SWARNANDHRA

COLLEGE OF ENGINEERING & TECHNOLOGY

(Autonomous)

ACADEMIC REGULATIONS, COURSE STRUCTURE & SYLLABUS

R20

For

Master of Computer Applications

Two year PG Course

(Applicable for batches admitted from 2020-21)

SWARNANDHRA

COLLEGE OF ENGINEERING & TECHNOLOGY

DEPARTMENT OF MASTER OF COMPUTER APPLICATIONS

SEETHARAMAPURAM, NARSAPUR-534 280, W.G.DT., A.P.

ACADEMIC REGULATIONS

1. INTRODUCTION

Swarnandhra College of Engineering & Technology (Subsequently referred to as SCET) will be followed the norms of Jawaharlal Nehru Technological University Kakinada and Govt. of Andhra Pradesh.

Academic Program of the institute are governed by rules and regulations approved by the Academic Council, which is the highest Academic body of the Institute. These academic rules and regulations are applicable for the students of MCA (Regular) course admitted from the academic year 2020-21 onwards.

The MCA Degree shall be conferred on candidates who are admitted to the program and fulfill all the requirements for the award of the Degree. Swarnandhra College of Engineering & Technology, an autonomous institution, follows Semester pattern for all the two years of its Postgraduate MCA programme with internal and external evaluation.

Semester Pattern: Each academic year shall be divided into two semesters: each semester consists of 22 weeks duration with a minimum of 110 working days which includes instruction, mid examinations and final examinations.

2. ADMISSIONS:

2.1 **Admission into first year MCA:** Admissions into first year of MCA Program of SCET will be as per the norms stipulated by Jawaharlal Nehru Technological University Kakinada & Govt. of Andhra Pradesh. Admissions into the program in the Institution are classified into CATEGORY – A, through convener, ICET and CATEGORY- B filled by the college management.

2.2 Admissions with advance standing:

(Transfer from other Colleges/ Re-admission due to dis-continuation)

These may arise in the following cases:

a) When a student seeks transfer from other colleges to SCET and desirous to pursue the study at SCET .

b) When students of SCET get transferred from one regulation to another regulation.

In all such cases, approval is mandatory from the statutory bodies

3. AWARD OF MCA DEGREE

3.1 A student shall be declared eligible for the award of MCA degree, if he pursues a course of study and completes it successfully for not less than two academic years and not more than four academic years.

3.2 A Student, who fails to fulfill all the academic requirements for the award of the degree with in four academic years from the year of their admission, shall forfeit his seat in MCA course.

3.3 For MCA course a student shall register for 80 credits and secure the same.

4. ATTENDANCE

4.1 A candidate shall be deemed to have eligibility to write End Semester examinations if he has put in a minimum of 75% of attendance in aggregate of all the subjects.

4.2 Condonation of shortage of attendance up to 10% i.e. 65% and above, and below 75% may be given by the College academic committee.

4.3 Condonation of shortage of attendance shall be granted only on genuine and valid reasons representations by the candidate with supporting evidence

4.4 Shortage of attendance below 65% shall in NO case be condoned

4.5 A candidate shall not be promoted to the next semester unless he fulfills the attendance requirements of the previous semester.

4.6 A stipulated fee shall be payable towards Condonation of shortage of attendance

5. DISTRIBUTION AND WEIGHTAGE OF MARKS:

5.1 The performance of the candidate in each semester shall be evaluated subject-wise, with a maximum of 100 marks for theory and Laboratory, on the basis of Internal Evaluation and End Semester Examination.

(a) External Evaluation

For the theory subjects 70 marks shall be awarded based on the performance in the End Examination Marks. External examination shall be conducted for duration of 180 minutes with 5 questions carrying 14 marks each. Each of these questions may contain sub questions. For each question there will be an “either” “or” choice, which means that there will be two questions from each unit and the student should answer either of the two questions.

(b) Internal Evaluation

30 marks shall be awarded based on the Internal Evaluation. Internal Evaluation shall be made based on the weighted Average of the marks secured in the two Mid Term –Examinations conducted, one in the middle of the Semester and the other immediately after the completion of instruction. The weights are 80% for the mid in which the students secured highest marks and 20% for mid in which the student secured lowest marks. Each mid examination shall be conducted for duration of 90 minutes with 3 questions to be answered out of 3 questions from two and half units (without choice) and each question for 10 marks

5.2 For practical subjects, 30 marks for Internal Evaluation and 70 for external examination. Out of 30 Internal marks 15 marks shall be awarded for day-to-day work including Record work and the remaining 15 marks to be awarded by conducting internal laboratory test. The External Laboratory examination for MCA course must be conducted with two examiners. One of them is the Laboratory Class Teacher, and the second examiner will be external examiner. External examiner will be appointed by the COE.

5.3 A Candidate shall be deemed to have secured the minimum academic requirement in a subject if he secures a minimum of 40% of marks in the End Examination and a minimum aggregate of 50% of the total marks in the End Semester Examination and Internal Evaluation taken together.

5.4 A Candidate shall be given one chance to re-register for each course provided the internal marks secured by a candidate are less than 50 per cent and he has failed in the end examination after completion of the two years. In such case, the candidate must reregister for the subject(s) and secure required minimum attendance. Attendance in the re-registered subject(s) should be calculated separately to become eligible to write the end examination in the re-registered subject(s). The attendance of re-registered subject(s) shall be calculated separately to decide his eligibility for taking the end examination in those subject(s). In the event of taking another chance, the internal marks and end examination marks obtained in the previous attempt are nullified. At a given time a candidate is permitted to re-register for a maximum of two subject(s). For re-registration the candidates have to apply to the Institute by paying the requisite fees and get approval from the concern authorities before the start of the semester in which re-registration is required. In case the candidate secures less than the required attendance in any re-registered course(s), he/she shall not be permitted to write the End Examination in that course.

5.5 A candidate shall be allowed to submit the project report only after fulfilling the attendance requirements of all the semesters. The viva–voce examination shall be conducted at the end of the course work (4th semester).

5.6 Mini Project/Internship guidelines: There shall be a Mini Project/Summer Internship, in collaboration with an industry of their specialization. Students will register for this immediately after II Semester examinations and purse it during summer vacation. Mini Project/Summer Internship shall be submitted in a technical report form and presented before the committee in II year I semester. It shall be evaluated for 50 internal marks. The Committee consists of Head of the Department, supervisor and senior faculty member of the department. A Minimum of 50% of maximum marks shall be obtained to earn the corresponding credits.

5.7 Employability Skills: This course is internal evaluation and will be evaluated for 50 marks. A minimum of 50% of maximum marks shall be obtained to earn the corresponding credits.

5.8 Bridge Course: The Course shall be taken on MOOCS platform/through college, the candidate has to get a satisfactory report upon successful completion.

5.9 MOOCs (NPTEL/SWAYAM) for Elective Papers: A student shall be permitted to pursue up to a maximum of two elective courses under MOOCs during the programme. Students are advised to register for only for minimum 12 weeks in duration MOOCs courses. Student has to pursue and acquire a certificate for a MOOC course only from the SWAY/NPTE through online with the approval of Head of the Department in order to earn the 3 credits. The Head of the department shall notify the list of such courses at the beginning of the semester. The Head of the Department shall appoint a mentor for each of the MOOC subjects registered by the students to monitor the student's assignment submissions given by SWAYAM/NPTEL. The student needs to submit all the assignments given and needs to take final exam at the proctor center. The student needs to earn a certificate by passing the exam. The student will be awarded the credits given in curriculum only by submission of the certificate. In case if student does not pass subjects registered through SWAYAM/NPTEL, the same or alternative equivalent subject may be registered again through SWAYAM/NPTEL in the next semester with the recommendation of HOD and shall be passed.

6 EVALUATION OF PROJECT WORK

6.1 Every candidate shall be required to submit thesis or dissertation after taking up a topic approved by the Project Review Committee.

6.2 A Project Review Committee (PRC) shall be constituted with Head of the Department and two other senior faculty members of the concerned department.

6.3 Registration of Project Work: A candidate is permitted to register for the project work after satisfying the attendance requirement of all the courses (theory and practical subjects) up to III semester.

6.4 After satisfying 6.3, a candidate has to submit, in consultation with his project supervisor, the title, objective and plan of action of his project work to the Project Review Committee for its approval. After obtaining the approval of the Committee the student can initiate the Project work after the third semester end examinations.

6.5 Every candidate shall work on projects approved by the PRC of the College

6.6 The duration of the project is for one semester.

6.7 If a candidate wishes to change his supervisor or topic of the project he can do so with approval of the PRC. However, the Project Review Committee (PRC) shall examine whether the change of topic/supervisor leads to a major change of his initial plans of project proposal. If so, his date of registration for the project work starts from the date of change of Supervisor or topic as the case may be.

6.8 A candidate shall submit status report in two stages at least with a gap of one month between them.

6.9 The work on the project shall be initiated in the beginning of the fourth semester and the duration of the project is for one semester. A candidate shall be allowed to submit the project report only with the approval of PRC and not earlier than 16 weeks from the date of registration of the project work. For the approval of PRC the candidate shall submit the draft copy of thesis to the Principal (through Head of the Department) and shall make an oral presentation before the PRC.

6.10 Three copies of the Project Thesis certified by the supervisor & HOD shall be submitted to the College / Department.

6.11 The project work carried out by the candidate during 4th semester is evaluated for internal assessment and external examination.

a) **Internal Assessment**: Internal Assessment will be carried out by the Project Review Committee consisting of 1) Head of the Department 2) Supervisor and 3) Senior faculty member. Internal Assessment shall be on the basis of two seminars given by the each student on the topic of his project.

b) **External Examination**: External Examination (Viva – Voce) will be conducted by Project External Examination committee consisting of 1) Head of the Department 2) Supervisor and 3) External member. External examiner will be appointed by the COE from the panel of examiners submitted by the HOD.

6.12 Out of a total of 200 marks for the project work, 50 marks shall be for internal assessment and 150 marks External examination (Viva-Voce). A minimum of 50% of maximum marks shall be obtained to earn the corresponding credits.

6.13 If he/she fails to secure those marks he/she will retake the viva-voce examination after three months. If he/she fails to secure those marks at this second viva-voce examination, he will not be eligible for the award of the degree unless the candidate is asked to revise and resubmit. If he/she fails to secure those marks again, the project shall be summarily rejected. Head of the Department shall coordinate and make arrangements for the conduct of Viva-Voce examination.

SEMESTER WISE CREDITS DISTRIBUTION

Semester	Theory	Lab	Total Credits
I Semester	5	3	20
II Semester	5	4+Bridge Course	20
III Semester	5	3+Internship/Mini Project	22
IV Semester	2	Project	18
Total			80

7 GRADING SYSTEM

7.1 Award of Grade:

(i) Grade Point Average (GPA):

a) The Grade Point Average (GPA) will be calculated according to the formula.

GPA = $\frac{\sum C_i G_i}{\sum C_i}$

Where C_i = number of credits for the subject i

G_i = grade points obtained by the student in the subject.

b) To arrive at Cumulative Grade Point Average (CGPA), the formula is used considering the student's performance in all the courses taken in all the semesters completed up to the particular point of time.

CGPA = $\frac{\sum C_i G_i}{\sum C_i}$

Where C_i = number of credits for the subject i

G_i = grade points obtained by the student in the subject.

(ii) After a student satisfies the requirements prescribed for the award of UG/PG Program he/she shall be placed in one of the following four grades. The award of the degree is based on CGPA on a grade point scale of 10.

CGPA	Award of Division
≥ 7.75	First Class with Distinction
≥ 6.75	First Division
≥ 5.75	Second Division
< 5.75	Unsatisfactory

7.1 **Award of Grade in Each Semester:**

i) Based on the student performance during a given semester, a final letter grade will be awarded at the end of the semester for each subject. The letter grades and the corresponding grade points are as given in the Table.

Percentage of Marks Scored	**Letter Grade**	**Level**	**Grade Points**
>=90	O	Outstanding	10
80– 89	S	Excellent	9
70-79	A	Very Good	8
60-69	B	Good	7
50-59	C	Fair	6
< 50	F	Fail	0
	AB	Absent	0

ii) A student earns a minimum of 6 grade points (D grade) in a subject is declared to have successfully completed the subject, and is deemed to have earned the credits assigned to that subject. However it should be noted that a pass in any subject/term paper/project/mini project/labs shall be governed by the rules mentioned in against them

iii) Grade Sheet: A grade sheet (memorandum) will be issued to each student indicating his/her performance in all courses taken in that semester and also indicating the grades.

iv) Transcripts: After successful completion of the total programme of study, a Transcript containing performance of all academic years will be issued as a final record. Duplicate transcripts will also be issued up to any point of study to any student on request and by paying the stipulated fee in force.

v) Candidates shall be permitted to apply for recounting/revaluation within the stipulated period with payment of prescribed fee.

vi) The Academic Council has to approve and recommend to the JNTUK, Kakinada for the award of a degree to any student.

8 WITH HOLDING OF RESULTS:

If the candidate has not paid any dues to the College or if any case of indiscipline is pending against him, the result of the candidate will be withheld. The issue of degree is liable to be withheld in such cases.

9 TRANSISTORY REGULATIONS

Candidates who have discontinued or have been detained for want of attendance or who have failed after having undergone the course are eligible for admission to the same or equivalent subjects as and when subjects are offered.

GENERAL

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

9.2 In case of any doubt or ambiguity in the interpretation of the above rules, the decision of the Principal is final and which is to be ratified by the Chairman of the Governing Body.

