Academic Regulations

Program structure & Detailed Syllabus

2017

For

Under Graduate Programme (B.Tech)

Computer Science and Engineering

(Applicable For Batches Admitted From 2017 – 2018)

VIGNAN'S INSTITUTE OF INFORMATION TECHNOLOGY
(AUTONOMOUS)

DUVVADA - VISAKHAPATNAM – 530 049

(An Autonomous Institute, Accredited by NAAC, Affiliated to JNTUK, Kakinada, AP)

VIGNAN'S INSTITUTE OF INFORMATION TECHNOLOGY

(AUTONOMOUS)

INDEX

ACADEMIC REGULATIONS

(VR 17)

VIGNAN'S INSTITUTE OF INFORMATION TECHNOLOGY (AUTONOMOUS)
VISAKHAPATNAM

ACADEMIC REGULATIONS for B. Tech. (Regular)
(Applicable for the batches admitted 2017-18 onwards)

The Admission of students into B. Tech. course shall be as per the Govt. of Andhra Pradesh rules.

1. Award of B. Tech. Degree

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

a. Pursue a program of study for not less than four academic years and not more than eight academic years.
b. For lateral entry scheme admission: Pursue a program of study for not less than three academic years and not more than six academic years.
c. For the award of a degree, regular candidate has to register for 189 credits and shall secure 189 credits.
d. Lateral entry candidate has to register for all the courses from second year onwards and secure all the credits registered for.

2. Courses of Study

The following courses of study are offered at present for specialization in the B. Tech. Course.

S. No.	Course Code	Programme & Abbreviation
01	01	Civil Engineering (CE)
02	02	Electrical and Electronics Engineering (EEE)
03	03	Mechanical Engineering (ME)
04	04	Electronics and Communication Engineering (ECE)
05	05	Computer Science and Engineering (CSE)
06	12	Information Technology (IT)
07	19	Electronics and Computer Engineering (E.Com E)

And any other Course as approved by the authorities of the Institute from time to time.

3. Registration: A student shall register for courses in each semester as per the courses offered by the concerned department.

4. Curricular Program

The Curriculum of the four-year B. Tech Course has been designed to achieve a healthy balance between theory & lab hours, industry experience and to develop technical skills required for a career in the industry or a career in research.

5. 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 courses and 100 marks for practical course. The project work shall be evaluated for 200 marks.

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

Distribution of marks for theory course, practical course and Design/Drawing is detailed below:

5.1. Internal 40 marks for theory course shall be awarded as follows:

i) 25 marks for MID exams
ii) 10 marks for continuous assessment
iii) 5 marks for Attendance

MID marks shall be calculated with 80% weightage for best of the two MIDs and 20% weightage for other MID exam.

5.2. For practical courses (Laboratory): There shall be continuous evaluation during the semester. Each Lab exam is evaluated for 100 marks. 50 marks shall be awarded for internal examination and 50 marks shall be awarded for external examinations.

5.2.1. Internal marks shall be awarded as follows

i) Day to day assessment– 20 Marks
ii) Record – 10 Marks
iii) Internal laboratory exam– 20 Marks

5.2.2.The semester end examinations shall be conducted by the teacher concerned and external examiner

5.3. For the courses having design and/or drawing, (Such as Engineering Graphics, Engineering Drawing, Machine Drawing) and estimation, the distribution shall be 40 marks for internal evaluation.

5.3.1. Internal marks shall be awarded as follows:

i) 20 marks for Day-to-day assessment
ii) 15 marks for internal exam
iii) 5 marks for Attendance

There shall be two internal examinations in a semester and the internal marks shall be calculated with 80% weightage for best of the two internals and 20% weightage for other internal exam.
5.3.2. External examination shall be conducted for 60 marks.

5.4. Industrial Visit: The industrial visit shall be carried out in their domain during the summer vacation after the second year second semester. A student has to submit a report which will be evaluated for 100 marks and will be submitted to an internal evaluation committee comprising Head of the Department or his / her nominee and two senior faculty of the department including the industrial visits coordinator/ supervisor.

5.5. Industry- Oriented Mini Project: The Industry oriented mini project is carried out during the third year second semester vacation. The students have an option to choose their own area of interest which may be related to the course work. Mini project report is evaluated for 100 marks in fourth year first semester before the first mid-term exam. Assessment is done by the supervisor /guide for 40 marks based on the work and mini project report. The remaining 60 marks are allocated for presentation by the student to a committee comprising of the project supervisor and senior faculties members nominated by Head of the Department.

5.6. MOOCs: It is an online course (Minimum of 12 weeks) to promote advanced knowledge suitable for placement and research.

To award credits, the student should get certificate after they have registered for written exam and successfully passed

(Or)

College will conduct the written examination/Viva-voce and award the credits and grades.

In case a student fails in any online course, he/she may be permitted to register for the same course or an alternate course decided by the department committee. The internal marks secured earlier will be nullified if the course is changed. The assessment procedure of MOOCs course remains same as general theory course.

Note: The registered course must not be same as any of the courses listed in the program structure of their regulation till final year including electives.

5.7. Technical Seminar: For Technical seminar, the student shall collect the information on a specialized topic and prepare a technical report, showing his/her understanding over the topic, and submit to the department, which shall be evaluated by the Departmental Committee consisting of Head of the Department, seminar supervisor and a senior faculty member. The seminar report shall be evaluated for 50 marks. There shall be no external examination for seminar.

5.8. Comprehensive Viva: The Comprehensive Viva aims to assess the students' understanding in various subjects he / she studied during the B.Tech course of study. Comprehensive Viva is conducted for a total of 50 marks. It shall be conducted in IV Year II Semester. The Comprehensive Viva–Voce will be conducted by a Committee consisting of Head of the Department, & senior faculty members of the Department.

5.9. Internship: Internships help students to acquire in depth knowledge about a particular topic related to the program of study. Such extensive work is expected to create a platform for a job or further research in the chosen area. Interested students may opt for a full semester Internship during the fourth year second semester. Such students shall be exempted for equivalent theory course credits during that semester and the corresponding credits are awarded through the Internship. A self-study report, duly authorized by the industry supervisor / guide, shall be submitted at the end of the fourth year second semester. Internship report is evaluated for 400 marks in total. Internal assessment is done by the academic supervisor/guide for 100 marks based on the work and presentation of the internship report. The assessment for 300 marks is evaluated and awarded by a panel of members consisting of Head of the Department, Senior Faculty and Industry Expert.

5.10. Main Project: Out of total 200 marks for the project work, 80 marks shall be for Internal Evaluation and 120 marks for the external assessment. The Internal Evaluation shall be on the basis

of two mid-term project reviews conducted during the progress of the project. The End Semester Examination (Viva-Voce) shall be conducted by the committee consists of an External Examiner, Head of the Department (internal examiner) and a senior faculty of the Department. The evaluation of project work shall be conducted at the end of the IV year.

5.11. Audit courses: All audit courses will be "Pass/Fail" type with no credit points allotted. The result of the student in the audit course will be notified in the marks memo. A student must pass all the audit courses registered to be eligible for the award of B. Tech. degree.

List of audit courses will be notified from time to time. An indicative list of courses is as shown below.

a) Professional Ethics & Human Values b) Any Foreign Language c) Journalism d) Finance e) Legal Sciences f) Social Sciences g) English for Special Purposes h) Fine Arts i) Clinical Psychology j) Intellectual Property Rights & Patents etc.

6. Attendance Requirements:

6.1. It is desirable for a candidate to have 100% attendance in the class in all the courses. However, a candidate shall be permitted to appear for the end semester examination if he/she has a minimum of 75% aggregate attendance in the semester. Student will not be permitted to write Mid examination if the attendance percentage is less than 75 % during the stipulated instruction duration. However, Academic Monitoring Committee shall review the situation and take appropriate decision.
Note: Special cases for students having extraordinary performance at National and International level will be considered by the Academic monitoring committee.

6.2. Condonation of shortage of attendance may be considered on Medical grounds maximum up to 10%, if the student provides the medical certificate to the HOD immediately after he / she recovers from the illness. Medical Certificate submitted afterwards shall not be permitted. Shortage of attendance equal to or above 65% and below 75%will be condoned on payment of fee as fixed by the competent authority and the student concerned will be permitted to take the end semester examination. ***This privilege is given only three times for regular student and only two times for lateral entry student during the entire program of study***.

6.3. Shortage of attendance may be considered for the students who participate in prestigious sports, co and extra-curricular activities if their attendance is in the minimum prescribed limit.

6.4. A student will be promoted to the next semester if satisfies attendance and credits requirement.

7. Academic Requirements:

The following academic requirements have to be satisfied in addition to the attendance requirements.

For any course, student is considered to be passed upon securing minimum 35% marks in the external examination alone and minimum 50% marks from both internal and external examination put together

8. Promotion Policy:

To promote to III year, a student has to secure minimum 50% of total credits from I & II-year courses
To promote to IV year, a student has to secure minimum 50% of total credits from I, II & III-year courses

In case of Lateral entry students, to promote to IV year, a student has to secure minimum 50% of total credits from II & III-year courses

9. Supplementary examinations: Supplementary examinations for the odd Semester shall be conducted with the regular examinations of even semester and vice versa. In case a student fails in online courses/ industrial lecture(s), he/she may be permitted to register for another course/lecture(s).

10. Examinations and Evaluation

10.1. General guidelines

i. All the semester end examinations are conducted for duration of three hours
ii. External examination shall be conducted for 60 marks consist of five questions of internal choice carrying 12 marks each.
iii. For laboratory examinations, the evaluation is done by internal examiner and one external examiner.

10.2. Revaluation

There is a provision for revaluation of theory courses if student fulfils the following norms.

The request for revaluation must be made in the prescribed format duly recommended by the Chief Superintendent of Examinations through Additional Controller along with the prescribed revaluation fee.

11. Grading System:

CGPA

Marks Range (in %)	Letter Grade	Level	Grade Point
≥ 90	O	Outstanding	10
≥80 to <90	A	Excellent	9
≥70 to <80	B	Very Good	8
≥60 to <70	C	Good	7
≥50 to <60	D	Satisfactory	6
<50	F	Fail	0
		Absent	-1
		Withheld	-2
		Malpractice	-3

Computation of SGPA

The following procedure is to be adopted to compute the Semester Grade Point Average. (SGPA) and Cumulative Grade Point Average (CGPA):

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

SGPA (Si) = Σ(Ci x Gi) / ΣCi

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 CGPA

- The CGPA is also calculated in the same manner taking into account all the courses undergone by a student over all the semesters of a programme, i.e.

$$CGPA = \Sigma(Ci \times Si) / \Sigma Ci$$

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

- Equivalent Percentage = (CGPA – 0.75) X 10

12. Award of Class

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

Class Awarded	**CGPA to be secured**	**CGPA secured from 189 Credits.**
First Class with Distinction	≥ 7.75 without course failures during entire duration of study	
First Class	≥ 6.75 to <7.75	
Second Class	≥ 5.75 to < 6.75	

13. General Instructions

i. Where the words ‘he’, ‘him’, ‘his’, occur, they imply ‘she’, ‘her’, ‘hers’, also.

ii. The academic regulations should be read as a whole for the purpose of any interpretation.

iii. In case of any doubt or ambiguity in the interpretation of the above rules, the decision of the Chairman, Academic Council is final.

iv. The college may change or amend the academic regulations or syllabi from time to time and the changes or amendments made shall be applicable to all the students with effect from the dates notified by the Institution.

14. Transitory Regulations

i. The student has to continue the course work along with the regular students of the respective semester in which the student gets re-admission.

ii. The student has to register for Substitute / Compulsory courses offered in place of courses studied earlier.

iii. The mode of internal evaluation and end-semester examinations shall be on par with the regular students, i.e., the student has to follow the mode of internal evaluation and the then question paper model for the end-semester examinations along with the regular students of the respective semester in which the student gets re-admission. The marks secured in the internal and end-semester examinations will be pro-rated in accordance with the regulations under which the student was first admitted.

iv. For the courses studied under earlier regulations but failed, the student has to appear, pass and acquire credits from the supplementary examinations as and when conducted. The question paper model shall remain same as the one in which the student took examination during previous regulations.

v. The promotion criteria based on attendance as well as credits shall be in accordance with the regulations under which the student was first admitted.

vi. All other academic requirements shall be in accordance with the regulations under which the student was first admitted.

vii. The decision of the Principal is final on any other clarification in this regard.

viii. Transcripts: After successful completion of the entire program of study, a transcript containing performance of all academic years will be issued as a final record. Partial transcript will also be issued up to any point of study to a student on request, after payment of requisite fee.

15. Minimum Instruction Days

The minimum instruction days for each semester shall be 16 weeks

There shall be no branch transfers after the completion of the admission process.

16. Withholding of Results

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

Note: All other regulations including attendance requirements related to four year

B. Tech Regular program will be applicable for B.Tech. Lateral Entry Scheme.

17. Malpractices Rules

DISCIPLINARY ACTION FOR MALPRACTICES / IMPROPER CONDUCT IN EXAMINATIONS

S.No	Nature of Malpractices/ Improper conduct	Punishment
1 (a)	If the candidate 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 course of the examination (theory or practical) in which he is appearing but has not made use of (material shall include any marks on the body of the candidate which can be used as an aid in the course of the examination)	Expulsion from the examination hall and cancellation of the performance in that course only.
(b)	If the candidate 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 course 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	If the candidate has copied in the examination hall from any paper, book, programmable calculators, palm computers or any other form of material relevant to the course of the examination (theory or practical) in which the candidate is appearing.	Expulsion from the examination hall and cancellation of the performance in that course and all other courses the candidate has already appeared including practical examinations and project work and shall not be permitted to appear for the remaining examinations of the courses of that Semester/year. The Hall Ticket of the candidate is to be cancelled.
3	If the candidate 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 courses of the examination (including practical's and project work) already appeared and shall not be allowed to appear for examinations of the remaining courses of that semester/year. The candidate is also debarred for two consecutive semesters from class

		work and all 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	If the candidate 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 course and all the other courses the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the courses of that semester/year. The candidate is also debarred for two consecutive semesters from class work and all examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat.
5	If the candidate uses objectionable, abusive or offensive language in the answer paper or in letters to the examiners or writes to the examiner requesting him to award pass marks.	Cancellation of the performance in that course.
6	If the candidate 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 the officer on duty amounts to use of unfair means or misconduct or has the tendency to disrupt the orderly conduct of the examination.	In case of students of the college, they shall be expelled from examination halls and cancellation of their performance in that course and all other courses the candidate(s) has (have) already appeared and shall not be permitted to appear for the remaining examinations of the courses 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.

7	If the candidate 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 course and all the other courses the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the courses of that semester/year. The candidate is also debarred for two consecutive semesters from class work and all examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat.
8	If the candidate possesses any lethal weapon or firearm in the examination hall.	Expulsion from the examination hall and cancellation of the performance in that course and all other courses the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the courses 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 college, expulsion from the examination hall and cancellation of the performance in that course and all other courses the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the courses 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	If the candidate comes in a drunken condition to the examination hall.	Expulsion from the examination hall and cancellation of the performance in that course and all other courses the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the courses of that semester/year.
11	Copying detected on the basis of internal evidence, such as, during valuation or during special scrutiny.	Cancellation of the performance in that course and all other courses the candidate has appeared including practical examinations and project work of that semester/year examinations.

12	If any malpractice is detected which is not covered in the above clauses 1 to 11 shall be reported to the Academic committee of the Institute for further action to award suitable punishment.	

18. UGC RECOMMENDED PUNISHMENT FOR RAGGING

i. Suspension from attending classes and academic privileges
ii. Withholding/withdrawing scholarships/fellowship and other benefits.
iii. Debarring from appearing in any test/examination or other evaluation process
iv. Withholding results
v. Debarring from representing the institution in any regional, national or international meet, tournament, youth festival etc.
vi. Suspension/expulsion from the hostel
vii. Cancellation of admission
viii. Rustication from the institution for period ranging from 1 to 4 semesters.
ix. Expulsion from the institution and consequent debarring from admission to any other institution for a specified period.
x. Fine may extend up to Rs. 2.5 lakh.

***

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
PROGRAM STRUCTURE (VR 17)

I B.Tech **I Semester**

S. No	Course Code	Course Title	L	T	P	Credits
1	1000171101	English -I	3	1*	0	3
2	1000171102	Engineering Mathematics -I	3	1*	0	3
3	1000171103	Engineering Mathematics -II	3	1*	0	3
4	1000171105	Computer Programming using C	3	1*	0	3
5	1000171106	Engineering Drawing	3	1*	0	3
6	1000171107	Applied Physics	3	1*	0	3
7	1000171121	English - Communication Skills Laboratory - I	0	0	3	2
8	1000171122	Engineering Physics Laboratory	0	0	3	2
9	1000171128	Computer Programming Laboratory	0	0	3	2
Total Credits						**24**

I B.Tech **II Semester**

S. No	Course Code	Course Title	L	T	P	Credits
1	1000171201	English - II	3	1*	0	3
2	1000171203	Engineering Mathematics - III	3	1*	0	3
3	1000171211	Applied Chemistry	3	1*	0	3
4	1000171212	Environmental Studies	3	1*	0	3
5	1000171215	Object Oriented Programming through C++	3	1*	0	3
6	1000171216	Engineering Mechanics	3	1*	0	3
7	1000171221	English Communication Skills Lab - 2	0	0	3	2
8	1000171227	Engineering Chemistry Laboratory	0	0	3	2
9	1000171229	Object Oriented Programming Lab	0	0	3	2
Total Credits						**24**

II B.Tech **I Semester**

S.No	Course Code	Course Title	L	T	P	Credits
1	1005172101	Statistics and R Programming	3	1*	0	3
2	1005172102	Mathematical Foundations of Computer Science	3	1*	0	3
3	1005172103	Digital Logic Design	3	1*	0	3
4	1005172104	Java Programming	3	1*	0	3
5	1005172105	Data Structures through C	3	1*	0	3
6	1005172106	Problem Solving and Program Design Through C	3	1*	0	3
7	1005172121	Data Structures through C Lab	0	0	3	2
8	1005172122	Java Programming Lab	0	0	3	2
Total Credits						**22**

II B.Tech **II Semester**

S.No	Course Code	Course Title	L	T	P	Credits
1	1005172201	Database Management Systems	3	1*	0	3
2	1005172202	Software Engineering	3	1*	0	3
3	1005172203	Advanced Data Structures	3	1*	0	3
4	1005172204	Computer Organization	3	1*	0	3
5	1005172205	Formal Languages and Automata Theory	3	1*	0	3
6	1005172206	Operating Systems	3	1*	0	3
7	1005172221	Advanced Data Structures Lab	0	0	3	2
8	1005172222	Database Management Systems Lab	0	0	3	2
9	1005172231	Industrial Visit/ Online certification course from Nptel or equivalent on any one programming language (C, C++, Java, Python or any state-of-art Courses).	0	0	0	2
Total Credits						**24**

III B.Tech **I Semester**

S.No	Course Code	Course Title	L	T	P	Credits
1	1005173101	Compiler Design	3	1*	0	3
2	1005173102	Python Programming	3	1*	0	3
3	1005173103	Data Mining Techniques	3	1*	0	3
4	1005173104	Unix and Shell Programming	3	1*	0	3
5	1005173105	Design and Analysis of Algorithms	3	1*	0	3
6	1005173121	Data Mining with R Lab	0	0	3	2
7	1005173122	Python Programming Lab	0	0	3	2
8	1005173123	Operating Systems and Compiler Design Lab	0	0	3	2
9	1099172103	Professional Ethics & Human Values	2	0	0	0
Total Credits:						**21**

III B.Tech **II Semester**

S.No	Course Code	Course Title	L	T	P	Credits
1	1005173201	Computer Networks	3	0	0	3
2	1005173202	Web Technologies	3	0	0	3
3	1005173203	Object Oriented Analysis and Design using UML	3	0	0	3
4	1005173204	Artificial Intelligence	3	0	0	3
5	**Open Elective-I**					
	1004173207	Digital image Processing	3	0	0	3
	1005173205	Embedded Systems				
	1004173208	Microprocessors & Microcontrollers-A				
	1003173203	Robotics				
6	**Open Elective-II (CBCS)(MOOCS)**					
	1005173291	*Any available online course approved by department committee at the time of semester commencement)	3	0	0	3
7	1005173221	Computer Networks Lab	0	0	3	2
8	1005173222	Unified Modeling Lab	0	0	3	2
9	1005173223	Web Technologies Lab	0	0	3	2
10	1099173101	IPR & Patents	2	0	0	0
Total Credits:						**24**

S.No	Course Code	Course Title	L	T	P	Credits
1	1005173241	Industry Oriented Mini Project	0	0	0	2

IV B.Tech **I Semester**

S.No	Course Code	Course Title	L	T	P	Credits
1	1005174101	Cryptography and Network Security	3	0	0	3
2	1099172106	Managerial Economics and Financial Analysis	3	0	0	3
3	1005174102	Machine Learning	3	0	0	3
4	1005174103	Big Data Analytics	3	0	0	3
5	**Elective-I**					
	1005174104	Mobile Ad-hoc Networks	3	0	0	3
	1005174105	Software Project Management				
	1004174105	IoT & its Applications				
6	**Elective-II**					
	1012172201	Computer Graphics	3	0	0	3
	1005174106	Cloud Computing				
	1012173203	Software Testing Methodologies				
7	1005174121	Cryptography and Network Security Lab	0	0	3	2
8	1005174122	Big Data Analytics Lab	0	0	3	2
Total Credits:						**22**

IV B.Tech **II Semester**

S.No	Course Code	Course Title	L	T	P	Credits
1	1005174201	Fundamentals of Block Chain Technology	3	0	0	3
2	1099172203	Management Science	3	0	0	3
3	1005174202	Software Architecture and Design Patterns	3	0	0	3
4	**Elective-III**					
	1005174203	Distributed Systems	3	0	0	3
	1005174204	Optimization Techniques				
	1005174205	Concurrent and parallel programming				
(OR)						
	1005174281	Internship	0	0	0	12
5	1005174251	Technical seminar	0	0	0	2
6	1005174261	Comprehensive Viva	0	0	0	2
7	1005174231	Main Project	0	0	0	10
Total Credits :						**26**

GRAND TOTAL CREDITS: 24+24+22+24+21+26+22+26=189

PROGRAM STRUCTURE

FOR

I-B.Tech

I & II SEMESTER

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
PROGRAM STRUCTURE

I B.Tech **I Semester**

S. No	Course Code	Course Title	L	T	P	Credits
1	1000171101	English -I	3	1*	0	3
2	1000171102	Engineering Mathematics -I	3	1*	0	3
3	1000171103	Engineering Mathematics -II	3	1*	0	3
4	1000171105	Computer Programming using C	3	1*	0	3
5	1000171106	Engineering Drawing	3	1*	0	3
6	1000171107	Applied Physics	3	1*	0	3
7	1000171121	English - Communication Skills Laboratory - I	0	0	3	2
8	1000171122	Engineering Physics Laboratory	0	0	3	2
9	1000171128	Computer Programming Laboratory	0	0	3	2
Total Credits						**24**

I B.Tech **II Semester**

S. No	Course Code	Course Title	L	T	P	Credits
1	1000171201	English - II	3	1*	0	3
2	1000171203	Engineering Mathematics - III	3	1*	0	3
3	1000171211	Applied Chemistry	3	1*	0	3
4	1000171212	Environmental Studies	3	1*	0	3
5	1000171215	Object Oriented Programming through C++	3	1*	0	3
6	1000171216	Engineering Mechanics	3	1*	0	3
7	1000171221	English Communication Skills Lab - 2	0	0	3	2
8	1000171227	Engineering Chemistry Laboratory	0	0	3	2
9	1000171229	Object Oriented Programming Lab	0	0	3	2
Total Credits						**24**

DETAILED SYLLABUS FOR

I-B.Tech

I-SEMESTER

Course Code		L	T	P	Credits
1000171101	**ENGLISH-I**	3	1	0	3

Course Objectives:

1. To improve the language proficiency of the students in English with emphasis on LSRW skills.
2. To enable the students to study and comprehend the prescribed lessons and subjects more effectively related to their theoretical and practical components.
3. To develop the communication skills of the students in both formal and informal situations.
4. Practice critical thinking to develop innovative and well-founded perspectives related to the students' emphases. Build and maintain healthy and effective relationships.
5. To convey a credible message and create concise messages using a structured writing process.
6. To develop effective interpersonal communication skills.

Course Outcomes:

After completing this Course, the student should be able to:

1. Use English language, both written and spoken, competently and correctly.
2. Improve comprehension and fluency of speech.
3. Gain confidence in using English in verbal situations.
4. Display competence in oral, written, and visual communication.
5. Communicate ethically.
6. Demonstrate positive group communication exchanges.

	Course outcome	**Cognitive Level as per Bloom's Taxonomy**	**PO**
CO1	Enhance English Language by relating the ideas of eminent personalities.	Level-2	PO6,PO9, PO10, PO12
CO2	Articulate the technological advancements fluently. .	Level-3	PO10, PO12
CO3	Inculcate the art of thinking and writing clearly and logically. .	Level-3	PO10, PO12
CO4	Enact various themes through team work and learn the usage of vocabulary through humourous texts.	Level-4	PO10, PO12

Detailed Text: English Essentials

1. In London - M. K. Gandhi
2. The Knowledge Society - A. P. J. Abdul Kalam
3. Principles of Good Writing - L. A. Hill
4. Man's Peril - Bertrand Russell
5. Luck - Mark Twain

Non-Detailed Text: Panorama

1. War – Luigi Pirandello
2. The Verger – Somerset Maugham

Prescribed Textbooks:

1. English Essentials by Ravindra Publishing House
2. Panorama by Oxford University Press

Suggested Text Books:

1. You Can Win by Shiv Khera
2. English for Engineers and Technologists by Orient Black Swan
3. Objective English by R. S. Agarwal, S.Chand.co

Reference Books:

1. "Practical English Usage" by Michael Swan, 3rd Edition, OUP.
2. "Intermediate English Grammar" by Raymond Murphy, CUP.
3. "Study: Reading" by Eric H .Glendinning, 2ndEdition CUP.
4. "Business Correspondence and Report writing" by R.C Sharma, Tata Mc Grawhill

Course Code		L	T	P	Credits
1000171102	**ENGINEERING MATHEMATICS-I**	3	1	0	3

Course Overview:

This course deals with differential equations and its application with more focus on advanced Engineering Mathematics. This course helps the students to learn relevant mathematical tools which are required in the analysis of problems in engineering and scientific professions. Topics included in this course are differential equations of first order and their applications, higher order linear differential equations and their applications, functions of single variable and their applications and multiple integrals, Laplace transforms and their applications. The mathematical skills derived from this course form a necessary base for analytical and design concepts encountered in the program.

Course Objectives:

1. To explain mathematical modeling with the knowledge of differential equations.
2. To discuss higher order differential equations and its applications to solve engineering problems.
3. To evaluate maxima and minima of function of several variables.

Course Outcomes:

1. Solve basic engineering problems described by first order differential equations.
2. Determine solutions to higher order linear homogeneous and non homogeneous differential equations with constant coefficients.
3. Apply the techniques of multivariable differential calculus to determine extrema and series expansions etc. of functions of several variables.
4. Extend the concept of integration to two and three dimensions and support it through applications in engineering mechanics
5. Appraise the Laplace transform technique and use it to solve various engineering problems

	Course outcome	**Cognitive Level as per Bloom's Taxonomy**	**PO**
CO1	Solve linear differential equations of first and higher order.	Understanding	PO1,PO2, PO3, PO12
CO2	Determine extrema and series expansions of functions of several variables.	Applying	PO1, PO2,PO4, PO12
CO3	Determine double integral, triple integral to find area and volume..	Applying	PO1, PO2, PO3, PO12
CO4	Appraise Laplace transform to solve various engineering problems.	Analyzing	PO1, PO2, PO12

UNIT-I: MEAN VALUE THEOREMS:

Mean Value Theorems - Rolle's Theorem - Lagrange's mean value theorem – Cauchy's mean value theorem(without proofs)

ORDINARY DIFFERENTIAL EQUATIONS:

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

Applications: Orthogonal trajectories ,Simple Electric Circuits

UNIT-II: LINEAR DIFFERENTIAL EQUATIONS OF HIGHER ORDER:

Linear differential equations of second and higher order with constant coefficients, Non-homogeneous term of the type $Q(x) = e^{ax}$, sin ax, cos ax, x^n , $e^{ax}V(x)$, $x^nV(x)$ – Method of variation of parameters.

Applications: LCR Circuits

UNIT-III: FUNCTIONS OF SEVERAL VARIABLES:

Functions of several variables – Partial Differentiation –Euler's Theorem-Total Derivative –Change of variables - Jacobian -Functional dependence – Taylors theorem for functions of two variables.

Applications: Maxima and Minima of functions of two variables – Lagrange's method of undetermined multipliers.

UNIT-IV: MULTIPLE INTEGRALS:

Introduction: Review of Coordinate Systems (Cartesian, Polar, Parametric, Spherical, Cylindrical) - multiple integrals - double and triple integrals – change of variables – Change of order of Integration.

Applications: Areas and Volumes of Simple curves (Cartesian)

UNIT-V: LAPLACE TRANSFORMS:

Introduction - Laplace transforms of standard functions – Shifting Theorems - Transforms of derivatives and integrals - multiplication by t^n - division by t – Unit step function –Dirac delta function. Laplace transform of Periodic functions.

Introduction - Inverse Laplace transforms–Properties- Convolution theorem (without proof).

Applications: Solution of ordinary differential equation with constant coefficients (Initial Value Problems)using Laplace transforms.

TEXT BOOKS:

1. Higher Engineering Mathematics – 43rd Edition by Dr. B. S. Grewal, Khanna Publishers, New Delhi.

REFERENCE BOOKS:

1. Advanced Engineering Mathematics, Erwin Kreszig, 8thEd, Wiley Student Edition.
2. Engineering Mathematics, Greenburg, 2nd Ed, Pearson education.
3. A Text book of Engineering Mathematics, N.P.Bali, Laxmi Publications (P) Ltd.
4. Advanced Engineering Mathematics, B. V. Ramana, Tata McGraw Hill Publishing Co. Ltd.
5. Engineering Mathematics, P.Sivarama krishna Das, C.Vijaya kumari , 2017 Pearson Education Services Pvt.Ltd

Course Code		L	T	P	Credits
1000171103	**ENGINEERING MATHEMATICS-II**	3	1	0	3

Course Overview:

The entire course material is divided into 5 modules covering duly recognized areas of theory and study. This course includes the topics of advanced Engineering Mathematics with more focus on the mathematical tools required to analyze the problems of Engineering & Scientific Professions. Some important topics of this course are Solutions of Algebraic and Transcendental Equations, Interpolation, Numerical integration and Numerical solution of ordinary differential equations, Fourier series and Fourier transforms. The main aim of this course is to provide a platform to the students to think, design, formulate and derive any problem encountered in real life situation.

Course Objectives:

1. To formulate and apply numerical techniques for root finding, interpolation.
2. To estimate definite integrals using Newton-Cotes quadrature formula.
3. To compute numerical solution of ordinary differential equations.
4. To determine the Fourier coefficients of a given function.
5. To analyze the characteristics and properties of Fourier transforms.

Course Outcomes:

Upon successful completion of this course, student will be able to:

1. Determine numerical solution of non Linear equation
2. Compute Interpolating polynomial for the given data
3. Explain Numerical Solution of ODE and Numerical Integration.
4. Construct Fourier series expansion of periodic functions
5. Determine Fourier transform, Fourier sine and cosine transform of function.

	Course outcome	**Cognitive Level as per Bloom's Taxonomy**	**PO**
CO1	Determine numerical solution of non Linear equation	Understanding	PO1,PO2, PO12
CO2	Compute Interpolating polynomial for the given data	Applying	PO1, PO2,PO12
CO3	Explain Numerical Solution of ODE and Numerical Integration.	Applying	PO1, PO2, PO4, PO12
CO4	Construct Fourier series and Fourier transforms for functions	Analyzing	PO1, PO2, PO3, PO5, PO6, PO12

UNIT-I: SOLUTIONS OF ALGEBRAIC AND TRANSCENDENTAL EQUATIONS & INTRODUCTION TO FINITE DIFFERENCES:

Bisection method - Regula-falsi method - Iteration method - Newton-Raphson method. Finite differences: Forward, Backward and Central differences - Other difference operators and relations between them - Differences of a polynomial – To find missing terms.

UNIT-II: INTERPOLATION:

Newton's forward interpolation, Newton's backward interpolation, Gauss Forward and Backward interpolation, Interpolation with unequal intervals – Newton's divided difference - Lagrange's interpolation.

UNIT-III: **NUMERICAL SOLUTION OF ORDINARY DIFFERENTIAL EQUATIONS AND NUMERICAL INTEGRATION:**

Single step methods: Taylor's series method –Picard's Method - Euler's and modified Euler's Methods - Fourth order Runge-Kutta method for solving first order equations. Numerical Integration: Trapezoidal Rule, Simpson's $1/3^{rd}$ Rule, Simpson's $3/8^{th}$ Rule.

UNIT-IV: FOURIER SERIES:

Introduction- Determination of Fourier coefficients – Even and Odd functions –Change of interval– Half-range sine and cosine series-Practical Harmonic Analysis.

UNIT-V : FOURIER TRANSFORMS:

Fourier integral theorem (only statement) – Fourier sine and cosine integrals – Fourier transforms-Fourier Sine and Cosine transforms – Properties – Inverse transforms – Finite Fourier transforms.

TEXT BOOKS:

1. Higher Engineering Mathematics – 43^{rd} Edition by Dr. B. S. Grewal, Khanna Publishers, New Delhi.

REFERENCE BOOKS:

1. Advanced Engineering Mathematics, Erwin Kreszig, 8thEd, Wiley Student Edition.
2. Engineering Mathematics, B.V.Ramana, Tata McGraw Hill.
3. Mathematical Methods – Dr. Ravindranath & Dr. P. Vijaya Lakshmi, Himalaya Publications.
4. Engineering Mathematics, P. Sivaramakrishna Das, C.Vijayakumari , 2017 Pearson Education Services Pvt. Ltd.

Course Code	COMPUTER	L	T	P	Credits
1000171105	PROGRAMMING USING C	3	1	0	3

Course Objectives:

1. Understanding the basics of the computers and background.
2. Drawing flowcharts and Formulating algorithmic solutions to problems and implementing in C language.
3. Understanding branching, iteration and data representation using arrays.
4. Modular programming and recursive solution formulation. Understanding pointers and dynamic memory allocation.

Course Prerequisites:

Students should have knowledge of

1. Basics of Computer Components.
2. Distinction between software and hardware.

Course Outcomes:

Students will be able to:

1. Understand the fundamentals of computers, solving the problems using flow charts, algorithms and pseudo code.
2. Write, compile and execute simple programs in C language.
3. Use different data types and operators in C language.
4. Design programs involving decision structures, loops, functions and passing parameters to functions.
5. Develop programs using arrays, structures and pointers.
6. Understand the dynamic memory allocation functions using pointers.
7. Understand the basics of file operations, reading, writing and updating the files.

	Course outcome	Skill	PO
CO1	Write compile and debug Programs in C language	Understand	PO1,PO2, PO3
CO2	Use operators, data types and write programs	Understand	PO1,PO2
CO3	Select the best loop construct for a given problem	Analyzing	PO3,PO5
CO4	Design and implement C programs	Analyzing	PO1,PO2 PO3,PO4, PO12

UNIT-I:

Computer Basics – What is a computer, History of computers, Characteristics of computers, Classification of computers, Applications of computers, Components and functions of a Computer System: hardware and software concept, input/output devices, memory concept and secondary memories, Number System, Computer languages, Flow Charts, algorithms and pseudo code.
Introduction to C programming- Background and characteristics of C, Structure of a C Program, Input/ Output Statements in C, writing C programs, compiling and executing C programs.

