I 10 Hours

Introduction: Algorithm Definition, Algorithm Specification, Performance Analysis, Performance Measurement, Asymptotic notations.

Divide and Conquer: General Method, Binary Search, Finding the Maximum and Minimum, Quick Sort.

UNIT - II 10 Hours

The Greedy Method: The General Method, Knapsack Problem, Single Source Shortest Path Problem, Optimal Storage on Tapes Problem, Optimal Merge Patterns Problem.

UNIT - III 10 Hours

Dynamic Programming: The General Method, 0/1 Knapsack Problem, Single Source Shortest Path – General Weights, All Pairs-Shortest Paths Problem, Traveling Salesperson Problem, String Editing Problem.

UNIT - IV 10 Hours

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

UNIT - V 10 Hours

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

NP-Hard and NP-Complete problems: Basic concepts, Cook's Theorem.

Text Books:

1. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, "Fundamentals of Computer Algorithms", 2nd Edition, Universities Press.

Reference Books:

1. Harsh Bhasin, "Algorithms Design & Analysis", Oxford University Press.
2. S. Sridhar, "Design and Analysis of Algorithms", Oxford University Press.

Micro Syllabus of Design and Analysis of Algorithms

UNIT I: Introduction: Algorithm Definition, Algorithm Specification, Performance Analysis, Performance Measurement, Asymptotic notation.
Divide and Conquer: General Method, Binary Search, Finding the Maximum and Minimum, Quick Sort.

Unit	Module	Micro Content
UNIT I	Algorithm Analysis	Definition of Algorithm, Properties of algorithm
		Algorithm Specification – Pseudo code Conventions
		Performance Analysis – time and space complexity
		Performance Measurement – step count and frequency count
		Asymptotic Notations – Big Oh, Omega, Theta
	Divide and Conquer	General Method
		Binary Search – Procedure, Example, Algorithm and Computing Time Complexity
		Finding the Maximum and Minimum - Procedure, Example, Algorithm and Computing Time Complexity
		Quick Sort - Procedure, Example, Algorithm and Computing Time Complexity

UNIT – II: The Greedy Method: The General Method, Knapsack Problem, Job Sequencing With Deadlines Problem, Single Source Shortest Path Problem, Optimal Merge Patterns Problem.

Unit	Module	Micro Content
UNIT II	Greedy Method	General Method
		Knapsack Problem - Description, Example, Algorithm.
		Single Source Shortest Path Problem - Description, Example, Algorithm.
		Optimal Storage on Tapes Problem - Description, Example, Algorithm.
		Optimal Merge Patterns Problem - Description, Example, Algorithm.

UNIT – III: Dynamic Programming: The General Method, 0/1 Knapsack Problem, Travelling Salesperson Problem, All Pairs-Shortest Paths Problem, Travelling Salesperson Problem, String Editing Problem.

Unit	Module	Micro Content
UNIT III	Dynamic Programming	The General Method
		0/1 Knapsack Problem - Description, Example.
		Single Source Shortest Path – General Weights - Description, Example, Algorithm.

		All Pairs-Shortest Paths Problem - Description, Example, Algorithm.
		Travelling Salesperson Problem - Description, Example.
		String Editing Problem - Description, Example.

UNIT – IV: Backtracking: The General Method, The N-Queens Problem, Sum of Sub-sets Problem, Graph Coloring Problem, Hamiltonian cycles Problem.

Unit	Module	Micro Content
UNIT IV	Backtracking	The General Method
		The N-Queens Problem - Description, State Space Tree, Algorithm.
		Sum of Subsets Problem - Description, Example, State Space Tree, Algorithm.
		Graph Coloring Problem - Description, Example, State Space Tree, Algorithm.
		Hamiltonian Cycles Problem - Description, Exam-ple, State Space Tree, Algorithm.

UNIT V: Branch and Bound: The General Method, FIFO Branch-and-Bound, LC Branch-and-Bound, 0/1 Knapsack Problem, Travelling Salesperson Problem.
NP-Hard and NP-Complete problems: Basic concepts, Cook's Theorem.

Unit	Module	Micro Content
UNIT V	Branch and Bound	The General Method
		FIFO Branch and Bound
		LC Branch and Bound
		0/1 Knapsack Problem - Description, Example.
		Traveling Salesperson Problem - Description, Example.
	NP-Hard and NP-Complete problems	Basics Concepts
		Cook's Theorem

III Year I Semester

L	T	P	C
3	0	0	3

Managerial Economic and Financial Analysis

Course Objective:

1. To equip the students with the basic inputs of managerial economics and demand concepts.
2. To understand the concepts of production and cost for various business decision.
3. To understand the different types of market, market structures & pricing strategies and their applications in business decision making and to know the different forms of Business organization and the concept of Business Cycles.
4. To understand the fundamental of accounting and analysis of accounting statements for managerial decision making.
5. To understand the concept of Capital, Capital Budgeting and the techniques used to evaluate Capital Budgeting proposals.

Course Outcomes: After completion of the course, students will be able to
CO1: To equipped with the knowledge of estimating the Demand and demand elasticities for a product.
CO2: The knowledge of understanding of the Input-Output-Cost relationships and estimation of the least cost combination of inputs.
CO3: To understand the nature of different markets and Price Output determination under various market conditions and also to have the knowledge of different Business Units.
CO4: To prepare Financial Statements and the usage of various Accounting tools for analysis.
CO5: To evaluate various investment project proposals with the help of capital budgeting techniques for decision making.

UNIT – I Introduction to Managerial Economics and demand Analysis: 10 Hrs
Definition of Managerial Economics –Scope of Managerial Economics and its relationship with other subjects –Concept of Demand, Types of Demand, Determinants of Demand- Demand schedule, Demand curve, Law of Demand and its limitations- Elasticity of Demand, Types of Elasticity of Demand and Measurement- Demand forecasting and Methods of forecasting.

UNIT - II Theory of Production and Cost Analysis: 10 Hrs
Production Function – Isoquant and Isocost, MRTS, Least Cost Combination of Inputs - Laws of Returns to scale - Internal and External Economies of Scale, Cost Analysis: Cost concepts, Cost & output relationship in short run & long run - Break-even Analysis (BEA)-Determination of Break-Even Point - Significance and limitations.

UNIT – III Introduction to Markets, Pricing Policies & Types of Business Organization and Business Cycles: 10 Hrs
Market Structures: Perfect Competition, Monopoly, Monopolistic competition and Oligopoly – Features – Price and Output Determination – Methods of Pricing: Average cost pricing, Limit Pricing, Market Skimming Pricing, and Internet Pricing: Flat Rate Pricing, Usage sensitive pricing and Priority Pricing. Features and Evaluation of Sole Trader, Partnership, Joint Stock Company – Business Cycles: Phases of Business Cycles.

UNIT – IV Introduction to Financial Accounting & Analysis: 10 Hrs
Financial Accounting and analysis: Accounting –significance -- Book Keeping-Double entry system –Journal- Ledger- Trial Balance- Final Accounts with simple adjustments.

Financial Statement Analysis through ratios: Ratio-analysis of financial statement using different ratios (Liquidity -Profitability- Solvency -Activity ratios).

UNIT - V Capital and Capital Budgeting: 10 Hrs

Capital Budgeting: Meaning of Capital-Capitalization-Meaning of Capital Budgeting-Time value of money- Methods of appraising Project profitability: Traditional Methods (payback period, accounting rate of return) and modern methods (Discounted cash flow method, Net Present Value method, Internal Rate of Return Method and Profitability Index).

Text Books:

1. Dr. A. R. Aryasri – Managerial Economics and Financial Analysis, TMH 2011.
2. Dr. N. Appa Rao, Dr. P. Vijay Kumar: 'Managerial Economics and Financial Analysis', Cengage Publications, New Delhi – 2011.
3. Prof. J.V. Prabhakara rao, Prof. P. Venkatarao. 'Managerial Economics and Financial Analysis', Ravindra Publication.

Reference Books:

1. V. Maheswari : Managerial Economics, Sultan Chand.
2. Suma Damodaran : Managerial Economics, Oxford 2011.
3. Dr. B. Kuberudu and Dr. T. V. Ramana : Managerial Economics & Financial Analysis, Himalaya Publishing House 2011.
4. Vanitha Agarwal : Managerial Economics, Pearson Publications 2011.
5. Sanjay Dhameja : Financial Accounting for Managers, Pearson.
6. Maheswari: Financial Accounting, Vikas Publications.
7. S. A. Siddiqui & A. S. Siddiqui: Managerial Economics and Financial Analysis, New Age International Publishers, 2012.

Micro Syllabus for Managerial Economics and Financial Analysis

UNIT – I Introduction to Managerial Economics and demand Analysis:
Definition of Managerial Economics –Scope of Managerial Economics and its relationship with other subjects –Concept of Demand, Types of Demand, Determinants of Demand- Demand schedule, Demand curve, Law of Demand and its limitations- Elasticity of Demand, Types of Elasticity of Demand and Measurement- Demand forecasting and Methods of forecasting.

Unit	Module	Micro Content
Unit I	Concept of Economics	Economics, Definitions of Economics
		Micro economics, Macro economics
		Scope of Micro & Macro Economics
		Difference Between Micro & Macro Economics
		Meaning & Definitions of Managerial Economics
	Concept of Managerial economics	Nature & scope of Managerial Economics
		Importance of Managerial Economics
		Difference between Economics & Managerial Economics
	relationship with other subjects	Linkage with other Disciplines
	Basic Economic tools of Managerial economics	Opportunity cost Principle, Incremental principle, Time perspective principle, Discounting Principle, Eqi marginal Principle
	Concept of Demand	What is Demand, Demand Analysis & Objectives
	Types of Demand	Demand distinctions, Demand function
		Factors determining demand
	Demand Schedule	Individual demand schedule, Market demand schedule

	Demand Curve	Individual demand curve, Market demand curve
	Law of Demand	Assumption of law of demand, Change in demand, Exceptions of law of demand, why does demand curve slope downwards.
	Elasticity of Demand, Types of Elasticity of Demand & Measurement	Meaning of elasticity of demand, types of Price and income elasticity of demand, factors effecting elasticity of demand, measurements of elasticity of demand, significance of elasticity of demand
	Demand fore casting	types of demand forecasting

UNIT - II Theory of Production and Cost Analysis:

Production Function – Isoquant and Isocost, MRTS, Least Cost Combination of Inputs - Laws of Returns to scale - Internal and External Economies of Scale, Cost Analysis: Cost concepts, Cost & output relationship in short run & long run - Break-even Analysis (BEA)-Determination of Break-Even Point - Significance and limitations.

Unit II	Theory of Production	Production function, Production process, importance of production, assumptions
	Isoquant and Isocost	Meaning and Types, properties
	MRTS, Least Cost Combination of Inputs	Schedule of Marginal rate of technical substitution, combination of different inputs
	Laws of Returns to scale	Schedule and graph
	Economies of scale	Internal and external
	Cost Analysis	Types of costs, cost & output relationship in short run and long run
	Break even Analysis	Uses, limitations of Break even analysis, Key terminology in Break analysis, Simple problems on BEP, graphical representation of Break even analysis.

UNIT – III Introduction to Markets, Pricing Policies & Types of Business Organization and Business Cycles:

Market Structures: Perfect Competition, Monopoly, Monopolistic competition and Oligopoly – Features – Price and Output Determination – Methods of Pricing: Average cost pricing, Limit Pricing, Market Skimming Pricing, and Internet Pricing: Flat Rate Pricing, Usage sensitive pricing and Priority Pricing. Features and Evaluation of Sole Trader, Partnership, Joint Stock Company – Business Cycles: Phases of Business Cycles.