9.3 The College may change or amend the academic regulations and syllabus at any time and the changes and amendments made shall be applicable to all the students with effect from the date notified by the college.

9.4 Wherever the word he, him or his occur, it will also include she, her and hers.

Course Structure
I SEMESTER

S.NO	Course Code	Course Title	L	P	C	IM	EM	TM
1	20MC1T01	Data Structures	3		3	30	70	100
2	20MC1T02	Computer Organization	3		3	30	70	100
3	20MC1T03	Database Management Systems	3		3	30	70	100
4	201MCT04	Operating Systems	3		3	30	70	100
5	20MC1T05	Probability and Statistical Applications	3		3	30	70	100
6	20MC1L01	Database Management Systems Lab		3	1.5	30	70	100
7	20MC1L02	Data Structures using C Lab		4	2	30	70	100
8	20MC1L03	Operating Systems and Linux Lab		3	1.5	30	70	100
		Total	15	10	20	240	560	800

II SEMESTER

S.NO	Course Code	Course Title	L	P	C	IM	EM	TM
1	20MC2T01	Computer Networks	3		3	30	70	100
2	20MC2T02	Data Warehousing and Data Mining	3		3	30	70	100
3	20MC2T03	Object oriented Programming using JAVA	3		3	30	70	100
4	20MC2T04	Object Oriented Software Engineering	3		3	30	70	100
5	---	**Program Elective-1**	3		3	30	70	100
6	20MC2L01	JAVA Programming Lab		3	1.5	30	70	100
7	20MC2L02	UML Lab		3	1.5	30	70	100
8	20MC2L03	Computer Networks Lab		3	1.5	30	70	100
9	20MC2L04	Employability Skills		1	0.5	50	-	50
10	20MC2B01	Bridge Course(Python Programming to be taken through MOOCs/College)	2	-	-	-	-	-
		Total	15	10	20	290	560	850

III SEMESTER

S.NO	Course Code	Course Title	L	P	C	IM	EM	TM
1	20MC3T01	Machine Learning with Python	3		3	30	70	100
2	20MC3T02	Web Technologies	3		3	30	70	100
3	20MC3T03	Principles of Cryptography and Network Security	3		3	30	70	100
4	20MC3T04	Human Resource Management	3		3	30	70	100
5	---	**Program Elective-2**	3		3	30	70	100
6	20MC3L01	Machine Learning with Python Lab		3	1.5	30	70	100
7	20MC3L02	Web Technologies Lab		4	2	30	70	100
8	20MC3L03	Network Security Lab		3	1.5	30	70	100
9	20MC3P01	Internship / Mini Project			2	50	-	50
Total			15	10	22	290	560	850

# This can be done during Semester break and evaluated at the end of 3rd Semester

IV SEMESTER

S.NO	Course Code	Course Title	L	P	C	IM	EM	TM
1	---	**Program Elective - 3**	3	0	3	30	70	100
2	---	**Program Elective -4**	3	0	3	30	70	100
3	20MCP02	Project Work / Dissertation			12	50	150	200
		Total			18	110	290	400

Program Elective-1

S. NO	Course Code	Course Title
1	20MC2TE1	Human Computer Interaction
2	20MC2TE2	Design and Analysis of Algorithms
3	20MC2TE3	No SQL Databases
4	20MC2TE4	Mobile Application Development
5	20MC2TE5	Internet of Things

Program Elective-2

S. NO	Course Code	Course Title
1	20MC3TE1	Cloud Computing
2	20MC3TE2	Software Project Management
3	20MC3TE3	Artificial Intelligence
4	20MC3TE4	E-Commerce
5	20MC3TE5	Cyber Security

Program Elective-3

S. NO	Course Code	Course Title
1	20MC4TE1	Digital Marketing
2	20MC4TE2	Block Chain Technologies
3	20MC4TE3	Deep Learning
4	20MC4TE4	Software Testing Methodologies
5	20MC4TE5	MOOCs-1 (NPTEL/SWAYAM) 1. Full Stack Technologies 2. Any recommended course

Program Elective-4

S. NO	Course Code	Course Title
1	20MC4TE6	Big Data Analytics
2	20MC4TE7	Soft Computing
3	20MC4TE8	Software Defined Networks
4	20MC4TE9	Network Programming

I Semester	L	P	C
	3	0	3

20MC1T01 DATA STRUCTURES

5	20MC4TE10	MOOCs-2 (NPTEL/SWAYAM) 1. Data Science 2. Any recommended course

Course Objectives:

The objective of this course is to explore basic data structures such as stacks and queues, introduce a variety of data structures such as hash tables, search trees, tries, heaps, graphs, sorting and pattern matching algorithms

Course Outcomes (CO): *At the end of the course, student will be able to*

Course Outcomes		Knowledge Level (K)#
CO1	Implement basic programs by using C concepts.	**K2**
CO2	Implement C Program using Functions, Structures and Unions, Pointers	**K3**
CO3	Design advanced Data Structures using Non Linear Data Structures	**K4**
CO4	Create Hash Table for storing data	**K4**
CO5	Apply appropriate Sorting technique for a problem	**K5**

UNIT - I :

Introduction to C: Constants and variables, Operators and Expressions, Managing Input and Output operators, Decision making-branching and looping, Arrays.

UNIT-II:

Functions, Structures and Unions, Pointers File handling in C

UNIT – III :

Data structure: Definition, types of data structures Recursion Definition, Design Methodology and Implementation of recursive algorithms. Linear list – Singly linked list, Double linked list and circular linked list- implementation, insertion, deletion and searching operations on linear list.

UNIT - IV :

Stacks-Operations, array and linked representations of stacks, stack applications, **Queues-operations**, array and linked representations.

Hash Table Representation: Hash functions, collision resolution-separate chaining, open addressing-linear probing, quadratic probing, double hashing and rehashing, extendible hashing.

UNIT - V:

Sorting Techniques: Insertion sort, selection sort, exchange-bubble sort, quick sort and merge sort Algorithms. **Trees**: Binary Trees, terminology, representation and traversals- pre, post & in order traversals. **Search Trees:** Binary Search Trees, Definition, Implementation, Operations-Searching, Insertion and Deletion

Text Books:

1. E. Balaguruswamy, Programming in ANSI C 8th ed. TMH, 2010
2. Let Us C: Authentic Guide to C Programming Language, 17th ed., Yashavant Kanetkar, BPB Publications, 2016
3. Data Structures Using C. 2nd Edition, Reema Thareja, Oxford
4. Data Structures and Algorithm Analysis in C, 2nd ed, Mark Allen Weiss

Reference Books:

1. Data Structures: A Pseudocode Approach with C, 2nd Edition, R. F. Gilberg and B.A. Forouzan, Cengage Learning.
2. Programming in ANSI C, 5th ed, E. Balaguruswamy, TMH Data Structures using C – A. S. Tanenbaum, Y. Langsam, and M.J. Augenstein, PHI/Pearson Education.
3. Let Us C: Authentic Guide to C Programming Language, 17th ed., Yashavant Kanetkar, BPB Publications.

I Semester	L	P	C
	3	0	3
20MC1T02 COMPUTER ORGANIZATION			

Course Objectives

The objectives of this course are to

1. Conceptualize the basics of organizational and architectural issues of a digital computer.
2. Learn the function of each element of a memory hierarchy.
3. Study various data transfer techniques in digital computer.

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

Course Outcomes		Knowledge Level (K)#
CO1	Understand the basic organization of computer and different instruction formats and addressing modes	**K2**
CO2	Analyze the concept of pipelining, segment registers and pin diagram of CPU.	**K4**
CO3	Understand and analyze various issues related to memory hierarchy	**K2**
CO4	Evaluate various modes of data transfer between CPU and I/O devices	**K5**
CO5	Examine various inter connection structures of multi processors	**K4**

UNIT I:

Basic Structure Of Computers: Computer Types, Functional units, Basic Operational concepts, Bus structures, Software, Performance, multiprocessor and multi computers, Historical perspective.

UNIT II:

Machine Instructions and Programs: Numbers, Arithmetic Operations, and Characters, Memory locations and addresses, Memory operations, Instructions and Instruction sequencing, Addressing Modes, Assembly Languages, stacks and Queues Basic Input/output Operations, role of Stacks and Queues Additional Instructions

UNIT III:

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

UNIT IV:

The Memory Systems: Some Basic concepts, Semiconductor RAM memories, Memory System Consideration, Read-Only Memories**,** Speed, Size, and cost, Cache Memories, Performance considerations, Virtual Memories, Memory Management Requirements, Secondary Storage

UNIT IV:

Large Computer Systems: Forms of Parallel Processing, Array Processors, Structure of General-Purpose Multiprocessors, Interconnection of Networks

Text Books:

1. Computer Organization, Carl Hamacher, Zvonks Vranesic, Safea Zaky, 5th Edition, McGraw Hill.
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, Sivarama Dandamudi Springer Int. Edition.

I Semester	**L**	**P**	**C**
	3	**0**	**3**
20MC1T03 DATABASE MANAGEMENT SYSTEMS			

Course Objectives:

This Course will enable students to

- Explain the concept of databases, database management systems, database structures and how they work.
- Make use of Entity-Relationship Modeling and Relational Modeling for creating simple databases from the real world scenarios.
- Write relational algebra and structured query language (SQL) statements.
- Normalize a database using Normalization Rules.
- Discuss the issues associated with Transaction Management and Recovery, Tree Structured and Hash-Based Indexing

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

CO	**Course Outcomes**	**Knowledge Level (K)#**
CO1	Illustrate the concept of databases, database management systems, database languages, database structures and their work	K2
CO2	Apply ER modeling and Relational modeling for designing simple databases.	K3
CO3	Summarize the concepts related to relational model and SQL and Write database queries using relational algebra and structured query language.	K2
CO4	Design and develop databases from the real world by applying the concepts of Normalization.	K6
CO5	Outline the issues associated with Transaction Management and Recovery, Tree Structured and Hash-Based Indexing	K2

Unit-I:

Introduction to Databases: Introduction, An Example, Characteristics of the Database Approach, Actors on Scene, Workers behind the scene, Advantages of Using the DBMS Approach, A Brief History of Database Applications, When Not to Use a DBMS **[TB-3]**

Overview of Database Languages and Architectures: Data Models, Schemas and Instances, Three-Schema Architecture and Data Independence, Database Languages and Interfaces, The Database System Environment, Centralized and Client/Server Architecture for DBMSs, Classification of Database Management Systems **[TB-3]**

Unit-II:

Introduction to Database Design: Database Design and ER Diagrams, Entities, Attributes and Entity Sets, Relationships and Relationship Sets, Additional Features of the ER Model, Conceptual Design with the ER Model, Conceptual Design for Large Enterprises **[TB-1]**

Relational Model: Introduction to the Relational Model, Integrity Constraints over Relations, Enforcing Integrity Constraints, Querying Relational Data, Logical Database Design: ER to Relational, Introduction to Views, Destroying/Altering Tables and Views **[TB-1]**

Unit-III:

Relational Algebra: Selection and Projection, Set Operations, Renaming, Joins, Division, More Examples of Algebra Queries **[TB-1]**

SQL: Queries, Constraints, Triggers: The Form of a Basic SQL Query, UNION, INTERSECT and EXCEPT, Nested Queries, Aggregate Operators, Null Values, Complex Integrity Constraints in SQL, Triggers and Active Databases, Designing Active Databases **[TB-1]**

Unit-IV:

Introduction to Normalization Using Functional and Multivalued Dependencies: Informal Design Guidelines for Relation Schema, Functional Dependencies, Normal Forms Based on Primary Keys, General Definitions of Second and Third Normal Forms, Boyce-Codd Normal Form, Multivalued Dependency and Fourth Normal Form, Join Dependencies and Fifth Normal Form **[TB-3]**

Unit-V:

Transaction Management and Concurrency Control: Transaction Concept, A Simple Transaction Model, Storage Structure, ACID Properties, Serializability, Transaction Isolation Levels, Concurrency Control, Lock-Based Protocols, Validation- Based Protocols **[TB-2]**

Note: For Practical Examples Please Go Through Reference 1

Text Books:

1. Data base Management Systems, 3/e, Raghurama Krishnan, Johannes Gehrke, Mc Graw-Hill
2. Data base System Concepts, 6/e, Abraham Silberschatz, Henry F. Korth, S. Sudarshan, Mc Graw-Hill
3. Database Systems, 6/e Ramez Elmasri, Shamkant B. Navathe, Pearson

Reference Books:

1. Database Systems, 9/e, Carlos Coronel, Steven Morris, Peter Rob, Cengage
2. Introduction to Database Systems, 8/e, C J Date, Pearson

I Semester	L	P	C
	3	0	3
20MC1T04 OPERATING SYSTEMS			

Course Objectives:

This course enables the student to

- Introduce different types of operating systems.
- Learn process management techniques.
- Learn various memory management techniques.
- Introduce the architecture of Linux operating system.
- Learn multiple operating system like Unix and Windows.

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

Course Outcomes		Knowledge Level (K)#
CO1	Understand the basics of operating systems like kernel, shell, types and views of operating systems	**K2**
CO2	Understands CPU scheduling algorithms and compare the results using Gantt chart.	**K5**
CO3	Explain various memory management techniques and concept of thrashing	**K2**
CO4	Apply disk scheduling algorithms for better utilization of external memory	**K3**
CO5	Understand the architecture of UNIX operating system	**K1**
CO6	Write and execute shell programs	**K1**

UNIT–I:

Introduction to Operating System Concept: Types of Operating Systems, Operating Systems Concepts, Operating System Operations. Operating Systems Structures- Operating System Services, User Operating-System Interface, Introduction to System calls, Types of System Calls.