UNIT-II: Programming Style:
Tokens of C, Keywords, Variables, Constants and rules to form variables and constants, Data Types, Declaration of Variables and initialization, Operators, Expression Types, Operator Precedence and Associativity. Implicit Type Conversions, Explicit Type Conversions (Casts), Assignment Variations, Mathematical Library Functions, Interactive Input, Formatted Output, Format Modifiers.

Flow of Control:
Selection: if and if-else Statements, if-else if statement and switch case, nested if, examples.
Repetition and Unconditional Control Statements: Pretest and Posttest Loops, Counter-Controlled and Condition-Controlled Loops, while Statement, do while statement, for Statement, Nested Loops. Break, continue and go to statements.

UNIT-III: Modular Programming:
Function and Parameter Declarations: Function definition, types of functions, declaration and definition of user defined functions, its prototypes and parameters, calling a function. Function stubs and Functions with and without Parameters. Variable Scope, Variable Storage Class, Local Variable Storage Classes, Global Variable Storage Classes.
Parameter passing Techniques: Pass by Value, recursive functions.

UNIT-IV: Arrays and Strings:
Arrays: One-Dimensional Arrays, Declaration, Array Initialization, Input and Output of Array Values, Arrays as Function Arguments, Two-Dimensional Arrays, linear search, and bubble sort.
Strings: String Fundamentals, String Input and Output, String manipulation functions, String Processing, String manipulation operations without Library Functions.

UNIT-V: Pointers, Structures and Unions, Data Files :
Pointers: Concept of a Pointer, Initialization of pointer variables, pointers as function arguments, Dangling memory, address arithmetic, character pointers and functions, pointers to pointers, Pointers and arrays, Pointers and strings, Array of Pointers, Dynamic memory management functions, parameter passing by address, command line arguments.

Structures and Unions: 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, bit- fields.

Data Files: Declaring, Opening, and Closing File Streams, Reading from and Writing to Text Files, Random File Access.

Text Books:

1. ANSI C Programming, Gary J. Bronson, Cengage Learning.
2. Programming in C, ReemaThareja, Oxford.
3. Programming in C, Bl Juneja Anita Seth, Cengage Learning.

Reference Books:

1. C Programming-A Problem Solving Approach, Forouzan, Gilberg, Cengage.
2. Programming with C, Bichkar, Universities Press.
3. The C Programming Language, Dennis Richie and Brian Kernighan, Pearson Education.
4. C by Example, Noel Kalicharan, Cambridge.

Course Code		L	T	P	Credits
1000171106	**ENGINEERING DRAWING**	3	1	0	3

Course Overview: Engineering drawing being the principle method of communication for engineers, the objective to introduce the students, the techniques of constructing the various types of polygons, curves and scales. The objective is also to visualize and represent the 3D objects in 2D planes with proper dimensioning, scaling etc.

Course Objectives:

1. To introduce the use and the application of drawing instruments and to make the students construct the polygons, curves and various types of scales. The student will be able to understand the need to enlarge or reduce the size of objects in representing them.
2. To introduce orthographic projections and to project the points and lines parallel to one plane and inclined to other.
3. To make the students draw the projections of the lines inclined to both the planes.
4. To make the students draw the projections of the plane inclined to both the planes.
5. To make the students draw the projections of the various types of solids in different positions inclined to one of the planes.
6. To represent the object in 3D view through isometric views. The student will be able to represent and convert the isometric view to orthographic view and vice versa.

	Course outcome	**Cognitive Level as per Bloom's Taxonomy**	**PO number mapped**
CO1	Understand the use of drawing instruments to construct the polygons and curves	Understanding	PO1,PO2,PO3
CO2	Learn the principle of orthographic projections. Draw Orthographic projections of points, lines.	Analyzing	PO1,PO2,PO3,PO12
CO3	Draw the various types of planes and solids its views in different Positions	Analyzing	PO1,PO2,PO3,PO12
CO4	Draw isometric views of simple objects	Analyzing	PO1,PO2,PO3,PO12

UNIT I: Polygons, Construction of regular polygons using given length of a side; Ellipse, arcs of circles and Oblong methods; Scales – Vernier and Diagonal scales.

UNIT II: Introduction to orthographic projections; projections of points; projections of straight lines parallel to both the planes; projections of straight lines – parallel to one plane and inclined to the other plane.

UNIT III: Projections of straight lines inclined to both the planes, determination of true lengths, angle of inclinations and traces.

UNIT IV: Projections of planes: regular planes perpendicular/parallel to one plane and inclined to the other reference plane; inclined to both the reference planes.

UNIT V: Projections of Solids – Prisms, Pyramids, Cones and Cylinders with the axis inclined to one of the planes. Conversion of isometric views to orthographic views; Conversion of orthographic views to isometric views.

Text Books:

1. Engineering Drawing, N. D. Butt, Chariot Publications
2. Engineering Drawing, K. L. Narayana & P. Kannaiah, Scitech Publishers.
3. Engineering Graphics, P.I. Varghese, McGraw Hill Publishers

Reference Books:

1. Engineering Graphics for Degree, K. C. John, PHI Publishers
2. Engineering Drawing, Agarwal & Agarwal, Tata McGraw Hill Publishers
3. Engineering Drawing + AutoCad – K Venugopal, V. Prabhu Raja, New Age.

Course Code	APPLIED PHYSICS	L	T	P	Credits
1000171107		3	1	0	3

Course Overview:
To enhance the fundamental knowledge in Physics and its applications relevant to various Streams of Engineering and Technology.

Course Objectives:
1. Impart Knowledge of Physical Optics phenomena like Interference, Diffraction and Polarization involving required to design instruments with higher resolution.
2. Teach Concepts of coherent sources, its realization and utility optical instrumentation.
3. Study the concepts regarding the bulk response of materials to the EM fields and their analytically study in the back-drop of basic quantum mechanics.
4. Understand the physics of Semiconductors and their working mechanism for their utility in sensors.

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Describe the wave phenomena and working principle of optical instruments.	Understanding	PO1,PO2, PO3, PO9, PO12
CO2	Apply the knowledge of basic quantum mechanics, to set up one-dimensional Schrodinger's wave equation .	Applying	PO1, PO2, PO9, PO12
CO3	Identify the importance of classical and quantum mechanical treatment of materials.	Applying	PO1, PO2, PO9, PO12
CO4	Make use of the basic concepts of energy bands in crystalline solids to understand semiconductor physics.	Analyzing	PO1, PO2, PO9, PO12

UNIT-I:
INTERFERENCE: Principle of Superposition – Coherent Sources – Interference in thin films (reflection geometry) – Newton's rings – construction and basic principle of Interferometers.

UNIT-II
DIFFRACTION: Fraunhofer diffraction at single slit cases - Circular Aperture (Qualitative treatment only) - Grating equation - Resolving power of a grating, Telescope and Microscopes.
POLARIZATION: Types of Polarization – Double refraction - Quarter wave plate and Half Wave plate – Working principle of Polarimeter (Sacharimeter).

UNIT-III:

ELECTRO MAGNETIC FIELDS: Scalar and Vector Fields – Electric potential- Gradient, Divergence of fields – Gauss and Stokes theorems – Maxwell's equations in differential forms

UNIT-IV:

QUANTUM MECHANICS: Introduction – Matter waves – Schrodinger time independent and time dependent wave equations – Particle in a box.

FREE ELECTRON THEORY: Defect of classical free electron theory – density of states – Quantum free electron theory – concept of Fermi energy.

UNIT-V:

BAND THEORY OF SOLIDS: Kronig- Penney model – energy bands in crystalline solids – classification of crystalline solids- effective mass of electron and concept of hole.

SEMICONDUCTOR PHYSICS: Conduction – Density of carriers in Intrinsic and Extrinsic-Semiconductors – Fermi energy in intrinsic and extrinsic semiconductors- Drift & Diffusion – Einstein's equation- Hall effect in semiconductors.

Outcome: Construction and working details of instruments, ie., Interferometer, Diffractometer and Polarimeter are learnt. Study EM-fields and semiconductors under the concepts of Quantum mechanics paves way for their optimal utility.

Text Books:

1. A Text book of Engineering Physics – by Dr. M.N.Avadhanulu and Dr. P.G.Kshira Sagar, S.Chand & Company Ltd., (2014)
2. Physics by David Halliday and Robert Resnick – Part I and Part II

Reference Books:

1. Applied Physics by P.K.Palanisamy, Scitech publications (2014)
2. Lasers and Non-Linear optics by B.B.Laud, New Age International Pub.
3. Engineering Physics by M. Arumugam, Anuradha Publication (2014)
4. Modern Engineering Physics by A.S. Vasudeva
5. University Physics by Young and Freedman
6. Engineering Physics by D.K.Bhattacharya and Poonam Tandon, Oxford press
7. Engineering Physics by R.K. Gaur and S.L. Gupta

Course Code	ENGLISH - COMMUNICATION SKILLS	L	T	P	Credits
1000171121	LABORATORY-I	0	0	3	2

Course Overview: The Language Lab focuses on the production and practice of sounds of language and familiarizes the students with the use of English in everyday situations and contexts.

Course Objective:

1. To sensitize the students to the nuances of English speech sounds, word accent, intonation and rhythm.
2. To bring about a consistent accent and intelligibility in their pronunciation of English by providing an opportunity for practice in speaking.
3. To improve the fluency in spoken English and neutralize mother tongue influence. To train students to use language appropriately to enhance Oratory Skills.

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Demonstrate the usage of phonemes while referring to the dictionary	Understanding	PO5,PO9, PO10, PO12
CO2	Articulate with others by using proper functions.	Applying	PO5,PO9, PO10, PO12
CO3	Enact the roles with proper body language.	Applying	PO5,PO9, PO10, PO12
CO4	Communicate fluently with proper pronunciation	Analyzing	PO5,PO9, PO10, PO12

Unit -1: Letters and Sounds
Unit-2: Interaction 1
Unit-3: The Sounds of English
Unit-4: Interaction 2

Prescribed Lab Manual:

1. Speak Well - Orient Black Swan Publishers

Suggested Books/ Manuals and Softwares:

1. Interact - Orient Black Swan
2. Strengthen your Communication Skills by Maruthi Publishers
3. Personality Development and Soft Skills (Oxford University Press, New Delhi)
4. GRE-Barons-12th Edition
5. Objective English-R.S.Agarwal-S.Chand Publishers
6. The Rossetta stone
7. English in Mind

Course Code 1000171122	ENGINEERING PHYSICS LABORATORY	L	T	P	Credits
		0	0	3	2

Course Objectives:

1. The Objective of this course is to make the students gain practical knowledge to co-relate with the theoretical studies. To achieve perfectness in experimental skills and the study of practical applications will bring more confidence and ability to develop and fabricate engineering and technical equipments.
2. Design of circuits using new technology and latest components and to develop practical applications of engineering materials and use of principle in the right way to implement the modern technology.

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Experimentation of laws of vibrations in strenteched string	Understanding	PO1,PO2, PO9, PO12
CO2	Determination of velocity of sound, rigidity modulus of a wire, acceleration due to gravity, radius of gyration and Planck's constant.	Applying	PO1, PO2, PO9, PO12
CO3	Analyze the voltage vs. current characteristics of Zener diode and temperature vs. resistance characteristics of a thermistor	Applying	PO1, PO2, PO9
CO4	Demonstration of formation Newton's rings, diffraction pattern using grating and induced magnetic field in a cirucuclar coil.	Analyzing	PO1, PO2, PO9

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 Rigidity modulus of a material- Torsional Pendulum.
4. Determination of Acceleration due to Gravity and Radius of Gyration- Compound Pendulum.
5. Melde's experiment – Transverse and Longitudinal modes.
6. Verification of laws of stretched string – Sonometer.
7. Determination of velocity of sound – Volume resonator.
8. L C R Senes Resonance Circuit.
9. Study of I/V Characteristics of Semiconductor diode.
10. I/V characteristics of Zener diode.
11. Thermistor characteristics – Temperature Coefficient.
12. Magnetic field along the axis of a current carrying coil – Stewart and Gee's apparatus.
13. Determination of wavelength of laser source using dirreraction grating.
    Determination of Planck's constant using photocell

Course Code	COMPUTER PROGRAMMING	L	T	P	Credits
1000171128	LABORATORY	0	0	3	2

Course Objectives:

1. Understand the basic concept of C Programming, and its different modules that include conditional and looping expressions, Arrays, Strings, Functions, Pointers, Structures and File programming.
2. Acquire knowledge about the basic concepts of writing a program in C language
3. Demonstrate Role of constants, variables, identifiers, operators, type conversion and other building blocks of C Language.
4. Demonstrate Use of conditional expressions and looping statements to solve problems associated with conditions and repetitions.
5. Demonstrate Role of Functions involving the idea of modularity.

Course Outcomes:

1. Apply and practice logical ability to solve the problems.
2. Understand C programming development environment, compiling, debugging, and linking and executing a program using the development environment
3. Analyzing the complexity of problems, Modularize the problems into small modules and then convert them into programs
4. Understand and apply the in-built functions and customized functions for solving the problems.
5. Understand and apply the pointers, memory allocation techniques and use of files for dealing with variety of problems.

Exercise – 1: Basics

a) What is an OS Command, Familiarization of Editors - vi, Emacs.
b) Using commands like mkdir, ls, cp, mv, cat, pwd, and man.
c) C Program to Perform Adding, Subtraction, Multiplication and Division of two numbers From Command line.

Exercise - 2: Basic Math

a) Write a C Program to Simulate 3 Laws at Motion
b) Write a C Program to convert Celsius to Fahrenheit and vice versa

Exercise - 3: 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 Add Digits & Multiplication of a number

Exercise - 4: Control Flow - II

a) Write a C Program to Find Whether the Given Number is
    i) Prime Number
    ii) Armstrong Number
b) Write a C program to print Floyd Triangle
c) Write a C Program to print Pascal Triangle

Exercise - 5: 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 - 6: Control Flow - III

a) Write a C Program to make a simple Calculator to Add, Subtract, Multiply or Divide Using switch…case

b) Write a C Program to convert decimal to binary and hex (using switch call function the function)

Exercise - 7: Functions - Continued

Write a C Program to compute the values ofsin x and cos x and e^x values using Series expansion. (use factorial function)

Exercise - 8: Arrays Demonstration of arrays

a) Search-Linear.

b) Sorting-Bubble, Selection.

c) Operations on Matrix.

Exercises - 9: Structures

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

b) Write a C Program to Store Information Using Structures with Dynamically Memory Allocation

c) Write a C Program to Add Two Complex Numbers by Passing Structure to a Function

Exercise - 10: Arrays and Pointers

a) Write a C Program to Access Elements of an Array Using Pointer

b) Write a C Program to find the sum of numbers with arrays and pointers.

Exercise - 11: Dynamic Memory Allocations

a) Write a C program to find sum of n elements entered by user. To perform this program, allocate memory dynamically using malloc () function.

b) Write a C program to find sum of n elements entered by user. To perform this program, allocate memory dynamically using calloc () function. Understand the difference between the above two programs

Exercise - 12: Strings

a) Implementation of string manipulation operations **with** library function.

i) Copy
ii) Concatenate
iii)Length
iv)Compare

b) Implementation of string manipulation operations **without** library function.

i) Copy
ii) Concatenate
iii)Length
iv)Compare

Exercise -13: 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 files

Exercise - 14: Files (Continued)

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

b) Write a C program to delete a file

DETAILED SYLLABUS FOR

I-B.Tech

II-SEMESTER

Course Code		L	T	P	Credits
1000171201	ENGLISH-II	3	1	0	3

Course Objectives:

1. To improve the language proficiency of the students in English with emphasis on LSRW skills.
2. To enable the students to study and comprehend the prescribed lessons and subjects more effectively related to their theoretical and practical components.
3. To develop the communication skills of the students in both formal and informal situations.
4. Practice critical thinking to develop innovative and well-founded perspectives related to the students' emphases. Build and maintain healthy and effective relationships.
5. How to convey a credible message and create concise messages using a structured writing process.
6. Develop effective interpersonal communication skills.

Course Outcomes:

After completing this Course, the student should be able to:

1. Use English language, both written and spoken, competently and correctly.
2. Improve comprehension and fluency of speech.
3. Gain confidence in using English in verbal situations.
4. Display competence in oral, written, and visual communication.

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Enhance English Language skills through the concept of Technological advancements.	Understanding	PO6,PO9, PO10, PO12
CO2	Illustrate the life of eminent personalities for developing the skills of vocabulary and grammar	Applying	PO10, PO12
CO3	Emphasize the relevance of cultures and traditions for enhancing writing skills through literature	Applying	PO10, PO12
CO4	Correlate the importance of Environment and sustainability with an emphasis on language skills	Analyzing	PO10, PO12

Detailed Text: English Encounters

1. **A Dilemma**- A Layman looks at Science
2. **Culture Shock**
3. **Lottery**
4. **Health Threats of Climate Change**
5. **A Chief Architect of Microsoft**

Non-Detailed Text: Panorama

1. **A Scarecrow** - Satyajit Ray
2. **A Village Lost to the Nation** - Krishna Chandra Pujari

Prescribed Books:

1. **English Encounters** by Maruthi Publications
2. **Panorama** by Oxford University Press

Course Code		L	T	P	Credits
1000171203	**ENGINEERING MATHEMATICS-III**	3	1	0	3

Course Overview:

This course focuses on basic theoretical concepts and advanced Engineering Mathematics. This course helps the students to understand mathematical tools required in the analysis of problems in Engineering and Scientific Professions. The topics included are Solution for linear systems, Eigen values & Eigen vectors, linear transformations, partial differential equations, Vector integral theorems(Green's, Stoke's and Gauss's divergence theorems). Thus mathematical skills derived from this course enables the students to design and solve the problems.

Course Objectives:

1. To explain the concepts of matrix algebra and methods of solving system of linear equations.
2. To compute Eigen values and Eigen vectors of real and complex matrices.
3. To apply properties of partial differential equations to obtain solution for science and engineering problems.
4. Classify and Solve partial differential equations
5. Generalize calculus to vector functions and to compute line, surface and volume integrals.

Course Outcomes:

Up on successful completion of this course, student will be able to:

1. Apply elementary transformations to reduce matrices to echelon form, normal form and hence find their rank.
2. Solve the system of linear equations and compute Eigen values and Eigen vectors of a square matrix.
3. Compute directional derivative and the gradient of functions of several variables.
4. Infer vector integral theorems to evaluate line, surface and volume integrals.

	Course outcome	**Cognitive Level as per Bloom's Taxonomy**	**PO**
CO1	Solve simultaneous linear equations numerically using rank of a matrix and compute Eigen values and Eigen vectors of a square matrix.	Understanding	PO1,PO2, PO3, PO12
CO2	Identify/classify and solve partial differential equations.	Applying	PO1, PO2, PO3, PO6, PO8
CO3	Calculate gradient of a scalar function, divergence and curl of a vector function.	Applying	PO1, PO2, PO3, PO12
CO4	Determine line, surface and volume integrals using appropriate integral theorems.	Analyzing	PO1, PO2, PO6, PO12

UNIT-I: LINEAR SYSTEMS OF EQUATIONS:

Introduction-Rank-Echelon form-Normal form-Solution of Linear systems - Gauss elimination-Gauss Seidel methods-Applications of matrix methods to finding current in the circuits.

UNIT-II: EIGEN VALUES-EIGEN VECTORS AND QUADRATIC FORMS:

Introduction-Eigen values-Eigen vectors-Properties(without proofs)-Cayley Hamilton theorem (without proof) - Inverse and power of a matrix by using Cayley Hamilton theorem, Diagonalisation of matrix-Quadratic forms-Reduction of Quadratic form to Canonical form-Rank-Index-Signature-Nature-Applications of Eigen value and Eigen vectors to Free Vibrations of two mass system.

UNIT-III: PARTIAL DIFFERENTIAL EQUATIONS:

Formation of partial differential equation by elimination of arbitrary constants and arbitrary functions- Solutions of first order linear (Lagrange's) equation and nonlinear (standard type) equations- Equations reducible to standard forms.

UNIT-IV: VECTOR DIFFERENTIATION:

Differentiation of vectors-Scalar and Vector point functions- Gradient of a scalar field and directional derivatives- Divergence and Curl of a vector field and its physical interpretation- Solenoidal and Irrotational of a vector- Vector identities.

UNIT-V: VECTOR INTEGRATION:

Line integral- Circulation, Work done, Surface and Volume integrals-Vector integral theorems: Green's, Stoke's and Gauss's Divergence theorems(without proofs) and related problems.

Text Books:

1. Higher Engineering Mathematics – 43rd Edition by Dr. B. S. Grewal, Khanna Publishers, New Delhi.

Reference Books:

1. Advanced Engineering Mathematics, Erwin Kreszig, 8thEd, Wiley Student Ed.
2. Advanced Engineering Mathematics, Greenburg, 2nd Ed, Pearson education.
3. Engineering Mathematics, N.P.Bali, Laxmi Publications (P) Ltd.
4. Engineering Mathematics, B. V. Ramana, TataMcGrawHill Publishing Co.
5. Engineering Mathematics, P.Sivaramakrishna Das, C.Vijayakumari , 2017 Pearson India Education Services Pvt. Ltd

Course Code		L	T	P	Credits
1000171211	**APPLIED CHEMISTRY**	3	1	0	3

Course Overview:

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.

Course Objectives:

1. Plastics are nowadays used in household appliances; also they are used as composites (FRP) in aerospace and automotive industries.
2. Fuels as a source of energy are a basic need of any industry, particularly industries like thermal power stations, steel industry, fertilizer industry etc., and hence they are introduced.
3. To know the mechanism of Corrosion for its control and prevention.
4. Water is a basic material in almost all the industries, more so where steam is generated and also where it is supplied for drinking purposes.
5. With the increase in demand, a wide variety of materials are coming up; some of them have excellent engineering properties and a few of these materials are introduced.

Course Outcomes: The student

1. Gains basic knowledge of polymer materials and their engineering applications.
2. Understands fuels which are used commonly and their advantages and limitations.
3. Extends the principles involved in corrosion to predict and prevent the corrosion in real life system
4. The advantages and limitations of semiconducting materials and their use in design would be understood.
5. Recalls the principles, working and design of energy storage devises and Acquires knowledge of advanced materials and their applications.

	Course outcome	**Cognitive Level as per Bloom's Taxonomy**	**PO**
CO1	Identify various polymers like Polythene, PVC, Teflon, Bakelite and their Engineering applications.	Understanding	PO1,PO7, PO8
CO2	Desribe various renwable and non-renwable energy resources.	Applying	PO1, PO2, PO4, PO7, PO8
CO3	Aquire the knoweldge of principles and reaction mechanism Of corrosion.	Applying	PO1, PO7, PO8
CO4	Illustrate green Synthesis, semiconductors, advanced materials and their applications in industry.	Analyzing	PO1, PO2, PO12

UNIT I: Polymer Technology

Polymerization: Introduction - Types of polymerization (Addition, Condensation & Co-polymerizstion) – Physical and mechanical properties – advantages and limitations – **Plastics**: Thermoplastics and Thermosetting plastics – Compounding, Moulding techniques (Compression, Injection & Blow film moulding) - Preparation, properties and applications of polyethene, PVC, Bakelite and Teflon.

Elastomers – Natural rubber- compounding and vulcanization – Synthetic rubbers - Buna S, Buna N and Thiokol – Applications. Composite materials & Fiber reinforced plastics (CFRP & GFRP) – Biodegradable polymers – Conducting polymers.

UNIT II: FUEL TECHNOLOGY

Introduction – Classification – Calorific value - Bomb calorimeter – Numerical problems – Coal — Proximate and ultimate analysis and its Significance – Liquid fuels – Petroleum - Refining – Cracking – knocking - Octane and Cetane numbers - Natural gas - LPG and CNG – Combustion – Flue gas analysis – Orsat apparatus – Numerical problems on combustion. Energy scenario in India – working of thermal power plant – Advantages and disadvantages –Renewable energy – Solar energy – Harnessing of solar energy – solar heaters – photo voltaic cells – Bio energy – Bio diesel.

UNIT III: ELECTROCHEMICAL CELLS & CORROSION

Galvanic cells - Reversible and irreversible cells, Electrode potential – Standard electrodes (Hydrogen and Calomel electrodes). Electro chemical series and its applications,

Batteries-: Dry Cell, lead acid battery and Ni-Cd battery - H_2-O_2 fuel cell & H_3PO_4 fuel cells.

Corrosion: Introduction – Theories of Corrosion (dry and wet) – Types of corrosion – galvanic, pitting, stress, differential aeration and waterline corrosion – Factors influencing corrosion – controlling methods – Design and material selection – Cathodic protection - inhibitors - Protective coatings – Metallic coatings (cathodic and anodic) - Methods of application on metals (Galvanizing, Tinning & Electroplating) .

UNIT IV: SOLID STATE CHEMISTRY

Types of solids - close packing of atoms and ions - BCC , FCC, structures of rock salt - cesium chloride- spinel - normal and inverse spinels, Non-elemental **semiconducting Materials**: Stoichiometric, controlled valency & Chalcogen photo/semiconductors, Preparation of Semiconductors - Semiconductor Devices:- p- n junction diode as rectifier – junction transistor.

Insulators (electrical and thermal applications)

Magnetic materials: Ferro and ferri magnetism - Hall- Effect and its applications.

UNIT V: CHEMISTRY OF ADVANCED ENGINEERING MATERIALS

Nano materials: Introduction – Preparation, Properties and engineering applications of Carbon nano tubes and fullerenes.

Liquid crystals: Introduction – Types – Applications.

Superconductors: Type-I & Type-2, properties & applications.

Green Chemistry: Principles, any three methods of synthesis – engineering applications.

Sensors & Biosensors: Classification, working principle & applications.
Explosives & Propellants: Introduction, classification & applications.

Prescribed books:

1. Engineering Chemistry (16^{th} edn.) by Jain and Jain; Dhanpat Rai Pub.Co.
2. A Text book of Engineering Chemistry by S. S. Dara; S. Chand & Co Ltd., Latest Edition.

Reference Books:

1. Chemistry for Engineers by Teh Fu Yen, Imperial college press, London.
2. Engineering Chemistry, Wiley India Pvt. Ltd., Vairam and others, 2014 edition (second).
3. Engineering Chemistry by Shikha Agarwal; Cambridge University Press, 2015 edition.
4. Text book of Nano-science and Nanotechnology by B.S. Murthy, P. Shankar and others, University Press, IIM.

Course Code		L	T	P	Credits
1000171212	ENVIRONMENTAL STUDIES	3	1	0	3

Course Overview:

The course gives a broad view on the importance of environment and its conservation. It deals with distribution of biotic and abiotic components on the Earth, their over exploitation and its associated problems. It provides knowledge on different types of environmental pollutions and their control aspects. It develops practical orientation towards environmental concerns.

Course Objectives:

The objectives of the course are:

1. Classify, describe and explains the concept of Ecosystems and Environmental Engineering.
2. Overallunderstanding of different types of natural resources and its conservation.
3. Acquaintance on various environmental challenges induced due to unplanned anthropogenic activities.
4. An understanding of the environmental impacts of developmental activities and the importance of Environmental Management
5. Awareness on the social issues, environmental legislations and global treaties

Course Outcomes

1. Give an outline of the natural resources and their importance for the sustenance of life and recognize the need to conserve the natural resources.
2. Explain the concepts of the ecosystem and its function in the environment; explains the need for protecting the producers and consumers in various ecosystems and their role in the food web
3. Elucidate the biodiversity of India and threats to biodiversity and conservation practices to protect the biodiversity
4. Give a broad view on various attributes of pollution and their impacts and measures to reduce or control the pollution along with waste management practices.

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Elucidate the natural resource & their importance for the sustenance of life and recognises the need to conserve natural resource	Understanding	PO2,PO5, PO6, PO7, PO12
CO2	Gives the broad view on the various attributes of pollution & and their impact & measure to reduce he pollution along with waste management	Applying	PO2,PO3, PO5, PO6, PO7, PO12

CO3	Debates on social issues both rural and urban environment possible means to combat the challenges and trace the legislation of India towards sustainability	Applying	PO1, PO2,PO5, PO6, PO7, PO12
CO4	Educates about Environmental Impact Assessment, Environmental Impact Statement & Environmental Audit	Analyzing	PO1, PO2, PO4, PO5, PO6, PO7, PO12

UNIT - I: Multidisciplinary nature of Environmental Studies & Natural Resource

Definition, Scope and Importance of Environmental Engineering – Sustainability: Stockholm and Rio Summit–Global

Forest resources– Use and over– exploitation, deforestation – 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, conflict over water, dams– benefits and problems

Mineral resources: Use and exploitation, environmental effects of extracting and using mineral resources

Food resources: World food problems, changes caused by non-agriculture activities- effects of modern agriculture, fertilizer-pesticide problems, water logging, salinity

Energy resources: Growing energy needs, renewable and non-renewable energy sources use of alternate energy sources.

Land resources: Land as a resource, land degradation, Wasteland reclamation, man induced landslides, soil erosion and desertification. Role of an individual in conservation of natural resources. Equitable use of resources for sustainable lifestyles.

UNIT - II: Environmental Pollution

Definition, Cause, effects and control measures of Air pollution,

Water pollution, Soil pollution, Noise pollution, Nuclear hazards. Role of an individual in prevention of pollution. - Pollution case studies.

Environmental Challenges: Global warming and climate change, acid rains, ozone layer depletion, population growth and explosion, effects. Role of information Technology in Environment and human health.

Solid Waste Management: Sources, classification, effects and control measures of urban and industrial solid wastes. Consumerism and waste products.

UNIT - III: Social Issues and the Environment

Urban problems related to energy -Water conservation, rain water harvesting- Resettlement and rehabilitation of people; its problems and concerns.

Environmental ethics: Issues and possible solutions. Environmental Protection Act -Air(Prevention and Control of Pollution) Act. –Water (Prevention and control of Pollution) Act -Wildlife Protection Act -Forest Conservation Act-Issues involved in enforcement of environmental legislation. -Public awareness.

UNIT - IV: Ecosystems, Biodiversity & 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 Forest ecosystem, Grassland ecosystem, Desert ecosystem, Aquatic ecosystems.

Biodiversity and its conservation

Definition: Levels of Biodiversity, Value of biodiversity: consumptive use, productive use, social-Biodiversity at national and local levels. India as a mega- diversity nation - Hot-spots of biodiversity -Threats to biodiversity: habitat loss, man-wildlife conflicts. -Endangered and endemic species of India – Conservation of biodiversity.

UNIT - V: Environmental Management and Field Studies

Impact Assessment and its significance various stages of EIA, Preparation of EMP and EIS, Environmental audit. Eco-tourism, Environmental Economics & Study of a Ecotourism spot in a local area, Visit to some Polluted site. Environmental diary.

Text Books:

1. Environmental Studies by R. Rajagopalan, 2nd Edition, 2011, Oxford University Press.
2. A Textbook of Environmental Studies by ShaashiChawla, TMH, New Delhi
3. Environmental Studies by P.N. Palanisamy, P. Manikandan, A. Geetha, and K. Manjula Rani; Pearson Education, Chennai.

References:

1. Text Book of Environmental Studies by Deeshita Dave & P. Udaya Bhaskar, Cengage Learning.
2. Environmental Studies by K.V.S.G. Murali Krishna, VGS Publishers, Vijayawada.
3. Environmental Studies by Benny Joseph, Tata McGraw Hill Co, New Delhi
4. Environmental Studies by PiyushMalaviya, Pratibha Singh, Anoopsingh: Acme Learning, New Delhi.

Course Code	OBJECT ORIENTED PROGRAMMING	L	T	P	Credits
1000171215	THROUGH C++	3	1	0	3

Course Objectives:

1. To write efficient, maintainable and portable code.
2. To learn and acquire art of computer programming.
3. To know about some popular programming languages and how to choose Programming language for solving a problem.

Course Outcomes:

Students will be able to:

1. Understand the basic terminology of C++.
2. Write, compile and debug programs in C++ language. Use different data types in a computer Program.
3. Design programs involving decision structures, loops and functions.
4. Explain the difference between call by value and call by reference.
5. Usage of generic programming, over loading of functions and operators, over riding and exception handling in various contexts.

UNIT-I: Introduction to C++

Difference between C and C++- Evolution of C++- The Object Oriented Technology- Disadvantage of Conventional Programming- Key Concepts of Object Oriented Programming- Advantage of OOP- Object Oriented Language

UNIT-II: Classes and Objects &Constructors and Destructor

Classes in C++-Declaring Objects- Access Specifiers and their Scope- Defining Member Function-Overloading Member Function- Nested class, Constructors and Destructors, Introduction-Constructors and Destructor- Characteristics of Constructor and Destructor- Application with Constructor- Constructor with Arguments (parameterized Constructor- Destructors- Anonymous Objects**.**

UNIT-III: Operator Overloading and Type Conversion & Inheritance

The Keyword Operator- Overloading Unary Operator- Operator Return Type- Overloading Assignment Operator (=)- Rules for Overloading Operators, Inheritance, Reusability- Types of Inheritance- Virtual Base Classes- Object as a Class Member- Abstract Classes- Advantages of Inheritance-Disadvantages of Inheritance.

UNIT-IV: Pointers & Binding Polymorphisms and Virtual Functions

Pointer to Object- The this Pointer- Pointer to Derived Classes and Base Class, Binding Polymorphisms and Virtual Functions, Introduction- Binding in C++- Virtual Functions- Rules for Virtual Function- Virtual Destructor.

UNIT-V: Generic Programming with Templates & Exception Handling & Overview of Standard Template Library

Generic Programming with Templates, Need for Templates- Definition of class Templates-

Normal Function Templates- Over Loading of Template Function, Difference Between Templates and Macros- Linked Lists with Templates, Exception Handling-Principles of Exception Handling- The Keywords try throw and catch- Multiple Catch Statements – Specifying Exceptions

Overview of Standard Template Library- STL Programming Model- Containers- Sequence Containers- Associative Containers- Algorithms- Iterators- Vectors- Lists- Maps.

Text Books:

1. A First Book of C++, Gary Bronson, CengageLearing
2. The Complete Reference C++, Herbert Schildt, TMH.
3. Programming in C++, Ashok N Kamathane, Pearson 2nd Edition .

Reference Books:

1. Object Oriented Programming C++, Joyce Farrell, Cengage.
2. C++ Programming: from problem analysis to program design, DS Malik, Cengage Learing.

Course Code		L	T	P	Credits
1000171216	**ENGINEERING MECHANICS**	3	1	0	3

Course Overview: The students completing this course are expected to understand the concepts of forces and its resolution in different planes, resultant of force system, Forces acting on a body, their free body diagrams using graphical methods. They are required to understand the concepts of centre of gravity and moments of inertia and their application, Analysis of frames and trusses, different types of motion, friction and application of work – energy method.

Course Objectives:

1. The students are to be exposed to the concepts of force and friction , direction and its application.
2. The students are to be exposed to application of free body diagrams. Solution to problems using graphical methods and law of triangle of forces.
3. The students are to be exposed to concepts of centre of gravity
4. The students are to be exposed to concepts of moment of inertia and polar moment of inertia including transfer methods and their applications.
5. The students are to be exposed to motion in straight line and in curvilinear paths, its velocity and acceleration computation and methods of representing plane motion.
6. The students are to be exposed to concepts of work, energy and particle motion.

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO number mapped
CO1	Analyze the force systems for equilibrium conditions and able to draw free body diagram.	Analyzing	PO1,PO2,PO3
CO2	Evaluate the frictional forces between contact surfaces.	Applying	PO1,PO2,PO3
CO3	Able to differentiate between centroid and centre of gravity and determine Centroid, centre of gravity and second moment of area for composite Sections.	Applying	PO1,PO2,PO3
CO4	Analyse the motion and calculate trajectory characteristics.	Analyzing	PO1,PO2,PO3

UNIT-I: Introduction to Engineering Mechanics – Basic Concepts

Systems of Forces: Coplanar Concurrent Forces – Components in Space – Resultant – Moment of Force and its Application – Couples and Resultant of Force Systems. Free Body Diagrams, Equations of Equilibrium of Coplanar Systems, Spatial Systems for concurrent forces. Lamis Theorm, Graphical method for the equilibrium of coplanar forces, Converse of the law of Triangle of forces, converse of the law of polygon of forces condition of equilibrium.