Unit III	Market Structures	Meaning, definitions, types of market
	Perfect Competition	Features, price output determination under perfect competition
	Monopoly	Features, price output determination under perfect competition
	Monopolistic competition	Features, price output determination under perfect competition
	Oligopoly	features
	pricing	Methods of pricing and internet pricing
	Type of business organization: Sole trader	Features, Advantages & disadvantages, suitability
	Partnership	Features, Advantages & disadvantages, suitability
	Joint stock company	Features, Advantages & disadvantages, suitability
	Business cycle	Phases of business cycle

UNIT – IV Introduction to Financial Accounting & Analysis:

Financial Accounting and analysis: Accounting –significance -- Book Keeping-Double entry

system –Journal- Ledger- Trial Balance- Final Accounts with simple adjustments. **Financial Statement Analysis through ratios:** Ratio-analysis of financial statement using different ratios (Liquidity -Profitability- Solvency -Activity ratios).		
Unit IV	Financial Accounting	Meaning, definitions, objectives & significance, users of accounting, accounting cycle, GAAP.
	Book Keeping	Single and double entry book keeping, types of Accounting
	Journal	Features, Pro-forma, Advantages & Limitations, preparation of journal entries, simple problems
	ledger	Features, Pro-forma, Advantages & Limitations, preparation of ledger, simple problems.
	Trial Balance	Features, Pro-forma, Advantages & Limitations, preparation of Trial balance, simple problems.
	Final accounts	Trading account- Pro-forma, Simple problems
		Profit & Loss account- Pro-forma, Simple problems
		Preparation of balance sheet with simple adjustments
	Financial Statement Analysis through ratios	Ratio Analysis, uses and types of ratios, significance, analysis of financial statements using Liquidity -Profitability- Solvency -Activity ratios
UNIT - V Capital and Capital Budgeting: Capital Budgeting: Meaning of Capital-Capitalization-Meaning of Capital Budgeting-Time value of money- Methods of appraising Project profitability: Traditional Methods (payback period, accounting rate of return) and modern methods (Discounted cash flow method, Net Present Value method, Internal Rate of Return Method and Profitability Index).		
Unit V	Capital	What is capital, need of capital types of capital
		Types of fixed capital, types of working capital
	Capital Budgeting	Meaning, Nature & scope of capital budgeting
		Capital budgeting procedure, capital budgeting decisions, method of capital budgeting.
	Payback period	Meaning, formula, advantages & disadvantages, simple problems
	Accounting rate of return (ARR)	Meaning, formula, advantages & disadvantages, simple problems
	Net present value (NPV)	Meaning, formula, advantages & disadvantages, simple problems
	Profitability index (PI)	Meaning, formula, advantages & disadvantages, simple problems
	Internal rate of return (IRR)	Meaning, formula, advantages & disadvantages, simple problems

Co- Po mapping

	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**
CO 1	-	-	-	-	-	**1**	-	-	**1**	-	-	-
CO 2	-	-	-	-	-	**1**	-	**2**	**1**	**2**	**2**	**2**
CO 3	-	-	-	-	-	**1**	-	-	**1**	-	-	-
CO 4	-	-	-	-	-	**1**	-	**2**	**1**	**2**	**2**	-
CO 5	-	-	-	-	-	**1**	-	**2**	**1**	**2**	**3**	**3**

III Year I semester

L	T	P	C
3	0	0	3

Unix and Shell Programming

Course Objectives:

1. Introduce Unix Operating System and its features while exploring file system and security
2. Learn UNIX Filters related to text processing, communication and search utilities
3. Learn programming filters and interactive shell scripting
4. Learn shell programming constructs writing advanced scripts
5. Learn kernel programming on file operations and managing processes

Course Outcomes:

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

CO1: Infer the importance of UNIX operating system by learning salient features and using basic utilities **(Understand)**

CO2: Develop apt programming and non-programming filters **(Apply)**

CO3: Devise shell scripts using the syntactic constructs of shell for producing the desired effects. **(Create)**

CO4: Compose advanced shell scripts for string and array processing (**Create**)

CO5: Construct functions using system calls related to file and process control (**Create)**

UNIT-I **10 Hours**

Introduction to Unix OS, File Systems, Security and File Permissions, Introduction to Shells.

UNIT-II **10 Hours**

Filters, Communications, Regular Expressions, global regular expression and print (grep)

UNIT-III **10 Hours**

Stream editor (sed), Programming filter (awk), Interactive shell programming

UNIT- IV **10 Hours**

Shell Programming concepts, Advanced Shell Programming

UNIT-V **08 Hours**

Introduction System calls and Signals, File I/O, Files & Directories, Process control

Text Books:

1. UNIX and Shell Programming, Behrouz A, Forouzan and Richard F.Gilberg, Cengage Learning, 2003.
2. Advanced Programming in UNIX Environment, W.Richard Stevens, Stephen A Rago, 3rd Edition, Addison-Wesley Professional, 2013.

Reference Books:

1. UNIX and shell programming by B.M. Harwani, OXFORD university press.
2. Unix essentials by Sumitabha Das
3. Unix Shell Programming, Stephen G.Kochan, Patrick Wood, 3/e, Pearson

Micro Syllabus of UNIX and Shell Programming

UNIT I: Introduction to Unix OS, File Systems, Security and File Permissions, Introduction to Shells.		
Unit	**Module**	**Micro Content**
I	Introduction to Unix Operating System	Why Unix, Computer Systems, Unix Environment,
		Unix Structure, Accessing Unix
		Command basics, common commands,
		other useful commands
	File Systems	File names, file types, regular files, directories
		File system implementation
		Operations unique to directories
		Operations unique to regular files
		Operations common to both files and directories
	Security and Permissions	Users and groups, security levels
		Changing permissions
		User masks, changing ownership and group
	Introduction to Shells	Unix session, standard streams, redirection
		Pipes, tee command, command execution, command line editing
		Quotes, command substitution, job control, aliases
		Variables, predefined variables, options, shell/environment customization
UNIT – II : Filters, Communications, Regular Expressions, global regular expression and print(grep)		
Unit	**Module**	**Micro Content**
II	Filters	Filters and Pipes, concatenating pipes,
		displaying beginning and ending of files, cut, paste, sorting,
		Translating characters, files with duplicate lines, count characters, words or lines, comparing files
	Communications	User communication, Electronic mail
		Remote access
		File transfer
	Regular Expressions & grep	Atoms, operators
		grep operation, grep family
		Examples, searching for file content
UNIT – III : Stream editor(sed), Programming filter (awk), Interactive shell programming		
Unit	**Module**	**Micro Content**
III	Sed	Scripts, operation, addresses, commands-1
		Commands-part 2
		Applications, grep and sed
	Awk	awk execution, fields and records, scripts
		awk operation, patterns, actions
		Associative arrays, string functions, math functions, User-defined functions

		Using system commands in awk, applications
		awk and grep, sed and awk
	Interactive shells	Shell features, two special files, variables, output, input,
		exit status of a command, eval, environmental variables, options,
		Command history and execution process

UNIT - IV : Shell Programming, Advanced Shell Programming

Unit	**Module**	**Micro Content**
IV	Shell programming	Basic script concepts, expressions
		Decisions: making selections, repetition
		Special parameters and variables
		Changing positional parameters
		Argument validation
		Debugging scripts,
		Script examples
	Advanced shell programming	Variable evaluation and substitution
		String manipulation
		Here document , functions,
		arrays, signals
		Built-in commands, scripting techniques,
		Shell environment and script, script examples

UNIT V:
Introduction System calls and Signals, File I/O, Files & Directories, Process control

Unit	**Module**	**Micro Content**
V	Introduction System calls and Signals	System call and library functions, signals
	File I/O	Introduction to file I/O
		create, open, close
		Lseek(), read(), write()
		dup dup2, fcntl, ioctl
	Files and directories	File types, stat, lstat, fstat,
		File size, system calls operating on file/directories
	Process control	Introduction to unix processes, process identifiers, fork(), vfork(), exit()
		Wait(), waitpid(), exec family of functions

Subject Title: Advanced Java and Web Technologies					
Course Code:	**Year and Semester:** III Year I semester	L	T	P	C
Prerequisites: Prior knowledge of Java programming		2	2	0	3

Course Objectives:

1. To understand database connectivity through Java
2. To acquire knowledge about web application development
3. To introduce XML and processing of XML Data with Java.
4. To introduce Server-side programming with Java Servlets and JSP
5. To introduce Client-side scripting with Java script and AJAX.

Course Outcomes:

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

CO1: Summarise different JDBC drivers and their connectivity **(Remember)**

CO2: Quote different built-in and user defined tags used in HTML & XML **(Understand)**

CO3: Classify server side and client-side application development through Servlets **(Understand)**

CO4: Relate JSP tags with server sides codes **(Apply)**

CO5: Use client-side application development through Java Script. **(Apply)**

UNIT-I 8 Hours

JDBC Connectivity: JDBC connectivity, types of Jdbc Drivers, connecting to the database, JDBC Statements, JDBC Exceptions, Manipulations on the database.

UNIT-II 10 Hours

HTML Common tags- List, Tables, images, forms, Frames; Cascading Style sheets;

XML: Introduction to XML, Defining XML tags, their attributes and values, Document Type Definition, XML Schemas, Document Object Model, Extensible Style sheet Language and XSL Transformations, Parsing XML Data – DOM and SAX Parsers in java.

UNIT-III 10 Hours

Introduction to Servlets: Life cycle of a Servlet, deploying a servlet, The Servlet API, Reading Servlet parameters, Reading Initialization parameters, Handling Http Request & Responses, Using Cookies and Sessions.

UNIT- IV 10 Hours Introduction to JSP: The Anatomy of a JSP Page, JSP Processing, Declarations, Directives, Expressions, Code Snippets, implicit objects, Using Beans in JSP Pages, Using Cookies and session for session tracking.

UNIT-V 10 Hours

Client-side Scripting: Introduction to Javascript, Javascript language – declaring variables, scope of variables, functions. event handlers (onclick, onsubmit etc.), Document Object Model, Form validation, Simple AJAX application.

Text Books:

1. Internet and World Wide Web – How to program, Dietel and Nieto, Pearson.
2. Java Server Pages –Hans Bergsten, SPD O'Reilly.

Reference Books:

1. Chris Bates, "Web Programming, building internet applications", 2nd Edition, WILEY, Dreamtech, 2008.
2. Thomas A Powel, "The Complete Reference: AJAX", 1st Edition, Tata McGraw Hill, 2008.
3. Web Technologies, Uttam K Roy, Oxford University Press

Micro Syllabus of Advanced Java and Web Technologies

III B.Tech I Semester

UNIT I : JDBC Connectivity: JDBC connectivity, types of Jdbc Drivers, connecting to the database, JDBC Statements, JDBC Exceptions, Manipulations on the database.

.

Unit	Module	Micro Content
UNIT I	JDBC connectivity	Database Application
		Need and Objective of JDBC
		Steps to build JDBC Application
	types of Jdbc Drivers	Type1: JDBC-ODBC Bridge Driver
		Type2: partial Java driver.
		Type 3: pure Java driver for database middleware.
		Type 4: pure Java driver for direct-to-database.
	JDBC Statements	Statement
		Prepared Statement
		Callable Statement
	JDBC Exceptions	SQL Exception and Methods of SQL Exception
	Manipulations on the database	Execution of insert, delete and update queries

UNIT – II: HTML Common tags- List, Tables, images, forms, Frames; Cascading Style sheets;

XML: Introduction to XML, Defining XML tags, their attributes and values, Document Type Definition, XML Schemas, Document Object Model, and Extensible Style sheet Language and XSL Transformations, Parsing XML Data – DOM and SAX Parsers in java.

Unit	Module	Micro Content
UNIT II	HTML Common tags	Standard HTML Document Structure
		Basic Text Markup, Hyper Links
	List, Tables, images	Unordered List, Ordered -List and Definition List
		Table related tags, Formatting of Tables, Use of COL-SPAN and ROWSPAN
		<img> tag and its attributes
	forms, Frames	HTML 5 form elements, GET and POST Method
		Inline Frames
		Applications of Binomial Heap
	Cascading Style sheets	Levels of Style Sheets
		Style Specification Formats, Selector Forms
		The Box Model, Conflict Resolution
	Introduction to XML	Basic Building Blocks of XML
		Structure of XML Document, Well formed and Valid XML Document
		XML Parser
	Defining XML tags, their attributes and values	Defining XML tags, their attributes and values

	Document Type Definition	Elements of DTD
		Internal and External DTD
	Schemas	Defining XSD and Validating XML Document
		Type of Elements in XSD
		Compare DTD and Schema
	Document Object Model, Extensible Style sheet Language and XSL Transformations	Document Object Model
		Introduction to XSLT, XPATH
		XSLT Syntax, Elements of XSLT
		Dom Parser and Sax Parser
	Parsing XML Data – DOM and SAX Parsers in java.	Compare DOM and SAX Parser
		Implementation of DOM Parser and SAX Parser in Java

UNIT – III : Introduction to Servlets: Life cycle of a Servlet, deploying a servlet, The Servlet API, Reading Servlet parameters, Reading Initialization parameters, Handling Http Request & Responses, Using Cookies and Sessions.