UNIT–II:

Process Management: Process concept, Process State Diagram, Process control block, Process Scheduling, Inter process Communication, Threads-Threading Issues, Scheduling- Basic Concepts, Scheduling Criteria, Scheduling Algorithms.

UNIT-III:

Process Synchronization: The Critical-Section Problem, Peterson's Solution, Synchronization Hardware, Semaphores, Classic Problems of Synchronization, Monitors, **Principles of deadlock:** System Model, Deadlock characterization, Deadlock handling, Deadlock Prevention, Detection and Avoidance, Recovery Starvation, Critical Regions form Deadlock

UNIT–IV:

Memory Management: Swapping, Contiguous Memory Allocation, Paging, structure of the Page Table, Segmentation Virtual Memory Management-Demand Paging, Page-Replacement Algorithms, Thrashing.

File-System Interface: File Concept, Access Methods, Directory structure, File-System mounting, Files Sharing, Protection. File-System implementation-File-System Structure, Allocation Methods, Free-Space Management, Disk Structure, Disk Scheduling

UNIT–V:

Case Studies: Linux System: Design Principles, kernel Modules, Process Management, File Systems, Input and Output, Interprocess Communication, Network Structure, Security. Windows7: Design Principles, System Components, Terminal Services and Fast User, File System, Networking, Programmer Interface.

Text Books:

1. Operating system concepts, Abraham Siliberschatz, Peter Baer Galvin, Greg Gagne, John Wiley & Sons, Inc., Edition 9, 2011
2. Introduction to UNIX and Shell Programming, M. G. Venkateshmurhty, Pearson, 2005
3. UNIX & Shell Programming by B.M. Harwani, OXFORD University Press, 2013

Reference Books:

1. Advanced Programming in the UNIX Environment by W. Richard Stevens, Stephen Rago, Wesley Professional, 2013
2. UNIX Network Programming by W. Richard Stevens, 1990
3. Operating systems, William stallings, PHI/Pearson, 6/E, 2009
4. Operating systems, Dietal, Dietal, Pearson, 3/e, 2007
5. Operating systems, Dhamdhere, TMH, 2/e, 2009

Web Reference:
https://onlinecourses.swayam2.ac.in/cec20_cs06/preview

I Semester	L	P	C
	3	0	3
20MC1T05 PROBABILITY AND STATISTICAL APPLICATIONS			

Course Objectives:

This course enables the student to

- Understand the concept of probability, random variables and their distributions.
- Understand the concept of estimation and hypothesis testing for population averages and percentages.
- Select and produce the appropriate tabular and graphical formulas for displaying bivariate
- Data sets and carry out correlation, regression and chi-square analyses.

Course Outcomes:

After the completion of the course, student will be able to

CO	Course Outcomes	Knowledge Level (K)#
CO1	Understand and apply the concepts of probability and statistics to solve a range of problems	K1
CO2	Different problems, and understand their applications in a variety of situation	K2
CO3	Construct the probability distribution of random variables, based on real-world situation, and use it to compute expectation and variance.	K4
CO4	Identify the components of statistical hypothesis and can apply them to solve real life problems.	K3
CO5	Estimate the extent of correlation and Regression between bivariate data	K3

Unit-I

Probability: Sample space and events–Probability–The axioms of probability – some Elementary theorems–Conditional Probability–Baye's theorem

Unit-II PROBABILITY DISTRIBUTIONS

Random Variables: Discrete and continuous – distributions – Distribution function. Binomial, Poisson, Normal distributions – related properties

Unit-III MOMENT GENERATING FUNCTIONS AND SAMPLING DISTRIBUTIONS

Expectations and higher order moments. Moment generating function and characteristic function. Population and samples, Sampling distribution of mean (with known and unknown variance), proportion, Variances, estimation.

Unit-IV INFERENCE THEORY

Statistical Inference: Introduction to null hypothesis verses alternative hypothesis. Statistical Hypothesis – Errors of type I and type II. One tail, two tail tests. Procedure for testing of hypothesis. Test of significance for large sampling using Z-test(i.e single mean, difference of means & proportions.)

Test of Hypothesis (Small Samples) using student’s - t test and F-test. chi-square test-Test of independence of attributes and goodness of it.

Unit-V CORRELATION AND REGRESSION

Types of Correlations – Positive, Negative Correlations – Methods of Studying Correlation – Carl Pearsons Coefficient and Correlation rank method - Correlation of Grouped data and Regression Analysis.

Text Books:

1. Probability and Statistics by Mr. K.Murugesan and Mr. P.Gurusamy
2. Probability and Statistics for Engineers and scientists by R.E. Walpole and Raymond H. Myers.

References:

1. Probability and Statistics for Engineers by Miller & Freund’s
2. Higher Engineering Mathematics by Dr. B.S. Grewal.

I Semester	L	P	C
	0	3	1.5
20MC1L01 DATABASE MANAGEMENT SYSTEMS LAB			

Course Objectives:

This Course will enable students to

- Populate and query a database using SQL DDL/DML Commands.
- Declare and enforce integrity constraints on a database
- Writing Queries using advanced concepts of SQL
- Programming PL/SQL including procedures, functions, cursors and triggers

Course Outcomes:

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

CO	Course Outcomes	Knowledge Level (K)#
CO1	Utilize SQL to execute queries for creating database and performing data manipulation operations	K3
CO2	Examine integrity constraints to build efficient databases	K4
CO3	Apply Queries using Advanced Concepts of SQL	K3
CO4	Build PL/SQL programs including stored procedures, functions, cursors and triggers.	K6

1. Execute all DDL, DML and DCL commands on sample tables.
2. Implementation of different types of operators and built-in functions with suitable examples
3. Implementation of different types of joins with suitable examples
4. Create views, partitions, Sequence, Indexes and locks for a particular DB
5. Implement different types of constraints on relations.
6. Implementation of sub queries and nested queries.
7. Implement Queries on Group By & Having Clauses, ALIAS, Sequence By, Order By
8. Control Structure
   a) Write a PL/SQL block for Addition of Two Numbers
   b) Write a PL/SQL block for IF, IF and else condition
   c) Write a PL/SQL block for implementation of loops
   d) Write a PL/SQL block for greatest of three numbers using IF ANDELSEIF

9.Exception Handling- Implement the following with respect to exception handling. Raising Exceptions, User Defined Exceptions, Pre-Defined Exceptions

10. Write PL/SQL block for an application using exception handling Procedures
    a) Write a PL/SQL Procedure using Positional Parameters
    b) Write a PL/SQL Procedure using notational parameters
    c) Write a PL/SQL Procedure for GCD Numbers

d) Write a PL/SQL Procedures for cursor implementation (explicit and implicit cursors)

11. Functions:
    - a) Write a PL/SQL block to implement factorial using functions
    - b) Write a PL/SQL function to search an address from the given database

12. Write a DBMS program to prepare Pl/SQL reports for an application using functions.

13. Triggers:
    - a) Write a Trigger to pop-up the DML operations
    - b) Write a Trigger to check the age valid or not Using Message Alert.
    - c) Create a Trigger to Raise appropriate error code and error message.
    - d) Create a Trigger on a table so that it will update another table while inserting values

14. Write PL/SQL block for an application using cursors and all types of triggers.

15. Write a PL/SQL block for transaction operations of a typical application using package

Text Books / Suggested Readings:

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

I Semester	L	P	C
	0	3	1.5
20MC1L02 DATA STRUCTURES LAB			

Course Objectives: This Course will enable students to

- Design and implement various data structures.
- Implement operations like searching, insertion, and deletion, traversing mechanism
- Develop applications using data structure algorithms.

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

Course Outcomes		Knowledge Level (K)#
CO1	Implement various basic data structures and its operations.	**K2**
CO2	Apply sorting and searching algorithms to given numbers	**K3**
CO3	Implement various tree operations.	**K2**
CO4	Implement various graphs algorithms.	**K2**
CO5	Develop applications using various data structures.	**K6**

Experiment 1:

a) Write a program in C to display the n terms of even natural number and their sum.
b) Write a program in C to display the n terms of harmonic series and their sum. 1 + 1/2 + 1/3 + 1/4 + 1/5 ... 1/n terms.
c) Write a C program to check whether a given number is an Armstrong number or not.
d) Write a C program to calculate the factorial of a given number.

Experiment 2:

a) Write a program in C for multiplication of two square Matrices.
b) Write a program in C to find transpose of a given matrix.

Experiment 3:

a) Write a program in C to check whether a number is a prime number or not using the function.
b) Write recursive program which computes the nth Fibonacci number, for appropriate values of n.
c) Write a program in C to add numbers using call by reference.

Experiment 4:

a) Write a program in C to append multiple lines at the end of a text file.
b) Write a program in C to copy a file in another name.

Experiment 5:

Write recursive program for the following

a) Write recursive and non recursive C program for calculation of Factorial of an integer.
b) Write recursive and non recursive C program for calculation of GCD (n, m)

c) Write recursive and non recursive C program for Towers of Hanoi: N disks are to be transferred from peg S to peg D with Peg I as the intermediate peg.

Experiment 6:

a) Write C program that use both recursive and non recursive functions to perform Linear search for a Key value in a given list.
b) Write C program that use both recursive and non recursive functions to perform Binary search for a Key value in a given list.

Experiment 7:

a) Write C program that implement stack (its operations) using arrays.
b) Write C program that implement stack (its operations) using Linked list.

Experiment 8:

a) Write a C program that uses Stack operations to convert infix expression into postfix expression.
a) Write C program that implement Queue (its operations) using arrays.
b) Write C program that implement Queue (its operations) using linked lists.

Experiment 9:

Write a C program that uses functions to create a singly linked list and perform various operations on it.

Experiment 10:

Write a C program to store a polynomial expression in memory using linked list and perform polynomial addition.

Experiment 11:

a) Write a recursive C program for traversing a binary tree in preorder, inorder and postorder.
b) Write a non recursive C program for traversing a binary tree in preorder, inorder and postorder.

Experiment 12:

a) Write a C program to implement Prims' algorithm.
b) Write a C program to implement Kruskal's algorithm.

Experiment 13:

Implementation of Hash table using double hashing as collision resolution function.

Experiment 14:

Implementation of Binary Search trees- Insertion and deletion.

Experiment 15:

Implementation of AVL Tree **–** Insertion and Deletion

Experiment 16:

a) Write C program that implement Bubble sort, to sort a given list of integers in ascending order.
b) Write C program that implement Quick sort, to sort a given list of integers

in ascending order.

c) Write C program that implement merge sort, to sort a given list of integers in ascending order

I Semester	L	P	C
	0	3	1.5
20MC1L03 OPERATING SYSTEMS LAB			

Course Objectives:
This Course will enable students to implement CPU scheduling algorithms, Disk scheduling algorithms, Execute different types of Linux commands and Write shell scripts

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

Course Outcomes		Knowledge Level (K)#
CO1	Implement various CPU scheduling algorithms and compare results	**K5**
CO2	Implement various disk scheduling algorithms and compare results	**K5**
CO3	Implement page replace algorithms	**K2**
CO4	Implement various memory management techniques.	**K2**
CO5	Execute basic Linux commands	**K1**

List of Experiments:

UNIX Lab- Introduction to Unix

1. Study of Unix/Linux general purpose utility commands
2. Study of Bash shell, Bourne shell and C shell in Unix/Linux operating system .
3. Study of UNIX/LINUX File System(tree structure).
4. C program to emulate the UNIX ls –l command
5. C program that illustrates how to execute two commands concurrently with a command pipe. Ex: - ls –l | sort
6. Multiprogramming-Memory management-Implementation of fork (), wait (), exec() and exit (), System calls

Operating Systems Lab

1. Simulate the Following CPU Scheduling Algorithms
   A) FCFS B) SJF C) Priority D) Round Robin
2. Multiprogramming-Memory Management- Implementation of fork(), wait(), exec() and exit()
3. Simulate The Following
   a) Multiprogramming with A Fixed Number Of Tasks (MFT
   b) Multiprogramming with A Variable Number Of Tasks (MVT)
4. Write a program to implement first fit, best fit and worst fit algorithm for memory management.
5. Simulate Bankers Algorithm for Dead Lock Avoidance

6. Simulate Bankers Algorithm for Dead Lock Prevention.
7. Simulate The Following Page Replacement Algorithms.
   a) FIFO b) LRU c) LFU
8. Simulate the Following File Allocation Strategies
   a) Sequenced b) Indexed c) Linked

Linux Lab

1. Write a Shell program to check whether given number is prime or not.
2. Write a shell script which will display Fibonacci series up to the given range.
3. Write a shell script to check whether the given number is Armstrong or not.
4. Write a shell script to the calculate the value of
5. Write a shell script to accept student number, name, marks in 5 subjects.
6. Find total, average and grade using the following rules:
   Avg>=80 then grade A
   Avg<80&&Avg>=70 then grade B
   Avg<70&&Avg>=60 then grade C
   Avg<60&&Avg>=50 then grade D
   Avg<50&&Avg>=40 then grade E
7. Write a shell script to find minimum and maximum elements in the given list of elements.
8. Write a shell program to check whether the given string is palindrome or not.
9. Write an awk program to print sum, avg of students marks list
10. Write a shell script to compute no. of characters and words in each line of given file

II Semester	L	P	C
	3	0	3
20MC2T01 COMPUTER NETWORKS			

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

- To Understands the fundamental concepts of computer networking and OSI Reference model.
- To Familiarize the student with the basic taxonomy and terminology of the computer networking area.
- To learn and understand the advanced networking concepts, preparing the student for entry advanced courses in computer networking.
- To develop and gain expertise in some specific areas of networking such as the design and maintenance of individual networks.