UNIT-II: Friction

Introduction - limiting friction and impending motion, coulomb's laws of dry friction, coefficient of friction, cone of friction

UNIT-III: Centroid

Centroids of simple figures (from basic principles) – Centroids of Composite Figures **Centre of Gravity :**Centre of gravity of simple body (from basic principles), centre of gravity of composite bodies, pappus theorem.

UNIT IV : Area moments of Inertia : Definition – Polar Moment of Inertia, Transfer Theorem, Moments of Inertia of Composite Figures, Products of Inertia, Transfer Formula for Product of Inertia. **Mass Moment of Inertia: Moment** of Inertia of Masses, Transfer Formula for Mass Moments of Inertia, mass moment of inertia of composite bodies.

UNIT – V: Kinematics: Rectilinear and Curve linear motions – Velocity and Acceleration – Motion of Rigid Body – Types and their Analysis in Planar Motion. **Kinetics:** Analysis as a Particle and Analysis as a Rigid Body in Translation – Central Force Motion – Equations of Plane Motion – Fixed Axis Rotation – Rolling Bodies.
Work – Energy Method: Equations for Translation, Work-Energy Applications to Particle Motion, Connected System-Fixed Axis Rotation and Plane Motion. Impulse momentum method.

Text Books:

1. Engineering Mechanics - S. Timoshenko & D. H. Young., 4th Ed. , Mc Graw Hill publications.
2. Engineering Mechanics: Statics and Dynamics 3rd ed., Andrew Pytel and Jaan Kiusalaas, Cengage Learning publishers.

References:

1. Engineering Mechanics statics and dynamics, R.C. Hibbeler, 11th Ed. Pearson
2. Engineering Mechanics, statics, J. L. Meriam, 6th Edn – Wiley India Pvt Ltd.
3. Engineering Mechanics, dynamics, J. L. Meriam, 6th Edn – Wiley India.

Course Code		L	T	P	Credits
1000171221	ENGLISH COMMUNICATION SKILLS LAB-2	0	0	3	2

Course Objectives:

This Lab focuses on using computer-aided multimedia instruction for language development to meet the following targets:

To improve the students' fluency in English, through a well-developed vocabulary and enable them to listen to English spoken at normal conversational speed by educated English speakers and respond appropriately in different socio-cultural and professional contexts. Further, they would be required to communicate their ideas relevantly and coherently in writing.

Course outcomes: The proposed course to enable students to use 'good' English and perform the following: Gather ideas and information, to organize ideas relevantly and coherently.

Engage in debates. Participate in group discussions. Face interviews. Write project/research reports/technical reports. Make oral presentations.

Writing formal letters and to take part in social and professional communication.

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Demonstrate the usage of phonemes while referring to the dictionary	Understanding	PO5,PO9, PO10, PO12
CO2	Articulate with others by using proper functions.	Applying	PO5,PO9, PO10, PO12
CO3	Enact the roles with proper body language.	Applying	PO5,PO9, PO10, PO12
CO4	Communicate fluently with proper pronunciation	Analyzing	PO5,PO9, PO10, PO12

Unit-1: Pronouncing Words

Unit-2: Interaction 3

Unit-3: Stress & Intonation

Unit-4: Interaction 4

Prescribed Lab Manual:

Speak Well - Orient Black Swan Publishers

Suggested Books/ Manuals and Softwares:

1. Interact - Orient Black Swan
2. The Rosetta Stone English Library
3. Language in Use English in Mind

Course Code		L	T	P	Credits
1000171227	ENGINEERING CHEMISTRY LABORATORY	0	0	3	2

List of Experiments

1. Determination of hardness of water using standard EDTA solution
2. Determination of Total alkalinity of a water sample.
3. Determination of Ferrous iron using standard $K_2Cr_2O_7$ solution.
4. Determination of Copper using standard EDTA solution.
5. Determination of Iron in cement by Colorimetric method
6. Determination of Zinc by ferro cyanide method.
7. Determination of strong acid by Conducto metric titration
8. Determination of Acetic acid by Conducto metric titration
9. Determination of iron by Potentiometric method using $K_2Cr_2O_7$
10. Preparation of Phenol formaldehyde resin
11. Determination of Vitamin – C
12. Determination of flash and fire point of a lubricant oil.
13. Determination of viscosity of a lubricant by Red-wood viscometer.
14. Advanced design experiment - Preparation of Bio diesel.
15. Additional design experiment - Construction of Galvanic cell

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Learn and apply basic techniques used in Chemistry laboratory for small/medium scale water analysis.	Understanding	PO1,PO2, PO9
CO2	Estimate the metal ions present in a domestic/industry sample solutions.	Applying	PO1, PO2, PO7, PO12
CO3	Utilize the fundamental laboratory techniques for titrations and synthetic procedures.	Applying	PO1, PO2, PO9
CO4	Analyze data and gain experimental skills through instrumentation	Analyzing	PO1, PO2, PO5, PO9, PO12

***The student should carry out a minimum of 12 experiments. Outcomes**:
The student is able to acquire principles of various analytical techniques and their applications.

Reference Books:

1. A Textbook of Quantitative Analysis, Arthur J. Vogel.
2. Dr. Jyotsna Cherukuris (2012) Laboratory Manual of engineering chemistry-II, VGS,Techno Series
3. Chemistry Practical Manual, Lorven Publications
4. K. Mukkanti (2009) Practical Engineering Chemistry, B.S. Publication

Course Code	OBJECT-ORIENTED PROGRAMMING	L	T	P	Credits
1000171229	LAB	0	0	3	2

Course Objectives:

1. To strengthen their problem solving ability by applying the characteristics of object-oriented approach.
2. To introduce object oriented concepts in C++ and Java.

Course Outcomes:

1. Explain what constitutes an object-oriented approach to programming and identify potential benefits of object-oriented programming over other approaches.
2. Apply an object-oriented approach to developing applications of varying complexities.

Programming Exercise-1 (Basics)

Write a Simple Program on printing "Hello World" and "Hello Name" where name is the input from the user.

a) Convert any two programs that are written in C into C++
b) Write a description of using g++ (150 Words)

Exercise-2 (Expressions Control Flow)

a) Write a Program that computes the simple interest and compound interest payable on principal amount (Rs.) of loan borrowed by the customer from a bank for a given period of time (in years) at specific rate of interest. Further determine whether the bank will benefit by charging simple interest or compound interest.
b) Write a Program to calculate the fare for the passenger starveling in a bus. When a Passenger enters the bus, the conductor asks "What distance will you travel?" On knowing distance from passenger.
c) (As an approximate integer), the conductor mentions the fare to the passenger according to following criteria.

Exercise-3 (Variables, Scope, Allocation)

a) Write a program to implement call by value and call by reference using reference variable.
b) Write a program to illustrate scope resolution, new and delete Operators. (Dynamic Memory Allocation)
c) Write a program to illustrate Storage classes
d) Write a program to illustrate Enumerations

Exercises-4 (Functions)

Write a program illustrating Inline Functions

a) Write a program illustrate function overloading. Write 2 overloading functions for power.
b) Write a program illustrate the use of default arguments for simple interest function.

Exercise-5 (Functions –Exercise Continued)

a) Write a program to illustrate function overloading. Write 2 overloading functions for adding two numbers
b) Write a program illustrate function template for power of a number.
c) Write a program to illustrate function template for swapping of two numbers.

Exercise-6 (Classes Objects) Create a Distance class with:

- feet and inches as data members
- member function to input distance
- member function to output distance
- member function to add two distance objects

a) Write a main function to create objects of DISTANCE class. Input two distances and output the sum.
b) Write a C++ Program to illustrate the use of Constructors and Destructors (use the above

program.)

c) Write a program for illustrating function overloading in adding the distance between objects (use the above problem)

d) Write a C++ program demonstrating a Bank Account with necessary methods and variables

Exercise-7 (Access)

a) Write a program for illustrating Access Specifiers public, private, protected

b) Write a program implementing Friend Function

c) Write a program to illustrate this pointer

d) Write a Program to illustrate pointer to a class

Exercise -8 (Operator Overloading)

a) Write a program to Overload Unary, and Binary Operators as Member Function, and Non Member Function.

   i. Unary operator as member function
   ii. Binary operator as nonmember function

b) Write a c ++ program to implement the overloading assignment = operator

c) Write a case study on Overloading Operators and Overloading Functions (150 Words)

Exercise -9 (Inheritance)

a) Write C++ Programs and incorporating various forms of Inheritance

   i. Single Inheritance
   ii. Hierarchical Inheritance
   iii. Multiple Inheritances
      a. Multi-level inheritance
      b. Hybrid Inheritance

b) Write a program to show Virtual Base Class

c) Write a case study on using virtual classes (150 Words)

Exercise-10 (Inheritance –Continued)

a) Write a Program in C++ to illustrate the order of execution of constructors and destructors in inheritance

b) Write a Program to *show* how *constructors* are invoked in *derived class*

Exercise -11 (Polymorphism)

a) Write a program to illustrate runtime polymorphism

b) Write a program to illustrate this pointer

c) Write a program illustrates pure virtual function and calculate the area of different shapes by using abstract class.

d) Write a case study on virtual functions (150 Words)

Exercise -12(Templates)

a) Write a C++ Program to illustrate template class

b) Write a Program to illustrate class templates with multiple parameters

c) Write a Program to illustrate member function templates

Exercise -13 (Exception Handling)

a) Write a Program for Exception Handling Divide by zero

b) Write a Program to rethrow an Exception.

Exercise -14 (STL)

a) Write a Program to implement List and List Operations

b) Write a Program to implement Vector and Vector Operations

Exercise -15 (STL Continued)

a) Write a Program to implement Deque and Deque Operations

b) Write a Program to implement Map and Map Operations

PROGRAM STRUCTURE

FOR

II-B.Tech

I & II SEMESTERS

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
PROGRAM STRUCTURE

II B.Tech **I Semester**

S.No	Course Code	Course Title	L	T	P	Credits
1	1005172101	Statistics and R Programming	3	1*	0	3
2	1005172102	Mathematical Foundations of Computer Science	3	1*	0	3
3	1005172103	Digital Logic Design	3	1*	0	3
4	1005172104	Java Programming	3	1*	0	3
5	1005172105	Data Structures through C	3	1*	0	3
6	1005172106	Problem Solving and Program Design Through C	3	1*	0	3
7	1005172121	Data Structures through C Lab	0	0	3	2
8	1005172122	Java Programming Lab	0	0	3	2
Total Credits						**22**

II B.Tech **II Semester**

S.No	Course Code	Course Title	L	T	P	Credits
1	1005172201	Database Management Systems	3	1*	0	3
2	1005172202	Software Engineering	3	1*	0	3
3	1005172203	Advanced Data Structures	3	1*	0	3
4	1005172204	Computer Organization	3	1*	0	3
5	1005172205	Formal Languages and Automata Theory	3	1*	0	3
6	1005172206	Operating Systems	3	1*	0	3
7	1005172221	Advanced Data Structures Lab	0	0	3	2
8	1005172222	Database Management Systems Lab	0	0	3	2
9	1005172231	Industrial Visit/ Online certification course from Nptel or equivalent on any one programming language (C, C++, Java, Python or any state-of-art Courses).	0	0	0	2
Total Credits						**24**

DETAILED SYLLABUS

FOR

II-B.Tech

I-SEMESTER

Course Code	STATISTICS AND R PROGRAMMING	L	T	P	Credits
1005172101		3	1	0	3

Course Overview:
This course focusing about fundamental of R programming, handling and analysis of data, visualization techniques. determine the most appropriate approach

Course Objectives:
1. After taking the course, students will be able to
2. Use R for statistical programming, computation, graphics, and modeling
3. Write functions and use R in an efficient way
4. Fit some basic types of statistical models
5. Use R in their own research
6. Be able to expand their knowledge of R on their own.

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Understand the R workspace and Programming with R.	Understanding	PO:1,2,4,5, 12
CO2	Explain the control statements, Loops, Operators and Functions of programming structures using R	Analyzing	PO:1,2,3,4, 12
CO3	Apply math functions and simulation to calculate probability and statistical distributions using R.	Applying	PO:1,2,3,4, 5
CO4	Perform statistical tests using R to create and visualize graphics and Explore data-sets to create testable hypotheses and identify appropriate statistical tests.	Applying	PO:1,2,3,4, 5

Unit-I:
Introduction: How to run R, R Sessions and Functions, Basic Math, Variables, Data Types, Vectors, Conclusion, Advanced Data Structures, Data Frames, Lists, Matrices, Arrays, Classes
Outcome: Understanding the R studio software. Able to work with Basics of R Programming.
Activity: Installation of R Studio in Personal systems or Mobile APP

Unit-II:
R Programming Structures, Control Statements, Loops, - Looping Over Non-vector Sets,- If-Else, Arithmetic and Boolean Operators and values, Default Values for Argument, Return Values, Deciding Whether to explicitly call return- Returning Complex Objects, Functions are Objective. No Pointers in R, Recursion, A Quick sort Implementation-Extended Example: A Binary Search Tree.
Outcome: Understanding the programming structure of R. Functions, Recursion concepts.

Activity: Working with R studio, understanding the R Environment and running the R scripts.

Unit-III:

Doing Math and Simulation in R, Math Function, Extended Example Calculating Probability- Cumulative Sums and Products-Minima and Maxima- Calculus, Functions For Statistical Distribution, Sorting, Linear Algebra Operation on Vectors and Matrices, Extended Example: Vector cross Product- Extended Example: Set Operation, Input /output, Accessing the Keyboard and Monitor, Reading and writer Files,

Introduction to different Statistics tools, Graphics, Creating Graphs, The Workhorse of R Base graphs, the plot() ,par() Function – Customizing Graphs, Saving Graphs to Files. gplot usage.

Outcome:

Importance of Linear algebra and operations on vector data. Able to the various graphs

Activity:

Generating the graphs using different data sets and understanding particle approach of analysis based on graphs.

Unit-IV:

Conditional probability and Bayes theorem, Basic Statistics: Mean median, mode, standard deviation and Covariance, Random variables. Probability distributions: Binomial, Poisson and Normal, Markov process

Outcome: Understanding Probability distribution of various models.

Activity: Generating the graphs using different data sets for Probability distribution models.

Unit-V:

T –Test, ANOVA

Correlation and Regression**:** Linear Models, Simple Linear Regression, -Multiple Regression Generalized Linear Models, Survival Analysis, Nonlinear Models, Splines- Decision- Random Forests.

Outcome: Able to understand the regression models and importance of Survival analysis.

Activity: Implementation of Regression techniques with data set.

Text Books:

1. The Art of R Programming, Norman Matloff, Cengage Learning R for Everyone, Lander, Pearson
2. Discrete Mathematical Structures with Applications to Computer Science, J. P. Tremblay and P. Manohar, Tata McGraw Hill.
3. Probability and Statistics for Engineers: Miller and John E. Freund, Prentice Hall of India

Reference Books:

1. R Cookbook, PaulTeetor, Oreilly.
2. R in Action,Rob Kabacoff, Manning

Course Code	MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE	L	T	P	Credits
1005172102		3	1	0	3

Course Overview:

Mathematical foundation of computer science is a systematic way to implement the logic in programs and to construct the algorithm efficiently. Following terms are the foundation terms of a Mathematical foundation of computer science.

The purpose of this course is to provide the students with solid foundations in the basic concepts of mathematical logic, predicates, graph theory concepts, and algorithms. The main objective of the course is to teach the students how to implement the concept and how to design the given data and algorithms that are appropriate for problems that they might encounter. This course is also about showing the correctness of algorithms and studying their computational complexities. This course offers the students a mixture of theoretical knowledge and practical experience. The study of Mathematical foundation of computer science is carried out with Mathematical concepts like Mathematical methods, Engineering mathematics, etc,.

Course Objectives:

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

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	To demonstrate skills in solving counting problem	Applying	PO1, PO2, PO3, PO4
CO2	To develop reasoning skills to implement in programming	Analyzing	PO1, PO2, PO3
CO3	To understand knowledge of mathematical modeling and proficiency in using mathematical software.	Understanding	PO1, PO2
CO4	To solve recurrence relations and able to implement structural models, design concepts using graph theory	Applying	PO1, PO2, PO3, PO12

Unit-I: Combinatorics, Number Theory:

Basic of Counting, Permutations, Permutations with Repetitions, Circular Permutations, Restricted Permutations, Combinations, Restricted Combinations, Generating Functions of Permutations and Combinations, Binomial and Multinomial Coefficients, Binomial and Multinomial Theorems, The Principles of Inclusion–Exclusion, Pigeonhole Principle and its Application.

Number Theory:

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

Outcome: Student will be able to demonstrate skills in solving counting problem.

Activity: Solve counting problems for real-time applications.

Unit-II : 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

Outcome: Student will be able to comprehend mathematical principles and logic

Activity: Identify applications of mathematical principles in real-time and implement it with in programs.

Unit-III: Set Theory, And Algebraic Structures:

Introduction, Operations on 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, Recursive Functions, Lattice and its Properties.

Algebraic Structures:

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

Outcome: Student will be able to demonstrate knowledge of mathematical modeling and proficiency in using mathematical software.

Activity: Select an appropriate optimized sorting/searching technique for a real-time application and justify.

Unit-IV: Recurrence Relations:

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

Outcome: Students will be able to apply induction and other proof techniques towards solving recurrences and other problems in elementary algebra.

Activity: Recurrence relations are used when an exhaustive approach to problem solving is simply too arduous to be practical. Although it is not ideal to compute the terms in a sequence one at a time by using previous terms, this approach can be much more efficient than the alternative of exhaustive casework.

Unit-V: Graph Theory:

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

Outcome:

1. Student will be able to manipulate and analyze data numerically and/or graphically using appropriate Software.
2. Demonstrate the use of Trees, Binary Trees in various applications.

Activity: Construct Graphs for an real time Application.

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, Bernand Kolman, Robert C. Busby, Sharon Cutler Ross, PHI.
3. Discrete Mathematics, S. K. Chakraborthy and B.K. Sarkar, Oxford, 2011.

.

Course Code	DIGITAL LOGIC DESIGN	L	T	P	Credits
1005172103		3	1	0	3

Course Overview:
This course provides an introduction to logic design and the basic building blocks used in digital systems, in particular digital computers. It starts with a discussion of combinational logic: logic gates, minimization techniques, arithmetic circuits, and modern logic devices such as field programmable logic It also deals with sequential circuits: flip-flops, synthesis of sequential circuits, and case studies, including counters, registers, and random access memories.

Course Objectives:
1. To solve typical number base conversions
2. To optimize logic gates using various techniques
3. To introduce the basic tools for designing combinational and sequential digital logic.

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Understand the conversions in number system and Develop the logic circuits using logic gates.	Understand	PO1, PO2
CO2	Minimize the boolean logic circuits using K-maps.	Understand	PO1, PO2, PO3
CO3	Construct and analyze the operation of combinational logic circuits.	Apply	PO1, PO2, PO3, PO4
CO4	Develop the various types of sequential logic circuits like flip flops, registers and counters.	Create	PO1, PO2, PO3, PO5, PO12

Unit-I: Digital Systems and Binary Numbers:
Digital Systems, Binary Numbers, Octal and Hexadecimal Numbers, Complements of Numbers, Signed and unsigned Binary Numbers, Arithmetic addition and subtraction, Binary coding.
Outcome:
1. Analyze and examine logic circuits by applying the knowledge of number systems, codes.
2. To define different number systems, binary addition and subtraction, 2's complement representation and operations with this representation

Activity: Illustrate how number system is applicable in real time with an example.

Unit-II: Concept of Boolean algebra:
Basic Theorems and Properties of Boolean algebra, Boolean Functions, Canonical and Standard Forms, Minterms and Maxterms.
Outcome:
1. Demonstrate Gate-Level minimization through Boolean algebra**.**

2. Evaluate and simplify logical functions using Boolean algebra.

Activity: Reduce Boolean expression by applying Boolean algebra postulates and manipulate into SOP/POS form and verify it with Truth table.

Unit-III: Gate level Minimization & Combinational Logic:

Map Method, Two-Variable K-Map, Three or more -Variable K-Map. Products of Sum Simplification, Sum of Products Simplification, Don't – Care Conditions, Quine-McCluskey method, NAND and NOR Implementation, Exclusive-OR Function. Introduction, Analysis Procedure, Design Procedure, Binary Adder–Subtractor, Decimal Adder, Binary Multiplier, Decoders, Encoders, Multiplexers, HDL Models of Combinational Circuits

Outcome:

1. Analyze and design combinatorial circuits
2. Simplify combinatorial circuits using Karnaugh maps.
3. Implement functions with NAND-NAND and NOR-NOR logic.

Activity: Design digital circuit using HDL.

Unit-IV: Synchronous Sequential Logic:

Introduction to Sequential Circuits, Storage Elements: Latches, Storage Elements: FlipFlops, Analysis of Clocked Sequential Circuits, Mealy and Moore Models of Finite State Machines

Outcome: Analyze and design Sequential Logic Circuits and summarize the differences Combinational and Sequential Circuits.

Activity: Demonstrate the use of sequential circuits and storage elements in real-time applications.

Unit-V: Registers and Counters:

Registers, Shift Registers, Ripple Counters, Synchronous Counters, Ring Counter, Johnson Cc Ripple Counter

Outcome: Analyze and Design Registers and counters.

Activity: Demonstrate the use of registers and counters in real-time applications.

Text Books:

1. Digital Design, 5/e, M.Morris Mano, Michael D Ciletti, PEA.
2. Fundamentals of Logic Design, 5/e, Roth, Cengage.

Reference Books:

1. Digital Logic and Computer Design, M.Morris Mano, PEA.
2. Digital Logic Design, Leach, Malvino, Saha, TMH.
3. Modern Digital Electronics, R.P. Jain, TMH.

Course code	JAVA PROGRAMMING	L	T	P	Credits
1005172104		3	1	0	3

Course Overview:

1. Java has emerged as the object-oriented programming language of choice.
2. Some of the important concepts of Java include are:
   1) A Java virtual machine (JVM), which provides the fundamental basis for platform independence
   2) Automated storage management techniques, such as garbage collection, collection frameworks
   3) Language syntax that is similar to that of the C language.

Course Objectives:

1. To Understanding the object oriented programming concepts like Data Abstraction, Encapsulation, Inheritance and Polymorphism.
2. Gain the knowledge about the relationship between the classes and objects.
3. Understand the principles of Inheritance, Packages, Multithreading and Interfaces.
4. To understand and apply the concepts of Applets and AWT.

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Identify the concepts and features of object oriented programming in Java.	Understanding	PO1, PO2
CO2	Describe and implement the programs with command line arguments and Scanner Class.	Analyzing	PO1, PO2, PO3,PO5
CO3	Analyze and implement the concepts of Inheritances and Multithreading with real world scenario.	Applying	PO1, PO2, PO3,PO5
CO4	Develop GUI programs using Applets and Event Handling.	Applying	PO1, PO2, PO3,PO5

Unit-I:

Introduction to OOP, procedural programming language and object oriented language, principles of OOP, applications of OOP, history of java, java features, JVM, program structure.

Variables, primitive data types, identifiers, literals, operators, expressions, precedence rules and associativity, primitive type conversion and casting, flow of control.

Outcome:

After reading this Unit, student should be able to understand:

- Be familiar with Object oriented programming techniques.
- Explain the structure of the program
- Demonstrate various control structures in JAVA.

Activity: Simulate various control structures for real time applications.

Unit-II:

Abstract Data Type, Classes and objects, class declaration, creating objects, methods, constructors and constructor overloading, garbage collector, importance of static keyword and

examples, this keyword, arrays, command line arguments, nested classes.

Outcome:

After reading this Unit, student should be able to understand:

- Outline the relation between class and object.
- Illustrate the difference between method and constructor overloading.
- Make use of static keyword and this keyword.
- Analyze the Command Line arguments.

Activity: Develop real time applications using OOPs concepts through various ADT's.

Unit-III:

Inheritance, types of inheritance, super keyword, final keyword, overriding and abstract class. Interfaces, creating the packages, using packages, importance of CLASSPATH and java.lang package. Exception handling, importance of try, catch, throw, throws and finally block, user defined exceptions, Assertions, Collection inbuilt classes.

Multithreading: introduction, thread life cycle, creation of threads, thread priorities, thread synchronization, communication between threads. Reading data from files and writing data to files.

Outcome:

After reading this Unit, student should be able to understand:

- Classify various types of Inheritance.
- Illustrate the difference between method overloading and overriding.
- Demonstrate to usage of Packages.
- Make use of Exception Handling.
- Develop and make use of synchronization through multithreading.

Activity: Develop enhanced applications from existing versions to new versions.

Unit-IV:

Applet class, Applet structure, Applet life cycle, sample Applet programs. Event handling: event delegation model, sources of event, Event Listeners.

Outcome:

After reading this Unit, student should be able to understand:

- Explain the structure of Applet Program.
- Construct an approach for event delegation model.
- Build the frame based applications using event handling mechanism.

Activity: Develop client browser applications with Graphics

Unit-V:

Java Swing package and AWT package: introduction, components and containers, JButton, Jlabel, JCheckbox, JRadio Button, JListJBoxes, JChoice Boxes, JContainer class, JLayouts, JMenu and JScrollbar.

Outcome:

After reading this Unit, student should be able to understand:

- Extend the importance of AWT.
- Develop the components and containers in AWT.
- Develop the GUI application using checkboxes, radio buttons, List Boxes etc.
- Construct different types of Layouts.

Activity: Develop a client side module which contains checkboxes, text fields, text area, radio buttons etc.

Text Books:

1. The complete Reference Java, 8th edition, Herbert Schildt, TMH.
2. Programming in JAVA, Sachin Malhotra, Saurabh Choudary, Oxford.
3. Introduction to java programming, 7th edition by Y Daniel Liang, Pearson.

Reference Books:

1. Swing: Introduction, JFrame, JApplet, JPanel, Componets in Swings, Layout Managers in
2. Swings, JList and JScrollPane, Split Pane, JTabbedPane, JTree, JTable, Dialog Box.

Course Code		L	T	P	Credits
1005172105	DATA STRUCTURES THROUGH C	3	1	0	3

Course Overview:

Data Structure is a systematic way to organize data in order to use it efficiently. Following terms are the foundation terms of a data structure.

The purpose of this course is to provide the students with solid foundations in the basic concepts of programming: data structures and algorithms. The main objective of the course is to teach the students how to select and design data structures and algorithms that are appropriate for problems that they might encounter. This course is also about showing the correctness of algorithms and studying their computational complexities. This course offers the students a mixture of theoretical knowledge and practical experience. The study of data structures and algorithms is carried out with C Language.

Course Objectives:

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

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Understand data structures concepts for solving computing problems.	Understand	PO1, PO2
CO2	Implement standard data structures like stack, queue	Applying	PO1, PO2, PO3, PO5
CO3	Understand sorting and searching algorithms to the small and large data.	Understand	PO1, PO2
CO4	Apply AND Implement basic data structures such as trees for real-time applications	Applying	PO1, PO2, PO3, PO4, PO5

Unit-I: Arrays And Linked Lists:

Abstract Data Types(ADTs) , Dynamic allocation of Arrays, Structures and unions, Polynomials, Spares Matrices Representation of multidimensional Arrays. Single Linked List and Chains, Representing Chains in C, Polynomials, Polynomial Representation- Adding Polynomials- Circular List Representation of Polynomials, Equivalence Classes, Sparse Matrices, Sparse Matrix Representation- Sparse Matrix Input-Deleting a Sparse Matrix, Doubly Linked Lists.

Outcome:

1. Differentiate primitive and non primitive data structures.
2. Design and apply appropriate data structures for solving computing problems.
3. Real time applications of arrays and Linked Lists.

Activity: Construct an Abstract Data Type for real-time applications.

Unit-II: Stacks and Queues:

The Stack, Stacks using Dynamic Arrays, Recursion, Linked Stacks, The Queue, Linked Queues, Circular Queues using Dynamic Arrays, De-queue. Application of stacks and queues, Evaluation of Expressions, Expression- Postfix Notation- Infix to Postfix Towers Of Hanoi Problem

Outcome:

1. Implement standard data structures like stack, queue
2. Able to implement real time applications on Stacks and Queues.

Activity: Identify applications of stack/Queue in real-time and implement it with Stack/Queue.

Unit-III: Searching and Sorting:

Searching: Linear Search, Binary Search, Fibonacci search. Bubble Sort, Selection Sort, Insertion Sort, Quick Sort, Merge Sort Merging, Iterative Merge Sort, Recursive Merge Sort, Heap Sort

Outcome: Apply sorting and searching algorithms to the small and large data sets.

Activity: Select an appropriate optimized sorting/searching technique for a real-time application and justify.

Unit-IV: Trees:

Introduction, Terminology, Representation of Trees, Binary Trees, The Abstract Data Type, Properties of Binary Tress, Binary Tree Representations, Binary Tree Traversal and Tree Iterators, Introduction, Inorder Traversal Preorder Traversal, Postorder Traversal, Threaded Binary Trees, Inorder Traversal of a Threaded Binary Tree, Inserting a Node into a Threaded Binary Tree

Outcome:

1. Summarize basic tree concepts, operations and applications.
2. Apply basic data structures such as trees for real-time applications.

Activity: Construct an optimized binary tree with age of your family members and perform traversals on that tree. Outline the observations.

Unit-V: Advanced Concept of Trees:

Binary Search Trees, Definition, Searching a Binary Search Tree, Insertion into a Binary Search Tree, Deletion from a Binary Search Tree, Height of Binary Search Tree. Heaps, Priority Queues, Definition of a Max/Min Heap, Insertion into a Max/Min Heap, Deletion from a Max/Min Heap

Outcome:

1. Demonstrate the use of Heaps in various applications.
2. Demonstrate the use of Binary Search Trees in various applications.

Activity: Construct a Binary Search Tree for an Arithmetic Expression.

Text Books:

1. Fundamentals of Data Structures in C, Ellis Horowtiz, S.Sahni, Andrews Freed, University Press (India). Second Edition.
2. Data Structures and Algorithm Analysis in C, Mark Allen Weiss, Pearson Education. Ltd., Second Edition.

Reference Books:

1. Classic Data Structures, Debasis Samantha, PHI. (Second Edition)
2. Data Structures using C and C++, Langsam, Augenstein and Tanenbaum, PHI.
3. Data Structures using C, Reema Thareja, Oxford Home Publications, Second Edition.

Course Code		L	T	P	Credits
1005172106	PROBLEM SOLVING AND PROGRAM DESIGN THROUGH C	3	1	0	3

Course Overview:

Problem Solving and Program design in C presents a structure, process and the data structures to help solve programming problems.

Course Objectives:

Problem Solving and Program design in C teaches a disciplined approach to problem solving and to applying widely accepted software engineering methods to design program solutions as coupling, cohesion, modular programming and debugging

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Explain the step by step notation to solve simple mathematical and logical problems	Understand	PO1, PO2, PO5, PO6, PO7, PO12
CO2	Implement the "C" Programs for solutions of mathematical and logical problems	Applying	PO1, PO2, PO4, PO5, PO6, PO12
CO3	Apply code reading and debugging techniques to analyze and interpret and describe the purpose of program code	Applying	PO1, PO2, PO3, PO4, PO5, PO6, PO12
CO4	Apply the modular Programming techniques to simplify the programs	Applying	PO1, PO2, PO3, PO5, PO6, PO11, PO12

Unit-I: Top Down Design And Functions:

Top-down Design with Functions: Building programs from existing information- case studies, library functions, top down design and structure charts - case study, functions without arguments, functions with input arguments, Case Study. Functions with simple output parameters, Multiple calls to a function with input/output parameters, Scope of names, Formal output parameters and Actual Arguments,

Problems:

Add addition of two polynomials, Multiplication of two polynomials. Implement the following functions with input as string: is-alphabet, is-digit, convert-to-upper, and convert-to-lowercase

Outcome:

1. Demonstrate the step by step notation to solve simple mathematical and logical problems
2. Apply the modular programming techniques to simplify the programs.
3. Implement the "C" programs for solutions of mathematical and logical problems.

Activity: Draw a Structural Chart for Real Time Applications.

Unit-II: Recursions:
The nature of recursion, tracing a recursive function, Linear Recursion, Mutual recursion, Recursive Mathematical functions, comparing recursion and iteration.

Problems: Factorial, Fibonacci, GCD, LCM, permutations of a given string, Generate all strings of n bits of binary digits, TOWERS of HANOI, binary search, Recursive Bubble Sort , N queens problem, sum of digits of a number.

Outcome:
1. Demonstrate the step by step notation to solve simple mathematical and logical problems
2. Apply the modular/ recursion programming techniques to simplify the programs.
3. Implement the "C" programs for solutions of mathematical and logical problems.

Activity: Simulate Test and Debug the Various Recursion Problems in IDE.

Unit-III: Bit Manipulations:
Bitwise Operators, little and big Indian.

Problems:
Convert given number to binary, count set bit of a number, Count number of bits to be flipped to convert A to B, Find one extra character in a string, Maximum XOR-value of at-most k-elements from 1 to n, Divide two integers without using multiplication, division and mod operator, Toggle all even bits of a number, XNOR of two numbers, find whether a no is power of two.

Outcome:
1. Implement the "C" programs for solutions of mathematical and logical problems.
2. Apply code reading and debugging techniques to analyze and interpret and describe the purpose of program code

Activity: Develop Logical Circuits for real time applications.

Unit-IV: Array:
Array arguments, Parallel Arrays and Enumerated Types, Multidimensional arrays, representation of pointer using Arrays

Pointers:
Introduction to pointers, Pointers and Arrays, Pointers and Strings, Pointers and Structures, Pointers and data structures, Applications of pointers.

Array Problems:
Maximum possible difference of two subsets of an array, Program for array rotation, Find the minimum element in a sorted and rotated array, Find Second largest element in an array, Find the largest K elements in an array, Elements of an array that are not divisible by any element of another array.

Outcome:

1. Demonstrate the step by step notation to solve simple mathematical and logical problems
2. Implement the simple programs using derived and user defined data types to organize the date items.
3. Apply code reading and debugging techniques to analyze and interpret and describe the purpose of program code.

Activity: Develop a Module by using Arrays.

Unit-V: String Manipulations:

Strings: String library functions, Structures and Union types: User-defined structure types, structure type data as input and output parameters, functions whose result values are structured, Problem Solving with Structured Types

String Problems:

Caesar cipher encryption, count vowels and consents and special characters of string, counting consecutive vowels from string, String with maximum number of unique characters in a given set of strings.

Outcome:

1. Demonstrate the step by step notation to solve simple mathematical and logical problems
2. Apply code reading and debugging techniques to analyze and interpret and describe the purpose of program code.

Activity: Build a Pre-processor Directive.

Text Books:

1. Problem Solving and Program Design in C, Jeri R. Hanly, Ellot B. Koffman, 7th Edition, Pearson.
2. Kanetkar, Yashavant P. Understanding Pointers In C. Bpb Publications, 2003.
3. 101 Programming puzzle problems solved: High School Junior to Seniors Join us to win Informatics Olympiad, N.B.Venkateswarlu, Feb, 2015. (Kindle edition: http://www.amazon.com/dp/B00T1OK42K)
   1. https://www.geeksforgeeks.org/
   2. https://icpc.baylor.edu/

Reference Books:

1. Programming in C, PradipDey, Manas Ghosh, 2nd Edition, OxfordUniversityPress.
2. How to Solve it by Computer- R.G.Dromey,PHI.
3. A First Book of ANSI C, Gary J.Bronson, 3rd Edition,Cengage.
4. A Book on C, AL KELLY and IRA POHL, 4th Edition,Pearson.
5. The C Programming Language, Brain W.Kernighan& Dennis Ritchie, 2nd Edition,

Course Code		L	T	P	Credits
1005172121	DATASTRUCTURES THROUGH C LAB	0	0	3	2

Course Objectives:

1. To develop skills to design and analyze simple linear and non linear data structures
2. To Strengthen the ability to identify and apply the suitable data structure for the given real world problem
3. To Gain knowledge in practical applications of data structures

List of Experiments:

1. Develop C programs to implement the following using an array
   a) Linear search b) binary search
2. Develop a C Program to find number of comparisons and swapping for a given list of numbers
   a) Bubble Sort b) Selection Sort
3. Develop a c program to implement Merge Sort
4. Develop a c program to implement Quick Sort
5. Develop C programs to implement the following using an array.
   a) Stack b) Queue
6. Develop C program to Implement Multistack in a Single Array.
7. Develop a C program to do the following
   a) Infix to postfix conversion. b) Evaluation of postfix expression.
8. Implementation of single linked list.
9. Implementation of Doubly linked list.
10. Implement double ended queue using a doubly linked list and an array.
11. Write C programs that use non-recursive functions to traverse the given binary tree in
    a) Pre-order b) In-order c) Post-order.
12. Implementation of Binary Search trees.
13. Implementation of Heaps.