.

Unit	Module	Micro Content
UNIT III	**Introduction to Servlets**	What is Servlet
		Life cycle of Servlet
	deploying a servlet	What is Web server
		Installation of Tomcat
		Deploying a Servlet in Tomcat
	The Servlet API	Most useful interfaces and classes in javax.servlet package
		Most useful interfaces and classes in javax.servlet.http package
	Reading Servlet parameters, Reading Initialization parameters	ServletRequest Interface
		ServletConfig Interface
	Handling Http Request & Responses	HttpServlet
		HttpServletRequest and HttpSerletResponse Interfaces
	Using Cookies and Sessions.	Session Tracking
		What is Cookie, SessionTracking using Cookies in servlets
		HttpSession Interface, Session Tracking using HttpSession in servlets

UNIT - IV : Introduction to JSP: The Anatomy of a JSP Page, JSP Processing, Declarations, Directives, Expressions, Code Snippets, implicit objects, Using Beans in JSP Pages, Using Cookies and session for session tracking.

Unit	Module	Micro Content
UNIT-IV	Introduction to JSP	Advantages of JSP over Servlet
	The Anatomy of a JSP	Directive Elements in

	Page	JSP Standard Actions
		Scripting Elements
	JSP Processing	JSP Architecture
		Deploying JSP in Tomcat
	implicit objects, Using Beans in JSP Pages	9 implicit objects in JSP
		Standard action jsp:useBean
	Using Cookies and session for session tracking	Session Tracking using Cookies in JSP
		Session Tracking using HttpSession in JSP

UNIT V : Client-side Scripting: Introduction to Java script, Java script language – declaring variables, scope of variables, functions. event handlers (onclick, onsubmit etc.), Document Object Model, Form validation, Simple AJAX application.

Unit	Module	Micro Content
UNIT V	Client-side Scripting	Introduction to Java script
		History of Java script
		Data types and Variables in Java script
		Scope of Variables
		Control Statements in Javascript
		Java script Objects
		Event Handlers
		HTML Dom
		Form Validation using regular Expressions
	Simple AJAX application	What is Ajax
		Ajax Technologies, Understanding XMLHttpRequest
		How Ajax Works, Ajax Example in Java

III-I Semester

L	T	P	C
0	0	3	1.5

Design and Analysis of Algorithms Lab

Course Objectives:

1. To learn fundamental algorithmic problems.
2. To understand methods of designing and analysing algorithms.
1. To know various designing paradigms of algorithms for solving real world problems.

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

CO1: Identify and apply the suitable algorithm for the given problem.
CO2: Design and implement efficient algorithms for a specified application.

List of experiments:

1. Write a program to find maximum and minimum values in a list using divide and conquer method.
2. Write a program to find optimal profit of Knapsack problem using Greedy method.
3. Write a program to find Optimal Merge Pattern for merging a list of files using Greedy Method.
4. Write a program to find shortest paths from a node to all nodes in a graph using Dynamic Programming method.
5. Write a program to find shortest paths between all pairs of nodes in a graph using Dynamic Programming method.
6. Write a program to find all possible solutions of N-Queens problem using Backtracking method.
7. Write a program to find all possible solutions of Sum of Sub Sets problem using Backtracking method.
8. Write a program to find all possible ways of coloring a graph withgiven colors using Backtracking method.
9. Write a program to find all Hamiltonian Cycles in a connected undirected graph using Backtracking method.

III –I semester

L	T	P	C
0	0	3	1.5

Unix and Shell Programming Lab

Course Objectives:

1. Learn UNIX Filters related to text processing, communication and search utilities
2. Learn programming filters and interactive shell scripting
3. Learn shell programming constructs writing advanced scripts
4. Learn kernel programming on file operations and managing processes

Course Outcomes:

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

CO1: Develop scripts compatible with different shells available under UNIX environment **(Apply)**

CO2: Develop scripts for automating the tasks of programmer during deployment and maintenance **(Apply)**

CO3: Develop scripts to automate task using programmable filters **(Apply)**

List of Shell Scripts:

1. Create a script that, given a user name, finds the home directory of the user using the /etc/passwd file.

Preparation:

- None

Script:

- **Script Name: findHomeDirectory.scr**
- **Arguments:** One, The user name.
- **Validation:** The minimum validation requirements are :
    i. Ensure that there is only one argument.
- **Body Section:** Create a script that, given the name of a user (as the only argument), prints the absolute pathname of the user's home directory

Testing the Script:

- Test the script with two or more arguments.
- Test the script with no arguments.
- Test the script with one argument.

Testing the Effect of the Script:

- Verify the script by using your user name.

2. Write a script that creates a file out of the /etc/passwd file.

Preparation:

- None

Script:

- **Script Name: newEtcPasswd.scr**
- **Arguments:** One, The name of the file.
- **Validation:** The minimum validation requirements are:
    i. Ensure that there is only one argument.
- **Body Section:** Create a script that makes a file out of the information in the /etc/passwd file using the following format.

```
User Name User Id Group ID Home Directory
- - - - - - - -   - - - - - -  - - - - - - -  - - - - - - - - - - - -
ram 234 23 /etc/usr/student/ram
- - - -
```

Testing the Script:

- Test the script with two or more arguments.
- Test the script with no arguments.
- Test the script with one argument that is not the name of a file.
- Test the script with one argument that is the name of a file.

Testing the Effect of the Script:

- Verify the file was created and contains the correct information and format.

3. In a C Program, there is only one comment format. All comments must start with an open comment token, /*, and end with a close comment token, */. C++ programs use the C tokens for comments that span several lines. Single-line comments start with two slashes (//). In either case, the start token can be anywhere on the line.

Write a script to change every single-line comment in a C++ source file that uses C program start and end comment tokens to a single-line comment starting with a C++ single-line token. The comment itself is to be unchanged.

Preparation:

- Create at least five C++ source files in your home directory. The files do not have to be real C++ source files; they can contain only a few lines of comments, some with C program tokens and some with C++ single-line tokens. Each program should have at least one multiple comment and at least one single-line comment that uses the C program tokens. Use one or more blank lines between comments. The name of the files should have C++ extension (.c++), such as file1.c++.

Script:

- **Script Name: commentType.scr**
- **Arguments:** None
- **Validation:** The minimum validation requirements are:
    i. Ensure that there is no argument.
- **Body Section:** Create a script that finds all files with extension (.c++) under your directory and change only the lines with comments. The name of the files should be preserved. If a file has the name file1.c++, the name still should be file1.c++ after the change.

Testing the Script:

- Test the script with one or two arguments.
- Test the script with no arguments.

Testing the Effect of the Script:

- Check to see if the comments are changed in the files.

4. Write a script to backup and archive a list of files.

Preparation:

- Create a file and type in it the list of files (in your home directory) that you want to back and archive
- Create a directory in which you will store the backed-up files and archive file.

Script

- **Script Name:** backup.scr
- **Arguments:** A filename and a directory. The filename holds the list of the files that should be backed-up. The directory is where the backed-up files should be stored.
- **Validation:** The minimum validation requirements are:
    i. Ensure that exactly two arguments are entered.
    ii. Check that the first argument is the name of a file exists
    iii. Check that the second argument is the name of the directory that exists

- **Body Section:** Create backup files for files listed in the first argument. The backup files should have the same name as the original file with the extension bak. They should be copied to the directory given as the second argument.

Testing the Script:

- Test the script with no arguments
- Test the script with one argument
- Test the script with three arguments
- Test the script with two arguments in which the first one is not the name of the file
- Test the script with two arguments in which the second one is the name of a file rather than a directory.
- Test the script with name of the file and the name of the directory you created in the preparation section.

Testing the Effect of the Script:

- Check the contents of the directory to be sure that the files are copied

5. Write a script that finds all soft links to a specific file.

Preparation:

- Create a file and type some junk in it.
- Make at least five soft links to this file using completely arbitrary names.

Script:

- **Script Name: softLinkFinder.scr**
- **Arguments:** A filename. The file for which we want to find the soft links.
- **Validation:** The minimum validation requirements are:
    i. Ensure that exactly one argument is entered.
    ii. Check that only argument is the name of a file and that the specified file exists.
- **Body Section:** Use ls -l and grep command to find all the soft links attached to $1 positional parameter. Note that a file of type soft link is distinguished by lower case l. Be sure to find the soft links to the file defined in $1 and not other files.

Testing the Script:

- Test the script with no arguments
- Test the script with one argument
- Test the script with one argument that is not a file
- Test the script with one valid argument.

Testing the Effect of the Script:

- Check to make sure all the soft links you created are included in the list of soft links.

6. Create a script that simulates the ls -l command but prints only three columns of our choice.

Preparation:

- None

Script:

- **Script Name: ls.scr**
- **Arguments:** Three numeric arguments defining the column number of the ls -l output to be printed in the order we specify.
- **Validation:** The minimum validation requirements are :
    i. Ensure that exactly three arguments are entered.

ii. Ensure that all three arguments are numeric
iii. Ensure that each argument is less than or equal to the actual number of columns in the ls -l command output.

- **Body Section:** Creates a new command that shows the output of the ls -l command to be printed in three columns in the order we like.

Testing the Script:

- Test the script with no arguments.
- Test the script with one argument.
- Test the script with two arguments.
- Test the script with three arguments, one of them nonnumeric.
- Test the script with three arguments, two of them nonnumeric.
- Test the script with three arguments, one of them too large.
- Test the script with three arguments,1 4 5
- Test the script with three arguments,3 7 1

Testing the Effect of the Script:

- None

7. Create a script that sends contents of a message file to everybody who logged in.

Preparation:

- Create a file of a short friendly message and mention that this is a test message that should be discarded by the receiver

Script:

- **Script Name: message.scr**
- **Arguments:** One argument, a message file.
- **Validation:** The minimum validation requirements are:
  i. Ensure that exactly one argument is entered.
  ii. Ensure that the argument is a readable filename.
- **Body Section:** Create a script that uses awk to create a temporary file containing the usernames of those users who are logged into the system at this moment. Then send the message contained in the first argument to every logged-in user. Note that a user who has logged in more than once should receive only one message.

Testing the Script:

- Test the script with no arguments.
- Test the script with two arguments.
- Test the script with one argument that is not a readable file.
- Test the script with one valid argument.

Testing the Effect of the Script:

- You should include yourself in the recipient list. Check to see if you have received the message.

8. Create a script that can be executed only from a specific terminal. This is done for security purposes. For example, a superuser may write scripts that can only be executed from his or her office and nowhere else.

Preparation:

- None

Script:

- **Script Name: security.scr**
- **Arguments:** None.
- **Validation:** The minimum validation requirements are:
  i. Ensure that no argument is entered.
- **Body Section:** Create a script that prints a friendly message. However, the script can be executed only for one terminal. You can use the name of the terminal you are

using when you write the script. If somebody uses the script from a terminal that is not authorized, the script is to exit immediately. Hint: Use the tty command to show your current terminal.

Testing the Script:

- Test the script with one argument.
- Test the script from right terminal.
- Log into the system using another terminal and test the script.

Testing the Effect of the Script:

- None

9. Create a script that finds each line in a file that contains a specified string.

Preparation:

- Create a file of at least 20 lines and insert a double quoted string, such as "hello," in several lines.