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

Course Outcomes		Knowledge Level (K)#
CO1	Explain the network architecture, TCP/IP and OSI reference models	**K2**
CO2	Identify and understand various techniques and modes of transmission	**K3**
CO3	Demonstrate the data link protocols, multi-channel access protocols and IEEE 802 standards for LAN	**K2**
CO4	Describe routing and congestion in network layer with routing algorithms and classify IPV4 addressing scheme	**K5**
CO5	Discuss the elements and protocols of transport layer	**K6**
CO6	Develop network security and define various protocols such as FTP, HTTP, Telnet, DNS	**K3**

UNIT – I

Introduction: Network Topologies WAN, LAN, MAN. Reference models- The OSI Reference Model- the TCP/IP Reference Model - A Comparison of the OSI and TCP/IP Reference Models. **Physical Layer –**Introduction to physical layer-Data and Signals, Periodic analog signals, digital signals, transmission impairment, Data rate limits, performance -Introduction to Guided Media-Twisted-pair cable, Coaxial cable and Fiber optic cable and Unguided media: Wireless-Radio waves, microwaves, infrared.

Unit-II

The Data Link Layer - Services Provided to the Network Layer – Framing – Error Control – Flow Control, Error Detection and Correction – Error-Correcting Codes – Error Detecting Codes. **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-A One Bit Sliding Window Protocol-A Protocol Using Go-

Back-N- A Protocol Using Selective Repeat.

UNIT-III

The Medium Access Control Sub layer-The Channel Allocation Problem-Static Channel Allocation-Assumptions for Dynamic Channel Allocation, Multiple Access Protocols-Aloha-Pure aloha- slotted aloha-Carrier Sense Multiple Access Protocols- Collision-Free Protocols-Limited Contention Protocols. **Wireless LAN Protocols-** Ethernet-Classic Ethernet Physical Layer-Classic Ethernet MAC Sub-layer Protocol- Ethernet Performance-Fast Ethernet- Wireless LANs-The 802.11 Architecture and Protocol Stack-The 802.11 Physical Layer-The802.11 MAC Sub-layer Protocol- The 805.11 Frame Structure-Services.

Unit-IV

The Network Layer Design Issues – Store and Forward Packet Switching-Services Provided to the Transport layer- Implementation of Connectionless Service- Implementation of Connection Oriented Service- Comparison of Virtual Circuit and Datagram Networks, Routing Algorithms-The Optimality principle-Shortest path, Flooding, Distance vector, Link state, Hierarchical. **Congestion Control algorithms-** General principles of congestion control, Congestion prevention polices, Approaches to Congestion Control-Traffic Aware Routing- Admission Control-Traffic Throttling- Load Shedding. **Internet Working:** How networks differ- How networks can be connected-Tunneling, internetwork routing-, Fragmentation, network layer in the internet – IP protocols-IP Version 4 protocol-, IP addresses-, Subnets-IP Version 6- The main IPV6 header- Internet control protocols- ICMP-ARP-DHCP.

UNIT-V

The Transport Layer: Transport layer protocols: Introduction-services- port number-User data gram protocol-User datagram-UDP services-UDP applications- Transmission control protocol: TCP services- TCP features-Segment- A TCP connection- windows in TCP- flow control-Error control. **Application Layer –-** World Wide Web: HTTP , FTP-Two connections-control connection-Data connection-security of FTP-Electronic mail-Architecture-web based mail- email security- TELENET-local versus remote Logging. **Domain Name System:** Name Space, DNS in Internet, - Resolution-Caching-Resource Records- DNS messages- Registrars-security of DNS Name Servers.

Text Books:

1. Computer Networks: Andrew S Tanenbaum David J. Wetherall, 5/e, Pearson
2. Data communications and networking: Behrouz Forouzan, 5/e, McGraw Hill

Reference Books

1. Computer Networks – A System Approach, Peterson, Bruce Davie,2/e , Harcourt Asia

2. Compute communications and networking technologies, Gallo, Hancock, Cengage
3. An Engineering approach to compute networking, Kesha, Pearson

II Semester	L	P	C
	3	0	3
20MC2T02 DATA WAREHOUSING AND DATA MINING			

Course Objectives:

- Be familiar with mathematical foundations of data mining tools..
- Understand and implement classical models and algorithms in data warehouses and data mining
- Characterize the kinds of patterns that can be discovered by association rule mining, classification and clustering.
- Develop skill in selecting the appropriate data mining algorithm for solving practical problems.

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

Course Outcomes		Knowledge Level (K)#
CO1	An ability to understand the basics of types of data, quality of data, suitable techniques required for preprocessing and measures required to perform data analysis	**K2**
CO2	Describe the need of classification, identify suitable technique(s) to perform classification, model building and evaluation	**K3**
CO3	Identify the requirements and usage of association rule mining on categorical and continuous data.	**K3**
CO4	Compare and Identify suitable clustering algorithm(s) (apply with open source tools), interpret, evaluate and report the result	**K4**
CO5	Describe the requirements and the need of web mining	**K2**

UNIT-1:

Introduction to Data mining, types of Data, Data Quality, Data Processing, Measures of Similarity and Dissimilarity, Exploring Data: Data Set, Summary Statistics, Visualization, Data Warehouse, OLAP and multi dimensional data analysis.

UNIT-II:

Classification: Basic Concepts, Decision Trees and model evaluation: General approach for solving a classification problem, Decision Tree induction, Model over fitting: due to presence of noise, due to lack of representation samples, Evaluating the performance of classifier. Nearest Neighborhood classifier, Bayesian Classifier, Support vector Machines: Linear SVM, Separable and Non Separable case.

UNIT-III:

Association Analysis: Problem Definition, Frequent Item-set generation, rule generation, compact representation of frequent item sets, FP-Growth Algorithms. Handling Categorical, Continuous attributes, Concept hierarchy, Sequential, Sub graph patterns

UNIT-IV:

Clustering: Over view, K-means, Agglomerative Hierarchical clustering, DBSCAN, Cluster evaluation: overview, Unsupervised Cluster Evaluation using cohesion and separation, using proximity matrix, Scalable Clustering algorithm

UNIT-V:

Web data mining: Introduction, Web terminology and characteristics, Web content mining, Web usage mining, web structure mining, Search Engines: Characteristics, Functionality, Architecture, Ranking of WebPages, Enterprise search

Text Books:

1. Introduction to Data Mining, Tan, Steinbach and Vipin Kumar, Pearson Education, 2016
2. DataMining:ConceptsandTechniques,2ndEdition,JiaweiHanand MichelineKamber, ELSEVIER

Reference Books:

1. Data Mining: The Textbook, Springer, May 2015, Charu C. Aggarwal.

Web resources:

1. NPTEL: https://nptel.ac.in/courses/106/105/106105174/
   https://www.saedsayad.com/data_mining.htm

II Semester	L	P	C
	3	0	3
20MC2T03 OBJECT ORIENTED PROGRAMMING USING JAVA			

Course Objectives:

- To understand the basic concepts of object oriented programming concepts.
- To introduce the principles of inheritance and polymorphism and demonstrate how they are related to the design of abstract classes
- To understand the implementation of packages and interfaces
- To introduce the concept of multithreading and exception handling
- To learn and understand the design of Graphical User Interface using applets and swing controls

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

	Course Outcomes	**Knowledge Level (K)#**
CO1	Describe the uses OOP concepts	**K2**
CO2	Apply OOP concepts to solve real world problems	**K3**
CO3	Distinguish the concept of packages and interfaces	**K4**
CO4	Demonstrate the exception handing, multithread applications with synchronization	**K2**
CO5	Design the GUI based applications using AWT and Swings	**K6**
CO6	Discuss the Collection Framework	**K6**

UNIT–I:

Basics of Object Oriented Programming (OOP): Need for OO paradigm , A way of viewing world- Agents, responsibility, messages, methods, classes and instances, class hierarchies (Inheritance), method binding, overriding and exceptions, summary of OOP concepts, coping with complexity, abstraction mechanisms. **Java Basics:** Data types, variables, scope and life time of variables, arrays, operators, expressions, control statements, type conversion and costing, simple java program, classes and objects-concepts of classes, objects, constructors methods, access control, this keyword, garbage collection, overloading methods and constructors, parameter passing, recursion, string handling.

UNIT–II:

Inheritance: Hierarchical abstractions, Base class object, subclass, subtype, substitutability, forms of inheritance- specialization, specification, construction, extension, limitation, combination, benefits of inheritance costs of inheritance. Member access rules, super uses, using final with inheritance, polymorphism, abstract classes. **Packages and Interfaces:** Defining, Creating and Accessing a package, Understanding CLASSPATH, Importing packages, differences between classes and interfaces, defining an interface, Implementing interface, applying interfaces variables in interface and extending interfaces.

UNIT–III:
Exception handling and Multithreading: Concepts of exception handling, benefits of exception handling, Termination or presumptive models, exception hierarchy, usage of try, catch, throws and finally, built in exceptions, creating own exception sub classes. Differences between multi threading and multitasking, thread life cycle, creating threads, synchronizing threads, daemon threads, thread groups.

UNIT–IV:
Event Handling: Events, Event sources, Event classes, Event Listeners, Delegation event model, handling mouse and keyboard events, Adapter classes, inner classes. The AWT class hierarchy , user-interface components-labels, button, canvas, scrollbars, text components, check box, check box groups, choices, list panes- scroll pane, dialogs, menu bar, graphics, layout manager- layout manager types- boarder, grid, flow, card and grid bag.

UNIT-V:
Applets: Concepts of Applets, differences between applets and applications, lifecycle of an applet, types of applets, creating applets, passing parameters to applets, **Swings:** Introduction, limitations of AWT, MVC architecture, components, containers, exploring swing- JApplet, JFrame and JComponent, Icons and Labels, text fields, buttons-The JButton class, Check boxes, Radio Buttons, Combo boxes, Tabbed panes, Scroll panes, Trees and Tables.

Text Books:

1. Java-The complete reference,7/e, Herbert schildt, TMH
2. JAVA: How to program, 8/e, Dietal , Dietal,PHI
3. Introduction of programming with JAVA,S.Dean,TMH
4. Introduction to Java programming, 6/e, Y.Daniel Liang, Pearson

Reference Books:

1. Core Java 2, Vol 1(Vol 2) Fundamentals(Advanced), 7/e, Cay.S.Horstmann, Gary Cornell, Pearson
2. Big Java2,3/e, Cay.S. Horstmann, Wiley
3. Object Oriented Programming through Java, P.Radha Krishna, University Press
4. JAVA& Object Orientation an Introduction, 2/e, John Hunt, Springer
5. Introduction to JAVA Programming, 7/e, Y. Daniel Liang, Pearson., TMH

II Semester	L	P	C
	3	0	3
20MC2T04 OBJECT ORIENTED SOFTWARE ENGINEERING			

- To understand the nature of software development and software life cycle models.
- To understand methods of capturing, specifying, visualizing and analyzing software requirements.
- To know basics of testing and understanding concept of software quality assurance and software configuration management process.
- To learn to provide correctness proofs for algorithms.

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

Course Outcomes		Knowledge Level (K)#
CO1	Analyze software application domains and process models used in software development.	**K4**
CO2	Explain the software requirements collection and develop specifications and evaluate them.	**K2**
CO3	Convert the requirements model into the design model and evaluate the complexity metrics.	**K5**
CO4	Compare various testing strategies and tactics and their applications with the supporting tools.	**K4**
CO5	Adopt the activities of Software Project Development principles in project development.	**K3**

UNIT–I:

Introduction to Object Oriented Software Engineering: Nature of the Software, Types of Software, Software Engineering Activities, Software Quality

Introduction to Object Orientation: Data Abstraction, Inheritance & Polymorphism, Reusability in Software Engineering, Examples: Postal Codes, Geometric Points.

Requirements Engineering: Domain Analysis, Problem Definition and Scope, Types of Requirements, Techniques for Gathering and Analyzing Requirements, Requirement Documents, Reviewing Requirements, Case Studies: GPS based Automobile Navigation System, Simple Chat Instant Messaging System.

UNIT–II:

Unified Modeling Language & Use Case Modeling: Introduction to UML, Modeling Concepts, Types of UML Diagrams with Examples; User-Centred Design, Characteristics of Users, Developing Use Case Models Of Systems, Use Case Diagram, Use Case Descriptions, The Basics of User Interface Design, Usability Principles.

Class Design and Class Diagrams: Essentials of UML Class Diagrams, Associations And Multiplicity, Generalization, Instance Diagrams, Advanced Features of Class Diagrams, Process of Developing Class Diagrams, Interaction and Behavioral Diagrams: Interaction Diagrams, State Diagrams, Activity Diagrams, Component and Deployment Diagrams.

UNIT- III:

Software Design and Architecture: Design Process, Principles Leading to Good Design, Techniques for Making Good Design Decisions, Good Design Document, Software Architecture, Architectural Patterns: The Multilayer, Client-Server, Broker, Transaction Processing, Pipe & Filter And MVC Architectural Patterns.

Design Patterns: Abstraction-Occurrence, General Hierarchical, Play-Role, Singleton, Observer, Delegation, Adaptor, Façade, Immutable, Read-Only Interface and Proxy Patterns

UNIT– IV:

Software Testing: Effective and Efficient Testing, Defects in Ordinary Algorithms, Numerical Algorithms, Timing and Co-ordination, Stress and Unusual Situations, Testing Strategies for Large Systems.