Course outcomes:

1. At the end of this lab session, the student will be able to design and analyze the time and space efficiency of the data structure be capable to identity the appropriate data structure for given problem
2. Have practical knowledge on the application of data structures

Text books:

1. Fundamentals of Data structures 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.,Second Edition

Course Code		L	T	P	Credits
1005172122	JAVA PROGRAMMING LAB	0	0	3	2

Objectives:

1. To understand object oriented programming concepts, and apply them in problem solving.
2. To learn the basics of java Console based programming, GUI based programming and networking programming.

List of Experiments:

Exercise - 1 (Basics)

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

b) Write a java program that display the roots of a quadratic equation ax2+bx=0.
Calculate the discriminate D and basing on value of D, describe the nature of root.

c) Five Bikers Compete in a race such that they drive at a constant speed which may or may not be the same as the other. To qualify the race, the speed of a racer must be more than the average speed of all 5 racers. Take as input the speed of each racer and print back the speed of qualifying racers.

d) Write a case study on public static void main(250 words)

Exercise - 2 (Operations, Expressions, Control-flow, Strings)

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

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

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

d) Write a JAVA program using String Buffer to delete, remove character.

Exercise - 3 (Class, Objects)

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

b) Write a JAVA program to implement constructor.

Exercise - 4 (Methods)

a) Write a JAVA program to implement constructor overloading.

b) Write a JAVA program implement method overloading.

Exercise - 5 (Inheritance)

a) Write a JAVA program to implement Single Inheritance

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

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

Exercise - 6 (Inheritance - Continued)

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

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

Exercise - 7 (Exception)

a)Write a JAVA program that describes exception handling mechanism

b)Write a JAVA program Illustrating Multiple catch clauses

Exercise – 8 (Runtime Polymorphism)

a) Write a JAVA program that implements Runtime polymorphism

b) Write a Case study on run time polymorphism, inheritance that implements in above problem

Exercise – 9 (User defined Exception)

a) Write a JAVA program for creation of Illustrating throw

b) Write a JAVA program for creation of Illustrating finally

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

d)Write a JAVA program for creation of User Defined Exception

Exercise – 10 (Threads)

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

b) Write a program illustrating is Alive and join

c) Write a Program illustrating Daemon Threads.

Exercise - 11 (Threads continuity)

a)Write a JAVA program Producer Consumer Problem

b)Write a case study on thread Synchronization after solving the above producer consumer problem

Exercise – 12 (Packages)

a) Write a JAVA program illustrate class path

b) Write a case study on including in class path in your os environment of your package.

c) Write a JAVA program that import and use the defined your package in the previous Problem

Exercise - 13 (Applet)

a) Write a JAVA program to paint like paint brush in applet.

b) Write a JAVA program to display analog clock using Applet.

c) Write a JAVA program to create different shapes and fill colors using Applet.

Exercise - 14 (Event Handling)

a) Write a JAVA program that display the x and y position of the cursor movement using Mouse.

b) Write a JAVA program that identifies key-up key-down event user entering text in applet.

Course outcomes:

1. Understanding the basics of Java programming, Inheritance, Multithreading and Exception Handling.
2. The skills to apply OOP and Java programming in problem solving.
3. Use of GUI based concepts like Applets and AWT.
4. Should have the ability to extend his/her knowledge in Java programming with his/her own business logic.

Text books:

1. The complete Reference Java, 8th edition, Herbert Schildt, TMH.
2. Programming in JAVA, Sachin Malhotra, SaurabhChoudary, Oxford.

DETAILED SYLLABUS

FOR

II-B.Tech

II-SEMESTER

Course Code 1005172201	DATABASE MANAGEMENT SYSTEMS	L	T	P	Credits
		3	1	0	3

Course Overview:

This course introduces database design and creation using a DBMS product. Emphasis is on data dictionaries, normalization, data integrity, data modelling, and creation of simple tables, queries, reports, and forms. Upon completion, students should be able to design and implement normalized database structures by creating simple database tables, queries, reports, and forms.

Course Objectives:

1. Provide students with theoretical knowledge and practical skills in the use of database and database management systems in information technology applications.
2. The logical design, physical design and implementation of relational databases are covered.

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Describe ER model and normalization f or database design.	Analyzing	PO1, PO2, PO4
CO2	Create, maintain and manipulate a relational database using SQL	Applying	PO1,PO2,PO4,PO5
CO3	Design and build database system for a given real world problem	Applying	PO1,PO2,PO4,PO5
CO4	Examine issues in data storage and query processing and can formulate appropriate solutions.	Understand	PO1, PO2

Unit-I: Introduction to Database Systems, :

File System Vs DBMS, Advantages of DBMS, Structure of DBMS, Levels of Data Abstraction (Data Independence), Database Users and Administrators, Different Data Models.

E-R Model:

Overview of Database Design, Entities, Attributes, and Entity Sets, Relationships and Relationship Sets, Additional Features of the ER Model.

Outcome:

After Completion of the Unit, Student will Be able to:

1. Describe the Architecture of Database Management Systems

2. Design different ER Models
3. Understand the applications of dbms, difference between file systems vs dbms, identify the data models ,understand dbms structure

Activity: Draw ER Diagram for Various Real Time Systems.

Unit-II: Relational model:
Introduction to the Relational Model, Relational model constraints over relations. Relational Algebra and calculus

Outcome:
After Completion of the Unit, Student will Be able to:
1. To differentiate the knowledge in TRC & DRC
2. Compare relational model with the structured query language (SQL)
3. Understands the relational algebra concepts, selection ,projection ,relational calculus which helps in understanding queries

Activity: Tabulate Various Relational Models for Real Time Application.

Unit-III: SQL Queries:
The Form of Basic SQL Query, Union, Intersect and Except-Nested Queries-Aggregative Operators- Group By and Having Clauses-Null Values-Outer Joins.

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

Outcome: After Completion of the Unit, Student will Be able to:
1. Design the new database.
2. Master the basic concepts and appreciate the applications of database systems.
3. Master the basics of SQL and construct queries using SQL.

Activity: Design a new Database and normalize the data.

Unit-IV: Overview of Storage and Indexing:
Data on External Storage – File Organization and Indexing – Cluster Indexes, Primary and Secondary Indexes – Index data Structures – Hash Based Indexing – Tree base Indexing.

Outcome:
After Completion of the Unit, Student will Be able to:
1. Differentiate different indexing techniques in real time.
2. An ability to use and apply current technical concepts and practices in the core information technologies.
3. Be familiar with a commercial relational database system (Oracle) by writing SQL using the

system.

4. Be familiar with the relational database theory, and be able to write relational algebra expressions for queries

Activity: Create your own data base and connect the front-End and back-End.

Unit-V:

Query processing, Transaction Management, Concurrency Control and Crash recovery Transactions: Acid Properties of Transaction - Transaction States - Schedule: Serial Schedule Concurrent Schedules - Anomalies Associated With Concurrent Schedules (RW, WR - and WW Conflicts) -Serializability – Conflict Serializability - and View Serializability. Introduction to Lock Management-Lock Based Concurrency Control: 2pl-Strict 2pl Concurrency without Locking, Timestamp–Based Concurrency Control – Optimistic Concurrency Control. Introduction to ARIES - The Log - The Write-Ahead Log Protocol Check Pointing

Outcome:

After Completion of the Unit, Student will Be able to:

1. Make us of transactions for new concepts.
2. Understands the properties of transaction management.
3. Master the basics of query evaluation techniques and query optimization.
4. Be familiar with the basic issues of transaction processing and concurrency control

Activity: Perform Transaction on Various Real Time Concepts.

Text Books:

1. Database Management Systems, 3/e Raghuram Krishnan, Johannes Gehrke, TMH
2. Database System Concepts. 6/e Silberschatz, Korth, TMH
3. Database Management System, 6/e RamezElmasri, Shamkant B. Navathe, PEA

Reference Books:

1. Introduction to Database Systems, 8/e C J Date, PEA
2. The Database book principles & practice using Oracle/MySql NarainGehani, University Press.
3. Database Principles Fundamentals of Design Implementation and Management, Corlos Coronel, Steven Morris, Peter Robb, Cengage Learning.

Course Code		L	T	P	Credits
1005172202	SOFTWARE ENGINEERING	3	1	0	3

Course Overview:

Software engineering is an engineering branch associated with development of software product using well-defined scientific principles, methods and procedures. The outcome of software engineering is an efficient and reliable software product. Software project management has wider scope than software engineering process as it involves communication, pre and post delivery support etc.

Course Objectives:

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 modeling languages.
4. To design and develop correct and robust software products.
5. To understand the quality control and how to ensure good quality software.
6. To understand the planning and estimation of software projects.
7. To understand the implementation issues, validation and verification procedures.
8. To understand the maintenance of software

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Apply the appropriate process models for the application development of SDLC	Applying	PO1, PO2, PO3, PO5
CO2	Understand the phases of SDLC from requirement gathering phase to design phase via Analysis Phase	Understanding	PO1, PO2
CO3	Analyzing the strategies for coding and testing phase in Software product development	Analyzing	PO1, PO2, PO3
CO4	Apply the knowledge about estimation and maintenance of software systems and modeling the software project by using CASE tools	Applying	PO1, PO2, PO3, PO5

Unit-I:Software and Software Engineering:

The Nature of Software, The Unique Nature of Web Apps, Software Engineering, Software Process, Software Engineering Practice, Software Myths.

Process Models:

A Generic Process Model like Waterfall Models, Agile Model etc. Process Assessment and Improvement, PrescriptiveProcess Models, Specialized Process Models, The Unified Process, Personal and Team Process Models, Process Terminology, Product and Process.

Outcome: Define and develop a software project from requirement gathering to implementation.

Activity: For a given problem which process model is suitable justify that.

Unit-II: Requirements Analysis and Specification:
Requirements Gathering and Analysis, Software Requirement Specification (SRS), Formal System Specification.

Software Design:
Overview of the Design Process, How to Characterize of a Design?, Cohesion and Coupling, Layered Arrangement of Modules, Approaches to Software Design

Outcome: Define and develop a software project from requirement gathering to implementation.

Activity: Do the Requirement Analysis and Prepare SRS for a project

Unit-III: Function-Oriented Software Design:
Overview of SA/SD Methodology, Structured Analysis, Developing the DFD Model of a System, Structured Design, Detailed Design, Design Review, over view of Object Oriented design.

User Interface Design:
Characteristics of Good User Interface, Basic Concepts, Types of User Interfaces, Fundamentals of Component-based GUI Development, A User Interface Design Methodology.

Coding And Testing:
Coding, Code Review, Software Documentation, Testing, Unit Testing, Black-Box Testing, White-Box Testing, Debugging, Program Analysis Tool, Integration Testing, Testing Object-Oriented Programs, System Testing, Some General Issues Associated with Testing

Outcome: Obtain knowledge about principles and practices of software engineering.

Activity: Draw E-R diagrams, DFD, CFD and structured charts for the project

Unit-IV:Software Reliability And Quality Management:
Software Reliability, Statistical Testing Software Quality, Software Quality Management System, ISO 9000, SEI Capability Maturity Model.

Computer Aided Software Engineering:
Case and its Scope, Case Environment, Case Supportin Software Life Cycle, Other Characteristics of Case Tools, Towards Second Generation CASE Tool, Architecture of a Case Environment

Outcome: Focus on the fundamentals of modelling a software project.

Activity: Design of Test cases based on requirements and design

Unit-V: Software Maintenance:

Software maintenance, Maintenance Process Models, Maintenance Cost, Software Configuration Management, Devops.

Software Reuse:

What can be reused? Why almost No Reuse So Far? Basic Issues in Reuse Approach, Reuse at Organization Level.

Outcome: Obtain knowledge about estimation and maintenance of software systems

Activity: Using COCOMO model estimate effort for a project.

Text Books:

1. Software engineering A practitioner's Approach, Roger S. Pressman, Seventh Edition
2. Mc GrawHill International Edition.
3. Fundamentals of Software Engineering, Rajib Mall, Third Edition, PHI.
4. Software Engineering, Ian Sommerville, Ninth edition, Pearson education

Reference Books:

1. Software Engineering : A Primer, Waman S Jawadekar, Tata McGraw-Hill, 2008
2. Software Engineering, A Precise Approach, Pankaj Jalote, Wiley India, 2010.
3. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press.
4. Software Engineering1: Abstraction and modeling, Diner Bjorner, Springer International edition, 2006.

Course Code		L	T	P	Credits
1005172203	**ADVANCED DATA STRUCTURES**	3	1	0	3

Course Overview:

The purpose of this course is to provide the students with an exploration of advanced data structures (particularly persistent structures) using C. Course reviews main-memory data structures such as hash tables, Graphs and trees. Disk-based structures such as persistent hash tables and dictionaries. Digital Search Structures such as tries.

Course Objectives:

1. Describe and implement a variety of advanced data structures (hash tables, priority queues, balanced search trees, graphs).
2. Analyse the space and time complexity of the algorithms studied in the course.
3. Identify different solutions for a given problem; analyse advantages and disadvantages to different solutions

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	To understand graph representations, Minimum Spanning Trees and traversals	Understanding	PO1, PO2
CO2	Understand dictionaries, hashing mechanism which supports faster retrieval.	Understanding	PO1, PO2
CO3	Implement heaps, queues and their operations, B Trees and B+ Trees	Applying	PO1, PO2, PO3, PPO5
CO4	Illustration of tries which share some properties of table look up, various issues related to the design of file structures	Analyzing	PO1, PO2, PO3, PPO4

Unit-I: Graphs:

The Graph Abstract Data Type, Introduction, Definition, Graph Representation, Elementary Graph Operation, Depth First Search, Breadth First Search, Connected Components, Spanning Trees, Biconnected Components, Minimum Cost Spanning Trees, Kruskal's Algorithm, Prim s Algorithm Sollin's Algorithm, Shortest Paths and Transitive Closure, Single Source/All Destination: Nonnegative Edge Cost, Single Source/All Destination: General Weights, All-Pairs Shortest Path, Transitive Closure.

Outcome: To understand graph representations, Minimum Spanning Trees and traversals.

Activity: Construct a minimum spanning tree for a real time application.

Unit-II: Hashing:

Dictionary ADT, Introduction-Static Hashing- Hash Table- Hash Functions- Secure Hash Function- Overflow Handling- Theoretical Evaluation of Overflow Techniques, Dynamic Hashing- Motivation for Dynamic Hashing -Dynamic Hashing Using Directories- Directory less Dynamic, Hashing,

Outcome: Understand dictionaries, hashing mechanism which supports faster retrieval.

Activity: Implement Dictionary through Hashing with all the operations

Unit-III: Priority queues and efficient Binary search trees:

Priority Queue ADT, Model, Simple Implementation, Binary Heap, Applications of Priority Queues- The Selection Problem Event Simulation Problem, Binomial Queues- Binomial Queue Structure – Binomial Queue Operation- Implementation of Binomial Queues

Binary Search Tree, AVL Tree, Insertion into a AVL Tree, Deletion from a AVL Tree, Red-Black Trees, Definition- Representation of a Red- Black Tree- Searching a Red-Black Tree- Inserting into a Red Black Tree- Deletion from a Red Black Tree, Splay Trees.

Outcome:

1. Comprehension of heaps, queues and their operations
2. Illustration of Balanced trees and their operations

Activity: Construct an AVL tree for a sample data

Unit-IV: Multiway Search Trees:

M-Way Search Trees, Definition and Properties- Searching an M-Way Search Tree, B-Trees, Definition and Properties- Number of Elements in a B-tree- Insertion into B-Tree- Deletion from a B-Tree- B+-Tree Definition- Searching a B+-Tree- Insertion into B+-tree- Deletion from a B+-Tree.

Outcome: Incorporate data structures into the applications such as B Trees and B+ Trees

Unit-V: Digital Search Structures:

Digital Search Trees, Definition- Search, Insert and Delete- Binary tries and Patricia, Binary Tries, Compressed Binary Tries- Patricia, Multiway Tries- Definitions- Searching a Trie- Sampling Strategies- Insertion into a Trie- Deletion from a Trie- Keys with Different Length- Height of a Trie- Space Required and Alternative Node Structure- Prefix Search and Applications- Compressed Tries- Compressed Tries With Skip Fields- Compressed Tries With Labelled Edges- Space Required by a Compressed Tries, Tries and Internet Packet Forwarding, - IP Routing- 1-Bit Tries- Fixed-Stride Tries-Variable-Stride Tries.

Outcome: Illustration of tries which share some properties of table look up, various issues related to the design of file structures

Activity: Construct Tries for the implementation of English Dictionary and Perform Searching of a word in dictionary.

Text Books:

1. Data Structures, a Pseudocode Approach, Richard F Gilberg, Behrouz A Forouzan, Cengage.
2. Fundamentals of DATA STRUCTURES in C: 2nded, , Horowitz , Sahani, Anderson-freed, Universities Press
3. Data structures and Algorithm Analysis in C, 2nd edition, Mark Allen Weiss, Pearson

Reference Books:

1. Web : http://lcm.csa.iisc.ernet.in/dsa/dsa.html
2. http://utubersity.com/?page_id=878
3. http://freevideolectures.com/Course/2519/C-Programming-and-Data-Structures
4. http://freevideolectures.com/Course/2279/Data-Structures-And-Algorithms
5. File Structures :An Object oriented approach with C++, 3rded, Michel J Folk, Greg Riccardi, Bill Zoellick
6. C and Data Structures: A Snap Shot oriented Treatise with Live examples from Science and Engineering, NB Venkateswarlu & EV Prasad, S Chand, 2010.

Course Code		L	T	P	Credits
1005172204	COMPUTER ORGANIZATION	3	1	0	3

Course Overview:

This course is intended to give you a basic understanding of how computers execute programs. Understanding computers means understanding the hardware/software process of how you and the computer work together to have the computer carry out a task. In this course, building will not mean connecting chips and gates. Rather, you will describe the hardware in diagrams, finite-state machines, and hardware simulators.

Course Objectives:

1. To study the basic organization and architecture of digital computers (CPU, memory, I/O, software). Also the Performance measurement of the computer system.
2. To understand various data transfer techniques in digital computer.
3. Be familiar with functional units of processor such as register file and arithmetic logic unit.
4. To understand the stages in instruction set life cycle.

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	To conceptualize the basics of organizational and architectural issues of a digital computer and to perform computer arithmetic operations.	Understanding	PO1, PO2
CO2	To analyze performance issues in processor and can calculate the effective address of an operand by addressing modes.	Analyzing	PO1, PO2, PO3, PPO4
CO3	Ability to design memory organization that uses banks for different word size operations to understand the concept of cache memory techniques	Applying	PO1, PO2, PO3, PPO4, PO5
CO4	Understand the concept of Input / Output organization.	Understanding	PO1, PO2

Unit-I: Introduction to Computers:

Basic of Computer, internal organization of CPU, Functional Units, Software, Basic Operational Concepts, Von Neumann Architecture, Data Representation, Fixed-Point Representation, Floating-Point Representation.

Outcome: To conceptualize the basics of organizational and architectural issues of a digital computer and to perform computer arithmetic operations.

Activity: Outline the characteristics of any computer architecture.

Unit-II: Register Transfer and Micro operations:

Register Transfer Language, Bus and Memory Transfers, Arithmetic, Logic and Shift Micro operations, Arithmetic Logic Unit.

Basic Computer Organization: Instruction Codes, Computer Registers, Computer Instructions, Timing and Control, Instruction Cycle, Memory-Reference Instructions, Input-Output and Interrupt.

Outcome: To understand the concept of Input / Output organization.

Activity: Design a logic circuit for an arithmetic operation

Unit-III: Central Processing Unit:

Introduction, General Register Organization, Stack Organization, Instruction Formats, Addressing Modes with numerical examples, Data Transfer and Manipulation, Program Control, Program Interrupt, Types of interrupts, CISC Characteristics, RISC Characteristics. Introduction to Parallel Processing, Pipelining – General Considerations.

Control Design:

Hardwired & Micro Programmed (Control Unit), Control Memory, Address Sequencing, Conditional and Unconditional Branching, Micro program Example.

Outcome: To analyze performance issues in processor and can calculate the effective address of an operand by addressing modes.

Activity: Interpret any instruction and write various instruction formats.

Unit-IV: Memory Organization:

Memory Hierarchy, Main Memory, Auxiliary Memory, Associative Memory, Cache Memory, Virtual Memory.

Outcome:

Understand the concept of various memory organizations

Activity: Analyse the memory organization of your system and list the system configuration.

Unit-V: Input-Output Organization:

Peripheral Devices, Input-Output Interface, Asynchronous Data Transfer, Modes of Transfer, Priority Interrupt, Direct Memory Access. Multi Processors: Introduction, Characteristics or Multiprocessors, Interconnection Structures, Inter Processor Arbitration.

Outcome: Understand various data transfer mechanisms

Activity: Attend a webinar and write a report on input -output organization

Text Books:

1. Computer System Architecture, M.Moris Mano, 3rd Edition, Pearson/PHI
2. Computer Architecture and Organization, John P. Hayes, 3rd Edition, McGraw Hill.

Reference Books:

1. Computer Organization and Architecture – William Stallings Sixth Edition, Pearson/PHI
2. Structured Computer Organization – Andrew S. Tanenbaum, 4th Edition PHI/Pearson
3. Fundamentals or Computer Organization and Design, - Siva Raama Dandamudi Springer Int. Edition.
4. "Computer Organization and Design: The Hardware/Software Interface" by David A. Patterson and John L. Hennessy

Course Code	FORMAL LANGUAGES AND AUTOMATA	L	T	P	Credits
1005172205	THEORY	3	1	0	3

Course Overview:

The course introduces some fundamental concepts in automata theory and formal languages including grammar, finite automaton, regular expression, formal language, pushdown automaton, and Turing machine. Not only do they form basic models of computation, they are also the foundation of many branches of computer science, e.g. compilers, software engineering, concurrent systems, etc. The properties of these models will be studied and various rigorous techniques for analysing and comparing them will be discussed, by using both formalism and examples.

Course Objectives:

1. Introduce the student to the concepts of Theory of computation in computer science
2. The students should acquire insights into the relationship among formal languages, formal Grammars and automata.

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Employ finite state machines to solve problems in computing	Analyzing	PO1, PO2, PO4
CO2	Classify machines by their power to recognize languages	Understanding	PO1, PO2
CO3	To Design PDA for solving computational Problems	Applying	PO1, PO2, PO3
CO4	To design Turing Machine for arithmetic Operations	Applying	PO1, PO2, PO4

Unit-I:

Computation, Finite State Machine, Components of Finite State Automata, Elements of Finite State System, Mathematical representation of Finite State Machine Formal Language Theory-Symbols, Alphabets and Strings, Operations on Strings, Formal Languages, Operations on Languages, types of formal languages, Finite Automata: Introduction, Deterministic Finite Automata(DFA), Design of DFAs, Non Deterministic Finite Automata(NFA),

Outcome:

1. Classify machines by their power to recognize languages
2. Employ finite state machines to solve problems in computing

Activity: Design a DFA For real world problems

Unit-II:

Non-Deterministic Automata with Є-moves , Design of NFA- Є s, NFA Versus DFA, Equivalent Automata: Equivalent Finite-State Automata, Equivalence of NFA/NFA- and DFA, Equivalence of NFA, with Є moves to NFA, without Є - moves. Minimization/ Optimization of DFA.

Transducers:

Moore Machine, Mealy Machine, Difference between Moore and Mealy Machines, Properties / Equivalence of Moore and Mealy Machines.

Outcome: Explain deterministic and non-deterministic machines

Activity: Develop a transducer for real world problem.

Unit-III:

Regular Expressions and Languages: Regular languages, Regular expressions, Properties of Regular Expressions, Uses of Regular Expressions, Finite Automata and Regular

Expressions: Properties of Regular Sets and Regular Languages, Arden's Theorem, Equivalence of Finite Automata and Regular Expressions, Equivalence of DFA and Regular Expression, Equivalence of NFA and Regular Expression,

Types of Grammar, Ambiguous and Unambiguous Grammars, Noam Chomsky's Classification of Grammar and Finite Automata, Relation between Regular Grammar and Finite Automata.

Outcome: Classify machines by their power to recognize languages Explain the power and the limitations of regular languages and Lexical Analysis

Activity: Construct a minimized DFA for a regular expression

Unit-IV:

Context-Free Grammars and Context-Free Languages: Simplification of Context – Free Grammar: Simplification of Context-Free Grammars, Elimination of Є -Productions, Elimination of Unit Productions, Normal Forms for Context Free Grammars, Chomsky Normal Form, Greibach Normal Form.

PDA: Definition of PDA and Design of PDA, types of PDA

Outcome: Able to identify, understand and construct various parsing techniques for a given context free grammars.

Activity: Demonstrate how a strings are accepted by PDA

Unit-V: Turing Machine:

Introduction, Components of Turing Machine, Description of Turing Machine, Elements of TM, Moves of a TM, Language accepted by a TM, Role of TM's , Design of TM's, TM Extensions and Languages: TM Languages, Undecidable Problem, P and NP Classes of Languages

Outcome: Comprehend the hierarchy of problems arising in the computer science

Activity: Design a Turing machine for arthematic operations.

Text Books:

1. Elements of Theory of Computation, Harry R Lewis, Papadimitriou, PHI
2. Introduction to theory of computation, 2nd ed, Michel sipser, CENGAGE
3. A Text Book on Automata Theory, Nasir S.F.B, P.K. Srimani, Cambridge university Press
4. Introduction to Automata Theory, Formal languages and computation, Shamalendukandar, Pearson

Reference Books:

1. Formal Languages and automata theory, C.K. Nagpal, OXFORD
2. Theory of Computation , aproblem solving approach, kavi Mahesh, Wiley
3. Automata, computability and complexity, Theory and applications, Elaine rich, PEARSON
4. Theory of Computation, Vivekkulkarni, OXFORDAttachments area
5. Introduction to Automata Theory, Languages and Computation, J.E.Hopcroft, R.MotwaniAnd J.D.Ullman, 3rd Edition, Pearson, 2008.

Course Code		L	T	P	Credits
1005172206	**OPERATING SYSTEMS**	3	1	0	3

Course Overview:

This course will introduce the core concepts of operating systems, such as processes and threads, scheduling, synchronization, memory management, file systems, input and output device management and security.

Course Objectives:

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

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Summarize various concepts of Operating Systems	Understanding	PO1, PO2
CO2	Implement and Apply Process Scheduling Algorithms	Applying	PO1, PO2, PO4,PO5
CO3	Illustrate concepts of Paging, Segmentation and Apply Concurrency, Deadlock Mechanisms in real world	Applying	PO1, PO2, PO3
CO4	Analyze the concepts of file systems in operating systems	Analyzing	PO1, PO3, PO12

Unit-I:

Introduction to Operating System Concept: Types of operating systems, operating systems concepts, operating systems services, Introduction to System call, System call types.

Outcome: Define Operating System and describe types of Operating Systems.

Activity: Brainstorming method.

Unit-II:

Process Management – Process concept, The process, Process State Diagram ,Process control block, Process Scheduling- Scheduling Queues, Schedulers, Operations on Processes, Inter process Communication, Threading Issues, Scheduling-Basic Concepts, Scheduling Criteria, Scheduling Algorithms. Case Studies: UNIX, Linux, Windows

Outcome: Define the concept of process and apply process scheduling algorithms.

Activity: Problem solving related to CPU Scheduling algorithms.

Unit-III: Memory Management:

Swapping, Contiguous Memory Allocation, Paging, structure of the Page Table, Segmentation Case Studies: UNIX, Linux, Windows

Virtual Memory Management:
Virtual Memory, Demand Paging, Page-Replacement Algorithms, Thrashing
Outcome: Illustrate the concept of Paging and Segmentation.
Activity: Visualization of concepts using model charts.

Unit-IV: Concurrency:
Process Synchronization, The Critical- Section Problem, Synchronization Hardware, Semaphores, Classic Problems of Synchronization, Monitors, Synchronization Examples Case Studies: UNIX, Linux, Windows
Principles of deadlock:
System Model, Deadlock Characterization, Deadlock Prevention, Detection and Avoidance, Recovery form Deadlock
Outcome: Apply the concept of Concurrency to real world problems
Activity: Role play related to classic problems of synchronization

Unit-V: File system Interface:
The concept of a file, Access Methods, Directory structure, File system mounting, file sharing, protection. File System implementation- File system structure, allocation methods, free-space management Mass-storage structure overview of Mass-storage structure, Disk scheduling, Device drivers, Case Studies: UNIX, Linux, Windows
Outcome: Design and Implement a prototype file systems
Activity: Seminar method

Text Books:
1. Edition 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.
3. Operating Systems-S Halder, Alex A Aravind Pearson Education Second 2016.

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

Course Code		L	T	P	Credits
1005172221	ADVANCED DATA STRUCTURES LAB	0	0	3	2

Course Objectives:

1. To understand heap and various tree structures like AVL, Red-black, B trees.

Programming:

1) To implement operations on Graphs.
   - Vertex insertion
   - Vertex deletion
   - Finding vertex
   - Edge addition and deletion
2) To implement Prim's algorithm to generate a min-cost spanning tree.
3) To implement Krushkal's algorithm to generate a min-cost spanning tree.
4) To implement Dijkstra's algorithm to find shortest path in the graph.
5) Implement Depth first search
6) Implement Breadth first search
7) To implementation of Static Hashing (Use Linear probing for collision resolution)
8) To implement of Huffman coding.
9) To perform various operations i.e., insertions and deletions on AVL trees.
10) To implement of B-tree.
11) To Implement Red Black Trees

Course Outcomes:

1. Implement heap and various tree structures like AVL, Red-black, B Trees.
2. Implement various graph algorithms.

Course Code	DATABASE MANAGEMENT SYSTEMS LAB	L	T	P	Credits
1005172222		0	0	3	2

Objectives:

1. To provide a sound introduction to the discipline of database management as a subject in its own right, rather than as a compendium of techniques and product-specific tools.
2. To familiarize the participant with the nuances of database environments towards an information oriented data-processing oriented framework
3. To give a good formal foundation on the relational model of data
4. To present SQL and procedural interfaces to SQL comprehensively
5. To give an introduction to systematic database design approaches covering conceptual design, logical design and an overview of physical design

List of Experiments:

1. Creation, altering and dropping of tables and inserting rows into a table (use constraints While creating tables) examples using SELECT command.
2. Queries (along with sub Queries) using ANY, ALL, IN, EXISTS, NOTEXISTS, UNION,
3. INTERSET, Constraints.
4. Example:- Select the roll number and name of the student who secured fourth rank in the class.
5. Queries using Aggregate functions (COUNT, SUM, AVG, MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views.
6. Queries using Conversion functions (to_char, to_number and to_date), string functions(Concatenation, lpad, rpad, ltrim, rtrim, lower, upper, initcap, length, substr and instr), date functions(Sysdate, next day, add months, last day, months between, least, greatest, trunc, round, to char, to date)
7. Creation of simple PL/SQL program which includes declaration section, executable section and exception –Handling section (Ex. Student marks can be selected from the table and printed for those who secured first class and an exception can be raised if no records were found).
8. Insert data into student table and use COMMIT, ROLLBACK and SAVEPOINT in PL/SQL block. Develop a program that includes the features NESTED IF, CASE and CASE expression.
9. The program can be extended using the NULLIF and COALESCE functions.
10. Program development using WHILE LOOPS, numeric FOR LOOPS, nested loops using ERROR Handling, BUILT –IN Exceptions, USE defined Exceptions, RAISE-APPLICATIONERROR.
11. Programs development using creation of procedures, passing parameters IN and OUT of PROCEDURES.
12. Program development using creation of stored functions, invoke functions in SQL Statements and write complex functions.
13. Program development using creation of package specification, package bodies, private objects, package variables and cursors and calling stored packages.
14. Develop programs using features parameters in a CURSOR, FOR UPDATE CURSOR, WHERE CURRENT of clause and CURSOR variables.
15. Develop Programs using BEFORE and AFTER Triggers, Row and Statement Triggers and INSTEAD OF Triggers.

Course outcomes:

1. Understand, appreciate and effectively explain the underlying concepts of database technologies
2. Design and implement a database schema for a given problem-domain
3. Normalize a database
4. Populate and query a database using SQL DML/DDL commands.
5. Declare and enforce integrity constraints on a database using a state-of-the-art RDBMS.

Text books:

1. ORACLE PL/SQL by example. Benjamin Rosenzweig, Elena Silvestrova, Pearson Education 3rd Edition
2. ORACLE DATA BASE LOG PL/SQL Programming SCOTT URMAN, Tata Mc-Graw Hill.
3. SQL & PL/SQL for Oracle 10g, Black Book, Dr.P.S. Deshpande.
4. Data Base Management System, Oracle SQL and PL/SQL, Pranabkumar Das Gupta, P Radha Krishna, PHI.

Course Code		L	T	P	Credits
1005172231	INDUSTRIAL VISIT	0	0	0	2

Industrial Visit: The industrial visit shall be carried out in their domain during the summer vacation after the second year second semester. A student has to submit a report which will be evaluated for 100 marks and will be submitted to an internal evaluation committee comprising Head of the Department or his / her nominee and two senior faculty of the department including the industrial visits coordinator/ supervisor. The industrial visit report shall be evaluated at the beginning of third year first semester before the first mid-term exams**.** Industry oriented MOOCs course (including NPTEL/ Coursera) for not less than EIGHT weeks can be considered as equivalent. The list of courses in such case shall be approved by Head of the department concerned. The registered course must not be same as any of the courses listed in the program structure of their regulation till final year. Marks/grades are awarded based on the performance in viva voce or written examination conducted for Coursera courses and online courses other than SWAYAM/NPTEL where there is no end examination.

PROGRAM STRUCTURE

FOR

III-B.Tech

I & II SEMESTERS

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
PROGRAM STRUCTURE

III B.Tech **I Semester**

S.No	Course Code	Course Title	L	T	P	Credits
1	1005173101	Compiler Design	3	1*	0	3
2	1005173102	Python Programming	3	1*	0	3
3	1005173103	Data Mining Techniques	3	1*	0	3
4	1005173104	Unix and Shell Programming	3	1*	0	3
5	1005173105	Design and Analysis of Algorithms	3	1*	0	3
6	1005173121	Data Mining with R Lab	0	0	3	2
7	1005173122	Python Programming Lab	0	0	3	2
8	1005173123	Operating Systems and Compiler Design Lab	0	0	3	2
9	1099172103	Professional Ethics & Human Values	2	0	0	0
Total Credits:						**21**

III B.Tech **II Semester**

S.No	Course Code	Course Title	L	T	P	Credits
1	1005173201	Computer Networks	3	0	0	3
2	1005173202	Web Technologies	3	0	0	3
3	1005173203	Object Oriented Analysis and Design using UML	3	0	0	3
4	1005173204	Artificial Intelligence	3	0	0	3
5	**Open Elective-I**					
	1004173207	Digital image Processing	3	0	0	3
	1005173205	Embedded Systems				
	1004173208	Microprocessors & Microcontrollers-A				
	1003173203	Robotics				
6	**Open Elective-II (CBCS)(MOOCS)**					
	1005173291	*Any available online course approved by department committee at the time of semester commencement)	3	0	0	3
7	1005173221	Computer Networks Lab	0	0	3	2
8	1005173222	Unified Modeling Lab	0	0	3	2
9	1005173223	Web Technologies Lab	0	0	3	2
10	1099173101	IPR & Patents	2	0	0	0
11	1005173241	Industry Oriented Mini Project	0	0	0	2
Total Credits:						**26**

DETAILED SYLLABUS FOR

III-B.Tech

I-SEMESTER

Course Code		L	T	P	Credits
1005173101	COMPILER DESIGN	3	1	0	3

Course Overview:

This course explores the principles, algorithms, and data structures involved in the design and construction of compilers. Topics include finite-state machines, lexical analysis, context-free grammars, push-down parsers, LR and LALR parsers, other parsing techniques, symbol tables, error recovery, and an introduction to intermediate code generation.