Script:

- **Script Name: search.scr**
- **Arguments:** Two arguments, the first is the string to be found; the second is the name of the file.
- **Validation:** The minimum validation requirements are:
  i. Ensure that exactly two arguments are entered.
  ii. Ensure that the second argument is the name of the file that exists and is not empty.
- **Body Section:** Create a script that uses grep and loops to find the line numbers in which the string is found. Note that grep should be applied to each line, not the whole file. The script should print the result in the following format:

Line Number: [Line contents]

Testing the Script:

- Test the script with no arguments.
- Test the script with one argument.
- Test the script with two argument but the second one is not a file.
- Test the script with two correct arguments.

Testing the Effect of the Script:

- Compare the results of your script with a printout of the file.

10. Create a script that compiles all C source files in your home directory and create executable files.

Preparation:

- Create at least five C source files in your home directory. The files do not have to be real C source files; at a minimum they should contain a comment line that contain a unique program name such as the following example:

/* .....file1.c .....*/

The name of the files should have a C source file extension (.c), such as file1.c.

Script:

- **Script Name: compile.scr**
- **Arguments:** Two arguments, the first is the string to be found; the second is the name of the file.
- **Validation:** The minimum validation requirements are :
  i. Ensure that there is no argument
- **Body Section:** Create a script that finds all files with extension (**.**c) under your home directory and compiles them one by one. Each executable file should have the same name as the source file except that the extension should be (**.**exe). For example,

if the source filename is file1.c, the executable filename should be file1.exe. Use the following command to compile:

cc -o executable_filename source_filename

Testing the Script:

- Test the script with one or two arguments.
- Test the script with no arguments.

Testing the Effect of the Script:

- Verify that executable files were created under your home directory.

11. Create a script that finds all files in subdirectories that have the same filename.

Preparation:

- Make several directories, at different levels, under your home directory. For example, make ~/A, ~/B, ~/C, ~/A/AA, ~/A/BB, ~/A/AA/AAA, and so on until you have at least 15 directories. Copy a small junk file named file1 under some of these directories; do not change its name. Copy another small junk file named file2 under some other directories. Copy a third junk file under several directories. Be sure that some directories get a combination of file1 and file2 or file1 and file3. In at least three of the directories, create a junk file with a unique name.

Script:

- **Script Name: duplicateName.scr**
- **Arguments:** None
- **Validation:** The minimum validation requirements are :
  i. Ensure that there is no argument.
- **Body Section:** Create a script that uses find and awk commands to create a list of files that are duplicated; use the full pathname for the duplicated filenames. Hint: Use a basename command and an array in awk. The output should look like the following example:

  file1: ~/A/file1 ~/A/AA/file1 ~/A/B/BB/BBB/file1
  file2: ~/B/file2 ~/C/file2
  . . .

Testing the Script:

- Test the script with one argument.
- Test the script with no arguments.

Testing the Effect of the Script:

- Use a recursive long list command to list the complete contents of your home directory. Verify the output of your script against the list command output.

12. Create a script that search for multiple occurrences of the specified string in each line.

Preparation:

- Create a file of at least 20 lines and insert a double quoted string, such as "hello," in several lines.
- Include two or three occurrences of the string in some lines.

Script:

- **Script Name: search.scr**
- **Arguments:** Two arguments, the first is the string to be found; the second is the name of the file.
- **Validation:** The minimum validation requirements are :
  i. Ensure that exactly two arguments are entered.
  ii. Ensure that the second argument is the name of the file that exists and is not empty.

- **Body Section:** Create a script that uses grep and loops to find the line numbers in which the string is found. Note that grep should be applied to each line, not the whole file. The script should print the result in the following format:

Line Number: [Line contents]

Testing the Script:

- Test the script with no arguments.
- Test the script with one argument.
- Test the script with two argument but the second one is not a file.
- Test the script with two correct arguments.

Testing the Effect of the Script:

- Compare the results of your script with a printout of the file.

III - I semester

L	T	P	C
0	0	3	1.5

Advanced Java and Web Technologies Lab

Course Objectives

At the end of the course the students will understand

1. Basic technologies to develop web documents.
2. Dynamic HTML Pages and Event handling mechanism.
3. XML and Web Servers.
4. Java Servlet technologies.

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

CO-1: Create static web pages using HTML, CSS, and JavaScript.
CO-2: Design dynamic Web Pages using client side scripting.
CO-3: Create XML documents and work with web servers to create web applications
CO-4: Write server side programs using Java Servlets and Jsp.

List of Programs

1. Develop and demonstrate a HTML5 document that illustrates the use of ordered list, unordered list, table, borders, padding, color, and the & tag.

2. Write HTML5 code to provide intra and inter document linking.

3. Create a web page with the following using HTML5:
   a. To embed an image map in a web page
   b. To fix the hot spots
   c. Show all the related information when the hot spots are clicked

4. Create a web page with all types of Cascading style sheets.

5. Create a web page with the following using CSS:
   a. Text shadows, rounded corners and box shadows.
   b. Linear and Radial gradients.
   c. Animation
   d. Transitions and Transformations.

6. Create a HTML5 form that interacts with the user. Collect first name, last name and date of birth and display that information back to the user.

7. Develop a HTML5 Form, which accepts any Mathematical expression. Write JavaScript code to evaluate the expression and Displays the result.

8. Create a HTML5 form that has number of Textboxes. When the form runs in the Browser fill the textboxes with data. Write JavaScript code that verifies that all textboxes has been filled. If a textboxes has been left empty, popup an alert indicating which textbox has been left empty.

9. Create a home page for "Cyber book stores" that will display the various books available, the authors and prices of the books. Include a list box that contains various subjects and a "submit" button, which displays information about the books on the subject required by the user.

10. Create a bank entry form using appropriate form elements. The account number must not be visible on the screen. The name and address must be stored in one place. There must be a text box showing the opening balance of the customer. The user should be able to make a choice of either a deposit (or) withdrawal transaction. Accordingly, when the user deposits (or) withdraws money, the opening balance must be updated using CREDIT/DEBIT button. The user should not be able to make any entries in the opening balance text box.

11. Using functions, write a JavaScript code that accepts user name and password from user. Check their correctness and display appropriate alert messages. Restrict the user to try only for a maximum of three times.

12. Create an HTML5 file for registration with three text fields name, mobile number and address. Write JavaScript to validate name, mobile number and address. Mobile number should be of 10 digits. Show alert message when user enter invalid entity.

13. Write a JavaScript code block using arrays and generate the current date in words, this should include the day, month and year.

14. Write a program to display a form that accepts student's name, age, father name. When age field receives its focus display message that age should be 18 to 25. After losing its focus from age field verify user entered in between given values or not display respective message

15. Create a web page using two image files, which switch between one another as the mouse pointer moves over the images. Use the mouseover and mouseut event handlers.

16. Perform the following using JavaScript
- To update the information into the array, in the "Click" event of the button "Update".
- To sort the elements of an array (Use array object)
- To find the duplicate elements of an array.

17. Demonstrate the following:
- String and Math objects
- Alphabetic and Numeric fields
- Calendar object.

18. Write an XML file which displays the book details that includes the following:
1) Title of book 2) Author name 3) Edition 4) Price
Write a DTD to validate the above XML file and display the details in a table (to do this use XSL).

19. Design an XML document to store information about a student in an engineering college. The information must include college id, Name of the College, Brach, Year of Joining, and e-mail id. Make up sample data for 3 students. Create a CSS style sheet and use it to display the document.

20. Create an XML document, which contains 10 users information. Implement a program, which takes User Id as an input and returns the user details by taking the user information from the XML document

21. Create tables in the database which contain the details of items (books in our case Like Book name, Price, Quantity, Amount) of each category. Modify your catalogue page in such a way that you should connect to the database and extract data from the tables and display them in the catalogue page using JDBC.

22. Using java servlets and JDBC store and retrieve the following information from a database:
a. Name b. Password c. Email id d. Phone number

23. Write a JSP program to conduct online examination and to display student mark list available in a database.

24. Demonstrate Cookie and Session Management in Servlets.

******

III Year II semester

L	T	P	C
2	2	0	3

Data Warehousing & Data Mining

Course Objectives:

1. Distinguishes the certainty of various classical approaches for mining data in warehouse.
2. Prepares students in identifying various problems and its corresponding approaches for mining data.
3. Outlines a student about merits and demerits of mining approaches contextually.

Course Outcomes:

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

CO1: Infers about Data Warehousing & Data Mining. **(Understand)**
CO2: Demonstrates Pre-processing Techniques before Data Mining. **(Applying)**
CO3: Infers Classification & recite different approaches. **(Analyzing)**
CO4: Infers Association Analysis & recite different approaches. **(Analyzing)**
CO5: Infers Cluster Analysis & recite different approaches. **(Analyzing)**

UNIT - I **8 Hours**
Introduction: Data Warehousing, Data Mining.

UNIT - II **10 Hours**
Data Pre-processing: An Overview, Data Cleaning, Data Integration, Data Reduction, Data Transformation and Data Discretization.

UNIT - III **10 Hours**
Classification: Basic concepts, algorithms, alternative techniques.

UNIT - IV **10 Hours**
Association Analysis: Basic Concepts and Algorithms.

UNIT - V **10 Hours**
Cluster Analysis: Basic Concepts and Algorithms.

TEXT BOOKS:

1. Introduction to Data Mining: Pang-Ning Tan & Michael Steinbach, Vipin Kumar, Pearson.
2. Data Mining concepts and Techniques, 3/e, Jiawei Han, Michel Kamber, Elsevier.

REFERENCE BOOKS:

1. Data Mining Techniques and Applications: An Introduction, Hongbo Du, Cengage Learning.
2. Data Mining : VikramPudi and P. Radha Krishna, Oxford.
3. Data Mining and Analysis - Fundamental Concepts and Algorithms; Mohammed J. Zaki, Wagner Meira, Jr, Oxford
4. Data Warehousing Data Mining & OLAP, Alex Berson, Stephen Smith, TMH.

Micro Syllabus of Data Warehousing & Data Mining

UNIT I : Introduction: Data Warehousing, Data Mining		
Unit	**Module**	**Micro Content**
UNIT I	Data Warehousing	Introduction
		Architecture
		OLAP vs OLTP
		Data cube
		Data cube Operations
	Data Mining	Introduction
		Kinds and Patterns of Data Mining
		Issues of Data Mining
		Statistical Descriptions
		Data Visualizations
		Similarity and Dissimilarity
UNIT – II: Data Pre-processing: An Overview, Data Cleaning, Data Integration, Data Reduction, Data Transformation and Data Discretization		
Unit	**Module**	**Micro Content**
UNIT II	Data Pre-processing	Data Cleaning: • Handling Missing values • Handling Noisy data
		Data Integration: • For nominal attribute • For numeric attribute
		Data Reduction: • Wavelet Transforms • PCS • Attribute subset selection • Numerosity reduction
		Data Transformation: • Strategies: smoothing, normalization etc • Normalization (Min Max, Z-score, Decimal Scaling) • Data Discretization
UNIT – III : Classification: Basic concepts, algorithms, alternative techniques		
Unit	**Module**	**Micro Content**
UNIT III	Classification Analysis	Basic Concepts
		General Approach to solving a classification problem
		Decision Tree Induction
		Working of Decision Tree
		building a decision tree
		methods for expressing an attribute test conditions
		measures for selecting the best split
		Algorithm for decision tree induction
		Bayes' Theorem
		Naïve Bayesian Classification

		Bayesian Belief Networks

UNIT - IV : Association Analysis: Basic Concepts and Algorithms

Unit	Module	Micro Content
UNIT IV	Association Analysis	Problem Definition
		Frequent Item Set generation
		Rule generation
		compact representation of frequent item sets
		FP-Growth Algorithm

UNIT V : Cluster Analysis: Basic Concepts and Algorithms.

Unit	Module	Micro Content
UNIT V	Cluster Analysis	Overview; Classification vs Clustering
		Different Types of Clustering
		Different Types of Clusters
		K-means
		K-Means: The Basic K-means Algorithm, K-means Additional Issues, Bisecting K-means, Strengths and Weaknesses
		Agglomerative Hierarchical Clustering
		Basic Agglomerative Hierarchical Clustering Algorithm
		DBSCAN: Traditional Density Center-Based Approach, DBSCAN Algorithm, Strengths and Weaknesses.