UNIT-V:

Software Project Management: Introduction to Software Project Management, Activities of Software Project Management, Software Engineering Teams, Software Cost Estimation, Project Scheduling, Tracking And Monitoring.

Software Process Models: Waterfall Model, The Phased Released Model, The Spiral Model, Evolutionary Model, The Concurrent Engineering Model, Rational Unified Process.

Text Books:

1. Object Oriented Software Engineering Practical Software Development using UML and Java, Timothy C. Lethibridge and Robert Laganiere Mc Graw Hill, Second Edition, 2004
2. Object Oriented Software Engineering using UML, Patterns and Java, Bernd Bruegge, Allen H. Dutoit, Pearson, 3rd edition, 2009
3. Software Engineering: A Practitioner's Approach by R. S. Pressman, McGraw Hill, 9th Edition, Sept 2019

Reference Books:

1. Zero Defect Software, G. G. Schulmeyer, Published by McGraw Hill, 1992
2. Object Oriented Modeling and Design, J. Rumbaugh, Published by Prentice Hall, 1991
3. Software Engineering K.K. Aggarwal, Yogesh Singh, Published by New Age International Publishers, Third Edition, 2007
4. Software Engineering , Ian Sommerville, Published by Addison Welsley, 9th Edition, 2010.
5. An Integrated Approach to Software Engineering, Pankaj Jalote, Published by Narosa Publishing House, 3rd Edition, 2007

II Semester	L	P	C
	3	0	3
20MC2TE1 HUMAN COMPUTER INTERACTION			

Course Objectives:

- The main objective is to get student to think constructively and analytically about how to design and evaluate interactive technologies.

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

	Course Outcomes	Knowledge Level (K)#
CO1	Explain the capabilities of both humans and computers from the viewpoint of human information processing.	**K4**
CO2	Describe typical human–computer interaction (HCI) models, styles, and various historic HCI paradigms	**K2**
CO3	Apply an interactive design process and universal design principles to designing HCI systems	**K5**
CO4	Analyze and identify user models, user support, socio-organizational issues, and stakeholder requirements of HCI systems	**K4**
CO5	Discuss tasks and dialogs of relevant HCI systems based on task analysis and dialog design.	**K3**

UNIT - I

Introduction: Importance of user Interface, definition, importance of good design. Benefits of good design. A brief history of Screen designs

The graphical user interface: Popularity of graphics, the concept of direct manipulation, graphical system, Characteristics, Web user – interface popularity, characteristics-Principles of user interface.

UNIT - II

Design Process: Human interaction with computers, importance of human characteristics, human consideration, Human interaction speeds, understanding business junctions.

UNIT – III

Screen Designing : Design goals, Screen planning and purpose, organizing screen elements, ordering of screen data and content, screen navigation and flow, Visually pleasing composition, amount of information, focus and emphasis, presentation information simply and meaningfully, information retrieval on web, statistical graphics, Technological consideration in interface design.

UNIT - IV

Windows: Windows new and Navigation schemes selection of window, selection of devices based and screen based controls. Components: Components text and messages, Icons and increases, Multimedia, colors, uses problems, choosing colors.

UNIT - V

Software tools: Specification methods, interface, Building Tools. Interaction Devices: Keyboard and function keys, pointing devices, speech recognition digitization and generation, image and video displays, drivers.

TEXT BOOKS:

1. The Essential guide to user interface design,2/e, Wilbert O Galitz, Wiley DreamaTech.

REFERENCE BOOKS:

1. Human Computer Interaction. 3/e, Alan Dix, Janet Finlay, Goryd, Abowd, Russell Beal, PEA,2004

II Semester	L	P	C
	3	0	3
20MC2TE2 DESIGN AND ANALYSIS OF ALGORITHMS			

Course Objectives:

1. To analyze the asymptotic performance of algorithms.
2. To understand the write rigorous correctness proofs for algorithms.
3. To familiarity with major algorithms and data structures.
4. Apply important algorithmic design paradigms and methods of analysis.
5. Synthesize efficient algorithms in common engineering design situations.

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

CO	Course Outcomes	Knowledge Level (K)#
CO1	Explain the basic concepts of time and space complexity	**K2**
CO2	Explain the basic concepts of Divide-and-Conquer Strategy, Greedy and approximate algorithm	**K3**
CO3	Describe the methodologies of how to analyze the applications by Dynamic Programming algorithm	**K5**
CO4	Discuss the concept of graph coloring and back tracking	**K3**
CO5	Develop solve problems using Branch and Bound Techniques	**K4**

UNIT-I:

Introduction: Algorithm, Pseudo code for expressing algorithms, performance Analysis-Space complexity, Time complexity, Asymptotic Notation- Big oh notation, Omega notation, Theta notation and Little oh notation, probabilistic analysis, Amortized analysis. Disjoint Sets- disjoint set operations, union and find algorithms, spanning trees, connected components and bi- connected components.

UNIT-II:

Divide and conquer: General method, applications-Binary search, Quick sort, Merge sort, Stassen's matrix multiplication. Greedy method: General method, applications-Job sequencing with deadlines, 0/1 knapsack problem, Minimum cost spanning trees, Single source shortest path problem.

UNIT-III:

Dynamic Programming: General method, applications-Matrix chain multiplication, Optimal binary search trees, 0/1 knapsack problem, All pairs shortest path problem, Travelling sales person problem, Reliability design.

UNIT-IV:

Backtracking: General method, applications-n-queen problem, sum of subsets problem, graph coloring, Hamiltonian cycles.

UNIT-V:

Branch and Bound: General method, applications - Travelling sales person problem, 0/1 knapsack problem- LC Branch and Bound solution, FIFO Branch and Bound solution. NP-Hard and NP-Complete problems: Basic concepts, non deterministic algorithms, NP - Hard and NP Complete classes, Cook's theorem.

Text Books:

1. Fundamentals of Computer Algorithms, Ellis Horowitz, Satraj Sahni and Rajasekharam, Universities Press
2. The Algorithm Design Manual, 2nd edition, Steven S. Skiena, Springer
3. Introduction to Algorithms, second edition, T.H.Cormen, C.E.Leiserson, R.L.Rivest and C.Stein, PHI Pvt. Ltd

Reference Books:

1. Introduction to the Design and Analysis of Algorithms, Anany Levitin, PEA
2. Design and Analysis of Algorithms, Pearson Education, Parag Himanshu Dave, Himansu Balachandra Dave
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, Pearson education, Aho, Ullmanand Hopcroft

II Semester	L	P	C
	3	0	3
20MC2TE3 NO SQL DATABASES			

Course Objectives:

The objective of the course is to:

- Define, compare and use the four types of NoSQL Databases (Document-oriented, Key Value Pairs, Column oriented and Graph)
- Demonstrate an understanding of the detailed architecture, define objects, load data, query data and performance tune Column-oriented NoSQL databases
- Explain the detailed architecture, define objects, load data, query data and performance tune Document oriented NoSQL databases
- Ability to design entity relationship model and convert entity relationship diagrams into RDBMS and formulate SQL queries on the data

Course Outcomes		Knowledge Level (K)#
CO1	Understand basic concepts of NoSql	**K1**
CO2	Apply NoSQL data modeling from application specific queries	**K5**
CO3	Understand NoSQL Architecture	**K1**
CO4	Apply HBASE data modeling from application specific queries	**K1**
CO5	Defining Indexing and Ordering Data Sets.	**K3**

UNIT I:

Introduction to NoSQL: Definition And Introduction, Sorted Ordered Column-Oriented Stores, Key/Value Stores, Document Databases, Graph Databases, Examining Two Simple Examples, Location Preferences Store, Car Make And Model Database, Working With Language Bindings.

UNIT II:

Interacting with NoSQL: If NoSql Then What, Language Bindings For NoSQL Data Stores, Performing Crud Operations, Creating Records, Accessing Data, Updating And Deleting Data

UNIT III:

NoSQL Storage Architecture: Working With Column-Oriented Databases, Hbase Distributed Storage Architecture, Document Store Internals, Understanding Key/Value Stores In Memcached And Redis, Eventually Consistent Non-Relational Databases.

UNIT IV:

NoSQL Stores: Similarities Between Sql And Mongodb Query Features, Accessing Data From Column-Oriented Databases Like Hbase, Querying Redis Data Stores, Changing Document Databases, Schema Evolution In Column-Oriented Databases, Hbase Data Import And Export, Data Evolution In Key/Value Stores.

UNIT V

Indexing and Ordering Data Sets: Essential Concepts Behind A Database Index, Indexing And Ordering In Mongodb, Creating and Using Indexes In Mongodb, Indexing And Ordering In Couchdb, Indexing In Apache Cassandra.

Text Books:

1) Pramod Sadalage and Martin Fowler, NoSQL Distilled, Addison-Wesley Professional, 2012.
2) Dan McCreary and Ann Kelly, Making Sense of NoSQL, Manning Publications, 2013.

Reference Books:

1) Shashank Tiwari, Professional NoSQL, Wrox Press, Wiley, 2011, ISBN: 978-0-470-94224-6
2) Gaurav Vaish, Getting Started with NoSQL, Packt Publishing, 2013.

II Semester	L	P	C
	3	0	3
20MC2TE4 MOBILE APPLICATION DEVELOPMENT			

Course Objectives:

1. To demonstrate the introduction and characteristics of mobile applications
2. Application models of mobile application frameworks. Managing application data and User- interface design for mobile applications
3. Integrating **networking**, the OS and hardware into mobile-applications
4. Addressing enterprise requirements in mobile applications – performance, scalability, modifiability, availability and security
5. Testing methodologies for mobile applications**–** Publishing, deployment, maintenance and management. To demonstrate their skills of using Android software development tools
6. To demonstrate their ability to deploy software to mobile devices

Course Outcomes		Knowledge Level (K)#
CO1	Install and configure Android application development tools	**K1**
CO2	Design and develop user Interfaces for the Android platform	**K5**
CO3	Save state information across important operating system events	**K1**
CO4	Apply Java programming concepts to Android application development	**K3**
CO5	Developing Android Application using GPS and Sensors	**K4**

UNIT I:

Introduction to mobile devices: Introduction to Mobile Computing, Introduction to Android Development Environment, Mobile devices vs. desktop devices, ARM and Intel architectures, Screen resolution, Touch interfaces, Application deployment, App Store, Google Play, Windows Store.

Development environments: XCode, Eclipse, VS2012, PhoneGAP, etc.; Native vs. web applications. **Factors in Developing Mobile Applications:** Mobile Software Engineering, Frameworks and Tools, Generic UI Development, Android User.

UNIT II:

Android User Interface: Measurements – Device and pixel density independent measuring units User Interface (UI) Components – Editable and non editable Text Views, Buttons, Radio and Toggle Buttons, Checkboxes, Spinners, Dialog and pickers Fragments – Creating fragments, Lifecycle of fragments, Fragment states, Adding fragments to Activity, adding, removing and replacing fragments with fragment transactions, interfacing between fragments and Activities, Multi-screen Activities.

UNIT III:

Back Ground Running Process, Networking and Telephony Services: Introduction to services local service, remote service and binding the service, the communication between service and activity, Intent Service.

MultiThreading: Handlers, AsyncTask.

Broad cast receivers: Local Broadcast Manager, Dynamic broadcast receiver, System Broadcast. Pending Intent, Notifications.

UNIT IV:

Android: Introduction – Establishing the development environment – Android architecture – Activities and views – Interacting with UI – Persisting data using SQLite – Packaging and deployment – Interaction with server side applications – Using Google Maps, GPS and Wifi – Integration with social media applications. **Android network programming:** Http Url Connection, Connecting to REST-based and SOAP based Web services.

UNIT V:

Advanced Topics: Power Management: Wake locks and assertions, Low-level OS support, Writing power-smart applications.

Augmented Reality via GPS and other sensors: GPS, Accelerometer, Camera. Mobile device security in depth: Mobile malware, Device protections, iOS "Jailbreaking", Android "rooting" and Windows' "defenestration"; Security and Hacking: Active Transactions, More on Security, Hacking Android.

Text Books:

1) Bill Phillips, Chris Stewart, Brian Hardy, and Kristin Marsicano, Android Programming: The Big Nerd Ranch Guide, Big Nerd Ranch LLC, 2nd edition, 2015.
2) Valentino Lee, Heather Schneider, and Robbie Schell, Mobile Applications: Architecture, Design and Development, Prentice Hall, 2004.
3) Professional Android 4 Application Development, Reto Meier, Wiley India, (Wrox) , 2012
4) Android Application Development for Java Programmers, James C Sheusi, Cengage Learning, 2013
5) Dawn Griffiths, David Griffiths,*"Head First: Android Development"*

,OReilly2015,ISBN: 9781449362188

6) Jeff McWherter and Scott Gowell, "Professional Mobile Application Development", Wrox, 2012

Reference Books:

1) Beginning Android 4 Application Development, Wei-Meng Lee, Wiley India (Wrox), 2013
2) Tomasz Nurkiewicz and Ben Christensen, Reactive Programming with RxJava, O'Reilly Media, 2016.
3) Brian Fling, Mobile Design and Development, O'Reilly Media, Inc., 2009.
4) Maximiliano Firtman, Programming the Mobile Web, O'Reilly Media, Inc., 2nd ed., 2013.
5) Cristian Crumlish and Erin Malone, Designing Social Interfaces, 2nd ed., O'Reilly Media, Inc., 2014.
6) Suzanne Ginsburg, Designing the iPhone User Experience: A User-Centered Approach to Sketching and Prototyping iPhone Apps, Addison-Wesley Professional, 2010.