Course Objectives:

1. To understand the theory and practice of compiler implementation.
2. To learn finite state machines and lexical scanning.
3. To learn context free grammars, compiler parsing techniques, construction of abstract syntax trees, symbol tables, intermediate machine representations and actual code generation
4. Over a series of four projects, you will create an extremely simple compiler C as a project in this course. The projects are Parsing, Scanning, Semantic Analysis, and Code Generation. The text provides a good deal of the structure and the best solutions for each project will be available for all students in succeeding projects.

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Acquire knowledge in different phases and passes of Compiler	Understanding	PO1
CO2	Understand Parser and its types i.e. Top-down and Bottom-up parsers.	Understanding	PO1, PO2
CO3	Construct LL, SLR, CLR and LALR parse table.	Applying	PO3, PO4
CO4	Syntax directed translation, synthesized and inherited attributes and analyze techniques for code optimization	Analyzing	PO1,PO2, PO3, PO4

Unit-I:

Introduction Language Processing, Structure of a compiler the evaluation of Programming language, The Science of building a Compiler application of Compiler Technology. Programming Language Basics.

Lexical Analysis-: The role of lexical analysis, specification of tokens. Recognitions of tokens, the lexical analyzer generator lexical.

Outcome:

1. To learn the process of translating a modern high-level language to executable code.
2. To draw the flow graph for the intermediate codes.
3. To apply the optimization techniques to have a better code for code generation

4. Able to convert any instruction of a program to convert from source language to target language and should be recognize what happens at each and every phase of a compiler

Activity:
1. List out types and explain roles of different Translators.
2. For a given sample code, show how compiler will processes the given code. Write results for every phases.

Unit-II:

Syntax Analysis: The Role of a parser, Context free Grammars, Writing A grammar, top down parsing, bottom up parsing, Ambiguous grammar, Recursive Descent parser, LL parser, Operator Grammar, Operator precedence parser.

Outcome: To understand the different types of parsing techniques and should be in a position to solve the problem

Activity: Identification whether given grammar contains Ambiguity, Left recursion and Left Factoring. Write a code in C language to implement Recursive Descent parser.

Unit-III:

Bottom up parsing:- Shift reduce parsing, LR parser, More Powerful LR parser (CLR, LALR) Using Ambiguous Grammars, Error Recovery in LR parser

Outcome:

1. To understand Bottom up parsing techniques.
2. To know how to resolve conflicts occur in parsing

Activity: Identification of whether given grammar will supported by LR(0) or not. Identification of type of conflict and by which parsing technique it will be resolved

Unit-IV:

Syntax Directed Transactions Definition, Evolution order of SDTs Application of SDTs. Syntax Directed Translation Schemes. Attribute Grammars, types of attributes.

Intermediated Code: Generation Variants of Syntax trees 3 Address code, Types and Deceleration, Translation of Expressions, Type Checking. Canted Flow Back patching

Outcome:
1. To understand how Intermediate code generation will works.
2. Clear idea about SDTs

Activity: To implement 3 Address code representations.
Applications of SDTs

Unit-V:

Runtime Environments, Stack allocation of space, access to Non Local data on the stack, Heap Management code generation – Issues in design of code generation the target Language Address in the target code Basic blocks and Flow graphs. A Simple Code generation.

Machine Independent Optimization. The principle sources of Optimization peep hole Optimization, Introduction to Data flow Analysis.

Outcome:
1. To understand the importance of code optimization
2. Analyze the program and minimize the code by using optimizing techniques which helps in reducing the no. of instructions in a program and also utilization of registers in an effective way

Activity: Implementation of Data structures for Runtime Environment.
Identification of Issues that are involved in code generation.

Text Books:

1. Compilers, Principles Techniques and Tools.Alfred V Aho, Monical S. Lam, Ravi Sethi Jeffery D. Ullman,2nd edition,pearson,2007
2. Compiler Design K.Muneeswaran, OXFORD
3. Principles of compiler design,2nd edition,Nandhini Prasad,Elsebier.

Reference Books:

1. Compiler Construction, Principles and practice, Kenneth C Louden, CENGAGE
2. Implementations of Compiler, A New approach to Compilers including the algebraic methods, Yunlinsu ,SPRINGER

Course Code		L	T	P	Credits
1005173102	PYTHON PROGRAMMING	3	1	0	3

Course Overview:

This course introduces computer programming using the Python programming language which will help you to master the Programming with Python by introducing the Object Oriented programming concepts, creation of Data Structures, Implementation of Functions, and Visualization libraries. Lastly you will get into design, code, test, and debug Python programming Language Scripts.

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Install Python IDE and run basic Python scripts.	Understand	PO1
CO2	Understand the operators, functions, key Concepts of Object Oriented Programming in python.	Understand	PO1,PO2
CO3	Access Python from various online resources and import packages to the current working environment.	Applying	PO5
CO4	Develop front end GUI using Visualization Libraries and Multithreading techniques.	Analyzing	PO12

Unit-I:

Introduction:

History of Python, Need of Python Programming, 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 and order of evaluations

Outcome:

1. Understand the Introduction of Python IDE.
2. Learn the basics building blocks of python.
3. Write the basic programs in python.
4. Learn the different types of operators in python

Activity: Install Python on PCs or through Mobile applications run basic Python Scripts for a given data.

Unit-II:

Control Flow-:if, if-elif-else, for, while, break, continue, pass

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

Outcome:

1. Understand the syntax of conditional statements in python
2. Understand the syntax of Data Structures in python

Activity: Identify Operators and types in Python. Implement Data Structure concepts by writing Python Scripts.

Unit-III:

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, name spacing.

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

Outcome:

1. Understanding Functions implementation using Python.
2. Learn the scope or life time of variables in a function.
3. Usage of import statement in modules.
4. Create a package, import and install PIP package in python

Activity: Using Functions develop simple scripts in Python Programming.

Unit-IV:

Object Oriented Programming OOP in Python: Classes, 'self variable', Methods, Constructor Method, Inheritance, Overriding Methods, Data hiding.

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

Outcome:

1.Implement the OOP concepts using python
2.Understand the Exception handling in python

Activity: Implement OOP concepts in Writing Python Scripts

Unit-V:

Multithreading: Understanding threads, Forking threads Synchronizing the threads Programming using multithreading

File handling: Python File(doc and csv) Operation Reading config files in python, Writing log files in python, Understanding read functions, Understanding write functions, Manipulating file pointer using seek ,Programming using file operations

Standard Libraries: Introduction to NumPy and Pandas

Outcome: Understand standard Libraries and GUI visualization in Python.

Activity: Write various test cases and implement specific test for a given case study.

Text Books:

1. Python Programming: A Modern Approach, Vamsi Kurama, Pearson
2. Learning Python, Mark Lutz, Orielly
3. Complete Reference to Python – TMH-2018
4. Python Programming - Using Problem Solving Approach, Reema Thereja, Oxford University Press

Reference Books:

1. Think Python, Allen Downey, Green Tea Press
2. Core Python Programming, W.Chun, Pearson.
3. Introduction to Python, Kenneth A. Lambert, Cengage
4. http://nptel.ac.in/courses/117106113/34
5. https://www.python.org/

Course Code		L	T	P	Credits
1005173103	DATA MINING TECHNIQUES	3	1	0	3

Course Overview:
This course includes the design of a Data warehouse system; perform business analysis with OLAP tools. By applying suitable pre-processing and visualization techniques for data analysis. Classification and regression techniques are also included.

Course Objectives:
1. To understand data warehouse concepts, architecture, business analysis and tools
2. To understand data pre-processing and data visualization techniques
3. To study algorithms for finding hidden and interesting patterns in data
4. To understand and apply various classification and clustering techniques using tools.

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Understand why there is a need for data warehouse in addition to traditional operational database systems And Understand why there is a need for data mining and in what ways it is different from traditional statistical techniques.	Understand	PO1, PO2
CO2	Identify components in typical data warehouse architectures.	Analyze	PO1, PO2, PO3
CO3	Design a data warehouse and understand the process required to construct one.	Apply	PO1, PO2, PO3, PO4
CO4	Understand the details of different algorithms made available by popular commercial data mining software and Solve real data mining problems by using the right tools to find interesting patterns.	Create	PO1, PO2, PO3, PO5, PO12

Unit-I: Introduction:
Need of Data Mining, Types of Data, Pattern can be mined, Tools for Mining, Applications of Mining. Major Issues in Data Mining. Data Objects and Attribute types, Basic Statistical Descriptions of Data, Data Visualization, Measuring Data Similarity and Dissimilarity. (Han & Kamber)

Outcome:
1. Understand the importance of data mining and the principles of business intelligence
2. To introduce the concept of data Mining as an important tool for enterprise data management and as a cutting edge technology for building competitive advantage.

Activity: Illustrate how does Data warehouse help in improving the business of an organization.

Unit-II: Data Pre-processing:

Need of Data Preprocessing an overview, Data Cleaning, Data Integration, Data Reduction, Data Transformation and Data Discretization.

Data Warehouse: An Overview: Data Warehouse A Multidimensional Data Model, Data Warehouse Architecture. (Han & Kamber)

Outcome:

1.Organize and prepare the data needed for data mining using preprocessing techniques.

2.Perform exploratory analysis of the data to be used for mining.

Activity:

1. Discuss various steps and approaches for data cleaning.
2. Briefly compare the following concepts. You may use an example to explain your point(s).
   (a) Snowflake schema fact constellation starnet query model
   (b) Enterprise warehouse data mart virtual warehouse.

Unit-III: Classification:

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. Classification:

Alterative Techniques:

Bayes' Theorem, Naïve Bayesian Classification, Bayesian Belief Networks. (Tan & Vipin)

Outcome:

1. To enable students to effectively identify sources of data and process it for data mining.
2. Perform exploratory analysis of the data to be used for mining.

Activity: Implement the Decision Tree Induction algorithm and find the best splitting attribute.

Unit-IV: Association Analysis:

Basic Concepts and Algorithms: Problem Defecation, Frequent Item Set generation, Rule generation, compact representation of frequent item sets, FP-Growth Algorithm. (Tan &Vipin)

Outcome:

1. To make students well versed in all data mining algorithms, methods of evaluation.
2. To impart knowledge of tools used for data mining.

Activity: A database has five transactions. Let min sup = 50% and min con f = 70%.
Find all frequent itemsets using Apriori and FP-growth respectively. Compare the efficiency of the two mining processes.

TID	*items_bought*
T100	{M, O, N, K, E, Y}
T200	{D, O, N, K, E, Y }
T300	{M, A, K, E}
T400	{M, U, C, K, Y}
T500	{C, O, O, K, I ,E}

Unit-V:

Cluster Analysis: Overview on Cluster Analysis, Different Types of Clustering, Different Types of Clusters;

K-means: The Basic K-means, Algorithm, K-means Additional Issues, Bisecting K-means, Strengths and Weaknesses; Agglomerative Hierarchical Clustering: Basic Agglomerative Hierarchical Clustering Algorithms.

DBSCAN: Traditional Density Center-Based Approach, DBSCAN Algorithm, Strengths and Weaknesses. (Tan &Vipin).

Outcome:

1. Define and apply metrics to measure the performance of various data mining algorithms.
2. Compare and analyze the implementation of different clustering algorithms.

Activity:

Generalize each of the following clustering algorithms in terms of the following criteria: (i) shapes of clusters that can be determined; (ii) input parameters that must be specified; and (iii) limitations.

(a) k-means

(c) DBSCAN

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: Introductory and Advanced topics : Dunham, Pearson.
3. Data Warehousing Data Mining & OLAP, Alex Berson, Stephen Smith, TMH.

Course Code		L	T	P	Credits
1005173104	UNIX AND SHELL PROGRAMMING	3	1	0	3

Course Overview:
In this course you learn about UNIX operating system commands and utilities, navigating through the UNIX file systems and to work with files, directories, and permissions, regular expressions to create powerful search strings, also to create advanced shell scripts using shell built-ins and conditionals, and powerful commands used to perform advanced text processing operations.

Course Objectives:
1. To provide knowledge of UNIX Operating System and its File System.
2. To develop the ability to formulate filtration techniques using filters.
3. To provide a comprehensive knowledge of SHELL programming, services and utilities.
4. To develop the ability to learn Distributed processing and multi-tasking.
5. To Create Productive environment for software development using rich set of tools.

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Explain the architecture and features of UNIX Operating System and differentiate it from other Operating Systems	Understand/Remember	PO1
CO2	Demonstrate UNIX commands for file handling and process control	Understand	PO1, PO10
CO3	Build Regular expressions for pattern matching and apply them to various filters for a specific task	Apply	PO, PO2, PO3, PO5
CO4	Analyze a given problem and apply requisite facets of SHELL programming in order to devise a SHELL script to solve the problem	Analyze	PO, PO2, PO4, PO12

Unit-I: Introduction to Unix-Brief History-What is Unix-Unix Components-Using Unix-Commands in Unix-Some Basic Commands-Command Substitution-Giving Multiple Commands.

The File system –The Basics of Files-What's in a File-Directories and File Names-Permissions-INodes-The Directory Hierarchy.

Outcome:

1. Demonstrate basic commands, command substitution, command chaining in UNIX.
2. Explain basics of files, directory hierarchy, permissions, and inodes in UNIX.

Activity: Illustrate how UNIX commands are applicable in real time with an example.

Unit-II: File Attributes and Permissions-The File Command knowing the File Type-The Chmod Command Changing File Permissions-The Chown Command Changing the Owner of a File-The Chgrp Command Changing the Group of a File.

Outcome:

1. Analyze file command to know the file type and chmod command to change file permissions.
2. Demonstrate the chown command and chgrp command to change the owner and group of a file.

Activity: Demonstrate the use of chmod, chown, chgrp commands in real time applications.

Unit-III: Using the Shell-Command Line Structure-Met characters-Creating New Commands-Command Arguments and Parameters-Program Output as Arguments-Shell Variables- -More on I/O Redirection-Looping in Shell Programs. Filters-The Grep Family-Other Filters-The Stream Editor Sed-The AWK Pattern Scanning and processing Language-Good Files and Good Filters.

Outcome:

1. Demonstrate command arguments, shell variables, looping in shell programs.
2. Illustrate filters, the stream editor, the awk pattern scanning and processing language.

Activity: Demonstrate the use of command arguments, shell variables, filters in real time applications.

Unit-IV:

Shell Programming-Shell Variables-The Export Command- The read Command-Positional parameters-The $? Variable knowing the exit Status-More about the Set Command-The Exit Command-The Expr Command: Performing Integer Arithmetic-Real Arithmetic in Shell Programs-The here Document(<<)-The Sleep Command-The Script Command-The Eval Command-The Exec Command.

Outcome:

1. Demonstrate the export, read, set, exit, expr commands.
2. Analyze The here Document(<<)-The Sleep Command-The Script Command-The Eval Command-The Exec Command

Activity: Demonstrate the use of export, exit, expr, sleep, script, eval, exec command in real time applications.

Unit-V:

The Process-The Meaning-Parent and Child Processes-Types of Processes-More about Foreground and Background processes-Internal and External Commands-Process Creation, synchronization and mutual exclusion-The Trap Command-The Stty Command-The Kill Command-Job Control.

Sockets- Socket system calls for connection oriented protocol and connectionless protocol, example- client/server program, socket options.

Activity: Demonstrate the use of types of processes and sockets in real-time applications.

Text Books:

1. The Unix programming Environment by Brain W. Kernighan & Rob Pike, Pearson.
2. Introduction to Unix Shell Programming by M.G.Venkateshmurthy, Pearson.

Reference Books:

1. Unix and shell programming by B.M. Harwani, OXFORD university press.

Course Code	DESIGN AND ANALYSIS OF	L	T	P	Credits
1005173105	ALGORITHMS	3	1	0	3

Course Overview:
This course introduces different techniques to design algorithms using Divide and Conquer, Greedy Approach, Dynamic Programming, Randomized techniques, Multi-Threading, Backtracking and Branch and Bound. It also focuses on how to measure the time and space complexities of algorithms

Course Objectives:
1. Explains how to design algorithms for problems.
2. Discusses on the performance measures of algorithms.
3. Improving perspective of the students while devising algorithms for new problems

Course outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Able to analyze the performance of an algorithm in terms of time and space.	Analysis	PO1, PO2, PO3
CO2	Give an intuition on how to find a solution to large problems by dividing them into smaller sub problems.	Understanding	PO2, PO3
CO3	Identifying which designing technique can be used to solve a particular problem.	Applying	PO1, PO2, PO12
CO4	Knowing how to explore the solution space by using Branch and Bound technique.	Understanding	PO1,PO2

UNIT-I:
Foundations of Algorithm: Algorithm, Algorithm Specification, Recursive Algorithm.
Analysis: Space Complexity and Time Complexity, Asymptotic Notations, Amortized Analysis, Masters Theorem with proof, Sorting in linear time: Radix sort, Counting sort and Bucket sort.
Outcome: Students have a basic understanding on the time and space complexities. Students know how to use Masters Theorem to find the time complexity.
Activity: For a set of basic problems, students have to develop programs and provide complexity analysis.

UNIT-II:
Divide and Conquer: General method, Applications: Binary search, Defective Chessboard. Finding the Maximum and Minimum, Quick sort, Merge sort, Matrix multiplication: Block and

Strassen's matrix multiplication

Outcome: Students should get an idea on what kind of problems a divide-and-conquer can be applied and know how to divide a problem into sub problems to find a solution.

Activity: Students have to develop programs for all the algorithms (problems) that are discussed in the classroom and know the importance of profiling.

UNIT-III:

Greedy method: General method, Applications: Job sequencing with deadlines, knapsack problem, Single source shortest path problem, Optimal Merge Patterns

Probabilistic analysis and Randomized algorithms: The Hiring Problem, Indicator random variables, Randomized Quick Sort, Randomly built binary search trees.

Outcome: Students should know what kind of problems a Greedy Approach can be used and how to choose a greedy strategy to get an optimal solution. Students have basics on how to apply probabilistic analysis for Randomized algorithms.

Activity: Students know how to develop a greedy solution for a problem and able to apply the concept of randomization while developing algorithms for few problems.

UNIT-IV:

Dynamic Programming: General method, Applications: Matrix chain multiplication, 0/1 knapsack problem, All pairs shortest path problem, Travelling salesperson problem, String Editing, Reliability design.

Multithreaded Algorithms: Basics of dynamic multithreading, multithreaded matrix multiplication, multithreaded merge sort.

Outcome: Students should know how to apply principle of optimality to get an optimal solution using Dynamic Programming approach and able to apply multithreading concepts to solve a few simple problems

Activity: Develop programs for a set of problems using Dynamic programing.
Write multithreaded algorithms and able to estimate their speedup and amount of work done

UNIT-V:

Backtracking: General method, Applications: n-queen problem, sum of subsets problem.

Branch and Bound: Control Abstraction for LC-Search, FIFO & LIFO Branch-and-Bound. 15-Puzzle Problem. Introduction to NP-Hard and NP- Completeness.

Outcome:

Students should be able to understand how to explore a solution space by using Backtracking and Branch-and-Bound. Students have to develop an intuition on what kind of problems can be solved with the help of Backtracking or Branch-and-Bound. Students are able to know how a problem can be classified into either P-class or NP-class.

Activity:

Students are able to generate all possible outcomes for a given problem either by using Backtracking or Branch-and-Bound.

Text Books:

1. Fundamentals of Computer Algorithms, Ellis Horowitz, Satraj Sahni and Rajasekharam, Universities Press.
2. Introduction to Algorithms, second edition, T.H.Cormen, C.E.Leiserson, R.L.Rivest and C.Stein, PHI Pvt. Ltd.

3. The Algorithm Design Manual, 2nd edition, Steven S. Skiena, Springer.
4. Design and Analysis of Algorithms, S. Sridhar, OXFORD UNIVERSITY PRESS.
5. Introduction to the Design and Analysis of Algorithms, Anany Levi, PEA

Reference Books:

1. Design and Analysis of Computer Algorithms, First Edition, V. AHO, Pearson
2. Design and Analysis of Algorithms, Parag Himanshu Dave, Himansu Balachandra Dave, Pearson Education.
3. Introduction to Design and Analysis of Algorithms A strategic approach, R.C.T. Lee, S.S.Tseng, R.C.Chang and T.Tsai, Mc GrawHill.
4. Design and Analysis of algorithms, Aho, Ullman and Hopcroft, Pearson education.

Algorithms: Fourth Edition, Robert Sedgewick, Addison-Wesley, 2008

Course Code	Professional Ethics & Human Values	L	T	P	Credits
1099172103		2	0	0	0

Course Overview:
Professional Ethics and Human Values subject provides character oriented education that in stills basic values and ethnic value in one's individual professionalism.

Course Objectives:

1. To encourages students to discover what they consider valuable.
2. To move from discrimination to commitment. It is to create an ability to act on any discrimination in a given situation.
3. To help the students appreciate the essential complementarily between 'VALUES' and 'SKILLS' to ensure sustained happiness and prosperity which are the core aspirations of all human beings.
4. To appreciate the rights of others.
5. Making the students aware and sensitive to value system in real life situations. To help the students to discriminate between ephemeral and eternal values.

Course Outcomes:

	Course Outcome	**Cognitive level as per Bloom's Taxonomy**	**PO**
CO1	Recognize importance of human values, harmony and ethical behavior in real life situations	Understanding	8
CO2	Describe the core values that shape the ethical behaviour of an engineer	Understanding	8
CO3	Recall basics of professional ethics and human values.	Remembering	8
CO4	Listing sustained happiness through identifying the essentials of human values and skills.	Remembering	8
CO5	Describe the practical importance of trust, mutually satisfying human behaviour and enriching interaction with nature	Understanding	8

Unit-I: Human Values
Morals, Values and Ethics – Integrity – Work Ethics – Service Learning – Civic Virtue – Respect for others – Living Peacefully - Caring – Sharing – Honesty –Courage – Value time – Co-operation – Commitment – Empathy – Self-confidence – Spirituality-Character

Outcome:

1. To understand different types of human values.
2. To Identify values linked to the ethical behavior

Activity/Event: Seminar

Unit-II: Engineering Ethics:
The History of Ethics-Purposes for Engineering Ethics-Engineering Ethics-Consensus and

Controversy – Professional and Professionalism –Professional Roles to be played by an Engineer –Self Interest, Customs and Religion-Uses of Ethical Theories-Professional Ethics-Types of Inquiry – Engineering and Ethics- Kohlberg's Theory – Gilligan's Argument –Heinz's Dilemma

Outcome:

1. To enable understanding on engineering ethics
2. To enable knowledge on professional level ethical theories

Activity/Event: Seminar

Unit-III: Engineering as Social Experimentation:

Comparison with Standard Experiments – Knowledge gained –Conscientiousness – Relevant Information – Learning from the Past – Engineers as Managers, Consultants, and Leaders – Accountability – Role of Codes – Codes and Experimental Nature of Engineering.

Outcome: Seminar

1. To Understand the application of ethics in social experimentation
2. To gain knowledge on a engineers as different roles

Activity/Event: Case Analysis

Unit-IV: Engineers' Responsibility for Safety and Risk:

Safety and Risk, Concept of Safety – Types of Risks – Voluntary v/s Involuntary Risk- Short term v/s Long term Consequences- Expected Probability- Reversible Effects- Threshold Levels for Risk- Delayed v/s Immediate Risk- Safety and the Engineer – Designing for Safety – Risk-Benefit Analysis-Accidents.

Outcome: Seminar

1. To understand the challenge for engineers to create safety to risk
2. To enable the knowledge on the risk bearable level

Activity/Event: Seminar

Unit-V: Engineers' Responsibilities and Rights:

Collegiality-Loyalty-Professionalism and Loyalty- Professional Rights –Professional Responsibilities – confidential and proprietary information-Conflict of Interest-Ethical egoism-Collective bargaining-Confidentiality-Acceptance of Bribes/Gifts- when is a Gift and a Bribe-examples of Gifts v/s Bribes-problem solving-interests in other companies- Occupational Crimes-industrial espionage-price fixing-endangering lives- Whistle Blowing-types of whistle blowing-when should it be attempted-preventing whistle blowing. Cross-culture Issues.

Outcome: Seminar

1. To enable understanding on engineers responsibilities
2. To enable knowledge on different types of rights of engineers

Activity/Event: Seminar

Text Books:

1. "Engineering Ethics and Human Values" by M.Govindarajan, S.Natarajan and V.S.SenthilKumar- PHI Learning Pvt. Ltd-2009
2. "Professional Ethics and Morals" by Prof.A.R.Aryasri, Dharanikota Suyodhana- Maruthi Publications
3. "Professional Ethics and Human Values" by A.Alavudeen, R.Kalil Rahman and M.Jayakumaran- Laxmi Publications
4. "Professional Ethics and Human Values" by Prof.D.R.Kiran

Reference Books:

1. "Indian Culture, Values and Professional Ethics" by PSR Murthy-BSP Publication
2. "EthicsinEngineering"byMikeW.MartinandRolandSchinzinger–TataMcGraw-Hill–2003.
3. "Engineering Ethics" by Harris, Pritchard and Rabins, CENGAGE Learning, India Edition, 2009.

Course Code	DATA MINING WITH R LAB	L	T	P	Credits
1005173121		0	0	3	2

Course Overview:

This course focusing about fundamental of R programming, handling and analysis of data, visualization techniques. Determine the most appropriate approach.

Course Objectives:

The student should be made to be:

1. Familiar with the algorithms of data mining,
2. Acquainted with the tools and techniques used for Knowledge Discovery in Databases.
3. Exposed to web mining and text mining.

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Able to installation of R studio and understand working with R	Understanding	PO1
CO2	Exercise the data mining techniques with varied input values for different parameter.	Applying	PO3, PO5
CO3	Understand the data sets and data pre-processing and implement various graphs using R software	Understanding	PO1, PO2
CO4	Demonstrate the working of algorithms for data mining tasks such association rule mining, classification, clustering and regression	Analyzing	PO2, PO3, PO5

List of Experiments:

1. Installation of R studio and working with various Arithmetic and Logical Operations.
2. Working with various data types and data structures in R (vector, array, list, matrix, data frame, factors and tables
3. Working with Control and loop statements illustration with appropriate examples
4. Working with user defined functions and implementation of Quick Sort
5. Generating the various graphs applies use legends for graph (linear, bar, pie, histogram, box plot) and save the graphs in various formats.
6. Categorize all the attributes listed in the table according to the NOIR topology? Apply the applicable central tendency measures to any four attributes taking one attribute from each category. Consider the attribute "peak-rpm" and "city-mpg"? Find which probability distribution(s) they are likely to follow?
   Reference: AUTOMOBILE data with 205 observations
7. Using R software perform the following operations on MASHROOMS data with 8124 observations

- Observe the mushroom data. Divide it in train data and test data.
- Build the Bayesian classification model with the train data.
- Predict the test data using the learned model
- Measure the performance of the classifier.

8. Demonstration of Association rule process on dataset contactlenses using apriori algorithm
9. Demonstration of classification rule process on dataset employee using naïve baye's algorithm
10. Demonstration of clustering rule process on dataset student using simple k-means

Text Books:

1. The Art of R Programming, Norman Matloff, Cengage Learning
2. R for Everyone, Lander, Pearson R Cookbook, PaulTeetor, Oreilly.
3. Introduction to Data Mining: Pang-Ning Tan & Michael Steinbach, Vipin Kumar, Pearson.
4. Data Mining concepts and Techniques, 3/e, Jiawei Han, Michel Kamber, Elsevier.

Course code	PYTHON PROGRAMMING LAB	L	T	P	Credits
1005173122		0	0	3	2

Course Objectives:

1. Write, test, and debug simple Basics of Python programming
2. Implement Python programs with conditionals and loops.
3. Use functions for structuring Python programs.
4. Object Oriented Programming using Python.
5. Represent compound data using Python lists, tuples, dictionaries.
6. Read and write data from/to files in Python
7. Learn GUI Programming and Databases operations in Python

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Understand how to write, test, and debug simple Python programs.	Understanding	PO1
CO2	Determine the Numbers, Math functions, Strings, List, Tuples and Dictionaries in Python and Develop Python programs step-wise by defining functions and calling them.	Applying	PO2, PO4, PO5
CO3	Explain how to design GUI Applications in Python and evaluate different database operations and Express different Decision Making statements and Functions	Understanding	PO1, PO5
CO4	Summarize different File handling operations	Evaluating	PO1, PO2, PO3, PO5

List of Experiments:

Exercise 1- Basics

a) Running instructions in Interactive interpreter and a Python Script

b) Write a program to purposefully raise Indentation Error and Correct it.

Exercise 2- Operations

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

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

Exercise 3- Control Flow

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

b) Using a for loop, write a program that prints out the decimal equivalents of 1/2, 1/3, 1/4, . ., 1/10

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

d) 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

a) 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, ...

b) 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

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

b) 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

a) Write a program combine lists that combines these lists into a dictionary.

b) 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

a) Write a program to print each line of a file in reverse order.

b) Write a program to compute the number of characters, words and lines in a file.

Exercise - 8 Functions

a) 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 centres) <= (sum of their radii) then (they are colliding)

b) Find mean, median, mode for the given set of numbers in a list.

Exercise - 9 Functions - Continued

a) 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 b.

b) Write a function dups to find all duplicates in the list.

c) Write a function unique to find all the unique elements of a list.

Exercise - 10 - Functions - Problem Solving

a) Write a function cumulative product to compute cumulative product of a list of numbers.

b) Write a function reverse to reverse a list. Without using the reverse function.

c) Write function to compute gcd, lcm of two numbers. Each function shouldn't exceed one line.

Exercise 11 - Multi-D Lists

a) Write a program that defines a matrix and prints.

b) Write a program to perform addition of two square matrices.

c) Write a program to perform multiplication of two square matrices.

Exercise - 12 – Modules

a) Install packages requests, flask and explore them. using (pip).
b) Write a script that imports requests and fetch content from the page. Eg.(Wiki).
c) Write a simple script that serves a simple HTTP Response and a simple HTML Page.

Exercise - 13 OOP

a) Class variables and instance variable and illustration of the self-variable
    i) Robot.
    ii) ATM Machine.

Exercise - 14 File access

a) Create a CSV file (roll no, subject1, subject2, subject 3) with 100 rows. All the marks randomly generated raving range (0-100). roll no are having range (1-100)
b) Read the above CSV file having(roll no,subject1, subject2,subject 3) and create new CSV (roll no,subject1, subject2,subject 3, average marks)

Exercise – 15

a) Give some use case scenario and design a system with python program on templates

Text Books:

1. Python Programming: A Modern Approach, Vamsi Kurama, Pearson
2. Learning Python, Mark Lutz, Orielly

Course code		L	T	P	Credits
1005173123	OPERATING SYSTEMS AND COMPILER DESIGN LAB	0	0	3	2

Course Overview:
This course focusing on simulation of various operating systems concepts and implementation of various phases in the compiler design.

Course Objectives:
1. To understand the design aspects of operating system.
2. To study the process management concepts & Techniques.
3. To study the storage management concepts.
4. Understand the working of lex and yacc compiler for debugging of programs
5. Understand and define the role of lexical analyzer, use of regular expression and transition diagrams.
6. Understand and use Context free grammar, and parse tree construction

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Understand operating systems concepts like deadlocks, multi programming	Understanding	PO1
CO2	Implement various concepts like page replacement algorithms, CPU scheduling algorithms	Applying	PO3,PO5
CO3	Implementation of Lexical analyzer	Applying	PO1,PO3,PO5
CO4	Implementation of various parsers	Applying	PO1,PO3,PO5

List of Experiments:

Operating Systems:

1. Simulate the following CPU scheduling algorithms
   a) Round Robin b) SJF c) FCFS d) Priority
2. Multiprogramming-Memory management- Implementation of fork (), wait (), exec() and exit (), System calls
3. Simulate the following
   a) Multiprogramming with a fixed number of tasks (MFT)

b) Multiprogramming with a variable number of tasks (MVT)

4. Simulate Bankers Algorithm for Dead Lock Avoidance
5. Simulate Bankers Algorithm for Dead Lock Prevention.
6. Simulate the following page replacement algorithms.
   a) FIFO b) LRU c) LFU

Compiler Design:

1. Design a lexical analyzer for given language and the lexical analyzer should ignore redundant spaces, tabs and new lines. It should also ignore comments. Although the syntax specification states that identifiers can be arbitrarily long, you may restrict the length to some reasonable value. Simulate the same in C language.
2. Implement the lexical analyzer using JLex, flex or other lexical analyzer generating tools.
3. Write a C program to test whether a given identifier is valid or not.
4. Write a C program for constructing recursive descent parsing.
5. Write a C program to implement operator precedence parsing.
6. Write a C program for constructing of LL(1) parsing

Text Books:

1. Operating Systems – Internals and Design Principles, William Stallings, 7th Edition, Prentice Hall, 2011.
2. Implementations of Compiler, A New approach to Compilers including the algebraic methods, Yunlinsu ,SPRINGER

DETAILED SYLLABUS FOR

III-B.Tech

II-SEMESTER

Course Code		L	T	P	Credits
1005173201	COMPUTER NETWORKS	3	0	0	3

Course Overview:

This course aims at hardware configuration of network and focusing on layer approach and their functionalities, connection establishment, data transfer, protocols, architectures and connection termination process. The detailed study help the student to settle their future in Hardware engineering.

Course Objectives:

1. Building a firm foundation for understanding fundamentals of Data Communications and Computer Networks.
2. Familiarize with the basic terminologies of Computer Networking area.
3. Understand the state of art in Network protocols, Architecture and Applications.
4. Acquire the knowledge of the basic protocols involved in Wired/Wireless communication process.
5. Understand Process of Networking Research, Approach and Analysis

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Identify the different types of network topologies and protocols.	Applying	PO1
CO2	Enumerate the layers of the OSI model and TCP/IP models	Understanding	PO1, PO4
CO3	Familiarity with the basic protocols of computer networks, and how they can be used to assist in network design and implementation and understand routing and congestion control algorithms	Understanding	PO1, PO3, PO4
CO4	Analyze MAC layer protocols and LAN technologies	Analyzing	PO1, PO2, PO4

Unit-I:

Introduction: Network Topologies, **Types of Networks**: WAN, LAN, MAN.

Reference models: The OSI Reference Model, the TCP/IP Reference Model, A Comparison of the OSI and TCP/IP Reference Models.

Physical Layer: Guided Transmission medium.

Multiplexing: Frequency Division multiplexing, Time Division Multiplexing, Code Division Multiplexing.

Outcome: Understand OSI and TCP/IP models.

Activity: Identify Network Topology of your laboratory and sketch it

UNIT-II:

The Data Link Layer – **Design issues**: Services Provided to the Network Layer, Framing, Error Control, Flow Control,

Error Detection: Parity check, Checksum, CRC,

Error Correction: Hamming Code, Linear block codes, FEC.

Elementary Data Link Protocols: A Utopian Simplex Protocol, A Simplex Stop and Wait Protocol for an Error free channel, A Simplex Stop and Wait Protocol for a Noisy Channel,

Sliding Window Protocols: One Bit Sliding Window Protocol, A Protocol Using Go-Back-N, Selective Repeat.