III - II Semester

L	T	P	C
3	0	0	3

Computer Networks

(Common to CSE & IT Branches)

Course Objectives:

1. To understand OSI and TCP/IP reference models and Example networks, characteristics of transmission media and classify multiplexing techniques
2. To understand the Error Control, Flow Control and Medium Access Control Protocols
3. To Compute optimal path using Routing Algorithms.
4. To understand the concepts of reliable unreliable transmission
5. To acquire the knowledge on various application layer protocols

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

CO1: Explain OSI and TCP/IP reference models and Example networks, characteristics of transmission media and classify multiplexing techniques (L2)
CO2: Summarize various Error Control, Flow Control techniques and Medium Access Control Protocols (L2)
CO3: Compute optimal path using Routing Algorithms. (L3)
CO4: Explain the concepts of reliable unreliable transmission (L2)
CO5: Illustrate the working of various application layer protocols (L3)

UNIT-I: Introduction to Computer Networks and Physical Layer 10 Hrs
Introduction: Network Topologies WAN, LAN, MAN. Reference models- The OSI Reference Model- the TCP/IP Reference Model - A Comparison of the OSI and TCP/IP Reference Models, Example Networks, Physical Layer – Fourier Analysis – Bandwidth Limited Signals – The Maximum Data Rate of a Channel Guided Transmission Media, Multiplexing: Frequency Division Multiplexing, Time Division Multiplexing, Code Division Multiplexing

UNIT-II: Data Link Layer 11 Hrs
Data Link Layer Design Issues, Error Detection and Correction, Elementary Data Link Control Protocols, Sliding Window Protocols, HDLC, PPP, Channel Allocation problem, Multiple Access Protocols, IEEE standards for Local Area Networks, WLAN, Bluetooth

UNIT– III: Network Layer 10 Hrs
Network Layer Design Issues, Routing Algorithms, Congestion Control Algorithms, Internet Protocol Header, IP Addresses, subnetting and super netting.

UNIT-IV: Transport Layer 8 Hrs
Transport Layer Design Issues, Connection Establishment, Connection Termination, Transport and User Datagram Protocols

UNIT – V: Application Layer 6 Hrs
Design Issues, DNS, WWW, HTTP/HTTPS, E-mail, FTP

Text Books:
1. Computer Networks, Andrew S Tanenbaum, Pearson, 5th Edition
2. Data Communications and Networking, Behrouz A Forouzan, Tata McGraw Hill, 4th Edition

Reference Book:
1. TCP/IP Protocol Suite, Behrouz A Forouzan**,** Tata McGraw Hill Edition, 3rd Edition

Web Resources:
1.https://youtube.com/playlist?list=PLbRMhDVUMngfpeFloB7kyiA40EptH1up
2. https://www.geeksforgeeks.org/computer-network-tutorials/
3.https://www.cisco.com/c/en/us/support/docs/ip/routing-information-protocol-rip/13788-3.html

MICRO SYLLABUS

UNIT-I: Introduction and Physical Layer
Introduction: Network Topologies WAN, LAN, MAN. Reference models- The OSI Reference Model- the TCP/IP Reference Model - A Comparison of the OSI and TCP/IP Reference Models, Example Networks, Physical Layer – Fourier Analysis – Bandwidth Limited Signals – The Maximum Data Rate of a Channel Guided Transmission Media, Digital Modulation and Multiplexing: Frequency Division Multiplexing, Time Division Multiplexing, Code Division Multiplexing

Unit	Module	Micro content	No of hrs
Introduction to Computer Networks and Physical Layer	Introduction	Uses of Computer Networks, Topologies, Types of Networks (LAN, MAN,WAN) Network Hardware, Network Software	2
	Reference Models	OSI and TCP/IP	2
	Example Networks	ARPANet, Novell Netware, ATM Networks	2
	Physical Layer	Design Issues, Maximum Data Rate of a Channel, Nyquist Theorem for a noiseless channel, Shannon Theorem for noisy channel	1
	Transmission Media	Guided and Unguided Transmission media	1
	Multiplexing	FDM,TDM,WDM,CDM	1

UNIT-II: Data Link Layer
Data Link Layer Design Issues, Error Detection and Correction, Elementary Data Link Control Protocols, Sliding Window Protocols, HDLC,PPP

Unit	Module	Micro content	No of hrs
Data Link Layer	Design Issues	Framing, Physical Addressing, Error Control, Flow Control, Access Control,	1
	Error Detection and Correction	VRC, LRC, CRC, Checksum, Single Bit Correction : Hamming Codes	1
	Flow Control	Elementary Data Link Control Protocols: An unrestricted Simplex, Simplex Stop and	3

		Wait, Stop Wait ARQ Sliding Window Protocols: 1-bit Sliding Win-dow, Sliding window using Go Back N, Sliding Window Using Selective Repeat	
	Example Data Link Control Protocols	HDLC, PPP	1
	Channel Allocation Problem	Static Channel Allocation, Dynamic Chan-nel Allocation	1
	Multiple Access Protocols	Aloha, CSMA, Collision Free Protocols,	1
	IEEE standards LAN Protocols	IEEE-802.3,802.4,802.5	2
	IEEE WLAN Proto-cols	IEEE 802.11, Bluetooth	1

UNIT-III: Network Layer

Network Layer Design Issues, Routing Algorithms, Congestion Control Algorithms, Internet Protocol Header, IP Addresses, subnetting and supernetting.

Unit	Module	Micro content	No of hrs
Network Layer	Design Issues	Connection Oriented and Connection less service, Comparison of Virtual Cir-cuit subnets and Datagram Networks	1
	Routing Algorithms	Shortest path, Flooding, Distance Vector Routing, Link State Routing, Hierar-chical Routing, Broadcast Routing, Mul-ticast Routing, Routing for Mobile Hosts	2
	IP Headers	IPV4 and IPV6	2
	IP Addresses	Classful IP Addressing, Classless IP Addressing, Types of IP Addresses Subnetting and Super netting	3

UNIT-IV: Transport Layer

Transport Layer Design Issues, Connection Establishment, Connection Termination, Transport and User Datagram Protocols,

Unit	Module	Micro content	No of hrs
Transport Layer	Design Issues	Design Issues, Pro-cess Addressing, Service Primitives	1
	TCP Phases	Connection Establishment, Connection Termination, Data Transfer	2
	Protocols	TCP, UDP, RTP	3

UNIT-V: Application Layer

Design Issues, DNS, WWW, HTTP/HTTPS, E-mail, FTP,

Unit	Module	Micro content	No of hrs
Application Layer	Design Issues	File Transfer and Access Manage-ment Network Virtual Terminals Mail Services	1
	DNS	DNS Name space, Resource Records, Name servers	1

	WWW	Architecture and overview, Static/Dynamic web pages,	1
	HTTP/HTTPS	HTTP Request and Response headers and methods	1
	E-mail	Architecture, User Agents, Message formats, Message Transfer Agents, SMTP, S/MIME, POP	1
	FTP	Communication over control Connection, Communication Over Data Connection, Anonymous FTP	1

CO-PO-PSO Mapping Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PPO2
CO-1	2										2		2	2
CO-2	2	2									2		2	2
CO-3	2	2	2								2	2	2	2
CO-4	-		2								2	2	2	2
CO-5	-		2								2	2	2	2

*****

III - II Semester

L	T	P	C
3	0	0	3

Artificial Intelligence

Course Objectives:

Course Objectives:

1) To have a basic proficiency in a traditional AI language including an ability to write simple to intermediate programs and an ability to understand code written in that language.
2) To improve analytical and problem-solving skills based on the characteristics of the problem using various heuristic search techniques and to improve designing and playing a game
3) To have knowledge on propositional calculus, proportional and predicate logic to understand few systems such as natural deduction, axiomatic system, etc.
4) To have an understanding of the basic issues of knowledge representation and blind and heuristic search, as well as an understanding of other topics such as minimax, resolution, etc. that play an important role in AI programs.
5) To have a basic understanding of some of the more advanced topics of AI such as learning, natural language processing, agents and robotics, expert systems, and planning
6) To have basic knowledge on probabilistic analysis and networks as well as fuzzy systems and fuzzy logics.

UNIT-I

Introduction to artificial intelligence: Introduction, history, intelligent systems, foundations of AI, applications, tic-tac-toe game playing, development of AI languages, current trends in AI

Problem solving: state-space search and control strategies: Introduction, general problem solving, characteristics of problem, exhaustive searches, heuristic search techniques, iterative deepening a*, constraint satisfaction

UNIT-II

Problem reduction and game playing: Introduction, problem reduction, game playing, alpha-beta pruning, two-player perfect information games

Logic concepts: Introduction, propositional calculus, proportional logic, natural deduction system, axiomatic system, semantic tableau system in proportional logic, resolution refutation in proportional logic, predicate logic

UNIT-III

Knowledge representation: Introduction, approaches to knowledge representation, knowledge representation using semantic network, extended semantic networks for KR, knowledge representation using frames

Advanced knowledge representation techniques: Introduction, conceptual dependency theory, script structure, CYC theory, case grammars, semantic web

UNIT-IV

Expert system and applications: Introduction, phases in building expert systems, expert system versus traditional systems, rule-based expert systems, blackboard systems truth maintenance systems, application of expert systems, list of shells and tools

Uncertainty measure: probability theory: Introduction, probability theory, Bayesian belief networks, certainty factor theory, Dempster-Shafer theory

UNIT-V

Fuzzy sets and fuzzy logic: Introduction, fuzzy sets, fuzzy set operations, types of membership functions, multi valued logic, fuzzy logic, linguistic variables and hedges, fuzzy propositions, inference rules for fuzzy propositions, fuzzy systems.

TEXT BOOKS:

1. Artificial Intelligence- Saroj Kaushik, CENGAGE Learning,
2. Artificial intelligence, A modern Approach , 2nded, Stuart Russel, Peter Norvig, PEA

REFERNCE BOOKS:

1. Artificial intelligence, structures and Strategies for Complex problem solving, - George F Lugar, 5thed, PEA
2. Introduction to Artificial Intelligence, Ertel, Wolf Gang, Springer
3. Artificial Intelligence, A new Synthesis, Nils J Nilsson, Elsevier

Micro Syllabus of Artificial Intelligence

UNIT-I

Introduction to artificial intelligence: Introduction, history, intelligent systems, foundations of AI, applications, tic-tac-toe game playing, development of AI languages, current trends in AI

Problem solving: state-space search and control strategies: Introduction, general problem solving, characteristics of problem, exhaustive searches, heuristic search techniques, iterative deepening a*, constraint

Unit	Module	Micro Content
UNIT I	Introduction to artificial intelligence	Introduction
		history
		Intelligent systems
		Foundations of AI
		Applications
		Tic-tac-toe game playing
		Development of AI languages
		Current trends in AI
	Problem solving: state-space search and control strategies	Introduction
		General problem solving
		Characteristics of problem
		Exhaustive searches
		Heuristic search techniques
		Iterative deepening a*
		Constraint satisfaction

UNIT – II

Problem reduction and game playing: Introduction, problem reduction, game playing, alpha-beta pruning, two-player perfect information games.

Logic concepts: Introduction, propositional calculus, proportional logic, natural deduction system, axiomatic system, semantic tableau system in proportional logic, resolution refutation in proportional logic, predicate logic.