II Semester	L	P	C
	3	0	3

20MC2TE5 INTERNET OF THINGS

Course Objectives:

1. Identify problems that are amenable to solution by AI methods, and which AI methods may be suited to solving a given problem.
2. Formalize a given problem in the language/framework of different AI methods (e.g., as a search problem, as a constraint satisfaction problem, as a planning problem, as a Markov decision process, etc).
3. Design and carry out an empirical evaluation of different algorithms on problem formalization, and state the conclusions that the evaluation supports.

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

Course Outcomes		Knowledge Level (K)#
CO1	Explain the definition and usage of the term 'the internet of things' in different contexts	**K2**
CO2	Discover the various network protocols used in IoT	**K2**
CO3	Define the role of big data, cloud computing and data analytics in a typical IoT system.	**K3**
CO4	Compare and contrast the threat environment based on industry and/or device type	**K2**
CO5	Design a simple IoT system made up of sensors, wireless network connection, data analytics and display /actuators, and write the necessary control software	**K6**

UNIT I:

The Internet of Things: An Overview of Internet of things, Internet of Things Technology, behind Io Ts Sources of the Io Ts, M2M Communication, Examples of IoTs, Design Principles For Connected Devices Internet Connectivity Principles, Internet connectivity, Application Layer Protocols: HTTP, HTTPS, FTP, Telnet.

UNIT II:

Business Models for Business Processes in the Internet of Things, IoT/M2M systems LAYERS AND designs standardizations ,Modified OSI Stack for the IoT/M2M Systems ,ETSI M2M domains and High-level capabilities, Communication Technologies, Data Enrichment and Consolidation and Device Management Gateway Ease of designing and affordability

UNIT III:

Design Principles for the Web Connectivity for connected-Devices, Web Communication protocols for Connected Devices, Message Communication protocols for Connected Devices, Web Connectivity for connected-Devices.

UNIT IV:

Data Acquiring, Organizing and Analytics in IoT/M2M, Applications / Services / Business Processes, IOT/M2M Data Acquiring and Storage, Business Models for Business Processes in the Internet Of Things, Organizing Data, Transactions, Business Processes, Integration and Enterprise Systems.

UNIT V:

Data Collection, Storage and Computing Using a Cloud Platform for IoT/M2M Applications/Services, Data Collection, Storage and Computing Using cloud platform Everything as a service and Cloud Service Models, IOT cloud-based services using the Xively (Pachube/COSM), Nimbits and other platforms Sensor, Participatory Sensing, Actuator, Radio Frequency Identification, and Wireless, Sensor Network Technology, Sensors Technology, Sensing the World.

Text Books:

1. Internet of Things: Architecture, Design Principles And Applications, 1st ed, Rajkamal,McGraw Hill Higher Education, 2017.
2. Internet of Things, 1st ed, A.Bahgya and V.Madisetti, Univesity Press, 2014

Reference Books:

1. Designing the Internet of Things, 1st ed, Adrian McEwen and Hakim Cassimally, Wiley, 2013.
2. Getting Started with the Internet of Things, 1st ed, CunoPfister , Oreilly, 2011.

II Semester	L	P	C
	0	3	1.5
20MC2L01 JAVA PROGRAMMING LAB			

Course Objectives:

- To understand how to design, implement, test, debug, and document programs that use basic data types and computation, simple I/O, conditional and control structures, string handling and functions.
- To understand the importance of Classes & objects along with constructors, Arrays and Vectors.
- Discuss the principles of inheritance, interface and packages and demonstrate though problem analysis assignments how they relate to the design of methods, abstract classes and interfaces and packages.
- To understand importance of Multi-threading & different exception handling mechanisms.
- To learn experience of designing, implementing, testing, and debugging graphical user interfaces in Java using applet and AWT that respond to different user events.
- To understand Java Swings for designing GUI applications based on MVC architecture

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

	Course Outcomes	Knowledge Level (K)#
CO1	Apply OOP concepts to solve real world problems	**K2**
CO2	Implement different forms of inheritance	**K3**
CO3	Create packages and to reuse them.	**K6**
CO4	Implement multi-threaded programs using synchronization concepts and exceptions	**K3**
CO5	Design GUI applications using AWT and SWINGS.	**K6**

List of Experiments:

1. The Fibonacci sequence is defined by the following rule. The first 2 values in the sequence are 1, 1. Every subsequent value is the sum of the 2 values preceding it. Write a Java Program that uses both recursive and non-recursive functions to print the nth value of the Fibonacci sequence.
2. Write a Java Program that prompts the user for an integer and then prints out all the prime numbers up to that Integer.
3. Write a Java Program that checks whether a given string is a palindrome or not. Ex. MALAYALAM is a palindrome.
4. Write a Java Program for sorting a given list of names in ascending order.
5. Write a Java Program that illustrates how runtime polymorphism is achieved.

6. Write a Java Program to create and demonstrate packages.
7. Write a Java Program, using StringTokenizer class, which reads a line of integers and then displays each integer and the sum of all integers.
8. Write a Java Program that reads on file name form the user then displays information about whether the file exists, whether the file is readable/ writable, the type of file and the length of the file in bytes and display the content of the using File Input Stream class.
9. Write a Java Program that displays the number of characters, lines and words in a text/text file.
10. Write an Applet that displays the content of a file.
11. Write a Java Program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +-*?% operations. Add a text field to display the result.
12. Write a Java Program for handling mouse events.
13. Write a Java Program demonstrating the life cycle of a thread.
14. Write a Java Program that lets users create Pie charts. Design your own user interface (with Swings & AWT).
15. Write a Java Program to implement a Queue, using user defined Exception Handling (also make use of throw, throws).

II Semester	L	P	C
	0	3	1.5
20MC2L02 OBJECT ORIENTED SOFTWARE ENGINEERING LAB			

The Unified Modeling Language (UML) is a standard language for specifying, visualizing, constructing and documenting the artifacts of software systems. The primary goal of UML is to provide users a ready-to-use, expressive visual modeling language so that they can develop and exchange meaningful models.

This lab deals with object oriented analysis and design of a software problem using UML concepts and notations. The tool used is Rational Rose Enterprise Edition. Any other open source tool is also recommended.

Document the Software Project Management and Software Engineering activities for any two of the following projects. Any other project of interest also can be chosen.

1. Student Result Management System
2. Library Management System
3. Payroll System
4. Bank Loan System
5. Railway Reservation System
6. Automatic Teller Machine
7. Hostel Management System
8. Hospital Management System
9. Online Shopping System
10. Blood Bank Management System
11. GPS
12. Journal Publication System
13. Chatroom Application
14. Social Media Application

Software Project Management and Software Engineering activities specified below can be customized according to the features of the project.

- Problem Statement
- Feasibility Study
- Software Requirements Specification Document
- Estimation of Project Metrics
- Entity Relationship Diagram
- Use Case Diagrams
- Class Diagram
- Sequence Diagrams
- Activity Diagrams
- State Chart Diagrams
- Test coverage

References:

1. The Unified Modeling Language User Guide. Grady Booch,James Rumbaugh and Ivar Jacobson. Addison-Wesley.
2. Object Oriented Software Engineering: Practical Software Development using UML and Java. Timothy C Lethbridge & Robert, Langaneire, Mc Graw Hill

II Semester	L	P	C
	0	3	1.5
20MC2L03 COMPUTER NETWORKS LAB			

PART – A

1) Implement the data link layer farming methods such as character stuffing and bit stuffing.
2) Implement on a data set of characters the three CRC polynomials – CRC 12, CRC 16 and CRC CCIP.
3) Implement Dijkstra‘s algorithm to compute the Shortest path through a graph.
4) Take an example subnet graph with weights indicating delay between nodes. Now obtain Routing table art each node using distance vector routing algorithm
5) Take an example subnet of hosts. Obtain broadcast tree for it.

PART – B

1) Implement the following forms of IPC.
   a) Pipes b) FIFO
2) Implement file transfer using Message Queue form of IPC
3) Write a programme to create an integer variable using shared memory concept and increment the variable
4) Simultaneously by two processes. Use semaphores to avoid race conditions
5) Design TCP iterative Client and server application to reverse the given input sentence
6) Design TCP client and server application to transfer file
7) Design a TCP concurrent server to convert a given text into upper case using multiplexing system call “select”
8) Design a TCP concurrent server to echo given set of sentences using poll functions
9) Design UDP Client and server application to reverse the given input sentence
10) Design UDP Client server to transfer a file
11) Design using poll client server application to multiplex TCP and UDP requests for converting a given text into upper case.
12) Design a RPC application to add and subtract a given pair of integers

II Semester	L	P	C
	0	1	0.5
20MC2M01 EMPLOYABILITY SKILLS			

Course Objectives:

The main of this course is

- To learn how to make effective presentations and impressive interviews
- To learn skills for discussing and resolving problems on the work site
- To assess and improve personal grooming
- To promote safety awareness including rules and procedures on the work site
- To develop and practice self-management skills for the work site

Course Outcomes:

By the end of this course, the student

- Recite the soft skills
- Make presentations effectively with appropriate body language
- Be composed with positive attitude
- Apply their core competencies to succeed in professional and personal life

A list of vital employability skills from the standpoint of engineering students with discussion how to potentially develop such skills through campus life.

1) Soft Skills: An Introduction – Definition and Significance of Soft Skills; Process, Importance and Measurement of Soft Skill Development.
2) Self-Discovery: Discovering the Self; Setting Goals; Beliefs, Values, Attitude, Virtue.
3) Positivity and Motivation: Developing Positive Thinking and Attitude; Driving out Negativity; Meaning and Theories of Motivation; Enhancing Motivation Levels.
4) Time Management – Concept, Essentials, Tips.
5) Personality Development – Meaning, Nature, Features, Stages, Models; Learning Skills; Adaptability Skills.
6) Decision-Making and Problem-Solving Skills: Meaning, Types and Models, Group and Ethical Decision-Making, Problems and Dilemmas in application of these skills.
7) Conflict Management: Conflict - Definition, Nature, Types and Causes; Methods of Conflict Resolution.
8) Stress Management: Stress - Definition, Nature, Types, Symptoms

and Causes; Stress Analysis Models and Impact of Stress; Measurement and Management of Stress

9) Leadership and Assertiveness Skills: A Good Leader; Leaders and Managers; Leadership Theories; Types of Leaders; Leadership Behaviour; Assertiveness Skills.

Note: The student shall be instructed to Record a 2 min video and add to profile before and after taking the course. Students are to be involved in Role Play, Team dynamics, Group Discussion and outcomes are to be recorded.

Reference Books:

1) Barun K. Mitra, Personality Development and Soft Skills, Oxford University Press, 2011.
2) S.P. Dhanavel, English and Soft Skills, Orient Blackswan, 2010.
3) R.S.Aggarwal, A Modern Approach to Verbal & Non-Verbal Reasoning, S.Chand & Company Ltd., 2018.
4) Raman, Meenakshi & Sharma, Sangeeta, Technical Communication Principles and Practice, Oxford University Press, 2011.
5) Managing Soft Skills for Personality Development – edited by B.N.Ghosh, McGraw Hill India, 2012.
6) English and Soft Skills – S.P.Dhanavel, Orient Blackswan India, 2010.

II Semester	L	P	C
	3	0	---
20MC1B01 Python Programming			

COURSE OBJECTIVES:

The objectives of this course are to

The student should be made to:

1. To develop skills on procedural oriented and object oriented programming in Python.
2. To understand and apply different data wrangling techniques using Python.
3. To perform data analysis using python libraries like NumPy, Pandas and exploratory data analysis using Matplotlib.

Course Outcomes (CO): *At the end of the course, student will be able to*

Course Outcomes		Knowledge Level (K)#
CO1	Simulate knowledge in Basics of python programming	**K2**
CO2	Acquire Programming knowledge on Searching and sorting using Python	**K3**
CO3	Acquire Programming knowledge on Text and File Handling and to calculate Mean, Median, Mode, Correlation	**K3**
CO4	Acquire Programming knowledge on NumPy, Pandas Library.	**K2**
CO5	Implement Graph Visualizations in Python and Data Analysis using Python	**K5**

UNIT-I: Introduction to Python: Data Types: Identifiers and Keywords, Integral Types, Floating Point Types, Strings: Strings, Comparing Strings, Slicing and Striding Strings, String Operators and Methods, String formatting with str.format, Collections Data Types, Control structures and functions. Introduction to Python Library Modules: random, math, num, time, os, shutil, sys, glob, re, statistics, requests, creating a custom module

UNIT-II: NumPy Arrays and Vectorized Computation: NumPy arrays, Array creation, Indexing and slicing, Fancy indexing, Numerical operations on arrays, Array functions, Data processing using arrays, Loading and saving data, Saving an array, Loading an array, Linear algebra with NumPy, NumPy random numbers. File Handling: Writing and Reading Binary Data, Writing and Parsing Text Files

UNIT-III: Data Analysis with Pandas: An overview of the Pandas package, The Pandas data structure-Series, The Data Frame, The Essential Basic Functionality: Reindexing and altering labels, Head and tail, Binary operations, Functional statistics, Function application Sorting, Indexing and selecting data, Computational tools, Working with Missing Data, Advanced Uses of Pandas for Data Analysis - Hierarchical indexing, The Panel data.

UNIT-IV: Interacting with Databases: Interacting with Data in Text Format: Reading data from text format, Writing data to text format, Interacting with data in binary format:HDF5, Interacting with Data in MongoDB, Interacting with data in Redis: The simple value, List, Set, Ordered set.