Outcome: Analyze Data Link Layer protocols and design issues.

Activity: Draw a time-line diagram for the sliding window algorithm.

UNIT-III:

The Medium Access Control Sub layer -The Channel Allocation Problem: Static Channel Allocation, Dynamic Channel Allocation,

Multiple Access Protocols: Aloha, pure ALOHA, Slotted ALOHA,**CSMA:**CSMA/CD,CSMA/CA, Collision Free Protocols, Limited Contention Protocols, Wireless LAN Protocols,

Ethernet: Classic Ethernet Physical Layer, Classic ,MAC Sub-layer,

Wireless LAN'S : The 802.11 Architecture and Protocol Stack, The 802.11 Physical Layer,The802.11 MAC Sub-layer Protocol.

Outcome: Analyze MAC layer protocols and LAN technologies.

Activity: With the help of flowchart differentiate between the Ethernet and IP address.

UNIT-IV:

The Network Layer- Design Issues – Store and Forward Packet Switching, Services Provided to the Transport layer, Implementation of Connectionless Service, Implementation of Connection Oriented Service, Comparison of Virtual Circuit and Datagram Networks,

Routing Algorithms: The Optimality principle, Shortest path Algorithm, **Congestion**

Control Algorithms: Approaches to Congestion Control, Traffic Throttling-Load Shedding.

Outcome: 1. Design applications using internet protocols, routing and congestion control algorithms.

Activity: Sketch routing tables in the case of direct and indirect routing.

UNIT-V:

Transport Layer: Transport Services, Connection management, TCP and UDP protocols; ATM AAL Layer Protocol.

Application Layer –The Domain Name System: The DNS Name Space, Resource Records, Name Servers, Electronic Mail: Architecture and Services.

Outcome: Understand how internet works.

Activity: Design resource allocation in TCP using any model.

Text Books:

1. **Computer Networks** (5th Edition) – Andrew S. Tanenbaum.Tanenbaum and David J Wetherall, Computer Networks, 5th Edition, Pearson Edu, 2010.
2. **Computer Networks**: A Top Down Approach, Behrouz A. Forouzan, Firouz Mosharraf, McGraw Hill Education.

Reference Books

1. Computer Networking: A Top-Down Approach (6th Edition) – Kurose and Ross
2. Internetworking with TCP/IP Vol.1: Principles, Protocols, and Architecture (4th Edition) – Douglas E. Comer.

Course Code	WEB TECHNOLOGIES	L	T	P	Credits
1005173202		3	0	0	3

Course Overview:
This course is designed to start a path toward future studies in web development and design. By the end of this course student can able to describe the structure and functionality of the world wide web, create dynamic web pages using a combination of HTML, CSS, and JavaScript, apply essential programming language concepts.

Course Objectives:
1. Design and develop the web application with client server architecture using latest technologies. Students will gain the skills and project-based experience needed for entry into web application and development careers.

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Understand HTML tags to design static web pages	Understand/ Remember	PO1
CO2	Describe the basic concepts of Java Scripts to design dynamic web pages	Understand	PO1, PO10
CO3	Familiarize the concepts of PHP and AJAX	Apply	PO1, PO2, PO3, PO5
CO4	Analyze a given problem and apply requisite appropriate tools for designing dynamic and interactive web applications	Analyze	PO1, PO2, PO4, PO12

Unit-I:

Introduction to HTML : Basic Syntax, Standard HTML Document Structure, Basic Text Markup, Images, Hypertext Links, Lists, Tables, Forms, HTML5

CSS: Levels of Style Sheets, Style Specification Formats, Selector Forms, The Box Model, Conflict Resolution

Outcome: Understand HTML tags to design static web pages

Activity: Design Static website using basic tags and Lists, Tables, Forms.

Unit-II: Java Script:

The Basic of Java script: Objects, Primitives Operations and Expressions, Screen Output and Keyboard Input, Control Statements, Object Creation and Modification, Arrays, Functions,

Constructors, Pattern Matching using Regular Expressions DHTML: Positioning Moving and Changing Elements

Outcome: Describe the basic concepts of Java Scripts to design dynamic web pages

Activity: Validate the login and registration pages.

Unit-III:

Introduction to XML, Document Type Definition, XML namespaces, XML SCHEMA Document object model, XML parsers, XSLT , DOM VS SAX, sample XML Documents

Introduction to AJAX: Integrating PHP and AJAX, Consuming web Services in AJAX,SOAP WSDL, UDDI, Sample Programs

Outcome: Familiarize the concepts of XML and AJAX

Activity: Validate the XML and DOM Files.

Unit-IV:

PHP Programming: Introducing PHP: Creating PHP script, Running PHP script. Working with variables and constants: Using variables, Using constants, Data types, Operators. Controlling program flow: Conditional statements, Control statements, Arrays, functions. Working with forms and Databases such as MySQL

Outcome:

1. Familiarize the concepts of PHP
2. Design the server side dynamic web application

Activity: Design web database application using PHP

Unit-V:

Introduction to PERL, Operators and if statements, Program design and control structures, Arrays, Hashs and File handling, Regular expressions, Subroutines, Retrieving documents from the web with Perl.

Outcome:

1. Analyze a given problem and apply requisite appropriate tools for designing dynamic and interactive web applications
2. Familiarize the concepts of PERL

Activity: Demonstrate the use of PERL in real-time applications.

Text Books:

1. Programming the World Wide Web, Robet W Sebesta, 7ed, Pearson.
2. Web Technologies, Uttam K Roy, Oxford
3. The Web Warrior Guide to Web Programming, Bai, Ekedahl, Farrelll, Gosselin, Zak, Karparhi, Maclntyre, Morrissey, Cengage

Reference Books:

1. Ruby on Rails Up and Running, Lightning fast Web development, Bruce Tate, Curt Hibbs, Oreilly (2006)
2. Web Technologies, HTML< JavaScript, PHP, Java, JSP, XML and AJAX, Black book, Dream Tech.
3. An Introduction to Web Design, Programming, Paul S Wang, Sanda S Katila, Cengage Learning

Course Code	OBJECT ORIENTED ANALYSIS AND DESIGN USING UML	L	T	P	Credits
1005173203		3	0	0	3

Course Overview:

The objective of this course is to use Unified Modeling Language (UML) by understanding the insight knowledge into analyzing and designing Complex software system using Object – Oriented approach and Provide the notations of Unified Modeling Language like Basic Behavioral, Advanced Behavioral and Architectural Modeling.

Course Objectives:

1. To understand how to solve complex problems
2. Analyze and design solutions to problems using object oriented approach
3. Study the notations of Unified Modeling Language

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Build solutions to the complex problems using object oriented approach	Creating	PO1, PO3
CO2	Identify classes and responsibilities of the problem domain	Understand	PO1, PO2
CO3	Apply UML tools for various case studies	Apply	PO4, PO5, PO12
CO4	Represent classes, objects, responsibilities and states using UML notations.	Remember and Understand	PO1, PO2

Unit-I: Introduction

The Structure of Complex systems, The Inherent Complexity of Software, Attributes of Complex System, Organized and Disorganized Complexity, Bringing Order to Chaos, Designing Complex Systems, Evolution of Object Model, Foundation of Object Model, Elements of Object Model, Applying the Object Model.

Outcome: Build solutions to the complex problems using object oriented approach

Activity: Design an object model for a real time system

Unit-II: Classes and Objects

Nature of object, Relationships among objects, Nature of a Class, Relationship among Classes, Interplay of Classes and Objects, Identifying Classes and Objects, Importance of Proper Classification, Identifying Classes and Objects, Key abstractions and Mechanisms.

Outcome: Identify classes and objects along with responsibilities of the problem domain

Activity: Identify classes and objects involved in a case study

Unit-III: Introduction to UML

Why we model, Conceptual model of UML, Architecture, Classes, Relationships, Common Mechanisms, Class diagrams, Object diagrams.

Basic Behavioural Modelling: Interactions, Interaction diagrams, Use cases, Use case Diagrams, Activity Diagrams.

Outcome: To Design Behavioral diagrams like Use Case and Interaction diagrams for an application

Activity: Identify use cases for a real time application and Draw Interaction diagrams, use case diagram and activity diagram for the same application.

Unit-IV: Advanced Behavioural Modelling

Events and signals, state machines, processes and Threads, time and space, state chart diagrams.

Outcome: To model state chart diagrams for an application

Activity: Identify States, events and Draw the state chart diagram for the application you worked in unit 3

Unit-V: Architectural Modeling

Component, Deployment, Component diagrams and Deployment diagrams.

Outcome: To model Component and Deployment diagrams for an application

Activity: Identify components, nodes and draw Component and Deployment diagram for the application you worked from unit 3 onwards.

Text Books:

1. "Object- Oriented Analysis And Design with Applications", Grady BOOCH, Robert Maksimchuk, Michael W. ENGLE, Bobbi J. Young, Jim Conallen, Kellia Houston, 3rd edition, 2013, PEARSON.
2. "The Unified Modeling Language User Guide", Grady Booch, James Rumbaugh, Ivar Jacobson, 12th Impression, 2012, PEARSON.

Reference Books:

1. Simon Bennett, Steve Mc Robb and Ray Farmer, "Object Oriented Systems Analysis and Design Using UML", Fourth Edition, Mc-Graw Hill Education, 2010.
2. Martin Fowler, "UML Distilled: A Brief Guide to the Standard Object Modeling Language", Third edition, Addison Wesley, 2003.

Course Code		L	T	P	Credits
1005173204	ARTIFICIAL INTELLIGENCE	3	0	0	3

Course Overview:

Artificial intelligence (AI) is a research field that studies how to realize the intelligent human behaviors on a computer. The ultimate goal of AI is to make a computer that can learn, plan, and solve problems autonomously. Although AI has been studied for more than half a century, we still cannot make a computer that is as intelligent as a human in all aspects.

Course Objectives:

The objectives of the course are:

1. Having a basic Knowledge about intelligent systems and Problem solving methods
2. Applying different search techniques for solving different problems
3. Understanding the knowledge representation techniques and uncertainty in data
4. Developing a Rule Based Expert Systems.

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Identify Methods in AI that may be suited to solving a given problem and Game Playing	Understanding	PO1,PO2
CO2	Analyze the basic issues of different types of knowledge representation techniques to build intelligent system	Analyze	PO2,PO3
CO3	Build Expert systems for real time applications	Create	PO2,PO3, PO5,PO7
CO4	Determination of uncertainty of data using different probability approaches for real time applications	Analyze	PO1,PO2,PO4

Unit-I:

Introduction to artificial intelligence: Introduction, history, intelligent systems, foundations of AI, Applications of AI, current trends in AI

Problem solving: Definition, characteristics of problem, types of Problem solving techniques, General Problem Solver (GPS), Water Jug Problem, Missionaries and Cannibals Problem.

Outcome: Understand need of AI and its applications

Activity: Write a Report on the use of artificial intelligence in every day

Unit-II:

Search Techniques: State Space Search**,** Definition, Examples, Exhaustive search techniques: BFS, DFS, IDDFS, Heuristic search techniques: Uniform Cost Search, Best First Search, A* algorithm & Constraint satisfaction Problem

Game playing: Introduction about game playing, Mini-Max Algorithm, Alpha-Beta pruning algorithm.

Outcome: Identify Methods in AI that may be suited to solving a given problem and Game Playing

Activity: Identify Methods in AI used to solve the tricky puzzles

Unit-III:

Logic concepts: Introduction, propositional logic, natural deduction system, axiomatic system, semantic tableau system in propositional logic, resolution refutation in propositional logic,

Predicate logic: Introduction, PNX Normal form, Resolution in Predicate Logic

Outcome: Able to understand the uses of logic concepts and predicate logic

Activity: Convert Natural Language Sentences To Predicate Logic

Unit-IV:

Knowledge representation: Introduction, approaches to knowledge representation, knowledge representation using semantic network, knowledge representation using frames.

Uncertainty measure: Introduction to probability theory, Bayesian belief networks, Certainty factor theory, Dempster-Shafer theory

Outcome:

1. Analyze the basic issues of different types of knowledge representation techniques to build intelligent system
2. Determination of uncertainty of data using different probability approaches for real time applications

Activity: Consider any data set Classify the test instances using BBN

Unit-V:

Expert system and applications: Introduction phases in building expert systems, expert system versus traditional systems, rule-based expert systems, application of expert systems, Black board Systems, TMS.

Fuzzy stand fuzzy logic: Introduction, fuzzy sets, fuzzy set operations, types of membership functions, multi valued logic, fuzzy logic, linguistic variables and hedges

Outcome: Build Expert systems for real time applications

Activity: Explain the use of fuzzy logic in the real time scenario

Text Books:

1. Artificial Intelligence- Saroj Kaushik, CENGAGE Learning,
2. Artificial intelligence, A modern Approach , 2nd ed, Stuart Russel, Peter Norvig, PEA
3. Artificial Intelligence- Rich, Kevin Knight, Shiv Shankar B Nair, 3rd ed, TMH
4. Introduction to Artificial Intelligence, Patterson, PHI

Course Code	DIGITAL IMAGE PROCESSING	L	T	P	Credits
1004173207		3	0	0	3

Course Overview:

Digital image processing including visual perception, image formation, spatial transformations, image enhancement, color image representation and processing, edge detection, image segmentation.

Course Objectives:

1. To introduce fundamentals of Image Processing.
2. To expose various intensity transformations in spatial and frequency domains.
3. To impart concepts various coding techniques for image compression.
4. To dissimilate various segmentation techniques for images.
5. To introduce the concepts of colour image segmentation.

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Analyze various types of images mathematically	Analyze	PO2, PO4, PO5
CO2	Compare image enhancement methods in spatial and frequency domains	Understand	PO2
CO3	Demonstrate various segmentation algorithms for given image	Apply	PO2
CO4	Justify different techniques for image compression	Analyze	PO2, PO4, PO5

Unit-I:

Digital Image Fundamentals-Elements of visual perception, image sensing and acquisition, image sampling and quantization, basic relationships between pixels – neighbourhood, adjacency, connectivity, distance measures.

Outcome:

1. Explain mathematical models of various types of images
2. Define image processing parameters such as adjacency and distance measures

Activity/Event: Seminar

Unit-II:

Image Enhancements and Filtering- Gray level transformations, histogram equalization and specifications, pixel-domain smoothing filters – linear and order-statistics, pixel-domain sharpening filters – first and second derivative, two-dimensional DFT and its inverse, frequency domain filters – low-pass and high-pass.

Outcome:

1. Compare image enhancement methods in spatial and frequency domains
2. Apply frequency Domain filtering techniques for image enhancement

Activity/Event: Executing programs using MATLAB software.

Unit-III:

Image Segmentation, Detection of discontinuities, edge linking and boundary detection, thresholding – global and adaptive, region-based segmentation.

Outcome:

1. Describe various Image segmentation techniques.
2. Illustrate detection of discontinuities in an image .

Activity/Event: Executing programs using MATLAB software.

Unit-IV:

Image Compression, -Redundancy, inter-pixel and psycho-visual; Loss less compression – predictive, entropy; Lossy compression- predictive and transform coding; Discrete Cosine Transform; Still image compression standards – JPEG and JPEG-2000.

Outcome:

1. Describe various transform techniques for lossy compression
2. Apply various coding techniques for lossless compression

Activity/Event: Executing programs using MATLAB software.

Unit-V:

Color Image Processing-Color models–RGB, YUV, HSI; Color transformations–formulation, color complements, color slicing, tone and color corrections; Color image smoothing and sharpening; Color Segmentation.

Outcome:

1. Describe various color models for color image processing
2. Apply various techniques for color image smoothing, sharpening and segmentation

Activity/Event: Executing programs using MATLAB software.

Text Books:

1. R.C. Gonzalez and R.E. Woods, Digital Image Processing, Second Edition, Pearson Education, 2008.
2. Anil Kumar Jain, Fundamentals of Digital Image Processing, Prentice Hall of India, 2nd edition 2004.

Reference Books:

1. Rafael C. Gonzalez, Richard E woods and Steven L. Eddins, “Digital Image processing using MATLAB”, Tata McGraw Hill, 2010.
2. S Jayaraman, S Esakkirajan, T Veerakumar, “Digital Image processing”, Tata McGraw Hill.

Course Code	EMBEDDED SYSTEMS	L	T	P	Credits
1005173205		3	0	0	3

Course Overview:

The objective of this course is to equip the students with the basic concepts of embedded system, It is intended for Designing, Implementation and Test of embedded applications. It provides RTOS concepts for coding the embedded system software routines. It tells what makes a system a real-time system and describes characteristics of it.

Course Objectives:

The objectives of the course are:

1. To have knowledge about the basic working of a microcontroller system and its programming in assembly language.
2. To provide experience to integrate hardware and software for microcontroller applications systems.

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Categorize embedded systems and Summarize 8051 microcontroller architecture	Understand/ Remember	PO1
CO2	Identify the unique characteristics of real-time systems	Understand	PO1, PO2
CO3	Apply synchronization tools in various real time scenarios	Apply	PO4, PO5, PO12
CO4	Define the unique design problems and challenges of real-time systems	Apply	PO2, PO3, PO4, PO5, PO12

Unit-I:

Introduction to Embedded systems: What is an embedded system Vs. General computing system, history, classification, major application areas, and purpose of embedded systems. Core of embedded system, memory, sensors and actuators, communication interface, embedded firmware, other system components, PCB and passive components.

Outcome: Understand the basics of an embedded system

Activity: List out various embedded system relevant components

Unit-II:

8-bit microcontrollers architecture: Characteristics, quality attributes application specific, domain specific embedded systems. Factors to be considered in selecting a controller, 8051 architecture

Outcome: Understand the INTEL 8051 architecture and instruction set.

Activity: 8051 interrupts

Unit-III:

Introduction to Real — Time Operating Systems : and Scheduling, Operating basics, types, RTOS, tasks, process and threads, multiprocessing and multitasking, types of multitasking, non preemptive, preemptive scheduling.

Task communication of RTOS, Shared memory, pipes, memory mapped objects, message passing, message queue, mailbox, signaling, RPC and sockets, task communication/synchronization issues, racing, deadlock, live lock, the dining philosopher's problem.

Outcome:

1. Identify the unique characteristics of real-time systems
2. Explain the general structure of a real-time system

Activity: Identify the unique characteristics of real-time systems

Unit-IV:

The producer-consumer problem, Reader writers problem, Priority Inversion, Priority ceiling, Task Synchronization techniques, busy waiting, sleep and wakery, semaphore, mutex, critical section objects, events, device, device drivers, how to clause an RTOS, Integration and testing of embedded hardware and fire ware.

Outcome: Able to describe synchronization techniques

Activity: Explain about process synchronization for any operating system

Unit-V:

Simulators, emulators, Debuggers, Embedded Product Development life cycle (EDLC), Trends in embedded Industry, Introduction to ARM family of processor.

Outcome:

Understand Embedded Product Development life cycle (EDLC)

Activity: Explain any practical application of embedded system

Text Books:

1. Introduction to embedded systems Shibu. K.V, TMH, 2009.
2. Embedded Systems, Rajkamal, TMH, 2009.

Reference Books:

1. Ayala &Gadre: The 8051 Microcontroller & Embedded Systems using Assembly and C, CENGAGE
2. Embedded Software Primer, David Simon, Pearson.
3. The 8051 Microcontroller and Embedded Systems, Mazidi, Mazidi, Pearson,.
4. Embedded System Design, Frank Vahid, Tony Givargis, John Wiley.

Course Code		L	T	P	Credits
1004173208	MICROPROCESSORS & MICROCONTROLLERS-A	3	0	0	3

Course Overview:

This Course brings out the basic hardware design concept of Intel 16 bit microprocessor 8086 and Intel 8 bit 8051 Microcontroller. It introduces the assembly language programming with 8086 and 8051 processors. I/O and memory interfacing is studied both with 8086 and 8051 Microcontrollers. Finally students are introduced to use the usage of these chips for industrial automation.

Course Objectives:

1. To understand the architecture, addressing modes, Instruction set of Intel 8086 Microprocessor and Intel 8051microcontroller.
2. To apply the instruction set in solving simple problems and create small assembly language programs.
3. Understand the hardware concepts of basic peripheral Chips and evaluate the interfacing requirements of the basic peripheral chips.

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Understand the concepts of architecture, memory organization of Intel 8086 microprocessor and Intel 8051 microcontrollers.	Understand	PO1
CO2	Understand the concepts of addressing modes, instruction set of Intel 8086 microprocessor and Intel 8051 microcontrollers.	Understand	PO2
CO3	Write assembly language programs for simple problem statements	Apply, Analyze	PO2, PO4, PO5
CO4	Design an interface between peripheral chips & processors and write programs for data transfer	Analyze	PO2, PO4, PO5

Unit-I:

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

Outcome: Student understand the internal block diagram of 16 bit 8086 chip and it's functions. Further student is introduced to the concept of interconnecting the 8086 microprocessor with memory and peripheral chips to make a system. Student Understands the timing and sequence of control signals between processor and other memory and peripheral chips for data transfer between them.

Activity/Event: Draw diagrams for the internal architecture, pin diagram and signal description, minimum mode system and maximum mode system, Timing diagram for bus signals during read and write operation.

Unit-II:

8086 Programming: Program development steps, addressing modes, instructions, assembler directives, interrupts and interrupt responses, writing simple programs with an assembler.

Outcome: Student understands the operations of instruction set and the operand addressing modes. Able to explain the concept of interrupt and processor response for the interrupt. Able to explain how to use interrupt mechanisms to execute some functions. Understand the facilities available for program development and able to make assembly language programs.

Activity/Event: Write assembly language programs with procedures and interrupt procedures for solving a given problem following program development steps.

Unit-III:

8086 Interfacing: Semiconductor memories interfacing(RAM,ROM), Intel 8259 programmable interrupt controller, Intel 8257 DMA controller, Intel 8255 programmable peripheral interface, interfacing of stepper motor, DAC, ADC to the 8086 microprocessor.

Outcome: Student explains the techniques for interfacing semiconductor memories(RAM and ROM) with system bus. Understand the functions of internal blocks and registers available in peripheral chips like 8259, 8255, 8257 and the interfacing of these chips to system bus.

Activity/Event: Draw interface circuits of the peripheral chips 8259, 8255, 8257 to system bus and peripheral devices to effect data transfer; draw the control words and internal architecture of above peripheral devices.

Unit-IV:

8051 Microcontroller:

Architecture, hardware concepts, input/output ports and circuits, external memory, counters/timers, serial data input/output, interrupts.

Outcome: Understand the internal architectural blocks and their functions including timer/counter operation and serial port operation.

Activity/Event: Draw architectural diagram and pin diagram of 8051.

Unit-V: 8051 Programming & Interfacing:

Assembly language programming: Instructions, addressing modes, simple Programs.

Interfacing: keyboard, displays (LED, 7-segment display unit), A/D and D/A converters, Stepper motor interfacing.

Outcome: Understand the data addressing modes available, operation of instructions and write assembly language programs for simple programs. Able to interface keyboards, display devices like LED, 7-segment display, LCD, stepper motor and A/D and D/A converters.

Activity/Event: Execution of simple programs on keil software

Text Books:

1. Advanced Microprocessors and Peripherals, K Bhurchandi, A. K. Ray, Tata McGraw Hill Education Private Limited, 3rd Edition, reprint 2013.
2. The 8051 Microcontroller and Embedded Systems Using Assembly and C, Muhammad Ali Mazidi, Janice Gillispe Mazidi, Rolin D. McKinlay, PHI, 2nd edition, reprint 2013.

Reference Books:

1. The 8051 Microcontroller & Embedded Systems Using Assembly and C KennethJ.Ayala, Dhananjay V. Gadre, Cengage Learninbg, India Edition.
2. Microprocessors and Microcontrollers by N. Senthil Kumar, M.Saravanan and S. Jeevananthan, Oxford University Press, Seventh Impression 2013

Course Code		L	T	P	Credits
1003173203	ROBOTICS	3	0	0	3

Course Overview:

The course is focused on robots in industrial automation with kinematic and dynamic analysis of manipulators. It also explains the need for automation in industry by discussing the concepts of robot actuators and feedback mechanisms with applications.

Course Objectives:

1. To give students practice in applying their knowledge of mathematics, science, and Engineering and to expand this knowledge into the vast area of robotics.
2. The students will be exposed to the concepts of robot kinematics, Dynamics, Trajectory planning.
3. Mathematical approach to explain how the robotic arm motion can be described.
4. The students will understand the functioning of sensors and actuators.

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Identify various robot configuration and components.	Understanding	PO1, PO5, PO6, PO7, PO8
CO2	Select appropriate actuators and sensors for a robot based on specific application.	Applying	PO2, PO3, PO4, PO6, PO11
CO3	Carry out kinematic and dynamic analysis for simple serial kinematic chains.	Analyzing	PO2, PO3, PO4, PO7, PO12
CO4	Perform trajectory planning for a manipulator by avoiding obstacles.	Evaluating	PO1, PO2, PO3, PO4, PO5, PO8, PO9, PO10, PO11, PO12

Unit-I:

Introduction: Automation and Robotics, – An over view of Robotics–Classification by coordinate system and control system. COMPONENTS OF THE

Industrial Robot: Common types of arms. Components, Architecture, number of degrees of freedom – Requirements and challenges of end effectors, determination of the end effectors.

Outcome: 1.Types of robots and arms.
2. Various classification of robots

Activity: Demo on mechatronics system in mechatronics lab.

Unit-II:

Motion Analysis: Homogeneous transformations as applicable to rotation and translation – problems. MANIPULATOR KINEMATICS: Specifications of matrices, D-H notation joint coordinates and world coordinates Forward and inverse kinematics – problems.

Outcome: 1.Transformations of robot arms.
2. Kinematics of manipulators

Activity: Demo on transformations using simulation mode in cad/cam lab.

Unit-III:

Differential transformation and manipulators, Jacobians – problems Dynamics: Lagrange – Euler and Newton – Euler formulations – Problems.

Outcome: Manipulator dynamics using jacobians.

Activity: Video representing of kinematic and dynamic analysis with simulation.

Unit-IV:

General Considerations in path description and generation. Trajectory planning and avoidance of obstacles, path planning, Skew motion, joint integrated motion –straight line motion – Robot programming, languages and software packages-description of paths with a robot programming language.

Outcome: 1.Trajectory planning with via points.
2. Determination of displacement, velocity and acceleration for a curve.

Activity: Demo of linear and circular interpolation with algorithms.

Unit-V:

Robot Actuators and Feed Back Components: Actuators: Pneumatic,Hydraulic Electrical actuators–comparison. Electric & stepper motors. Feedback components: position sensors – potentiometers, resolvers, encoders – Velocity sensors.

Robot Applications in Manufacturing: Material Transfer - Material handling, loading and unloading- Processing - spot and continuous arc welding & spray painting - Assembly and Inspection.

Outcome: 1. Difference between various actuating systems.
2. Planning for various manufacturing methods.

Activity: Demo of Electric, pneumatic and hydraulic system.

Text Books:

1. Industrial Robotics / Grover M P, M Weiss, R N Nagel, N G Odrey, Ashish Dutta/ Pearson Edu.

2. Robotics and Control / Mittal R K & Nagrath I J / TMH.

Reference Books:

1. Robotics / Fu K S/ McGraw Hill.

2. Robotic Engineering / Richard D. Klafter, Prentice Hall.

3. Robot Analysis and Intelligence / Asada and Slow time / Wiley Inter Science.

4. Introduction to Robotics / John J Craig / Pearson Edu.

Course Code		L	T	P	Credits
1099173101	IPR & PATENTS	2	0	0	0

Course Overview:

This Course provides the students with a wide perspective and in depth knowledge in intellectual property to enable them to get solid grounding in the legislative framework, practice and procedure of the intellectual property protected through patents, trademarks, copyrights, designs and geographical indications.

Course Objectives:

1. To know the importance of Intellectual property rights, which plays a vital role in advanced Technical and Scientific disciplines.
2. Imparting IPR protections and regulations for further advancement, so that the students can familiarize with the latest developments.

Course Outcomes:

1. The students once they complete their academic projects, shall get an adequate knowledge on patent and copyright for their innovative research works
2. During their research career, information in patent documents provide useful insight on novelty of their idea from state-of-the art search.
3. This provide further way for developing their idea or innovations Pave the way for the students to catch up Intellectual Property(IP) as an career option
4. R&D IP Counsel , Government Jobs – Patent Examiner , Private Jobs , Patent agent and Trademark agent and Entrepreneur

Unit-I:

Introduction to Intellectual Property Law – Evolutionary past – Types of Intellectual Property - Innovations and Inventions of Trade related Intellectual Property Rights – Agencies Responsible for Intellectual Property Registration –Geographical indications- Regulatory – Over use or Misuse of Intellectual Property Rights - Compliance and Liability Issues- India`s New National IP Policy, 2016 – Govt. of India step towards promoting IPR – Govt. Schemes in IPR – Career Opportunities in IP

Unit-II:

Introduction to Copyrights – Principles of Copyright – Subject Matters of Copyright – Rights Afforded by Copyright Law –Copyright Ownership – Transfer and Duration – Right to Prepare Derivative Works –Rights of Distribution – Rights of performers – Copyright Formalities and Registration– Limitations – Infringement of Copyright – International Copyright Law - Semiconductor Chip Protection Act- Fair use and Fair Dealing concepts.

Unit-III:

Introduction to Patent Law – Rights and Limitations – Rights under Patent Law – Patent Requirements – Ownership and Transfer – Patent Application Process and Granting of Patent – Patent Infringement and Litigation – International Patent Law – Double Patenting – Patent Searching – Patent Cooperation Treaty – New developments in Patent Law - Invention Developers and Promoters- Non patentable inventions

Unit-IV:

Introduction to Trade Mark – Trade Mark Registration Process – Post registration procedures – Trade Mark maintenance – Transfer of rights –Emblem Act– Inter parties Proceedings – Infringement – Concept of distinctiveness -Dilution of Ownership of Trade Mark – Likelihood of confusion – Trade Mark claims – Trade Marks Litigation – International Trade Mark Law.

Unit-V:

Cyber Law and Cyber Crime

Introduction to Trade Secrets – Maintaining Trade Secret – Physical Security – Employee Access Limitation – Employee Confidentiality Agreement – Trade Secret Law – Unfair Competition – Trade Secret Litigation – Breach of Contract – Applying State Law- Plant Variety Protection and Farmer's Right- Introduction to Cyber Law – Information Technology Act - Cyber Crime and E-commerce – Data Security – Confidentiality – Privacy - International aspects of Computer and Online Crime

Reference Books:

1. "Law Relating to Intellectual Property Rights" by V K Ahuja
2. "Intellectual Property Rights" by Neeraj Pandey and Khushdeep Dharni
3. "Intellectual Property Rights: Text and Cases" by R Radhakrishnan and S Balasubramanian
4. "Intellectual Property Rights-Infringement And Remedies" by Ananth Padmanabhan
5. "Intellectual Property Rights (IPRs): TRIPS Agreement and Indian Laws" by E T Lokganathan
6. .B.L.Wadehra; Law Relating to Patents, Trade Marks, Copyright, Designs & Geographical Indications Universal law Publishing Pvt. Ltd., India 2000
7. P. Narayanan; Law of Copyright and Industrial Designs; Eastern law House, Delhi ,2010
8. Lionel Bently & Brad Sherman, Intellectual Property Law, Oxford. P. Narayanan, Intellectual Property Law, Eastern Law House

Course Code		L	T	P	Credits
1005173221	COMPUTER NETWORKS LAB	0	0	3	2

Course Objectives:

This course aims at hardware configuration of network and focusing on layer approach and their functionalities, real time implementation of various protocols

Course Outcomes:

1. To Understand the functionalities of various layers of OSI model
2. To understand how to use TCP and UDP based sockets and their differences

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Explain how to Communicate between two desktop computers.	Understanding	PO1
CO2	Implement the different protocols	Applying	PO1, PO2, PO4
CO3	Implement and compare the various routing algorithms	Applying and Analyzing	PO1, PO2, PO4
CO4	Demonstrate Program using sockets	Understanding	PO1

Prerequisites

Knowledge of C Programming, Basic commands of UNIX.

List of Programs

1. Implement the data link layer framing methods such as character, character stuffing and bit stuffing.
2. Implementation of checksum.
3. Implement on a data set of characters the three CRC polynomials – CRC 12, CRC 16 and CRC CCIP .
4. Implement Dijkstra 's algorithm to compute the Shortest path thru a graph.
5. Take an example subnet graph with weights indicating delay between nodes. Now obtain Routing table at each node using distance vector routing algorithm
6. Take an example subnet of hosts . Obtain broadcast tree for it.
7. Implementation of Connection oriented concurrent service (TCP).
8. Implementation of Connectionless Iterative time service (UDP).
9. Implementation of FTP.
10. Implementation of HTTP.

Course Code		L	T	P	Credits
1005173222	UNIFIED MODELING LAB	0	0	3	2

Course Objectives:

- Construct UML diagrams for static view and dynamic view of the system.

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Understand the various Case studies	Understanding	PO1, PO2
CO2	Model Static View of a System using Use case diagram	Applying	PO1, PO2, PO3, PO4, PO5
CO3	Model Dynamic View of a System using UML diagrams	Applying	PO1, PO2, PO3, PO4, PO5
CO4	Represent user and programmatic interactions using UML	Applying	PO1, PO2, PO3, PO4, PO5

List of Experiments:

Take three case studies:

1. **Customer Support System (**in the Object-Oriented Analysis & Design with the Unified Process by Satzinger, Jackson & Burd Cengage Learning)
2. **Point-Of-Sale Terminal (** Larman textbook**)**
3. **Library Management System (**UML toolkit**)**

Week 1:
Familiarization with Rational Rose or Umbrello For each case study:
Week 2, 3 & 4: For each case study:

a) Identify and analyze events

b) Identify Use cases

c) Develop event table

d) Identify & analyze domain classes

e) Represent use cases and a domain class diagram using Rational Rose

f) Develop CRUD matrix to represent relationships between use cases and problem domain classes

Week 5 & 6: For each case study:
a) Develop Use case diagrams
b) Develop elaborate Use case descriptions & scenarios
c) Develop prototypes (without functionality)
d) Develop system sequence diagrams

Week 7, 8, 9 & 10: For each case study:
a) Develop high-level sequence diagrams for each use case
b) Identify MVC classes / objects for each use case
c) Develop Detailed Sequence Diagrams / Communication diagrams for each use case showing interactions among all the three-layer objects
d) Develop detailed design class model (use GRASP patterns for responsibility assignment)
e) Develop three-layer package diagrams for each case study

Week 11 & 12: For each case study:
a) Develop Use case Packages
b) Develop component diagrams
c) Identify relationships between use cases and represent them
d) Refine domain class model by showing all the associations among classes

Week 13: For each case study
a) Develop sample diagrams for other UML diagrams - state chart diagrams, activity diagrams and deployment diagrams

Text Books:

1. Object-Oriented Analysis & Design with the Unified Process by Satzinger, Jackson & Burd.

   Cengage Learning will be the primary source for finding templates for developing different artifacts / diagrams

2. Applying UML and patterns' by Craig Larman, Pearson
3. UML 2 Toolkit, by Hans-Erik Eriksson, Magnus Penker, Brian Lyons, David Fado: WILE\'-Dreamtech lndia Pvt. Lid.