Unit	Module	Micro Content
UNIT II	Problem reduction and game playing	Introduction
		Problem reduction
		Game playing
		Alpha-beta pruning
		Two-player perfect information games
	Logic concepts	Introduction
		Propositional calculus
		Proportional logic
		Natural deduction system
		Axiomatic system

		Semantic tableau system in proportional logic
		Resolution refutation in proportional logic
		Predicate logic

UNIT– III

Knowledge representation: Introduction, approaches to knowledge representation, knowledge representation using semantic network, extended semantic networks for KR, knowledge representation using frames

Advanced knowledge representation techniques: Introduction, conceptual dependency theory, script structure, cyc theory, case grammars, semantic web

Unit	Module	Micro Content
UNIT III .	Knowledge representation	Introduction
		Approaches to knowledge representation
		Knowledge representation using semantic network
		Extended semantic networks for KR
		Knowledge representation using frames
	Advanced knowledge representation techniques	Introduction
		Conceptual dependency theory
		Script structure
		CYC theory
		Case grammars
		Semantic web

UNIT – IV

Expert system and applications: Introduction, phases in building expert systems, expert system versus traditional systems, rule-based expert systems, blackboard systems truth maintenance systems, application of expert systems, list of shells and tools

Uncertainty measure: probability theory: Introduction, probability theory, Bayesian belief networks, certainty factor theory, Dempster-Shafer theory

Unit	Module	Micro Content
UNIT IV	Expert system and applications	Introduction
		Phases in building expert systems
		expert system versus traditional systems

		Rule-based expert systems
		Blackboard systems truth maintenance systems
		Application of expert systems
		List of shells and tools
	Uncertainty measure	Introduction
		Probability theory
		Bayesian belief networks
		Certainty factor theory
		Dempster-Shafer theory

UNIT V

Fuzzy sets and fuzzy logic: Introduction, fuzzy sets, fuzzy set operations, types of membership functions, multi valued logic, fuzzy logic, linguistic variables and hedges, fuzzy propositions, inference rules for fuzzy propositions, fuzzy systems.

Unit	Module	Micro Content
UNIT V	Fuzzy sets and fuzzy logic	Introduction
		Fuzzy sets
		Fuzzy set operations
		Types of membership functions
		Multi valued logic
		Fuzzy logic
		Linguistic variables and hedges
		Fuzzy propositions
		Inference rules for fuzzy propositions
		Fuzzy systems

***

III Year II semester

L	T	P	C
2	2	0	3

Compiler Design

Course Learning Objectives:

1. To impart the knowledge of compilers and their structure
2. To help students to design parsers - Language generators & recognizers.
3. To demonstrate the students' parsers can be equipped with translation Schemes
4. To make the student to understand storage allocations, Machine independent IR, Machine Dependent / independent - Code Generation
5. To make the students to understand different techniques in optimization of code

Course Outcomes:

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

CO1. Annotating Compilers, Grammars, Scanners, Types & structures of Compilers **(Understand)**

CO2. Inferring and Articulate different Parsers - can generate language & recognize it (Understand **& apply)**

CO3. Exemplifying semantic analyzer without the aid of automatic generators translation schemes (**Understand**)

CO4. Associating storage allocation strategies, IR forms & Code generation form **(Understand)**

CO5. Finally, will express how source code for a novel language converted into machine code for a novel computer (**Understand**)

UNIT-I **10 Hours**
Introduction: To Languages & Translators, Lexical Analysis, Grammars, Syntax Analysis

UNIT-II **12 Hours**
Parsers: Top-Down Parsers, Bottom-Up Parsers

UNIT-III **8 Hours**
Semantic Analysis: Syntax-Directed Definitions, Translation schemes

UNIT- IV **13 Hours**
Intermediate-Code Generation, **Run-Time Environments, Code Generation**

UNIT-V **5 Hours**
Optimization Techniques: Machine-Independent Optimizations

TEXT BOOKS

1. Compilers: Principles, Techniques and Tools, Second Edition, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffry D. Ullman, Pearson.

REFERENCE BOOKS

1. Compiler Construction-Principles and Practice, Kenneth C Louden, Cengage Learning.
2. Modern compiler implementation in C, Andrew W Appel, Revised edition, Cambridge University Press.
3. The Theory and Practice of Compiler writing, J. P. Tremblay and P. G. Sorenson, TMH
4. Writing compilers and interpreters, R. Mak, 3rd edition, Wiley student edition.
5. Lex & yacc – John R. Levine, Tony Mason, Doug Brown, O'reilly

Micro Syllabus of Compiler Design

UNIT - I: Introduction: To Languages & Translators, Lexical Analysis, Grammars, Syntax Analysis		
Unit	**Module**	**Micro Content**
UNIT I	**Introduction to Languages & Translators**	Importance of translators , Translator oriented languages, Language pre-processors
		Compiler, Cousins of compiler & the structure of a compiler
		Types of compilers, two-pass compilers & structure
	Lexical Analysis (Scanner)	The Role of the Lexical Analyzer
		Recognition of Tokens, Patterns
		The Lexical-Analyzer Generator, Input Buffering
		Lex, Yacc , A program on .lex
	Grammars	Introduction Noam Chomsky Classification of grammars, CFG, Generating Language from Grammars
		Generating Parse Tress (LMD&RMD), Ambiguous Grammars,
		Left recursion, Elimination of Left recursion,
		Left factoring, eliminating left factoring
	Syntax Analysis	Role of Syntax Analyzer in compiler
	Additional Topics	More Example of generating. lex program
UNIT - II: Parsers, Top-Down Parsers, Bottom-Up Parsers		
Unit	**Module**	**Micro Content**
UNIT - II	**Parsers**	Importance of Parsers, Types of Parsers
	Top-Down Parsers	Intro of top-down parsers, non-Backtracking, backtracking Parsers - Recursive Descendent Parsers,
		Recursive Descendent Parsers, example
		Finding FIRST
		Finding FOLLOW
		LL Parser – LL (1) , LL(k), Examples
		Examples of LL(1) Parsers, Predictive Parsers , Disadvantages of LL Parsers
	Bottom-Up Parsers	Introduction to Bottom-up Parsers - SLR, CLR,

		LALR
		Advantages of LR over LL Parsers
		Operator precedence Parsers - with Operator precedence Table
		Operator precedence Parsers - with Operator precedence function table
		SLR Parser & Example
		SLR Parser & Example, conflicts
		CLR Parsers & Example
		CLR Parsers & Example Conflicts
		LALR Parser & Example
		LALR Parser & Example
		Comparison of all Bottom-Up parsers, & top-down & bottom-up
UNIT - III: Semantic Analysis: Syntax-Directed Definitions, Translation schemes		
Unit	**Module**	**Micro Content**
UNIT - III	**Semantic Analysis**	Importance, Annotated Parse Tree, inherited & synthesized attributes, Syntax
		Syntax - Semantic-predicate rules associated with grammar symbols
		Types and Declarations, Type conversions, type checking.
	Syntax-Directed Definitions (SDDs)	Types of SDDs, Implementing L-Attributed SDD's
		S-Attribute SDDs, Evaluation Orders for SDD's,
	Translation schemes (SDTs)	Syntax-Directed Translation Schemes,
		Applications of Syntax-Directed Translation
UNIT - IV: Intermediate-Code Generation, Run-Time Environments, Code Generation		
Unit	**Module**	**Micro Content**
UNIT - IV	**Intermediate-Code Generation**	Role of Intermediate Code generation phase, Types of IR
		Three-Address Codes & Types
		Quadruples, triples& Indirect Triples
		Control Flow, Back patching,
		Switch-Statements, Intermediate Code for Procedures
	Run-Time Environments	Storage Organization, Static Allocation
		Stack Allocation of Space, Heap Management
		Access to Nonlocal Data on the Stack, , Introduction to Garbage Collection,
	Code Generation	Issues in the Design of a Code Generator,
		Role of Code generation phase
		The Target Language

		Basic Blocks and Flow Graphs
		A Simple Code Generator, Register Allocation and Assignment
		Dynamic Programming Code-Generation
UNIT V : Optimization Techniques, Machine-Independent Optimizations		
Unit	**Module**	**Micro Content**
UNIT- V	**Optimization Techniques**	Role of Optimization Phase, The Principal Sources of Optimization
		Structure preserving techniques & Algebraic Techniques - Constant propagation, constant folding, strength reduction, code motion, loop optimization,
		Peephole Optimization.
	Machine-Independent Optimizations	Introduction to Data-Flow Analysis, Foundations of Data-Flow Analysis,
		Optimization of Basic Blocks, Loops in Flow Graphs.

*****

III Year II semester

L	T	P	C
3	0	0	3

COMPUTER GRAPHICS
(Professional Elective - 1)

Course Objectives:

1. To develop, design and implement two- and three-dimensional graphical structures
2. To enable students to acquire knowledge Multimedia compression and animations.
3. To learn Creation, Management and Transmission of Multimedia objects.

UNIT - I **10 Hours**

Introduction to Computer Graphics: Applications of Computer Graphics,
2D Primitives: -Output Primitives: Points, Lines, Planes, Frame- Buffers, Video-display devices, Line Drawing Algorithms: DDA Line drawing, Bresenham's Line Drawing, Parallel Line Drawing, Circle and Ellipse Generation, Polygon Generation, Polygon Filling Algorithms, Attributes of Output Primitives.

UNIT - II **10 Hours**

2D Transformations & Viewing: Basic Transformations: Translation, Rotation, Scaling, Other Transformations: Reflection, Shear, Composite Transformations, Coordinate Transformation, Viewing Pipeline: Viewing Reference Frame, window, view-port, window-to-view-port Transformation, Multiple window transformation, Clipping: Line Clipping: cohen-sutherland line clipping algorithm, Polygon Clipping: Sutherland-Hodheman polygon clipping algorithm, Text Clipping.

UNIT - III **10 Hours**

3D Concepts: 3D Object Representation: Polygons, Curved Lines, Splines, Quadric Surfaces, **3D Transformations: Basic: Translation**, Coordinate-axis-Rotation, Arbitrary-axis Rotation, Scaling, Other: Reflection, Shear, Composition of 3D transformations, Projections: Parallel, Perspective, 3D Viewing, Visible-Surface Detection Algorithms: Back face removal, Z-Buffer, A-Buffer, Area-sub-division, Depth-Sorting (painter's), BSP-Tree, Octree,3D Clipping

UNIT - IV **10 Hours**

Graphics Programming Color Models – RGB, YIQ, CMY, HSV – Animations – General Computer Animation, Raster, Key frame Graphics programming using OPENGL – Basic graphics primitives –Drawing three dimensional objects - Drawing three dimensional scenes
Rendering Introduction to Shading models – Flat and Smooth shading – Adding texture to faces –Adding shadows of objects – Building a camera in a program – Creating shaded objects– Rendering texture – Drawing Shadows

UNIT - V **8 Hours**

Fractals Fractals and Self similarity – Peano curves – Creating image by iterated functions – Mandelbrot sets – Julia Sets – Random Fractals.
Overview of Ray Tracing Intersecting rays with other primitives – Adding Surface texture – Reflections and Transparency – Boolean operations on Objects.

Text Books:

1. Donald Hearn, Pauline Baker, Computer Graphics – C Version, second edition Pearson Education,2004.
2. F.S. Hill, Computer Graphics using OPENGL, Second edition, Pearson Education, 2003.

Reference Books:

1. James D. Foley, Andries Van Dam, Steven K. Feiner, John F. Hughes, Computer Graphics- Principles and practice, Second Edition in C, Pearson Education, 2007.

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

COs	STATEMENT
CO1	**Identify** Applications, video devices and **analyse** 2D Objects by learning output primitives
CO2	**Analyze** various 2D Object representation models by **learning** various visualization techniques
CO3	**Analyze** various 3D Object representation models by **learning** various visualization techniques
CO4	**Develop** programs in OPENGL by using apt functions for efficacy in Computer Graphics 2D/3D and Animation **Perform Rendering** of 2D/3D Objects by **learning** about shading, texture mapping techniques and drawing shadows
CO5	**Design** complicated Real-world Scenes by **learning** Iterated Function Systems for implementing Fractals **Apply** 3D Solid Geometric Techniques for representing 3D objects

Micro Syllabus of Computer Graphics

UNIT I: Applications of Computer Graphics, 2D Primitives: -Output Primitives, Line Drawing Algorithms ,Circle and Ellipse Generation, Polygon Generation, Attributes of Output Primitives.