UNIT-V: Data Visualization: The matplotlib API primer-Line properties, Figures and subplots, Exploring plot types-Scatter plots, Bar plots, Histogram plots, Legends and annotations, Plotting functions with Pandas, Additional Python data visualization tools-Bokeh, MayaVi.

Text Books:

1. Programming in Python 3: A Complete Introduction to Python Language, Mark Summerfield, Second Edition, Addison-WesleyPublications, 2009.
2. Python: End-to-End Data Analysis Learning Path, Module 1: Getting Started with Python Data Analysis, Phuong VothiHong , Martin Czygan, Packt Publishing Ltd, 2017

Reference Books:

1. Learning Python, Mark Lutz, Orielly, 5th edition, 2013.
2. Python for Data Analysis, Wes McKinney, Orielly Publications, 2012.
3. Think Python: How to think like a computer scientist, Allen B Downey, Green Tea Press, 2016.
4. Fundamentals of Python: First programs, Second edition, Kenneth A. Lambert, Cengage, 2019.

III Semester	L	P	C
	3	0	3
20MC3T01 MACHINE LEARNING WITH PYTHON			

Course Objectives:
From the course the student will learn

1. To learn patterns and concepts from data without being explicitly programmed in various IOT nodes.
2. To design and analyze various machine learning algorithms and techniques with a modern outlook focusing on recent advances.
3. Explore supervised and unsupervised learning paradigms of machine learning.
4. To explore Deep learning technique and various feature extraction strategies.

Course Outcomes(CO's): At the end of the course, student will be able to

Course Outcomes		Knowledge Level (K)#
CO1	Illustrate and comprehend the basics of Machine Learning with Python	**K2**
CO2	Demonstrate the algorithms of Supervised Learning and be able to differentiate linear and logistic regressions	**K2**
CO3	Demonstrate the algorithms of Unsupervised Learning and be able to understand the clustering algorithms	**K2**
CO4	Evaluate the concepts of binning, pipeline Interfaces with examples	**K5**
CO5	Apply the sentiment analysis for various case studies	**K3**

UNIT-I:
Introduction to Machine Learning with Python: Introduction to Machine Learning, basic terminology, Types of Machine Learning and Applications, Using Python for Machine Learning: Installing Python and packages from the Python Package Index, Introduction to NumPy, SciPy, matplotlib and scikit- learn, Tiny application of Machine Learning.

UNIT-II:
Supervised Learning: Types of Supervised Learning, Supervised Machine Learning Algorithms: k-Nearest Neighbors, Linear Models,

Naive Bayes Classifiers, Decision Trees, Ensembles of Decision Trees, Kernelized Support Vector Machines, Uncertainty Estimates from Classifiers.

UNIT-III:
Unsupervised Learning: Types of Unsupervised Learning, challenges, Preprocessing and scaling, Dimensionality Reduction, Feature Extraction, Manifold Learning, Clustering: K-Means Clustering, Agglomerative Clustering, DBSCAN, Comparing and Evaluating Clustering Algorithms.

UNIT-IV:
Representing Data and Engineering Features: Categorical Variables, Binning, Discretization, Linear Models, Trees, Interactions and Polynomials, Univariate Nonlinear Transformations, Automatic Feature Selection. Parameter Selection with Preprocessing, Building Pipelines, The General Pipeline Interface

UNIT-V:
Working with Text Data (Data Visualization) : Types of Data Represented as Strings, Example Application: Sentiment Analysis of Movie Reviews, Representing Text Data as a Bag of Words, Stop Words, Rescaling the Data with tf-idf, Investigating Model Coefficients, Approaching a Machine Learning Problem, Testing Production Systems, Ranking, Recommender Systems and Other kinds of Learning.

Text Books:

1. Introduction to Machine Learning with Python: A Guide for Data Scientists, Andreas C. Muller & Sarah Guido, Orielly Publications, 2019.
2. Python Machine Learning, Sebastian Raschka & Vahid Mirjalili, 3rd Edition, 2019.
3. Building Machine Learning Systems with Python, Luis Pedro Coelho, Willi Richert, 2nd Edition, 2015.

Reference Books:

1. Machine Learning, Tom M. Mitchell, Mc Graw-Hill Publication,2017

III Semester	L	P	C
	3	0	3
20MC3T02 WEB TECHNOLOGIES			

Course Objectives:

1. To Learn PHP language for server side scripting
2. To introduce XML and processing of XML Data with Java
3. To introduce Server side programming with Java Servlets and JSP
4. To introduce Client side scripting with JavaScript.

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

	Course Outcomes	**Knowledge Level (K)#**
CO1	Analyze a web page and identify its elements and attributes.	**K4**
CO2	To acquire knowledge of xml fundamentals and usage of xml technology in electronic data interchange	**K2**
CO3	Build dynamic web pages using JavaScript (client side programming).	**K3**
CO4	To design and develop web based enterprise systems for the enterprises using technologies like jsp, servlet.	**K6**
CO5	Build web applications using PHP	**K3**

Unit I:

Web Basics- Introduction, Concept of Internet- History of Internet, Protocols of Internet, World Wide Web, URL, Web Server, Web Browser. **HTML- Introduction**, History of HTML, Structure of HTML Document: Text Basics, Structure of HTML Document: Images and Multimedia, Links and webs, Document Layout, Creating Forms, Frames and Tables, Cascading style sheets.

Unit II:

XML Introduction- Introduction of XMLXML: Introduction to XML, Defining XML tags, their attributes and values, Document Type Definition, XML Schemes, Document Object Model, XHTML Parsing XML Data – DOM and SAX Parsers in java.

Unit III:

Introduction to Servlets: Common Gateway Interface (CGI), Life cycle of a Servlet, deploying a Servlet, The Servlet API, Reading Servlet parameters, Reading Initialization parameters, Handling Http Request & Responses, Using Cookies and Sessions, connecting to a database using JDBC.

Unit IV:

Introduction to JSP: The Anatomy of a JSP Page, JSP Processing, Declarations, Directives, Expressions, Code Snippets, implicit objects, Using Beans in JSP Pages, Using Cookies and session for session tracking, connecting to database in JSP. Client-side Scripting: Introduction to JavaScript, JavaScript language – declaring variables, scope of variables, functions. event handlers (onClick, onSubmit etc.), Document Object Model, Form validation.

Unit V:

Introduction to PHP: Declaring variables, data types, arrays, strings, operators, expressions, control structures, functions, reading data from web form controls like text boxes, radio buttons, lists etc., Handling File Uploads. Connecting to database (MySQL as reference), executing simple queries, handling results, Handling sessions and cookies File Handling in PHP: File operations like opening, closing, reading, writing, appending, deleting etc. on text and binary files, listing directories.

Text Books:

1. Web Technologies, Uttam K Roy, Oxford University Press.
2. The Complete Reference PHP — Steven Holzner, Tata McGraw-Hill.

Reference Books:

1. Web Programming, building internet applications, Chris Bates 2″ edition, Wiley Dreamtech.
2. Java Server Pages —Hans Bergsten, SPD O'Reilly.
3. Java Script, D.Flanagan
4. Beginning Web Programming-Jon Duckett WROX.

III Semester	L	P	C
	3	0	3
20MC3T03 PRINCIPLES OF CRYPTOGRAPHY AND NETWORK SECURITY			

Course Objectives:

- To learn various cryptographic algorithms including secret key cryptography, hashes and message digests, public key algorithms,
- To Familiar in design issues and working principles of various authentication protocols and various secure communication standards including Kerberos, IPsec, and S/MIME

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

Course Outcomes		Knowledge Level (K)#
CO1	Explain Basic Principles, different security threats, countermeasures, foundation course of cryptography mathematics and Symmetric Encryption.	**K2**
CO2	Classify the basic principles of Asymmetric key algorithms and operations of asymmetric key cryptography.	**K4**
CO3	Design Cryptographic Hash Functions as SHA-3 and Digital Signatures as Elgamal	**K6**
CO4	Explain the concept of Revise Key Management and Distribution and User Authentication	**K3**
CO5	Determine the knowledge of Network and Internet Security Protocols such as S/MIME	**K5**

UNIT I:

Basic Principles: Security Goals, Cryptographic Attacks, Services and Mechanisms, Mathematics of Cryptography. **Symmetric Encryption:** Mathematics of Symmetric Key Cryptography, Introduction to Modern Symmetric Key Ciphers, Data Encryption Standard, Advanced Encryption Standard.

UNIT II:

Asymmetric Encryption: Mathematics of Asymmetric Key Cryptography-Primes, primality Testing, Factorization, Asymmetric Key Cryptography-RSA Cryptosystem, Rabin Cryptosystem, ElGamal Cryptosystem, Elliptic Curve Cryptosystem

UNIT III:

Cryptographic Hash Functions: Applications of Cryptographic Hash Functions, Two Simple Hash Functions Requirements and Security Hash Functions Based on Cipher Block Chaining, Secure Hash Algorithm (SHA), SHA-3. **Digital Signatures:** Elgamal Digital Signature Scheme, Schnorr Digital Signature, NIST Digital Signature Algorithm

Unit IV:

Key Management and Distribution: Symmetric Key Distribution Using Symmetric Encryption, Symmetric Key Distribution Using Asymmetric Encryption, Distribution of Public Keys, X.509 Certificates. **User Authentication:** User Authentication, Remote User-Authentication Principle, Remote User-Authentication Using Symmetric Encryption, Kerberos, Remote User-Authentication Using Asymmetric Encryption

Unit V: Network and Internet Security

Electronic Mail Security: Internet Mail Architecture, Email Formats, Email Threats and Comprehensive Email Security, S/MIME. **IP Security:** IP Security Policy, Encapsulating Security Payload, Combining Security Associations Internet Key Exchange

Text Books:

1. Cryptography and Network Security, 3rd Edition Behrouz A Forouzan, Deb deep Mukhopadhyay, McGraw Hill,2015
2. Cryptography and Network Security, William Stallings, Global Edition, 7e Pearson, 2017

Reference Books:

2. Network Security and Cryptography, First Edition, Bernard Meneges, Cengage Learning, 2018

III Semester	L	P	C
	3	0	3
20MC3T04 HUMAN RESOURCE MANAGEMENT			

Course Objectives:

1. Contribute to the development, implementation, and evaluation of employee recruitment, selection, and retention plans and processes.
2. Administer and contribute to the design and evaluation of the performance management program.
3. Develop, implement, and evaluate employee orientation, training, and development programs.
4. Facilitate and support effective employee and labour relations in both non-union and union environments.

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

	Course Outcomes	**Knowledge Level (K)#**
CO1	Explain the importance of human resources and their effective management in organizations	**K2**
CO2	Demonstrate a basic understanding of different tools used in forecasting and planning, human resource need.	**K2**
CO3	Describe the meanings of terminology and tools used in managing employees effectively	**K4**
CO4	Make use of Record governmental regulations affecting employees and employers	**K3**
CO5	Analyze the key issues related to administering the human elements such as motivation, compensation, appraisal, career planning, diversity, ethics, and training	**K4**

UNIT I:

HRM: Significance - Definition and Functions – evolution of HRM- Principles - Ethical Aspects of HRM- - HR policies, Strategies to increase firm performance - Role and position of HR department – aligning HR strategy with organizational strategy - HRM at global perspective -challenges – cross-cultural problems – emerging trends in HRM.

UNIT II:

Investment perspectives of HRM: HR Planning – Demand and Supply forecasting - Recruitment and Selection- Sources of recruitment - Tests and Interview Techniques - Training and Development – Methods and techniques – Training evaluation - retention - Job Analysis – job description and specifications - Management development - HRD concepts.

UNIT III:

Wage and Salary Administration: Concept- Wage Structure- Wage and Salary Policies- Legal Frame Work- Determinants of Payment of Wages- Wage Differentials - Job design and Evaluation- Incentive Payment Systems. Welfare management: Nature and concepts – statutory and non-statutory welfare measures – incentive mechanisms.

UNIT IV:

Performance Evaluation: Importance – Methods – Traditional and Modern methods – Latest trends in performance appraisal - Career Development and Counseling- Compensation, Concepts and Principles- Influencing Factors- Current Trends in Compensation- Methods of Payments - compensation mechanisms at international level.

UNIT V:

Managing Industrial Relations: Trade Unions - Employee Participation Schemes-Collective Bargaining–Grievances and disputes resolution mechanisms – Safety at work – nature and importance – work hazards – safety mechanisms - Managing work place stress.

Text Books:

1. K Aswathappa: “Human Resource and Personnel Management”, Tata McGraw Hill, New Delhi, 2013
2. N.Sambasiva Rao and Dr. Nirmal Kumar: “Human Resource Management and Industrial Relations”, Himalaya Publishing House, Mumbai
3. Mathis, Jackson,Tripathy:“Human Resource Management:Asouth-Asin Perspective”, Cengage Learning, New Delhi, 2013
4. Subba Rao P: “Personnel and Human Resource Management-Text and Cases”, Himalaya Publications, Mumbai, 2013.
5. Madhurima Lall, Sakina Qasim Zasidi: “Human Resource Management”, Excel Books, New Delhi, 2010

III Semester	L	P	C
	3	0	3
20MC3TE1 CLOUD COMPUTING			

Course Objectives:

1. To explain the evolving computer model caned cloud computing.
2. To introduce the various levels of services that can be achieved by cloud.
3. To describe the security aspects in cloud.
4. To motivate students to do programming and experiment with the various cloud computing environments.