Course Code		L	T	P	Credits
1005173223	**WEB TECHNOLOGIES LAB**	0	0	3	2

Course Objectives:

1. To acquire knowledge of XHTML, Java Script and XML to develop web applications
2. Ability to develop dynamic web content using PHP, MySql
3. To understand Database connections and Mail API through PHP
4. To understand the design and development process of a complete web application

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Create a static web pages using HTML and CSS and Develop JavaScript code for data validation	Understand	PO1
CO2	Demonstrate how *XML* provides a standard method to access information	Applying	PO1, PO2
CO3	Demonstrate database connectivity for developing web applications	Applying	PO5
CO4	Summarize object oriented programming concepts	Analyzing	PO12

List of Experiments:

1. Design the following static web pages required for a Training and placement cell web site.

   1) Home Page 2) Login Page 3) Registration page 4) Company Details Page 5) Alumni Details Page 6) Placement Staff Details Page7) Student personal Info Page 8) Student Academic Info page 9) Semester Wise Percentage & their Aggregate page
2. Validate login page and registration page using regular expressions.
3. Apply different font styles, font families, font colors and other formatting styles to the above static web pages.
4. Write an XML file which will display the Student personal information which includes

the Book Information Fields

5. Install wamp server to access above developed static web pages using this server.
6. Write a PHP to connect to the database, Insert the details of the users who register with the web site, whenever a new user clicks the submit button in the registration.
7. Write a PHP to connect to the database, Insert the details of the student academic information with student academic info page
8. Write perl program takes set names along the command line and prints whether they are regular files or special files
9. Write a perl program to implement UNIX `passed' program
10. An example perl program to connect to a MySQl database table and executing simple commands.

Text Books:

1. Ruby on Rails Up and Running, Lightning fast Web development, Bruce Tate, Curt Hibbs, Oreilly (2006)
2. Programming Perl, 4ed, Tom Christiansen, Jonathan Orwant, Oreilly (2012)
3. Web Technologies, HTML< JavaScript, PHP, Java, JSP, XML and AJAX, Black book, Dream Tech.
4. An Introduction to Web Design, Programming, Paul S Wang, Sanda S Katila, Cengage.

Course Code	INDUSTRY ORIENTED MINI PROJECT	L	T	P	Credits
1005173241		0	0	0	2

Industry- Oriented Mini Project: The Industry oriented mini project is carried out during the third year second semester vacation. The students have an option to choose their own area of interest which may be related to the course work. Mini project report is evaluated for 100 marks in fourth year first semester before the first mid-term exam. Assessment is done by the supervisor /guide for 40 marks based on the work and mini project report. The remaining 60 marks are allocated for presentation by the student to a committee comprising of the project supervisor and two senior faculty members nominated by Head of the Department (or) Industry oriented MOOCs course (including NPTEL/ Coursera) for not less than EIGHT weeks can also be considered as equivalent. The list of courses in such case shall be approved by Head of the department concerned. The registered course must not be same as any of the courses listed in the program structure of their regulation till final year. Marks/grades are awarded based on the performance in viva voce or written examination conducted for Coursera courses and online courses other than SWAYAM/NPTEL where there is no end examination.

PROGRAM STRUCTURE

FOR

IV-B.Tech

I & II SEMESTERS

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
PROGRAM STRUCTURE

IV B.Tech **I Semester**

S.No	Course Code	Course Title	L	T	P	Credits
1	1005174101	Cryptography and Network Security	3	0	0	3
2	1099172106	Managerial Economics and Financial Analysis	3	0	0	3
3	1005174102	Machine Learning	3	0	0	3
4	1005174103	Big Data Analytics	3	0	0	3
5	**Elective-I**					
	1005174104	Mobile Ad-hoc Networks	3	0	0	3
	1005174105	Software Project Management				
	1004174105	IoT & its Applications				
6	**Elective-II**					
	1012172201	Computer Graphics	3	0	0	3
	1005174106	Cloud Computing				
	1012173203	Software Testing Methodologies				
7	1005174121	Cryptography and Network Security Lab	0	0	3	2
8	1005174122	Big Data Analytics Lab	0	0	3	2
Total Credits:						**22**

IV B.Tech **II Semester**

S.No	Course Code	Course Title	L	T	P	Credits
1	1005174201	Fundamentals of Block Chain Technology	3	0	0	3
2	1099172203	Management Science	3	0	0	3
3	1005174202	Software Architecture and Design Patterns	3	0	0	3
4	**Elective-III**					
	1005174203	Distributed Systems	3	0	0	3
	1005174204	Optimization Techniques				
	1005174205	Concurrent and parallel programming				
(OR)						
1005174281		Internship	0	0	0	12
5	1005174251	Technical seminar	0	0	0	2
6	1005174261	Comprehensive Viva	0	0	0	2
7	1005174231	Main Project	0	0	0	10
Total Credits :						**26**

DETAILED SYLLABUS FOR

IV-B.Tech

I-SEMESTER

Course Code	CRYPTOGRAPHY AND NETWORK SECURITY	L	T	P	Credits
1005174101		3	0	0	3

Course Overview: The objective of this course is to equip the students with principles and practice of cryptography and network security, Classical systems, symmetric block ciphers (DES, AES, other contemporary symmetric ciphers), Public-key cryptography (RSA, Discrete logarithms), Algorithms for factoring and discrete logarithms, cryptographic protocols, hash functions, authentication, key management, key exchange, signature schemes, Email and web security.

Course Objectives:

1. Understanding the requirement of security in modern communication and information systems.
2. Mastering the concept of security attack, services and mechanisms.
3. Mastering concepts of confidentiality using cryptography with mathematical background.
4. Mastering concept of authentication using hash algorithms and digital signature
5. To be familiar with network security designs using available secure solutions (PGP, SSL, IPSe c).

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Understand the principles and practices involved in cryptography and network security	Understand/ Remember	PO1
CO2	Understand the various symmetric and Asymmetric encryption algorithms.	Understand	PO1,PO2
CO3	Identifying cryptographic protocols, hash functions, authentication, key management, key exchange, signature schemes	Apply	PO2, PO3, PO4, PO12
CO4	Design of network security solutions for E-mail Security like PGP,S/MIME and web security like SSL,TLS .	Evaluate/Create	PO1, PO2, PO3, PO4, PO6, PO12

UNIT-I:

Security attacks, services & mechanisms, fundamental security principles, A Model for Network Security, Symmetric Cipher Model, Substitution Techniques Transportation Techniques, Rotor Machines, steganography.

Outcome: Demonstrate a systematic and critical understanding of the theories, principles and practices of Cryptography and network security.

Activity: Implement a simple cryptographic function

UNIT-II:

Secret Key Cryptography: Traditional Block Cipher Structure**,** Data Encryption Standard (DES), Block Cipher Design Principles, Triple DES, Blowfish, AES, Steam ciphers, RC4, Modes of Operation.

Outcome: Understand fundamentals of symmetric cryptographic algorithms like DES, AES, BLOWFISH etc.

Activity: Implement a AES algorithm in Python Language

UNIT-III:

Number Theory: Prime and Relatively Prime Numbers, Modular Arithmetic, Fermat's and Euler's Theorems, the Chinese Remainder Theorem, Discrete Logarithms.

Public Key Cryptography: Principles of Public Key Cryptosystems, RSA Algorithm, Diffie-Hellman Key Exchange, Introduction to Elliptic Curve Cryptography.

Outcome: Understand the asymmetric cryptography algorithms like RSA, Elliptical Cryptography etc

Activity: Bit coin and time stamp server

UNIT-IV:

Cryptographic Hash Functions: Applications of Cryptographic Hash Functions, Secure Hash Algorithm (SHA), Message Authentication Codes - Message Authentication Requirements and Functions, HMAC, Digital signatures, RSA Digital Signature Scheme, NIST Digital Signature Schemes(DSA approach)

Outcome: Understand the Authentication functions the manner in which Message Authentication Codes and Hash Functions works.

Activity: Generate digital signature for a given message

Unit-V:

Security at application layer: PGP and S/MIME, Security at the Transport Layer: SSL and TLS, Security at the Network Layer: IPSec, System Security

Outcome: Understand existing system security protocols like Kerberos, PGP, SSL and IPSEC

Activity: Develop a web page using a protocol HTTPS

TEXT BOOKS

1. Cryptography and Network Security, Behrouz A Forouzan, DebdeepMukhopadhyay, (3e) Mc Graw Hill.
2. Cryptography and Network Security, William Stallings, (7e) Pearson.

Reference Books:

1. Everyday Cryptography, Keith M.Martin, Oxford.
2. Network Security and Cryptography, Bernard Meneges, Cengage Learning.
3. Cryptography and Network Security: AtulKahate, Mc Graw Hill, 2nd Edition.
4. Information Security, Principles and Practice : Mark Stamp, Wiley India.
5. Principles of Computer Sceurity: WM.Arthur Conklin, Greg White, TMH
6. Introduction to Network Security: Neal Krawetz, CENGAGE Learning.
7. Principles of Information security by Michael E Whitman and Herbert J.Mattord.
8. Cryptography and Network Security: C K Shyamala, N Harini, Dr T R Padmanabhan, Wiley India, 1st Edition.

Course Code	MANAGERIAL ECONOMICS AND FINANCIAL ANALYSIS	L	T	P	Credits
1099172106		3	0	0	3

Course Overview: The present course is designed in such a way that it gives an overview of concepts of Economics. Managerial Economics enables students to understand micro environment in which markets operate how price determination is done under different kinds of competitions. Financial Analysis gives clear idea about concepts, conventions and accounting procedures along with introducing students to fundamentals of ratio analysis and interpretation of financial statements. Break Even Analysis is very helpful to the Business Concern for Decision Making, controlling and forward Strategic Planning. Ratio analysis gives an idea about financial forecasting, financial planning, controlling the business and decision making.

Course Objectives:

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

1. Understand the concepts of managerial economics and the market dynamics namely Demand, Elasticity of demand and pricing in different market structures.
2. Acquire the knowledge about production theories and cost analysis besides dealing with the production and factors of production.
3. Analyze the different market structures and understand various pricing methods which are adopted in attracting the customers under different markets.
4. To provide the basic knowledge on financial accounting
5. To understanding Capital budgeting decisions.

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Analyze the Demand, Price and Cost.	Understand	PO3, PO8, PO11, PO12
CO2	Identify the Nature of different markets to determine Price Output for different Business Units	Understand& Apply	PO5, PO8, PO11, PO12
CO3	Understand Various Business Forms	Understand	PO5, PO8, PO11, PO12
CO4	Evaluate investment project proposals	Apply & Analyze	PO3, PO11, PO12

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

Outcome: Describe the economic activities performed by the businessmen in the business for profit earning. Understand the significance of demand, its analysis, measurement of demand and its Forecasting

Activity: Presentations and object oriented tests

Unit-II:

Production and Cost Analyses: Concept of Production function- Cobb-Douglas Production function- Leontief production function - Law of Variable proportions-Isoquants and Isocosts and choice of least cost factor combination-Concepts of Returns to scale and Economies of scale- Different cost concepts: opportunity costs, explicit and implicit costs- Fixed costs, Variable

Costs and Total costs –Cost –Volume-Profit analysis-Determination of Breakeven point(simple problems)-Managerial significance and limitations of breakeven point.
Outcome: Evaluate the production theories and pricing policies of various enterprises

Activity: Presentations and object oriented tests

Unit-III:
Part:I: Introduction to Markets, Theories of the Firm & Pricing Policies: Managerial Theories of firm: Marris and Williamson's models – Significance of Pricing and various methods of pricing with contemporary examples. Market Structures: Perfect Competition, Monopoly, Monopolistic competition and Oligopoly – Features – Price and Output Determination.
Part: II: Types of Business Organization and Business Cycles: Features and Evaluation of Sole Trader, Partnership, Joint Stock Company – State/Public Enterprises and the irforms – Business Cycles : Meaning and Features – Phases of Business Cycle.
Outcome: Design and implement different structures of market covering how price is determined under different market structures. Also can able to take decisions using business cycles
Activity: Presentations and object oriented tests

Unit-IV:
Introduction to Accounting and Capital Budgeting Decisions: Part I: Introduction to Accounting, Double Entry Systems Journal, Ledger, Trail Balance, preparation of Financial Statements (Problems).
Outcome: Analyze different forms of business organizations existing in the modern business and able to choose suitable form of business.
Activity: Presentations and object oriented tests.

Unit-V:
Capital Budgeting Decisions: Classification of Capital- Methods of appraising Project profitability: Traditional Methods (Payback period, Accounting rate of return) and Time value of money- Modern methods (Net Present Value method, Internal Rate of Return Method and Profitability Index Method) – Problems
Outcome: Able to prepare financial statements and understand and implement the capital budgeting tools and techniques.

Activity: Presentations and object oriented tests

Text Books:
1. M.Kasi Reddy & Saraswathi, "Managerial Economics and Financial Analysis", PHI Publications, New Delhi, 10th Revised Edition, 2012.
2. Varshney & Maheswari, "Managerial Economics", Sulthan Chand Publishers, 1st Revised Edition, 2009.
3. S.N. Maheshwari & S.K.Maheshwari, "Financial Accounting", Vikas Publication House Pvt.Ltd, 4th Edition, 2012.

Reference Books:

1. D.N. Dwivedi, "Managerial Economics", Vikas Publication House Pvt.Ltd, 2nd Edition, 2012.
2. R.NarayanaSwamy, "Financial Accounting- A managerial Perspective", Pearson publications, 1st Indian Reprint Edition, 2012.
3. J.V.Prabhakar Rao &P.V.Rao, "Managerial Economics & Financial Analysis", Maruthi Publishers, 1st Revised Editon, 2011

Course Code		L	T	P	Credits
1005174102	MACHINE LEARNING	3	0	0	3

Course Overview: This course provides how to recognize the characteristics and applications of machine learning. The course provides brief information about supervised, unsupervised, clustering algorithms and Artificial neural networks.

Course Objectives:

1. Familiarity with a set of well-known supervised, unsupervised and semi-supervised learning
2. The ability to implement basic machine learning algorithms
3. Understanding of how machine learning algorithms are evaluated
4. Applying new concepts in machine learning

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Recognize the characteristics of machine learning that make it useful to real-world Problems	Understand/Remember	PO1,PO3,PO4
CO2	Characterize machine learning algorithms as supervised, semi-supervised, and Unsupervised	Understand/analyze	PO1,PO5,PO9,
CO3	Be able to use support vector machine, regularized regression algorithms	Create	PO3,PO5,PO9,PO2
CO4	Understand the concept behind neural networks for learning non-linear functions	Evaluate/Apply	PO2,PO3,PO5,PO12

UNIT -I: Introduction to Machine Learning:

Introduction to machine learning, Definition, traditional programming vs machine learning algorithms, learning a system, supervised learning, unsupervised learning and reinforcement learning, application areas.

UNIT- II: Classification and Regression Models

Linear separability and decision regions, linear discriminants, linear regression, logistic regression, decision trees-ID3 and C4.5, KNN

UNIT -III: Dimensionality reduction and Support vector machines

Dimensionality reduction and Feature selection, Dimensionality reduction algorithms: LDA and PCA, Margin of a classifier, Support Vector Machine, Learning nonlinear hypothesis using kernel functions.

UNIT- IV: Clustering and Ensemble Methods

Introduction to clustering: K-means clustering, Gaussian mixture model, Ensemble Methods: bagging and boosting, Random forest and Ada Boost algorithms and Bayesian learning algorithm.

UNIT- V:ARTIFICIAL NEURAL NETWORKS:

Introduction,The perceptron,the perceptron learning algorithm, Multilayer neural

networks,activation functions, Back Propagationalgorithm and introduction to Deep learning models: CNN

Text Books:

1. Tom Mitchell, *"Machine Learning"*, Mc GrawHill publications, 1997
2. Machine Learning: The art and science of algorithms that make sense of data, Peter Flach, Cambridge.
3. Introduction to Machine Learning with Python By Andreas C. Müller, Sarah Guido O'Reilly Media
4. Deep Learning by Josh Patterson, Adam Gibson ,O'Reilly Media

Reference Books:

1. Understanding Machine Learning: From Theory to Algorithms, Shai Shalev-Shwartz, Shai Ben- David, Cambridge.
2. Machine Learning in Action, Peter Harington, 2012, Cengage.

Course Code	BIG DATA ANALYTICS	L	T	P	Credits
1005174103		3	0	0	3

Course Overview: This course provides practical foundation level training that enables immediate and effective participation in big data projects. The course provides grounding in basic and advanced methods to big data technology and tools, including Map Reduce and Hadoop and its ecosystem

Course Objectives:

1. Optimize business decisions and create competitive advantage with Big Data analytics
2. Introducing Java concepts required for developing map reduce programs
3. Derive business benefit from unstructured data
4. Imparting the architectural concepts of Hadoop and introducing map reduce paradigm
5. To introduce programming tools PIG & HIVE in Hadoop echo system

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Preparing for data summarization, query, and analysis.	Understanding	PO1, PO2
CO2	Applying data modeling techniques to large data sets	Applying	PO1, PO2,PO4
CO3	Creating applications for Big Data analytics	Applying	PO1, PO2,PO4
CO4	Building a complete business data analytic solution	Applying	PO1, PO2,PO4, PO5

UNIT-I:

Working with Big Data: Google File System, Hadoop Distributed File System (HDFS) – Building blocks of Hadoop (Namenode, Datanode, Secondary Namenode, JobTracker, TaskTracker), Introducing and Configuring Hadoop cluster (Local, Pseudo-distributed mode, Fully Distributed mode), Configuring XML files.

Outcome:

1. Analyze the distinction between GFS and HDFS
2. Demonstrate understanding of different mode of Hadoop Installations.

Activity: Installation of Hadoop and configuring various XML files

Unit-II:

Writing Map Reduce Programs: A Weather Dataset, Understanding Hadoop API for Map Reduce Framework (Old and New), Basic programs of Hadoop Map Reduce: Driver code, Mapper code, Reducer code, Record Reader, Combiner, Partitioner

Outcome:

1. Analyze the distinction Map Reduce execution of old and new versions
2. Demonstrate understanding of different code blocks

Activity: Running of Map Reduce program to forecast weather.

Unit-III:

Hadoop I/O: The Writable Interface, Writable Comparable and comparators, Writable Classes: Writable wrappers for Java primitives, Text, Bytes Writable, Null Writable, Object Writable and

Generic Writable, Writable collections, Implementing a Custom Writable: Implementing a Raw Comparator for speed, Custom comparators

Outcome:
1. Understanding the I/O classes used for Hadoop MapReduce concept
2. Able to write wrapper classes and Generic class programs

Activity: Implementation of I/O operations using writable wrappers

Unit-IV:

Admiring the Pig Architecture, Going with the Pig Latin Application Flow, Working through the ABCs of Pig Latin, Evaluating Local and Distributed Modes of Running Pig Scripts, Checking out the Pig Script Interfaces, Scripting with Pig Latin

Outcome:
1. Understand the PIG Architecture and Modes of operations.
2. Able to write the PIG scripts

Activity : Installation of PIG and running pig scripts on different modes

UNIT-V:

Applying Structure to Hadoop Data with Hive: Saying Hello to Hive, Seeing How the Hive is Put Together, Getting Started with Apache Hive, Examining the Hive Clients, Working with Hive Data Types, Creating and Managing Databases and Tables, Seeing How the Hive Data Manipulation Language Works, Querying and Analyzing Data

Outcome:
1. Understand the HIVE Architecture and Modes of operations.
2. Able to create database on HIVE environment

Activity: Installation of HIVE and running running the queries on database

Text Books:
1. Big Java 4th Edition, Cay Horstmann, Wiley John Wiley & Sons, INC
2. Hadoop: The Definitive Guide by Tom White, 3rd Edition, O'reilly
3. Hadoop in Action by Chuck Lam, MANNING Publ.
4. Hadoop for Dummies by Dirk deRoos, Paul C.Zikopoulos, Roman B.Melnyk,Bruce Brown, Rafael Coss

Reference Books:
1. Hadoop in Practice by Alex Holmes, MANNING Publ.
2. Hadoop MapReduce Cookbook, SrinathPerera, ThilinaGunarathne

ELECTIVE - I

Course Code		L	T	P	Credits
1005174104	**MOBILE AD-HOC NETWORKS**	3	0	0	3

Course Overview: This course focusing about Mobile adhoc networks, introduction, features, data retrievals and security in WSN's.

Course Objectives:

After taking the course, students will be able to

1. To make the student understand the concepts of MOBILE AD HOC NETWORKS (Manets) as well as Wireless Sensor Networks (WSN), their characteristics, novel applications, and technical challenges.
2. To understand the issues and solutions of various layers of Manets, namely MAC layer, Network Layer & Transport Layer in Manets and WSN.
3. To understand the platforms and protocols used in Manets and WSN.
4. To make the student take up further research as part of his higher studies

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Able to think and develop new applications in Manets and WSN.	Understanding	PO:1,2,4,5, 12
CO2	To enable the student to understand the need for security and the challenges and also the role of cross layer design in enhancing the network performance	Understanding	PO:1,2,3,4, 12
CO3	Able to develop algorithms/protocols for Manets and WSN.	Applying	PO:1,2,3,4, 5
CO4	Able to take any new technical issue related to these new thrust areas and come up with a solution(s).	Analyzing	PO:1,2,3,4, 5

Unit-I:

Introduction to Ad Hoc wireless networks: Cellular and Ad Hoc Wireless Networks, Characteristics of of MANET's, Issues MANET's, Applications and Challenges of MANET's.

Routing in MANET's: Classification of Routing Protocols, Topology based versus Position based approaches, Topology based routing protocols, Position based routing, Other routing protocols.

Outcome: Understanding the characteristics of adhoc wireless networks and their applications

Activity: demonstrate an understanding of the trade-offs involved in the design of adhoc networks

Unit-II:

Networks **Data Transmission in MANET's:** The broadcast storm, Multicasting, Geocasting, TCP over Ad Hoc – TCP protocol overview, TCP and MANET's, solution for TCP over Ad Hoc.

Security in MANET's: Security in Ad Hoc wireless networks, key management, Secure routing, cooperation in MANET's, Intrusion Detection System.

Outcome: Understanding the data transmission and security in MANET's

Activity: design and implement protocols suitable to adhoc communication scenario

Unit-III:

Basics of wireless sensors and applications: The Mica Mote, Sensing and Communication Range, Design Issues, Energy Consumption, Clustering of Sensors, Applications.

Data Retrieval in Sensor Networks: Classification of WSNs, MAC Layer, Routing Layer, High-Level Application Layer Support, Adapting to the Inherent Dynamic Nature of WSNs.

Outcome: understand the basics of wireless sensors and data retrievals

Activity: demonstrate routing layer algorithms

Unit-IV:

Sensor Network Platforms and Tools: Sensor network Hardware, Sensor Network Programming Challenges, and Node-Level Software Platforms, Node – level simulators.

Outcome: able to understand sensor network platforms and tools

Activity: sensor network programming challenges

Unit-V:

Secure data aggregation in wireless sensor networks, Introduction to vehicular Ad Hoc networks, Introduction to wireless mesh networks **Security in WSN's**: Security in wireless sensor networks, Key management in wireless sensor networks.

Outcome: Able to understand security in WSN's

Activity: security in WSN's

Text Books:

1. Ad Hoc and Sensor Networks: Theory and Applications, Carlos de MoraisCordeiro and Dharma Prakash Agrawal, World Scientific Publications / Cambridge University Press, 2006. 2.
2. Wireless Sensor Networks: An Information Processing Approach, Feng Zhao, Leonidas Guibas, Elsevier Science Imprint, Morgan Kauffman Publishers, 2005.

Reference Books:

1. Ad Hoc Wireless Networks: Architectures and Protocols, C. Siva Ram Murthy and B. S. Manoj, Pearson Education, 2004.
2. Guide to Wireless Ad Hoc Networks, SudipMisra. Isaac Woungang, and Subhas Chandra Misra, Springer International Edition. 2011.
3. Guide to Wireless Sensor Networks. SudipMisra, Isaac Woungang. and Subhas Chandra Misra, Springer International Edition. 2012.
4. Wireless Mesh Networking. Thomas Krag and SebastinBuettrich. O'Reilly Publishers, 2007.
5. Wireless Sensor Networks - Principles and Practice, Fei Hu, Xiaojun Cao, An Auerbach book, CRC Press, Taylor &Francis Group, 2010.

Course Code	SOFTWARE PROJECT MANAGEMENT	L	T	P	Credits
1005174105		3	0	0	3

Course Overview: To study how to plan and manage projects at each stage of the software development lifecycle (SDLC). To train software project managers and other individuals involved in software project planning and tracking and oversight in the implementation of the software project management process. To understand successful software projects that support organization's strategic goals

Course Objectives:

1. To study how to plan and manage projects at each stage of the software development lifecycle (SDLC)
2. To train software project managers and other individuals involved in software project planning and tracking and oversight in the implementation of the software project management process.
3. To understand successful software projects that support organization's strategic goals

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	To understand the basic concepts and issues of software project management	understand	PO1
CO2	To conduct activities necessary to successfully complete and close the Software projects	Understanding	PO1,PO2
CO3	To implement the project plans through managing people, communications and change	Applying	PO2,PO3, PO4,PO12
CO4	To develop the skills for tracking and controlling software deliverables	Analyzing	PO1,PO2, PO3,PO4, PO6,PO12

UNIT-I: Introduction

Project, Management, Software Project Management activities, Challenges in software projects, Stakeholders, Objectives & goals

Project Planning: Step-wise planning, Project Scope, Project Products & deliverables, Project activities, Effort estimation, Infrastructure

Outcome: Develop basic knowledge of Project Management and Project Planning

Activity: Design a plan for the given project

UNIT-II: Project Approach:

Lifecycle models, Agile model, Choosing Technology, Prototyping Iterative & incremental Process Framework: Lifecycle phases, Process Artifacts, Process workflows.

Outcome: Obtain knowledge about various project Approaches and Process Framework

Activity: Choose a suitable Project Approach and define a Process Framework for a project

UNIT-III: Effort estimation & activity Planning

Estimation techniques, Function Point analysis, SLOC, COCOMO, Use case-based estimation, Activity Identification Approaches, Network planning models, Critical path analysis

Outcome: Obtain knowledge about various estimation techniques and Network planning models

Activity: Estimate Effort using various estimation techniques for the project

UNIT-IV:
Risk categories, Identification, Assessment, Planning and management, PERT technique, Monte Carlo approach
Outcome: Define various risk categories and assess it by Identifying the risk.
Activity: Identify and asses Risks in the project

UNIT – V:
Creating a framework for monitoring & control, Progress monitoring, Cost monitoring, Earned value Analysis, Defects Tracking, Issues Tracking, Status reports, Types of Resources, Identifying resource requirements, Resource scheduling
Outcome: Obtain knowledge about monitoring & controlling process, cost etc., and Resource scheduling
Activity: Monitor and control the progress of the project

Text Books:

1. Rajib Mall, Fundamentals of Software Engineering, PHI Publication, 3rd Edition.
2. Pankaj Jalote, Software Engineering, Narosa Publication, 3rd Edition
3. K. K. Aggarwal and Yogesh Singh, Software Engineering, New Age International Publishers, 3rd Edition.

Reference Books:

1. R. S. Pressman, Software Engineering: A Practitioners Approach, McGraw Hill, 6th Edition.
2. Ian Sommerville, Software Engineering, Addison Wesley, 8th Edition.
3. Carlo Ghezzi, M. Jarayeri, D. Manodrioli, Fundamentals of Software Engineering, PHI Publication

Subject Code		L	T	P	Credits
1004174105	INTERNET OF THINGS & ITS APPLICATIONS	3	0	0	3

Course Overview: The purpose of this course is to impart knowledge on IoT Technology and Architecture, Internal communication protocols, corrections with other technologies, real time applications ad study practical design and implementation issues

Course Objectives: The main objective of course make student to understand the IoT basic concepts, standards, communication protocols, technological relation and real time applications and their design, implementation and deployment issues.

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	To Understand the Architecture, protocols and applications of IoT.	Understand	PO1
CO2	To Analyse the communication protocols and standards used in IoT	Analyse	PO2
CO3	To analyse and design the simple IoT applications to monitor or control IoT devices using simulation or hardware	Design and Creative thinking	PO3,PO6, PO7
CO4	To implement the real time IoT applications.	Design and Deployment	PO4,PO5, PO11, PO12

Unit-I:

Introduction to IoT, Need of Internet of Things, Internet of Things ERA, Characteristics of Internet of Things, architectural view of Internet of Things, Technologies behind Internet of Things – Server- End Technology – Major Components of IoT system – Development Tools – API and device interfacing components, Sources of IoT, Examples of IoT – Smart Watch – Smart Home – Smart Phone.

Outcome: Understand the characteristics, physical and logical of IoT and their application.

Activity: Identify physical and logical components involved in IoT applications

Unit-II:

Introduction, Generic computing systems Vs. Embedded systems, Purpose of Embedded Systems - Typical Embedded System – Core of the Embedded System – Memory – Sensors and Actuators with I/O subsystems – Communication Interfaces – Wireless Interfaces and Wire Interfaces – Characteristics of Embedded Systems – Quality Attributes of an Embedded Systems.

Outcome: Analyze and design hardware and software components of IoT application

Activity: Identification day to day embedded devices along with hardware and software components

Unit-III:

M2M communication – M2M to IoT – M2M architecture – software development tools, Communication Technologies – Wireless communication technologies – Wired

Communication, Physical Design of IoT – Things in IoT – IoT Protocols, Logical design of IoT – IoT functional blocks – IoT communication models.

Outcome: Analyze and design the communication protocols of IoT applications

Activity: Identification of communication styles of various IoT Protocols

Unit-IV:

Basic building blocks of an IoT devices, Introduction about the Raspberry Pi Board, Operating systems for Raspberry Pi, Interfaces for IoT – Serial Interface – SPI – I2C, IoT Design Methodology – Requirements – Process – Domain Model – Information model – service – Functional View – Operational View – Device & components Integration – Application development.

Outcome: Construction of IoT systems with raspberry pi and simulation tools

Activity: Hands-on setup of IoT Systems using Raspberry pi

Unit-V:

Case Studies: Home Automation – Smart lighting – Home intrusion detection, Cities – smart parking, Environment – Weather monitoring system – Air Pollution Monitoring – Forest Fire Detection, Agriculture – smart irrigation system.

Outcome: Understand physical and logical aspects of real time IoT applications.

Activity: Mini project on IoT applications for monitoring/control devices

Text Books:

1. Internet of Things: A hands-On Approach, ArshdeepBahga, Vijay Madisetti, 2014 edition, University Press.
2. The Internet of Things: Enabling technologies, Platforms and Use cases, Pethuru Raj and Anupama C. Raman, 2017 edition, CRC Press, Taylor and Francis Group.
3. Introduction to Embedded Systems, Shibu K V, 2nd Edition, Tata Mc-Graw hill Edition.

Reference Books:

1. Internet of Things: Architecture and design Principles, Raj Kamal, Tata Mc-Graw hill Edition.
2. Embedded Systems: Architecture and applications, Raj Kamal, Tata Mc-Graw hill Edition.

ELECTIVE II

Course Code		L	T	P	Credits
1012172201	COMPUTER GRAPHICS	3	0	0	3

Course Overview: Computer graphics are an intrinsic component of many modern software applications and are often essential to the success of these applications. This course is to familiarize the student with fundamental algorithms and data structures that are used in today's interactive graphics systems as well as programming and architecture of high-resolution graphics computers. The principles and practice of computer graphics are described from their mathematical foundations to the modern applications domains of scientific visualization, computer games and film animation.

Course Objectives:

1. Develop, design and implement two and three dimensional graphical structures.
2. Acquire knowledge on usage of various algorithms used for drawing, filling and
    a. Animations.
3. Learn to create, manage various Multimedia techniques used in animation.

Course Outcomes:

1. Know and be able to describe the general software architecture of programs that use 3D computer graphics.
2. Know and be able to discuss hardware system architecture for computer graphics.
3. Know and able to understand graphics pipeline, frame buffers, and graphic accelerators / co processors.
4. Know and be able to select among models for lighting/shading: Color, ambient light; distant and light with sources; Phong reflection model; and shading (flat, smooth, Goraud, Phong).

Unit-I:

Basic Of Computer Graphics: Pixel, Buffer frame, raster scan and random scan, resolution, aspect ratio, rasterization and scan conversion

2D primitives: Output primitives – Line, Circle and Ellipse drawing algorithms

Outcome:

1. Able to describe the architecture used in computer graphics programming
2. Able to understand different types of hardware and software used in computer graphics

Activity: Implement a line drawing algorithm to draw a snowman.

Unit-II:

Polygon Filling Algorithms: Scan line polygon, boundary filling flood filling

Clipping Algorithms: Cohen Sutherland , Sutherland-Hodgemen and Cyrus-Beck

Outcome: Able to understand different types of filling algorithms used in graphics.

Activity: Add to your drawing algorithm implementing any one of the filling algorithms.

Unit-III:

2-D Geometrical Transforms: Translation, scaling, rotation, reflection and shear transformations, matrix representations and homogeneous coordinates, composite transformations between coordinate systems

3-D Geometric Transformations: Translation, rotation, scaling, reflection and shear transformations, composite transformations. 3D Viewing pipeline, clipping, projections (Parallel and Perspective).

Outcome: Able to understand key transformation algorithms used in real time scenario
Activity: Create a simple polygon that rotates in 3D on its y-axis.

Unit-IV:
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.
Fractals: Fractals and Self similarity – Peano curves – Creating image by iterated functions – Mandelbrot sets – Julia Sets – Random Fractals

Outcome:
1. Know and able to understand graphics pipeline, frame buffers, and graphic accelerators / co-processors.
2. Know and be able to select among models for lighting/shading: Color, ambient light; distant and light with sources; Phong reflection model; and shading (flat, smooth, Goraud, Phong).

Activity: Add a floor to your scene and add a shadow of your object on the floor. You should see the shadow change as the object spins.

Unit-V:
Visible Surface Detection Methods: Classification, back-face detection, depth-buffer, ray tracing method, depth sorting, BSP tree methods.
Graphics Programming: Colour 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
Outcome:
1. Know and understand various colour models applied in real world.
2. Know and able to select graphics accelerators / co-processors

Activity: Create a simple Screen saver program using basic OPENGL commands

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 Code	CLOUD COMPUTING	L	T	P	Credits
1005174106		3	0	0	3

Course Overview: This course provides the comprehensive study of key concepts of Cloud & its environment. Building software systems and components that scale to millions of users in modern internet. Cloud concepts capabilities across the various cloud service models including Iaas, Paas, Saas. Developing cloud based software applications on top of cloud platforms. Storage systems and backup strategies for cloud based data.

Course Objectives:

After taking the course, students will be able:

1. To learn the basics of Cloud computing
2. To know the key concepts of Virtualization
3. To gain knowledge on cloud computing service models
4. To develop cloud implementation, programming and mobile cloud computing
5. To learn key components of Amazon web services
6. To maintain the Cloud backup and solutions

Course Outcomes:

	Course Outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Define cloud computing and memorize the different cloud computing service and deployment models	Understand/Remember	PO1
CO2	Describe the importance of virtualization along with their technologies for data center automation	Understand	PO1,PO5
CO3	Identify emerging cloud programming paradigms and its software environments	Apply	PO1,PO3,PO5
CO4	Describe the key application of cloud resource management. Design and develop the backup strategies for cloud data based on features.	Analyze	PO1, PO3, PO10

UNIT-I: Systems modelling, Clustering, and virtualization of Clusters and Data Centers: Scalable Computing over the Internet, Technologies for Network based systems, System modelsfor Distributed and Cloud Computing, Software environments for distributed systems andclouds, Performance, Security And Energy Efficiency.

Implementation Levels of Virtualization, Virtualization Structures/ Tools and mechanisms, Virtualization of CPU, Memory and I/O Devices, Virtual Clusters and Resource Management, Virtualization for Data Center Automation.

Outcome: understand the basics of cloud computing and key concepts of virtualization

Activity: virtualization tools and mechanisms

UNIT-II: Cloud Platform Architecture:

Cloud Computing and service Models, Architectural Design of Compute and Storage Clouds, Public Cloud Platforms, Inter Cloud Resource Management, Cloud Security and Trust Management. Service Oriented Architecture, Message Oriented Middleware

Outcome: To gain knowledge on cloud computing service models, architectural design of

store clouds

Activity: Critique the consistency of services deployed from a cloud architecture

UNIT-III: Cloud Programming and Software Environments: Features of Cloud and Grid Platforms, Parallel & Distributed Programming Paradigms, Programming Support of Google App Engine, Programming on Amazon AWS and Microsoft Azure, Emerging Cloud Software Environments, Data sources and sinks with an emphasis on MS Azure, Google Cloud

Outcome:

1. Able to develop cloud implementation, programming and mobile cloud computing,
2. Able to learn key components of Amazon web services

Activity: Compare and contrast the economic benefits delivered by various cloud models based on application requirements, economic constraints and business requirements.