Unit	Module	Micro Content
UNIT I	Applications of Computer Graphics, Display Devices	Engineering, Art, Science, Presentation
		Visualization, Education, Entertainment
		CRT, DVST, LED, LCD
		Raster Scan
		Random Scan, Color Display's
	2D Output-Primitives	Points, Frame Buffer Loading, Line drawing Algorithms, DDA
		Bresenham's Examples
		Parallel Line Drawing, Circle & ellipse Generation
		Polygon Filled Algorithms, scan line, boundary fill, flood fill
		Attributes of output primitives
	Circle & Ellipse Generation	Circle & ellipse Generation Algorithm
		Example of mid-point circle generation
		Example of ellipse algorithm

UNIT – II: 2D Transformations ,2D Viewing & Clipping: Basic Transformations, Other Transformations, Composite transformations, Viewing Pipeline, Clipping.

Unit	Module	Micro Content

UNIT II	Transformations	Basic: Translation, Rotation, Scaling, Other: Reflection, Shear
	Composite transformations	Additive, commutative
		Coordinate transformation
	Viewing pipeline	Coordinate reference frame
		Window to view port transformation
		Multiple Windowing
	Clipping	Point, line, polygon, text

UNIT – III: 3D Concepts: 3D Object Representation, 3D Transformations, Projections, 3D Viewing, Visible Surface Detection Algorithms

Unit	Module	Micro Content
UNIT III	3D Object Representation	Boundary, Spatial
		Polygons, curves quadric surfaces
	3D Transformations	Basic: Translation, Rotation, Scaling Other: Reflection, Shear
		Rotations: coordinate axis, Arbitrary-axis
		Additive & commutative proveings on composite
	Projections	Parallel, Perspective
		View volumes
	3D Viewing	Projection planes
		Projection coordinate transformations
	3D Clipping & visible surface detection algorithms	Clipping against view volume boundaries, applying visible surface detection
		Operations on B^+ Tree: Insertion, Deletion, Search

UNIT - IV: Color Models: RGB, HSV, CMY, YIQ, Animation & Open GL Primitives, 3D Scenes, Shading models.

Unit	Module	Micro Content
UNIT IV	Color Models	RGB, CMY
		HSV, YIQ
	Animation, Open GL primitives	Key frame animation
		Basic primitives: Begin, end, polygon, vertex etc
		3D Scene representation
	Shading Models	Flat
		Smooth, surface renderings
	Shadows	Shadow buffer
		Textures

UNIT V: Fractals: Self similarity objects, random fractals, Mandelbrot set, Julia set, snowflake
Ray Tracing: Forward ray tracing, backward ray tracing, Boolean operations

Unit	Module	Micro Content
UNIT V	Fractals	Introduction, applications, random fractals

		Snowflakes
		Mandelbrot set
		Julia sets
		Created an image by using Iterated Functions
	Ray Tracing	Introduction, forward, backward
		Boolean Operations on CSG objects

******

III - II Semester

L	T	P	C
3	0	0	3

No-SQL

Professional Elective - II

(Common to CSE & IT Branches)

Course Objectives:

1. To make student understand about NoSQL, its characteristics and history, and the primary benefits for using NoSQL data
2. To explore students about various types of NO-SQL databases (wide-column, document, key-value, graph and object-oriented) in adding content and running queries
3. To make students in understanding the NoSQL data architecture patterns

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

CO1: Outlines the importance of NoSQL and types of NoSQL Databases. (L1)
CO2: Demonstrates the working environment of Column-oriented databases. (L3)
CO3: Demonstrates the working environment of Key Value Databases. (L3)
CO4: Demonstrates the working environment of Document based Databases. (L3)
CO5: Demonstrates the working environment of Graph Databases.(L3)

UNIT-I: Introduction to No-SQL **8 hrs**

What is No-SQL? NoSQL Overview, NoSQL Database Environment, NoSQL Options, When to use No-SQL?, Introduction to No-SQL development

UNIT-II: Column-Oriented Databases **10 hrs**

Column family, key and keyspace, Apache HBASE

Unit – III: Key Value Databases **10 hrs**

What is key value store? Key value databases, DynamoDB

UNIT-IV: Document based Databases **10 hrs**

What is document? Document Databases, MangoDB

UNIT-V: Graph Databases **10 hrs**

What is Graph Database? Graph Databases, Neo4J

Text Books:

1. NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence, Author: Sadalage, P. & Fowler, Publication: Pearson Education

2. NoSQL Databases A Complete Guide - 2020 Edition, Author: Gerardus Blo dyk, Publisher: 5starcooks

Reference Books

1. Name: Redmond, E. &Wilson, Author: Seven Databases in Seven Weeks: A Guide to Modern Databases and the NoSQL Movement Edition: 1st Edition.
2. NoSQL For Dummies, Author: Adam Fowler, Publisher: A wiley Brand

e- Resources & other digital material:

1. https://www.guru99.com/hbase-tutorials.html
2. https://docs.mongodb.com/manual/tutorial/
3. https://dynobase.dev/dynamodb/
4.https://neo4j.com/developer/graph-db-vs-nosql/

CO-PO-PSO Mapping Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSPO1	PSPO2
CO-1	2	1	-	3	-	-	-	-	-	-	-	-	3	-
CO-2	1	-	3	-	1	-	-	-	-	-	-	-	-	1
CO-3	1	-	3	-	1	-	-	-	-	-	-	-	-	1
CO-4	1	-	3	-	1	-	-	-	-	-	-	-	-	1
CO-5	1	-	3	-	1	-	-	-	-	-	-	-	-	1

MICRO SYLLABUS

UNIT-I: Introduction to No-SQL
What is No-SQL? NoSQL Overview, NoSQL Database Environment, NoSQL Options, When to use No-SQL?, Introduction to No-SQL development

Unit	Module	Micro content
Introduction to No-SQL	Introduction	• Introduction to NoSQL • What is NoSQL • NoSQL Overview • NoSQL Database Environment • NoSQL Options
	When to use No-SQL?	• Benefits to using NoSQL DB • Backend Management • Drawbacks to Using NoSQL DB • NoSQL vs. SQL
	Introduction to No-SQL development	• Data Models • Distribution Models • Consistency • Categories of NoSQL • NoSQL Scalability

UNIT-II: Column-Oriented Databases (12 hrs)
Column family, key and key space, Apache HBASE

Unit	Module	Micro content
Column-Oriented Databases	Column-Oriented Data-bases	• Column family • Key and Key Space • Overview of various models (Apache Hbase, Cassandra etc.)
	Apache HBASE	• Architecture of HBASE • Features, Consistency, Transactions, Avail-

		ability • Query Features, • Scaling, Suitable Use Cases, • Event Logging, • Content Management Systems, • Blogging Platforms, • Counters, • When Not to Use

Unit – III: Key Value Databases (12 hrs)

What is key value store? Key value databases, DynamoDB

Unit	Module	Micro content
Key Value Databases	Key Value Databases	• What is key value store? • Key value databases • Major & Minor keys • Overview of various models (DynamoDB, Redis etc.)
	DynamoDB	• What Is a Key-Value Store • Key Value Store Features, • Consistency, Transactions, • Query Features, • Structure of Data, Scaling, • Suitable Use Cases, • Storing Session Information, • When Not to Use, • Relationships among Data, • Multi operation Transactions, • Query by Data, • Operations by Sets

UNIT-IV: Document based Databases (12 hrs)

What is document? Document Databases, MangoDB

Unit	Module	Micro content
Document based Databases	Document based Databases	• What is document • Attributes • Metadata • Formats • XML • JSON and BSON • Overview of various models (MongoDB, CouchDB etc.)
	MongoDB	• Features, • Consistency, • Transactions, • Availability, • Query Features, • Scaling, Suitable Use Cases, • Content Management Systems, Blogging Platforms, • Web Analytics or Real-Time Analytics, • E-Commerce Applications, • When Not to Use, • Queries against Varying Aggregate Structure

UNIT-V: Graph Databases (12 hrs)
What is Graph Database? Graph Databases, Neo4J

Unit	Module	Micro content
Graph Databases	Graph Databases	• Edges • Nodes • Relationship • Overview of various models (Neo4J, InfoGrid etc.)
	Neo4J	• Database development tools and programming languages, • Graph Databases, • Features, • Consistency, • Transactions, • Availability, • Query Features, • Scaling, Suitable Use Cases, • Location-Based Services, • Recommendation Engines, • When Not to Use

******

III Year II semester

L	T	P	C
3	0	0	3

Full Stack Development

(Professional Elective - III)

Course Objectives:

1. To learn Client-side application development using HTML and CSS
2. To understand Java script ES6 features
3. To focus on contemporary front-end technologies like React
4. To understand data access through NodeJS

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

- Summarize Client-side design of the web.
- Explore different ES6 features in Java script.
- Implement components and props through React.
- Comprehend React Hooks
- Use NodeJs for data availability

Unit-1: Introduction to HTML 5, syntax, attributes, events, SVG, Web storage, Introduction to Canvas, Audio & Video, Geolocations, Drag & Drop, Web workers, working with Fonts, working with other graphics.
Style sheets: Introduction CSS, Applying CSS to HTML, Selectors, Properties and Values, CSS Colors and Backgrounds, CSS Box Model, CSS Margins, Padding, and Borders, CSS Text and Font Properties

Unit-2: Introduction to ES6 features, Arrow functions, default parameters, destructuring elements, Higher order functions, defining classes, accessing data members, constructors, inheritance, super.

Unit-3: ReactJS: Introduction, creating a simple react project, Templating using JSX, Components, Rendering, State and Props, Types of Components – Component Lifecycle, Forms and User Input, Event Handling, Communicate Between Components.

Unit-4: React **JS:** React Routing, Introduction to Hooks, State management, Types of Hooks - useState, useEffect, useContext. CORS policies, Usage of Web API calls- fetch and axios, Error Handling.

Unit-5: Node JS: Overview, Node js - Basics and Setup, Node js Console, Node js Command Utilties, Node js Modules, Node js Concepts, Node js Events, Node js with Express js.
Introduction to MongoDB, creating databases, Operations – insert, update, delete and Querying.

Text Books:

1. HTML5, Black book, Dreamtech Publications
2. Beginning React, Greg Lim
3. Learning AngularJS: A Guide to AngularJS Development, O' Reilly Publication

References:

1. React Cook Book, Carlos Santana Roldan
2. Learning React, 2nd Edition, O' Reilly publications.
3. React in Action by Mark Tielens Thomas

Web Resources:

https://developer.mozilla.org/en-US/docs/Web/JavaScript
https://reactjs.org/docs/getting-started.html
https://nodejs.org/en/docs/

Micro Syllabus of Full Stack Development

Unit-1: Introduction to HTML 5, syntax, attributes, events, SVG, Web storage, Introduction to Canvas, io & Video, Geolocations, Drag & Drop, Web workers, working with Fonts, working with other graphics.
Style sheets: Introduction CSS, Applying CSS to HTML, Selectors, Properties and Values, CSS Colors and kgrounds, CSS Box Model, CSS Margins, Padding, and Borders, CSS Text and Font Properties.

Unit No.	Topic	Sub Topic
I	Introduction to HTML 5	Syntax, attributes, events
		SVG, Web storage
		Introduction to Canvas, Audio & Video, Geolocations
		Drag & Drop, Web workers
		Working with Fonts, working with other graphics.
	Style Sheets	Introduction CSS
		Applying CSS to HTML
		Selectors, Properties and Values
		CSS Colors and Backgrounds
		CSS Box Model, CSS Margins
		Padding, and Borders
		CSS Text and Font Properties
II	Introduction	Primitive Types, operators, Type conversions, control flow – conditions & loops
	ES6 features	Let and const, Arrow functions, destructuring elements, multi-line strings, default parameters, promise.
	Higher order functions	Map(), filter(), reduce()
	OOPs	Classes & objects, properties, constructors, this, inher-itance, super.

Unit-3: ReactJS: Introduction, Component - types of Components – Component Lifecy-Rendering, State and Props, Forms and User Input, Event Handling, Communicate Between Compo-ts.

Unit No.	Topic	Sub Topic
III	Introduction	Introduction, Need of React, Advantages of React JS
		Creating a simple react application
		Directory structure, Package.json structures
		Templating using JSX

	Components	Class & functional components
		Component Life cycle – methods.
	Use Input & Event Handling	Text events, button events, mouse events, drop down
	Communication	From parent to child & child to parent – rendering props, context & call backs

ct **JS:** React Routing, Introduction to Hooks, State management, Types of Hooks -useState, useEf-, useContext. CORS policies, Usage of Web API calls- fetch and axios, Error Handling.

Unit No.	Topic	Sub Topic
IV	React Routing	Need of Routing
		Brower Router, Routes, Route
		Nested Routing
	Introduction to Hooks	Need of hooks, State management
		Types of Hooks
		Implementation of useState, useEffect & useContext
	CORs policies & Usage of Web API calls	Need of Web API calls
		Fetch
		axios
		Error handling

Unit-5: Node JS: Overview, Node js - Basics and Setup, Node js Console, Node js Command ties, Node js Modules, Node js Concepts, Node js Events, Node js with Express js.
oduction to MongoDB, creating databases, Operations – insert, update, delete and Querying.

Unit No.	Topic	Sub Topic
V	Overview	Introduction, Node js - Basics and Setup, Differences between NPM and NPX
		Installation of Node JS
		Architecture of Node JS
	Modules & Events	Node JS Modules – HTTP module, URL module
		Installing Express, Request & response, Basic Routing, Get and post.
	Introduction to MongoDB	Introduction, advantages of MongoDB
		Creating a database, create and drop collections
	Operations	Implementation of insert, delete, update, querying documents.

***

III Year II semester

L	T	P	C
3	0	0	3

Software project Management

(Professional Elective - IV)

COURSE OBJECTIVES:

1. To study how to plan and manage projects at each stage of the software development life Cycle (SDLC)
2. To train software project managers and other individuals involved in software project Planning tracking and oversight in the implementation of the software project management process.
3. Implement the project plans through managing people, communications and change
4. To understand successful software projects that support organization's strategic goals.
5. Conduct activities necessary to successfully complete and close the Software projects

Course Outcomes:

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

CO	OUTCOME
CO1	Recognize need of software models and software economics **(Remember)**
CO2	Summarize different principles of software engineering **(Remember)**
CO3	Infer various process models **(understand)**
CO4	Articulate iterative planning models **(Apply)**
CO5	Sketch different performance indicators and software metrics **(Apply)**

UNIT -I: **8 Hours**

Conventional Software Management: The waterfall model, conventional software Management performance. Evolution of Software Economics: Software Economics, pragmatic software cost estimation. Improving Software Economics: Reducing Software product size, improving software processes, improving team effectiveness, improving automation, Achieving required quality, peer inspections

UNIT -II: **10 Hours**

The Old Way and The New: The principles of conventional software Engineering, principles of modern software management, transitioning to an iterative process. Life Cycle Phases: Engineering and production stages, inception, Elaboration, construction, transition phases. Artifacts of The Process: The artifact sets, Management artifacts, Engineering artifacts, programmatic artifacts.

UNIT -III: **10 Hours**

Model Based Software Architectures: A Management perspective and technical perspective. Work Flows of the Process: Software process workflows, Iteration workflows. Checkpoints of the Process: Major mile stones, Minor Milestones, Periodic status assessments.

UNIT -IV: **10 Hours**

Iterative Process Planning: Work breakdown structures, planning guidelines, cost and schedule estimating, Iteration planning process, Pragmatic planning. Project Organizations and Responsibilities: Line-of-Business Organizations, Project Organizations, evolution of Organizations.

UNIT -V: **10 Hours**

Process Automation: Automation Building blocks, The Project Environment. Project Control and Process Instrumentation: The seven core Metrics, Management indicators, quality indicators, life cycle expectations, pragmatic Software Metrics, Metrics automation. Project Estimation and Management: COCOMO model, Critical Path Analysis, PERT technique, Monte Carlo approach (Text book 2)

Text Books:

1) Software Project Management, Walker Royce, Pearson Education, 2005.
2) Software Project Management, Bob Hughes, 4th edition, Mike Cotterell, TMH.

Reference Books:

1) Software Project Management, Joel Henry, Pearson Education.
2) Software Project Management in practice, Pankaj Jalote, Pearson Education, 2005.
3) Effective Software Project Management, Robert K.Wysocki, Wiley,2006

Micro Syllabus of Software Project Management

UNIT I: Conventional Software Management: The waterfall model, conventional software Management performance. Evolution of Software Economics: Software Economics, pragmatic software cost estimation. Improving Software Economics: Reducing Software product size, improving software processes, improving team effectiveness, improving automation, Achieving required quality, peer inspections.

Unit	Module	Micro Content
UNIT I	The waterfall model	Waterfall model for Software Development
	conventional software Management performance	Performance metrics evaluation
	Evolution of Software Economics	Software Economics
		pragmatic software cost estimation
	Improving Software Economics	Reducing Software product size
		improving software processes
		improving team effectiveness
		improving automation
		Achieving required quality
		peer inspections

UNIT – II: The Old Way and The New: The principles of conventional software Engineering, principles of modern software management, transitioning to an iterative process. Life Cycle Phases: Engineering and production stages, inception, Elaboration, construction, transition phases. Artifacts of The Process: The artifact sets, Management artifacts, Engineering artifacts, programmatic artifacts.

Unit	Module	Micro Content
UNIT II	The principles of conventional software engineering	Conventional principles of software
	Principles of modern software management	Modern principles of software

	transitioning to an iterative process	Iterative process models
	Life Cycle Phases	Engineering and production stages
		inception
		Elaboration,
		construction
		transition phases
	Artifacts of The Process	The artifact sets
		Management artifacts
		Engineering artifacts
		programmatic artifacts

UNIT – III: Model Based Software Architectures: A Management perspective and technical perspective. Work Flows of the Process: Software process workflows, Iteration workflows. Checkpoints of the Process: Major mile stones, Minor Milestones, Periodic status assessments.

Unit	Module	Micro Content
UNIT III	A Management perspective and technical perspective	Management perspective of software
		Technical perspective of software
	Work Flows of the Process	Software process workflows
		Iteration workflows
	Checkpoints of the Process	Major mile stones
		Minor Milestones
		Periodic status assessments.

UNIT - IV: Iterative Process Planning: Work breakdown structures, planning guidelines, cost and schedule estimating, Iteration planning process, Pragmatic planning. Project Organizations and Responsibilities: Line-of-Business Organizations, Project Organizations, evolution of Organizations.

Unit	Module	Micro Content
UNIT IV	Work breakdown structures	Construction of Work breakdown structure
	planning guidelines	Details about planning phase
	cost and schedule estimating	Forward looking and top down approach
	Iteration planning process	Stages of iteration process
	Pragmatic planning	Pragmatic software metrics
	Project Organizations and Responsibilities	Line-of-Business Organizations
		Project Organizations
		Evolution of Organizations.

UNIT V: Tries: Process Automation: Automation Building blocks, The Project Environment. Project Control and Process Instrumentation: The seven core Metrics, Management indicators, quality indicators, life cycle expectations, pragmatic Software Metrics, Metrics automation. Project Estimation and Management: COCOMO model, Critical Path Analysis, PERT technique, Monte Carlo approach.

Unit	Module	Micro Content
UNIT V	Automation Building blocks	Tools for automation of building blocks
	The Project Environment	Round trip engineering, change management
	Project Control and Process Instrumentation	The seven core Metrics
		Management indicators
		quality indicators
		life cycle expectations
		pragmatic Software Metrics
		Metrics automation
	Project Estimation and Management	COCOMO model
		Critical Path Analysis
		PERT technique
		Monte Carlo approach.

****

III - II Semester

L	T	P	C
0	0	3	1.5

Data Warehousing & Data Mining Lab

Course Objectives:

1. Demonstrates various Data Mining Tasks.
2. Relates students in differentiating Data Sets for analysis.
3. Illustrates students in evaluating the methods contextually.

Course Outcomes:

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

CO – 1 : Demonstrates Data Pre-processing Techniques.
CO – 2 : Demonstrates and Discovers Knowledge using Classification Methods
CO – 3 : Demonstrates and Discovers Knowledge using Association Methods
CO – 4 : Demonstrates and Discovers Knowledge using Clustering Methods

List of experiments

1. Demonstration of pre-processing on dataset student.arff [CO1]
2. Demonstration of pre-processing on dataset labor.arff [CO1]
3. Demonstration of Association rule process on datset contactlenses. arff using apriori algorithm [CO3]
4. Demonstration of Association rule process on dataset test.arff using apriori algorithm [CO3]
5. Demonstration of classification rule process on dataset student.arff using j48 algorithm [CO2]
6. Demonstration of classification rule process on dataset employee.arff using j48 algorithm [CO2]
7. Demonstration of classification rule process on dataset employee.arff using id3 algorithm [CO2]
8. Demonstration of classification rule process on dataset employee.arff using na-ïve bayes algorithm [CO2]
9. Demonstration of clustering rule process on dataset iris.arff using simple k-means [CO4]
10. Demonstration of clustering rule process on dataset student.arff using simple k- means. [CO4]

III - II Semester

L	T	P	C
0	0	3	1.5

Artificial Intelligence Lab

Course Objectives:

1. Study the concepts of Artificial Intelligence.
2. Learn the methods of solving problems using Artificial Intelligence.
3. Introduce the concepts of machine learning.

Experiments

1. Write a Program to Implement Tic-Tac-Toe game using Python.
2. Write a program to solve water jug problem
3. Write a Program to Implement Breadth First Search using Python.
4. Write a Program to Implement Depth First Search using Python.
5. Write a Program to Implement 8-Puzzle problem using Python
6. Implementation of Towers of Hanoi Problem
7. Write a Program to Implement Missionaries-Cannibals Problems using Python
8. Write a Program to Implement Travelling Salesman Problem using Python
9. Write a Program to Implement Monkey Banana Problem using Python
10. Write a Program to Implement N-Queens Problem using Python

Course Outcomes:

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

CO1: Identify problems that are amenable to solution by AI methods.

CO2: Recognize appropriate AI methods to solve a given problem.

CO3: Discuss a given problem in the language /framework of different AI methods.

CO4: Develop basic AI algorithms

CO-PO mapping Table

Mapping	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1						3	3		2	1			3	
CO2						3	3		2					2
CO3						1		2					3	2
CO4						1			1		1			2

III - II Semester

L	T	P	C
0	0	3	1.5

Computer Networks Lab

(Common to CSE & IT Branches)

Course Objectives:

1. To illustrate the working of data link layer and network layer protocols
2. To illustrate the use of client/server architecture in application development
3. To illustrate how to use TCP and UDP based sockets and their differences.
4. To get acquainted with Unix system networking commands and Socket system calls.
5. To Design reliable servers using both TCP and UDP sockets

List of Experiments

1. Implement the data link layer framing methods such as character stuffing and bit stuffing.
2. Implement on a data set of characters the three CRC polynomials
3. Implement Dijkstra‘s algorithm to compute the Shortest path thru a graph.
4. Take an example subnet graph with weights indicating delay between nodes. Now obtain Routing table art each node using distance vector routing algorithm
5. Understanding and using of commands like ifconfig, netstat, ping, arp, telnet, ftp, finger, traceroute, whois etc. Usage of elementary socket system calls (socket (), bind(), listen(), accept(), connect(), send(), recv(), sendto(), recvfrom()).
6. Implementation of Connection oriented concurrent service (TCP).
7. Implementation of Connectionless Iterative time service (UDP).
8. Implementation of Select system call.
9. Implementation of gesockopt(), setsockopt() system calls.
10. Implementation of getpeername() system call.
11. Implementation of remote command execution using socket system calls.
12. Implementation of Distance Vector Routing Algorithm.
13. Implementation of SMTP.
14. Implementation of FTP.
15. Implementation of HTTP.

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

CO1: Demonstrate the working of various Data link layer and Network layer protocols (L3)
CO2: Experiment the working of client/server communication through Sockets API (L3)
CO3: Experiment the usage of TCP and UDP based sockets and their differences (L3)
CO4: Demonstrate the working of Unix system networking commands and Socket system calls. (L3)
CO5: Design reliable servers using both TCP and UDP sockets (L6)

CO-PO-PSO Mapping Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO-1	2	2											2	2
CO-2	2	2											2	2
CO-3	2	2											2	2
CO-4	2	2	2										2	2
CO-5	2	2	2	2									2	2

****