UNIT-IV:

Cloud Resource Management and Scheduling-Policies and Mechanisms for Resource Management Applications of Control Theory to Task Scheduling on a Cloud, Stability of a Two Level Resource Allocation Architecture, Feedback Control Based on Dynamic Thresholds. Coordination of Specialized Autonomic Performance Managers, Resource Bundling, Scheduling Algorithms for Computing Clouds, Fair Queuing ,Start Time Fair Queuing, Borrowed Virtual Time, Cloud Scheduling Subject to Deadlines, Scheduling Map Reduce Applications Subject to Deadlines.

Outcome: understand the application of cloud resource management and scheduling algorithms for computing clouds.

Activity: cloud resource management and scheduling algorithms

Unit-V:

Storage Systems-Evolution of storage technology, storage models, file systems and database, distributed file systems, general parallel file systems. Google file system. Apache Hadoop, Big Table, Megastore, Amazon Simple Storage Service (S3).

Cloud Security and trust Management- Cloud Security defence strategies, Distributed intrusion/ Anomaly detection. Data and software protection techniques, Reputation guided protection of Data Centres.

Outcome: Analyze and develop the backup strategies for cloud data based on features

Activity: Critically analyze case studies to derive the best practice model to apply when developing and deploying cloud based applications

Text Books:

1. Distributed and Cloud Computing, Kai Hwang, Geoffry C. Fox, Jack J. Dongarra MK Elsevier.
2. Cloud Computing, Theory and Practice, Dan C Marinescu, MK Elsevier.
3. Cloud Computing, A Hands on approach, ArshadeepBahga, Vijay Madisetti, University Press

Reference Books:

1. Cloud Computing, A Practical Approach, Anthony T Velte, Toby J Velte, Robert Elsenpeter, TMH
2. Mastering Cloud Computing, Foundations and Application Programming, Raj Kumar Buyya, Christen vecctiola, S Tammaraiselvi, TMH.

Course Code		L	T	P	Credits
1012173203	**SOFTWARE TESTING METHODOLOGIES**	3	0	0	3

Course Overview: This course explains the fundamentals of software testing. It will provide a theoretical framework and also demonstrate how to practically implement software testing techniques. This course will explain how the testing process works and how it should be implemented. Candidates will gain an understanding of the different terminology involved in software testing and how each fits into the development of software.

Course Objectives:

- Fundamentals for various testing methodologies.
- Describe the principles and procedures for designing test cases.
- Provide supports to debugging methods.
- Acts as the reference for software testing techniques and strategies.

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Understanding the purpose of Software Testing.	Understand	PO1
CO2	Understand the Transaction Flow Testing and Dataflow testing	Understand	PO1, PO2
CO3	Test the software using domain testing and Logic Based Testing	Applying	PO1, PO2, PO3, PO4
CO4	Apply the software testing tools for real world applications	Applying, Create	PO1, PO2, PO3, PO4, PO5

UNIT-I:

Introduction: Purpose of Testing, Dichotomies, Model for Testing, Consequences of Bugs, Taxonomy of Bugs.

Flow graphs and Path testing: Basics Concepts of Path Testing, Predicates, Path Predicates and Achievable Paths, Path Sensitizing, Path Instrumentation, Application of Path Testing.

Outcome:

- Understanding the purpose of Software Testing.
- Develop Model for Testing.
- Able to use flow graphs and path testing.

Activity Write in detail about flow graphs and path testing.

Unit 2: Transaction Flow Testing: Transaction Flows, Transaction Flow Testing Techniques.

Dataflow testing: Basics of Dataflow Testing, Strategies in Dataflow Testing, Application of Dataflow Testing.

Domain Testing: Domains and Paths, Nice & Ugly Domains, Domain testing, Domains and

Interfaces Testing, Domain and Interface Testing, Domains and Testability.

Outcome: Obtain knowledge about Transaction Flow Testing and Dataflow testing.

Activity: Develop an application and test that application using different testing techniques.

Unit 3:

Paths, Path products and Regular expressions: Path Products & Path Expression, Reduction Procedure, Applications, Regular Expressions & Flow Anomaly Detection

Syntax Testing: Why, What and How, A Grammar for formats, Test Case Generation, Implementation and Application and Testability Tips.

Outcome:

- Knowledge of various testing techniques.
- Understanding path products and regular Expressions.

Activity: Test the software using domain testing and Logic Based Testing

Unit 4:

Logic Based Testing: Overview, Decision Tables, Path Expressions, KV Charts, and Specifications.

State, State Graphs and Transition Testing: State Graphs, Good & Bad State Graphs, State Testing, and Testability Tips.

Outcome:

- Understanding the use of state Graphs and Transition Testing.
- Knowledge of graph matrices and Application.

Activity: Discuss about Graph matrices and its applications.

UNIT – 5:

Graph Matrices and Application: Motivational overview, matrix of graph, relations, power of a matrix, node reduction algorithm.

Software Testing Tools: Introduction to Testing, Automated Testing, Concepts of Test Automation, Introduction to list of tools like Win runner, Load Runner, Jmeter, About Win Runner ,Using Win runner, Mapping the GUI, Recording Test, Working with Test, Enhancing Test, Checkpoints, Test Script Language, Putting it all together, Running and Debugging Tests, Analyzing Results, Batch Tests, Rapid Test Script Wizard.

Outcome:

- The need for automation and categorization of testing tools.
- Selection of testing tools and costs incurred.
- Guidelines for automated testing and an overview of some commercial testing tools

Activity: Discuss the usage of various testing tools.

Text Books:

1. Software testing techniques – Boris Beizer, Dreamtech, second edition.
2. Software Testing- Yogesh Singh, Camebridge

Reference Books:

1. The Craft of software testing - Brian Marick, Pearson Education.
2. Software Testing, 3rd edition, P.C. Jorgensen, Aurbach Publications (Dist.by SPD).
3. Software Testing, N.Chauhan, Oxford University Press.
4. Introduction to Software Testing, P.Ammann&J.Offutt, Cambridge Univ.Press.
5. Effective methods of Software Testing, Perry, John Wiley, 2nd Edition, 1999.
6. Software Testing Concepts and Tools, P.NageswaraRao, dreamtech Press
7. Win Runner in simple steps by Hakeem Shittu, 2007Genixpress

Course Code		L	T	P	Credits
1005174121	**CRYPTOGRAPHY AND NETWORK SECURITY LAB**	0	0	3	2

Course Overview: To give practical exposure on basic security attacks, encryption algorithms, authentication techniques. Apart from security algorithms, firewall configuration is also introduced.

Course Objectives:

1. To provide deeper understanding into cryptography, its application to network security, threats/vulnerabilities to networks and countermeasures.
2. To explain various approaches to Encryption techniques, strengths of Traffic Confidentiality, Message Authentication Codes.
3. To familiarize symmetric and asymmetric cryptography

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Identify basic security attacks and services	understanding	PO1, PO2
CO2	To understand the implementation of Symmetric cryptographic algorithms	applying	PO1, PO2,PO4
CO3	To understand the implementation of Asymmetric cryptographic algorithms	applying	PO1, PO2,PO4
CO4	To understand the implementation of Various Cryptographic Hash algorithms	applying	PO1, PO2,PO4, PO5

1. Working with sniffers for monitoring network communication using Wireshark
2. Implementation of Caesar Cipher technique
3. Decode the Given message which is encrypted using Caesar Cipher. Also Identify the key. M = "AOPZPZJYFWAVNYHWOFULADVYRZLJBYPAFSHI"
4. Implement implement Simple Substitution Cipher
5. Decode the Given message which is encrypted using Simple Substitution cipher. Also identify the key.M=

"ceiraoxnirvhrcacvcadfqaxeikarpqaxeikcepceprhiifafvriudkopfwevfgkigrduwipkrpfzqinnifcearcdkwarmasifafraodfrafmerceiqdgihddlachpraqpnnwqdfrarcrdurvhrcacvcafmisikwxnpafcizcqepkpqcikudkpgauuikifcqaxeikcizcqepkpqcikacgauuikrukdoceiqpirpkqaxeikafcepcceiqaxeikpnxephicarfdcraoxnwceipnxephicreaucigacarqdoxnicinwyvohnigceiraoxnirvhrcacvcadfqaxeikduuikrsikwnaccniqdoovfaqpcadfriqvkacwpfgactannhiredtfcepcacqpfhiipranwhkdlifisifhwepfgirxiqapnnwprceioirrpmirhiqdoindfmikodkicepfrisikpnevfgkigqaxeikcizcqepkpqcikr"

6. Implement the Play fair Cipher
7. Implement the Pure Transposition Cipher

8. Implement DES Encryption and Decryption
9. Implement the BLOWFISH Encryption and decryption
10. Implement the AES Encryption and decryption
11. Implement RSA Encryption and Decryption
12. Implementation of CRC Hash Functions
13. Implement SHA – 128 Hash Functions

Text Books:

1. Cryptography and Network Security, Behrouz A Forouzan, Debdeep Mukhopadhyay, (3e) Mc Graw Hill.
2. Cryptography and Network Security, William Stallings, (7e) Pearson.

Course Code	BIGDATA ANALYTICS LAB	L	T	P	Credits
1005174122		0	0	3	2

Course Overview: This course provides practical foundation level training that enables immediate and effective participation in big data projects. The course provides grounding in basic and advanced methods to big data technology and tools, including Map Reduce and Hadoop and its ecosystem

Course Objectives:

- Optimize business decisions and create competitive advantage with Big Data analytics
- Introducing Java concepts required for developing map reduce programs
- Derive business benefit from unstructured data
- Imparting the architectural concepts of Hadoop and introducing map reduce paradigm
- To introduce programming tools PIG & HIVE in Hadoop echo system

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Preparing for data summarization, query, and analysis.	Understanding	PO1, PO2
CO2	Applying data modeling techniques to large data sets	Applying	PO1, PO2,PO4
CO3	Creating applications for Big Data analytics	Applying	PO1, PO2,PO4
CO4	Building a complete business data analytic solution	Applying	PO1, PO2,PO4, PO5

Week 1:

1. (i)Perform setting up and Installing Hadoop in its three operating modes:
- Standalone,
- Pseudo distributed,
- Fully distributed.

(ii)Use web based tools to monitor your Hadoop setup.

Week 2:

2. Implement the following file management tasks in Hadoop:
- Adding files and directories
- Retrieving files
- Deleting files

Hint: A typical Hadoop workflow creates data files (such as log files) elsewhere and copies them into HDFS using one of the above command line utilities.

Week 3:

3. Run a basic Word Count Map Reduce program to understand Map Reduce Paradigm.

Week 4:

4. Write a Map Reduce program that mines weather data.
Weather sensors collecting data every hour at many locations across the globe gather a

large volume of log data, which is a good candidate for analysis with Map Reduce, since it is semi structured and record oriented.

Week 5:

5. Implement Matrix Multiplication with Hadoop Map Reduce

Week 6:

6. Install and Run Pig then write Pig Latin scripts to sort, group, join, project, and filter your data.

Week 7:

7. Install and Run Hive then use Hive to create, alter, and drop databases, tables, views,functions, and indexes

Text Books:

1. Big Java 4th Edition, Cay Horstmann, Wiley John Wiley & Sons, INC
2. Hadoop: The Definitive Guide by Tom White, 3rd Edition, O'reilly
3. Hadoop in Action by Chuck Lam, MANNING Publ.
4. Hadoop for Dummies by Dirk deRoos, Paul C.Zikopoulos, Roman B.Melnyk,Bruce Brown, Rafael Coss

Reference Books:

1. Hadoop in Practice by Alex Holmes, MANNING Publ.
2. Hadoop MapReduce Cookbook, SrinathPerera, ThilinaGunarathne

Course Code	FUNDAMENTALS OF BLOCK CHAIN TECHNOLOGY	L	T	P	Credits
1005174201		3	0	0	3

Course Overview: This course discusses about various types of agreements and abstract models of Black Chain. It also focuses on Bitcoin technology, various cryptographic payment systems both e-cash and credit card based technologies. Finally this course dealt with trends in Block chain technologies.

Course Objectives:

- To understand abstract models of block chain technology
- Ability to know how the various payment systems will work on real time.
- Understand the mathematical analysis of the bit coin technology.

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Familiarize the functional/operational aspects of crypto currency ECOSYSTEM	Understanding	PO1, PO2
CO2	Understand emerging abstract models for Blockchain Technology	Understanding	PO1, PO2
CO3	Identify major research challenges and technical gaps existing in between theory and practice in cryptocurrency domain	Analyzing	PO1, PO3, PO4
CO4	Develop a course project using a Bitcoin technology	Applying	PO1, PO3, PO4, PO5, PO11, PO12

UNIT-I

Cryptography, Digital Signatures, Public VS Private Key Cryptography, MD VS SHA, Hashing Algorithm (SHA-256), Merkle Tree

Outcome: Gives a brief introduction about cryptography and basic algorithms

Activity: Hashing algorithms

UNIT-II Tracking origin, Challenges of current transaction system, The emergence of bit coin, Revolutionizing the traditional business network, Explore, Benefits of block chain, Questions of trust

Outcome: Able to understand benefits and challenges of block chain technology

Activity: Bitcoin technology

UNIT-III

Introduction to Bit coin, Double Spending Problem, Transactions, Time-Stamp Server, Proof-of-Work

Outcome: Understand emerging abstract models of block chain technology

Activity: Bit coin and time stamp server

UNIT-IV:

Network, Incentive, Reclaiming Disk Space ,Usage of Merkel Tree, Simplified payment verification, Combining and Splitting Values, Privacy

Outcome: Identify major research challenges and technical gaps existing between theoretical and practise in cryptocurrency domain

Activity: Usage of Merkel trees

Unit-V:

Why the name, Business Suitability(Shared Ledger, Permissions, Consensus),Types of frictions, Achieving friction free business , Transforming Ecosystems, Use cases(Financing, Insurance, Supply chain management, Healthcare, IOT)

Outcome: Understand the mathematical analysis of the bit coin technology

Activity: Simple project on bit coin technology

TEXT BOOKS

1. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, and Steven Goldfeder. Bitcoin and cryptocurrency technologies: a comprehensive introduction. Princeton University Press, 2016.
2. Blockchain: Ultimate guide to understanding blockchain, bitcoin, cryptocurrencies, smart contracts and the future of money, Mark Gates, Wise Fox Publisher, 2017

Reference Books:

1. Mastering Bitcoin: Programming the Open Blockchain, Andreas M. Antonopoulos, O' Reilly
2. Selected papers from IEEE Symposium on security and Privacy, EUROCRYPT and PODC

Course Code	MANAGEMENT SCIENCE	L	T	P	Credits
1099172203		3	0	0	3

Course Overview:

This course is intended to familiarize the students with the framework for the managers and leaders available for understanding and making decisions relating to issues related organizational structure, production operations, marketing, Human resource Management, product management and strategy

Course Objectives:

1. Management Science is an approach to management decision-making that makes extensive use of quantitative methods
2. This course aims to introduce students to the application of quantitative techniques to problems where models capture problem structure and use it to help optimize the decision outcome.
3. The classes demonstrate how advances in imputing power have made these techniques more accessible to managers and how the techniques can be applied to a range of different situations.
4. Provide a basic understanding of management science and engineering principles, including analytical problem solving and communications skills.
5. Prepare for practice in a field that sees rapid changes in tools, problems, and opportunities.

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Illustrate basic insights of management principles	Understand	PO6, PO11, PO12
CO2	Summarize Production process, Quality control and Inventory techniques	Analyze	PO8, PO10, PO11, PO12
CO3	Apply Strategies and policies to functional areas	Apply	PO8, PO10, PO11, PO12
CO4	Understand Contemporary management Practices	Understand	PO6, PO11, PO12

Unit-I:

Introduction to Management: Concept –nature and importance of Management –Generic Functions of Management – Principles of Management – Evolution of Management thought-

Theories of Motivation (Maslow's, hertz berg and X-Y Theory) – Decision making process- Designing organization structure- Principles of organization.

Outcome: Gives an outline of management and its nature scope and functions and hierarchical levels and organizational structure and managing the culture

Activity: Case study on motivation

Unit-II:

Operations Management: Plant location, Principles and Types of plant layout, production methods (job, batch mass production) – Work study- Statistical Quality Control- Control Charts (X Bar chart &R-charts, P-chart and C-chart) Simple problems- Material Management: Need for Inventory control- Tools and techniques of Inventory Control - EOQ, ABC analysis, HML, SDE, VED, and FSN analyses

Outcome: Able to understand the various functions of production and inventory management

Activity: Group discussion

Unit-III: Strategic Management: Vision, Mission, Goals, Strategy – Elements of Corporate Planning Process –Environmental Scanning – SWOT analysis- Steps in Strategy Formulation and Implementation, Generic Strategy, Alternatives. Global strategies, theories of Multinational Companies. **Project Management**: (PERT/CPM): Development of Network – Difference between PERT and CPM Identifying Critical Path- Probability (Problems)

Outcome:
Bring out various concepts of strategic management and project management
Activity:
SWOT analysis
Unit-IV: Functional Management: Concept of HRM, HRD and PMIR- Functions of HR Manager- Wage payment plans (Problems) – Job Evaluation and Merit Rating – Salient features of The Factories Act 1948 - Marketing Management, Marketing Mix strategies – Product, Price, Place and Promotion.

Outcome:
Elucidate the process of matching manager qualifications with position requirements and concept of marketing mix
Activity:
case study and role-play
Unit-V: Contemporary Management Practices: Basic concepts of MIS, MRP, Justin-Time(JIT) system, Total Quality Management(TQM), Six sigma and Capability Maturity Model(CMM) Levies, Supply Chain Management ,Enterprise Resource Planning (ERP), Business Process outsourcing (BPO), Business process Re-engineering and Bench Marking, Balanced Score Card.

Outcome: Gives outline of various contemporary issues of management

Activity: Debate on Contemporary Management Practices

Text Books:

1. Dr. P. Vijaya Kumar &Dr. N. Appa Rao, '*Management Science'* Cengage, Delhi, 2012.
2. Dr. A. R. Aryasri, *Management Science'* TMH 2011.

Reference Books:

1. Koontz &Weihrich: 'Essentials *of management'* TMH 2011
2. Seth &Rastogi: Global Management Systems, Cengage learning , Delhi, 2011
3. Robbins: Organizational Behaviour, Pearson publications, 2011
4. KanishkaBedi: Production & Operations Management, Oxford Publications, 2011
5. Philip Kotler & Armstrong: Principles of Marketing, Pearson publications
6. Biswajit Patnaik: Human Resource Management, PHI, 2011

Course Code	SOFTWARE ARCHITECTURE AND DESIGN PATTERNS	L	T	P	Credits
1005174202		3	0	0	3

Course Overview: This course introduces how to create and analyse the architecture by making use of different types of patterns

Course Objectives:
The course should enable the student:

- To understand interrelationships, principles and guidelines governing architecture and evolution over time.
- To understand various architectural styles of software systems.
- To understand design patterns and their underlying object oriented concepts.
- To understand implementation of design patterns and providing solutions to real world software design problems

Course Outcomes:

	Course Outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Classify the architecture ,create it and moving from one to many, different structural patterns	Understanding	PO1,PO2,PO3
CO2	Illustrate the architecture and build the system the components	Applying	PO1,PO3
CO3	Design creational and structural patterns	Creating	PO1,PO3,PO5
CO4	Outline about behavioural pattern and case study in uitlizing the architectural structures	Understanding	PO1,PO3,PO5

UNIT-I:
Envisioning Architecture The Architecture Business Cycle, What is Software Architecture, Architectural patterns, reference models, reference architectures, architectural structures and views.
Creating and Architecture Quality Attributes, Achieving qualities, Architectural styles and patterns, designing the Architecture, Documenting software architectures, Reconstructing Software Architecture.

Outcome:
To obtain knowledge about how to create an architecture

Activity:
- Take an example of any architecture and try to evaluate the qualities of that architecture.

UNIT-II:

Analyzing Architectures Architecture Evaluation, Architecture design decision making, ATAM, CBAM

Moving from One System to Many: Software Product Lines, Building systems from off the shelf components, Software architecture in future.

Outcome:

Develop and analyze the architecture by using ATAM and CBAM methods

Activity:

- Apply the methods and try to implement the future software architecture

UNIT-III:

Patterns: Pattern Description, Organizing catalogs, role in solving design problems, Selection and usage.

Creational Patterns: Abstract factory, Builder, Factory method, Prototype, Singleton

Outcome:

Outline about various design patterns

Activity:

use anyone of the creational pattern and link up with realtime example

UNIT-IV:

Structural Patterns: Adapter, Bridge, Composite, Decorator, Façade, Flyweight, PROXY

Behavioral Patterns: Chain of responsibility, command, Interpreter, iterator, mediator, memento, observer, state, strategy, template method, visitor

Outcome:

Outline about various design patterns

Activity:

- use anyone of the structural and behavioral pattern and link up with realtime example

UNIT-V:

Case Studies

A-7E – A case study in utilizing architectural structures, The World Wide Web - a case study in Interoperability, Air Traffic Control – a case study in designing for high availability, Celsius Tech – a case study in product line development.

A Case Study (Designing a Document Editor): Design Problems, Document Structure, Formatting, Embellishing the User Interface, Supporting Multiple Look-and-Feel Standards, Supporting Multiple Window Systems, User Operations, Spelling Checking and Hyphenation

Outcome:

- obtain knowledge about the architecture and design of a software

Activity

- use the appropriate method to devlop the entire architecture with a case study

Text Books:

1. Software Architecture in Practice, second edition, Len Bass, Paul Clements & Rick Kazman, Pearson Education,2003.
2. Design Patterns, Erich Gamma, Pearson Education, 1995.

Reference Books:

1. Beyond Software architecture, Luke Hohmann, Addison wesley,2003.
2. Software architecture, David M. Dikel, David Kane and James
3. R. Wilson, Prentice Hall PTR, 2001
4. Software Design, David Budgen, second edition, Pearson education,2003
5. Head First Design patterns, Eric Freeman & Elisabeth Freeman, O'REILLY,2007.
6. Design Patterns in Java, Steven John Metsker& William C. Wake, Pearson education, 2006
7. J2EE Patterns, Deepak Alur, John Crupi& Dan Malks, Pearson education,2003.
8. Design Patterns in C#, Steven John metsker, Pearson education,2004.
9. Pattern Oriented Software Architecture, F.Buschmann& others, John Wiley &Sons

ELECTIVE-III

Course Code		L	T	P	Credits
	DISTRIBUTED SYSTEMS				
1005174203		3	0	0	3

Course Overview: This course provides an introduction to the fundamentals of distributed computer systems, assuming the availability of facilities for data transmission, IPC mechanisms in distributed systems, Remote procedure calls and expose students to current technology used to build architectures to enhance distributed Computing infrastructures with various computing principles

Course Objectives:

- Explain difference between failure model and security model
- Give outline of resource sharing concept
- Explain client server communication.
- Extract the features of Internet protocols
- Describe important characteristics of distributed systems and the salient architectural
- Features of such systems

Course Outcomes:

	Course outcome	**Cognitive Level as per Bloom's Taxonomy**	**PO**
CO1	Develop distributed file system	Understand	PO1
CO2	Outline importance characteristics of distributed systems and the salient architectural features of such systems	Understand	PO1,PO4
CO3	Explain the features and applications of important standard protocols which are used in the distributed system	Applying	PO5
CO4	Interpret inter-process communication in a distributed system	Applying	PO2,PO12

Unit-I:

Characterization of Distributed Systems: Introduction, Examples of Distributed Systems, Resource Sharing and the Web, Challenges.
System Models: Introduction, Architectural Models- Software Layers, System Architecture, Variations, Interface and Objects, Design Requirements for Distributed Architectures, Fundamental Models- Interaction Model, Failure Model, Security Model.

Outcome:

- List out the different examples of distributed system
- Explain difference between failure model and security model

- Give outline of resource sharing concept

Activity: Distributed systems usage in real time examples and resource sharing usage.

Unit-II:

Inter process Communication: Introduction, The API for the Internet Protocols- The Characteristics of Inter process communication, Sockets, UDP Datagram Communication, TCP
Stream Communication; External Data Representation and Marshalling; Client Server Communication; Group Communication- IP Multicast- an implementation of group Communication, Reliability and Ordering of Multicast Comprehensions.

Outcome:

- Differentiate between TCP and UDP.
- Explain client server communication.
- Extract the features of Internet protocols

Activity: Practice Example of TCP and UDP communication

Unit-III:

Distributed Objects and Remote Invocation: Introduction, Communication between Distributed Objects- Object Model, Distributed Object Modal, Design Issues for RMI, Implementation of RMI, Distributed Garbage Collection; Remote Procedure Call, Events and Notifications, Case Study: JAVA RMI
Operating System Support: Introduction, The Operating System Layer, Protection, Processes and Threads –Address Space, Creation of a New Process, Threads.

Outcome:

- Able to Explain the process of RPC
- List the mechanisms for RMI
- Explain the processes and thread.
- List out the thread states

Activity: JAVA RMI with JVM example study, examples of thread execution

Unit-IV:

Distributed File Systems: Introduction, File Service Architecture; Peer-to-Peer Systems: Introduction, Napster and its Legacy, Peer-to-Peer Middleware, Routing Overlays.
Coordination and Agreement: Introduction, Distributed Mutual Exclusion, Elections, Multicast Communication.
Outcome:

- Develop a familiarity with distributed file systems.
- Describe important characteristics of distributed systems and the salient architectural features of such systems.

Activity: Different shortest path applications using graph based approach

Unit-V:

Transactions & Replications: Introduction, System Model and Group Communication, Concurrency Control in Distributed Transactions, Distributed Dead Locks, Transaction Recovery; Replication-Introduction, Passive (Primary) Replication, Active Replication.

Outcome:

- Describe important characteristics of Distributed Deadlocks along with passive and active replication.

Activity: ATM application, Railway Reservation examples

Text Books:

1. Ajay D Kshemkalyani, MukeshSighal, "Distributed Computing, Principles, Algorithms and Systems", Cambridge
2. George Coulouris, Jean Dollimore, Tim Kindberg, "Distributed Systems- Concepts and Design", Fourth Edition, Pearson Publication

Reference Books:

1. Distributed-Systems-Principles-Paradigms-Tanenbaum PHI
2. https://nptel.ac.in/courses/106/106/106106168/

Course Code	OPTIMIZATION TECHNIQUES	L	T	P	Credits
1005174204		3	0	0	3

Course Overview: The goal of this course is to formulate, analyze, and solve mathematical models that represent real-world problems. It covers different models: linear programming, network flow, inventory models and etc. Operations research has many applications in science, engineering, economics, and industry.

Course Objectives:

- Understand the meaning, purpose, and tools of Operations Research
- Make a graphical analysis of the linear programming problem and describe various solutions of Simplex Method
- Determine basic feasible solution of Transportation and Assignment methods
- Distinguish between different types of games and interpret the results from the payoff matrix of a two-person zero sum game
- Describe different factors affecting inventory management.
- Have an idea about the algorithm for a shortest path problem

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Formulate a physical problem into a mathematical model using linear programming problem and solve LPP using appropriate techniques like simplex method and its extensions	Applying	PO1, PO2, PO4, PO5
CO2	Build and Solve Transportation and Assignment models	Creating	PO3, PO4, PO5
CO3	Interpret how a task in a machine has to be processed in a particular order using sequencing Models and solve two-person zero sum games	Analyzing	PO1, PO2, PO4
CO4	Illustrate the use of Simple Inventory Models in Practical Situations	Understanding	PO1, PO2

Unit-I:

Introduction to Operations Research: Basics definition, scope, objectives, phases, models and limitations of Operations Research. Linear Programming Problem – Formulation of LPP, Graphical solution of LPP. Simplex Method, Artificial variables, big-M method, two-phase method, degeneracy and unbound solutions.

Outcome:

Understand the meaning, purpose and tools of operations research

Activity: formulate the physical problem into a mathematical model using linear programming problem

Unit-II:

Transportation Problem: Formulation, solution, unbalanced Transportation problem. Finding basic feasible solutions – Northwest corner rule, least cost method and Vogel's approximation method. Optimality test: the stepping stone method and MODI method

Assignment model: Formulation. Hungarian method for optimal solution. Solving unbalanced problem.

Outcome:

Determine basic feasible solutions of transportation and assignment methods

Activity: build transportation and assignment models

Unit-III:

Sequencing models: Solution of Sequencing Problem – Processing n Jobs through 2 Machines – Processing n Jobs through 3 Machines – Processing 2 Jobs through m machines

Games Theory: Competitive games, rectangular game, saddle point, minimax (maxi, min) method of optimal strategies, value of the game. Solution of games with saddle points, dominance principle. Rectangular games without saddle point – mixed strategy for 2 X 2 games

Outcome:

- Distinguish between different types of games and interpret the results from the payoff matrix of a two-person zero sum game

Activity: Game theory

Unit-IV:

Project planning and network analysis: Introduction and basic definitions in Network Analysis, Rules for drawing Network Analysis ,Critical Path Method (CPM) ,Project Evaluation and Review Technique (PERT)

Outcome:

- Describe different factors affecting inventory management.

Activity: CPM, PERT

Unit-V:

Inventory models: Inventory costs. Models with deterministic demand – model (a) demand rate uniform and production rate infinite, model (b) demand rate non-uniform and production rate infinite, model (c) demand rate uniform and production rate finite.

Outcome:

- Have an idea about the algorithm for a shortest path problem

Activity:

- Inventory models

Text Books:

1. Hiller, F.S. and Lieberman, G.J., Introduction to Operations Research (9th ed.), McGraw-Hill, 2009
2. P. SankaraIyer,"Operations Research", Tata McGraw-Hill, 2008.
3. A.M. Natarajan, P. Balasubramani, A. Tamilarasi, "Operations Research", Pearson Education, 2005.

Reference Books:

1. Winston, W.L., Introduction to Mathematical Programming (4th ed.), Duxbury Press, 2002
2. J K Sharma. "Operations Research Theory & Applications, 3e", Macmillan India Ltd, 2007.
3. P. K. Gupta and D. S. Hira, "Operations Research", S. Chand & co., 2007.

Course Code		L	T	P	Credits
1005174205	CONCURRENT AND PARALLEL PROGRAMMING	3	0	0	3

Course Overview:

This course approaches in two directions. In the first part, the abstractions used in concurrent programming are explored. This part will build the student's intuition on understanding and reasoning about concurrency and parallel programming. The second part of the course will focus on performance. It is imperative for programmers who write concurrent code to understand the fundamentals and the limitations of the underlying architectures. Different implementation models of concurrent programs also explored.

Course Objectives:

1. Learn the abstractions of concurrent and parallel programming.
2. Knowing how to use different synchronization techniques.
3. Understand the limitations of architectures, operating systems, and programming languages
4. Quickly explore the solution space and reduce the response time.

Course Outcomes:

	Course outcome	Cognitive Level as per Bloom's Taxonomy	PO
CO1	Understand the art of multi core programming	Understanding	PO1, PO2
CO2	Building scalable multithreaded applications and learn how to reduce the energy consumption of the multithreaded applications.	Creating/Applying	PO1, PO3, PO4, PO11, PO12
CO3	For better designs one has to know the usage of different performance monitoring tools.	Design	PO1, PO3, PO4
CO4	Understand the limitations of executing environment	Analysis	PO1, PO3, PO4

Unit-I:

Introduction: Shared objects and synchronization; parallel programming; Threads and monitors, sleeping; Interconnect, memory, caches, cache-conscious programming; Multicore and multithreaded architectures, hardware synchronization instructions.

Outcome: Able to understand the introduction of parallel programming
Activity: understand the art of multi core programming

Unit-II:

Abstractions: Mutual exclusion: solutions to two-thread problems; locks, fairness and timestamps; Concurrent objects: different forms of consistency, linearizability, progress and memory models of modern programming languages; Foundations of shared memory, correctness arguments; Synchronization primitives.

Outcome: Knowing how to use different synchronization techniques

Activity: abstraction and concurrent objects

Unit-III:

Implementation of spin locks, test-and-set locks, exponential backoff locks, composite and hierarchical locks. Performance optimization tools: perf and jRAPL.

Outcome: Understand the limitations of architectures, operating systems, and programming languages

Activity: Implementation of spin locks

Unit-IV:

Concurrent implementation of linked lists using coarse grained, fine grained, optimistic and lazy and non-blocking synchronization techniques; Concurrent implementation of queues and stacks; Linearization points.

Outcome:

Quickly explore the solution space and reduce the response time

Activity: implementation of linked list, queues, stacks

Unit-V:

Concurrent hashing and priority queues; Barriers and transactional memory.

Outcome: Understand the limitations of executing environment

Activity: concurrent hashing and priority queues

Text Books:

1. "The Art of Multiprocessor Programming" by Maurice Herlihy and NirShavit, Morgan Kaufmman Publishers, 1st Edition, Indian Reprint 2012.
2. "An Introduction to Parallel Programming" by Peter Pacheco, Morgan Kaufmann, 1st Edition, 2011

Reference Books:

1. "The Art of Concurrency: A Thread Monkey's Guide to Writing Parallel Applications" by O' Reilly Media, 1st Edition, 2009
2. "Java Concurrency in Practice" by Brian Goetz, Tim Peierls, Joshua Block, Joseph Bowbeer, David Holmes and Doug Lea, Addison Wesley, 1st Edition, 2006
3. Selected reading from papers published in conferences like MICRO, ISCA, HPCA, IPDPS, ICDCS, PODC, SC, ICS, PPoPP, PLDI, OOPSLA, ASPLOS and PACT.

Course Code		L	T	P	Credits
1005174251	**INTERNSHIP**	0	0	0	12

Internship: Internships help students to acquire in depth knowledge about a particular topic related to the program of study. Such extensive work is expected to create a platform for a job or further research in the chosen area. Interested students may opt for a full semester Internship during the fourth year second semester. Such students shall be exempted for equivalent theory course credits during that semester and the corresponding credits are awarded through the Internship. A self-study report, duly authorized by the industry supervisor / guide, shall be submitted at the end of the fourth year second semester. Internship report is evaluated for 400 marks in total. Internal assessment is done by the academic supervisor/guide for 100 marks based on the work and presentation of the internship report. The assessment for 300 marks is evaluated and awarded by a panel of members consisting of Head of the Department, Senior Faculty and Industry Expert.

Course Code		L	T	P	Credits
1005174251	TECHNICAL SEMINAR	0	0	0	2

Technical Seminar: For Technical seminar, the student shall collect the information on a specialized topic and prepare a technical report, showing his/her understanding over the topic, and submit to the department, which shall be evaluated by the Departmental Committee consisting of Head of the Department, seminar supervisor and a senior faculty member. The seminar report shall be evaluated for 50 marks. There shall be no external examination for seminar.

Course Code		L	T	P	Credits
	COMPREHENSIVE VIVA				
1005174261		**0**	**0**	**0**	**2**

Comprehensive Viva: The Comprehensive Viva aims to assess the students' understanding in various subjects he / she studied during the B.Tech course of study. Comprehensive Viva is conducted for a total of 50 marks. It shall be conducted in IV Year II Semester. The Comprehensive Viva–Voce will be conducted by a Committee consisting of Head of the Department, & senior faculty members of the Department.

Course Code		L	T	P	Credits
1005174231	MAIN PROJECT	0	0	0	10

Main Project: Out of total 200 marks for the project work, 80 marks shall be for Internal Evaluation and 120 marks for the external assessment. The Internal Evaluation shall be on the basis of two mid-term project reviews conducted during the progress of the project. The End Semester Examination (Viva-Voce) shall be conducted by the committee consists of an External Examiner, Head of the Department (internal examiner) and a senior faculty of the Department. The evaluation of project work shall be conducted at the end of the IV year.