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

	Course Outcomes	Knowledge Level (K)#
CO1	Illustrate the key dimensions of the challenge of Cloud Computing	**K2**
CO2	Classify the Levels of Virtualization and mechanism of tools.	**K2**
CO3	Analyze Cloud infrastructure including Google Cloud and Amazon Cloud.	**K4**
CO4	Explain Cloud Programming and Software Environments.	**K5**
CO5	Apply authentication, confidentiality and privacy issues in Cloud resource management.	**K3**

UNIT–I:

Systems modeling, Clustering and virtualization: Scalable Computing over the Internet, Technologies for Network based systems, System models for Distributed and Cloud Computing, Software environments for distributed systems and clouds, Performance, Security And Energy Efficiency.

UNIT–II:

Virtual Machines and Virtualization of Clusters and Data Centers: 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 Centre Automation.

UNIT–III:

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.

UNIT–IV:

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. **Storage Systems**: Evolution of storage technology, storage models, file systems and database, distributed file systems, general parallel file systems. Google file system., Apache Hadoop, BigTable, Megastore, Amazon Simple Storage Service(S3).

UNIT-V:

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 MapReduce Applications Subject to Deadlines.

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, Arshadeep Bahga, Vijay Madisetti, University Press

Reference Books:

1. Cloud Computing: A Practical Approach. Anthony T.Velte. Toby J.VeFte, Robert Elsenpeter. Tata McGraw Hill. rp2Oll.
2. Enterprise Cloud Computing Gautam Shroif, Cambridge University Press. 2010.
3. Cloud Computing: Implementation, Management and Security, John W. Rittinouse, James F Ransome. CRC Press, rp2012.
4. Cloud Application Architectures: Building Applications and Infrastructure in the Cloud. George Reese, O'Really SPD, rp2Oll.
5. Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance, Tim Mather, Subra Ktriaraswamy, Shahed Latif, O'Redç SPD, rp2Oll.

III Semester	L	P	C
	3	0	3
20MC3TE2 SOFTWARE PROJECT MANAGEMENT			

Course Objectives:

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

1. To describe and determine the purpose and importance of project management from the perspectives of planning, tracking and completion of project
2. To compare and differentiate organization structures and project structures
3. To implement a project to manage project schedule, expenses and resources with the application of suitable project management tools

	Course Outcomes	Knowledge Level (K)#
CO1	Apply the process to be followed in the software development life-cycle models	**K2**
CO2	Apply the concepts of project management & plannING	**K2**
CO3	Implement the project plans through managing people, communications and change	**K4**
CO4	Conduct activities necessary to successfully complete and close the Software projects	**K5**
CO5	Implement communication, modeling, and construction & deployment practices in software development	**K3**

UNIT-I:

Conventional Software Management: The waterfall model, conventional software Management performance.

Evolution of Software Economics: Software Economics, pragmatic software cost estimation. **Improving Software Economics:** Reducing Software product size, improving software processes, improving team effectiveness, improving automation, Achieving required quality, peer inspections.

The old way and the new: The principles of conventional software Engineering, principles of modern software management, transitioning to an iterative process.

UNIT-II:

Life cycle phases: Engineering and production stages, inception, Elaboration, construction, transition phases.

Artifacts of the process: The artifact sets, Management artifacts, Engineering artifacts, programmatic artifacts.

UNIT- III:

Model based software architectures: A Management perspective and technical perspective.

Work Flows of the process: Software process workflows, Iteration workflows.

Checkpoints of the process: Major mile stones, Minor Milestones, Periodic status assessments. **Iterative Process Planning:** Work breakdown structures, planning guidelines, cost and schedule estimating, Iteration planning process, Pragmatic planning.

UNIT- IV:

Project Organizations and Responsibilities: Line-of-Business Organizations, Project Organizations, evolution of Organizations.

Process Automation: Automation Building blocks, The Project Environment.

Project Control and Process instrumentation: The seven core Metrics, Management indicators, quality indicators, life cycle expectations, pragmatic Software Metrics, Metrics automation.

UNIT-V:

Agile Methodology, adapting to Scrum, Patterns for Adopting Scrum, Iterating towards Agility. Fundamentals of DevOps: Architecture, Deployments, Orchestration, Need, Instance of applications, DevOps delivery pipeline, DevOps eco system. DevOps adoption in projects: Technology aspects, Agiling capabilities, Tool stack implementation, People aspect, processes

Text Books:

1) Software Project Management, Walker Royce, PEA, 2005.
2) Succeeding with Agile: Software Development Using Scrum, Mike Cohn, Addison Wesley.
3) The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations, Gene Kim , John Willis , Patrick Debois , Jez Humb,1st Edition, O’Reilly publications, 2016.

Reference Books:

1) Software Project Management, Bob Hughes,3/e, Mike Cotterell, TMH
2) Software Project Management, Joel Henry, PEA
3) Software Project Management in practice, Pankaj Jalote, PEA, 2005,
4) Effective Software Project Management, Robert K.Wysocki, Wiley,2006
5) Project Management in IT, Kathy Schwalbe, Cengage
6) Quality Software Project Management, Futrell,Donald F. Shafer, Donald I. Shafer, PEA

III Semester	L	P	C
	3	0	3
20MC3TE3 ARTIFICIAL INTELLIGENCE			

Course Objectives:

1. To learn the basic State space representation. Intelligent Systems Categorization of Intelligent concepts and techniques of AI and machine learning
2. To explore the various mechanism of Knowledge and Reasoning used for building expert system.
3. To become familiar with supervised and unsupervised learning models
4. To design and develop AI and machine learning solution using modern tools.

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

Course Outcomes		Knowledge Level (K)#
CO1	Demonstrate knowledge of the building blocks of AI as presented in terms of intelligent agents	**K6**
CO2	Analyze and formalize the problem as a state space, graph, design heuristics and select amongst different search or game based techniques to solve them.	**K5**
CO3	Develop intelligent algorithms for constraint satisfaction problems and also design intelligent systems for Game Playing	**K6**
CO4	Attain the capability to represent various real life problem domains using logic based techniques and use this to perform inference or planning.	**K1**
CO5	Solve problems with uncertain information using Bayesian approaches.	**K3**

UNIT– I

Introduction, history, intelligent systems, foundations of AI, applications, tic-tac-toe game playing, development of AI languages, current trends.

UNIT– II

Problem solving: state-space search and control strategies: Introduction, general problem solving, characteristics of problem, exhaustive searches, heuristic search techniques, iterative deepening A*, constraint satisfaction

Problem Reduction and Game Playing: Introduction, problem reduction, game playing, alpha beta pruning, two-player perfect information games.

UNIT – III

Logic concepts: Introduction, propositional calculus, proportional logic, natural deduction system, axiomatic system, semantic tableau system in proportional logic, resolution refutation in proportional logic, predicate logic.

UNIT –IV

Knowledge representation: Introduction, approaches to knowledge representation, knowledge representation using semantic network, extended semantic networks for KR, knowledge representation using frames Advanced knowledge representation techniques: Introduction, conceptual dependency theory, script structure, CYC theory, case grammars, semantic web.

UNIT –V

Expert system and applications: Introduction phases in building expert systems, expert system versus traditional systems

Uncertainty measure: probability theory: Introduction, probability theory, Bayesian belief networks, certainty factor theory, dempster-shafer theory.

Fuzzy sets and fuzzy logic: Introduction, fuzzy sets, fuzzy set operations, types of membership functions, multi valued logic, fuzzy logic, linguistic variables and hedges, fuzzy propositions, inference rules for fuzzy propositions, fuzzy systems..

Text Books:

1. Artificial Intelligence, Saroj kaushik Published by Cengage Learning India, 2011
2. Artificial Intelligence and Machine Learning By Vinod Chandra S.S., Anand Hareendran S
3. Stuart J. Russell and Peter Norvig, "Artificial Intelligence A Modern Approach “Second Edition" Pearson Education

Reference Books:

1. Ivan Bratko "PROLOG Programming for Artificial Intelligence", Pearson Education, Third Edition.
2. Elaine Rich and Kevin Knight "Artificial Intelligence "Third Edition
3. Han Kamber, “Data Mining Concepts and Techniques”, Morgann Kaufmann Publishers.
4. G. Luger, W. A. Stubblefield, "Artificial Intelligence", Third Edition, AddisoWesley Longman, 1998.

III Semester	L	P	C
	3	0	3
20MC3TE4 E-COMMERCE			

Course Objectives:

1. Identify the major categories and trends of e-commerce applications.
2. Identify the essential processes of an e-commerce system.
3. Identify several factors and web store requirements needed to succeed in e-commerce.
4. Discuss the benefits and trade-offs of various e-commerce clicks and bricks alternatives.
5. Understand the main technologies behind e-commerce systems and how these technologies interact.
6. Discuss the various marketing strategies for an online business.
7. Define various electronic payment types and associated security risks and the ways to protect against them.

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

Course Outcomes		Knowledge Level (K)#
CO1	Understand E-Commerce frame work	**K2**
CO2	Design E Payment Systems	**K4**
CO3	Implementation of Inter Organizational commerce	**K4**
CO4	Design and Development of E-Commerce	**K4**
CO5	Identify and analyze consumer needs	**K3**

UNIT-I

Electronic Commerce, Frame Work, Anatomy of E-Commerce Applications, E-Commerce Consumer Applications, E-Commerce organization applications. Consumer Oriented Electronic Commerce, Mercantile Process Models.

UNIT- II

Electronic Payment Systems - Digital Token-Based, Smart Cards, Credit Cards, Risks in Electronic Payment systems. Designing Electronic Payment Systems.

UNIT-III

Inter Organizational Commerce - EDI, EDI Implementation, Value added networks. Intra Organizational Commerce - work Flow, Automation Customization and internal Commerce, Supply chain Management.

UNIT- IV

Corporate Digital Library -Document Library, digital Document types, corporate

Data Warehouses. Advertising and Marketing, Information based marketing, Advertising on Internet, on-line marketing process, market research.

UNIT- V

Consumer Search and Resource Discovery, Information Search and Retrieval, Commerce Catalogues, Information Filtering. Multimedia - key multimedia concepts, Digital Video and electronic Commerce, Desktop Video Processing, Desktop video conferencing.

TEXT BOOK :

1. Frontiers of Electronic Commerce ,Kalakata, Whinston, PEA,2006.

REFERENCE BOOKS:

1. E-Commerce Fundamentals and Applications Hendry Chan, Raymond Lee, Dillon, Chang, John Wiley.
2. E-Commerce, A Managerial Perspective, Turban E, Lee J , King, Chung H.M.,PEA,2001.
3. E-Commerce An Indian Perspective , 3/e, P.T. Joseph, PHI,2009.
4. E-Commerce, S.Jaiswal, Galgotia.
5. Electronic Commerce, Gary P.Schneider, Thomson.

III Semester	L	P	C
	3	0	3
20MC3TE5 CYBER SECURITY			

Course Objectives:

1. To learn threats and risks within context of the cyber security architecture.
2. Student should learn and Identify security tools and hardening techniques.
3. To learn types of incidents including categories, responses and timelines for response.

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

Course Outcomes		Knowledge Level (K)#
CO1	Apply cyber security architecture principles.	K3
CO2	Demonstrate the risk management processes and practices.	K2
CO3	Appraise cyber security incidents to apply appropriate response	K5
CO4	Distinguish system and application security threats vulnerabilities.	K4
CO5	Identify security tools and hardening techniques	K3

UNIT–I:

Introduction to Cyber Security-Cyber security objectives, roles, differences between information security and cyber security, Cyber security principles- confidentiality, integrity, availability, authentication and non-repudiation

UNIT–II:

Information Security within Lifecycle Management-Lifecycle management landscape, Security architecture processes, Security architecture tools, Intermediate lifecycle management concepts, **Risks & Vulnerabilities-**Basics of risk management, Operational threat environments, Classes of attacks

UNIT–III:

Incident Response-Incident categories, Incident response,Incident recovery, **Operational security protection**-Digital and data assets, ports and protocols, Protection technologies, Identity and access Management, configuration management

UNIT–IV:

Threat Detection and Evaluation Monitoring-Vulnerability management, Security logs and alerts, Monitoring tools and appliances, **Analysis-**Network traffic analysis, packet capture and analysis

UNIT–V:

Introduction to backdoor System and security-Introduction to metasploit, backdoor, demilitarized zone (DMZ), Digital signature, Brief study on Harding of operating system.

Text Books:

1. NASSCOM: Security Analyst Student Hand Book, Dec 2015
2. Information Security Management principles, David Alexander, Amanda Finch, David Sutton, BCS Publishers, 2013

Reference Books:

1. Cyber Security Fundamentals-Cyber Security, Network Security and Data Governance Security, 2nd Edition, ISACA Publishers, 2019

III Semester	L	P	C
	0	3	1.5
20MC3L01 MACHINE LEARNING WITH PYTHON LAB			

Course Objectives:

1. Make use of Data sets in implementing the machine learning algorithms
2. Implement the machine learning concepts and algorithms in any suitable language of choice.
3. Design Python programs for various Learning algorithms.

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

Course Outcomes		Knowledge Level (K)#
CO1	Implement procedures for the machine learning algorithms	**K4**
CO2	Design Python programs for various Learning algorithms	**K6**
CO3	Apply appropriate data sets to the Machine Learning algorithms	**K3**
CO4	Identify and apply Machine Learning algorithms to solve real world problems	**K2**

Experiment 1:
Implement and demonstrate the FIND-S algorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .csv file

Experiment 2:
For a given set of training data examples stored in a .csv file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples

Experiment 3:
Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.

Experiment 4:
Write a Python program to implement k-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions.

Experiment 5:
Build an Artificial Neural Network by implementing the Backpropagation algorithm and test the same using appropriate data sets.

Experiment 6:
Write a program to implement the naive Bayesian classifier for a sample training data set stored as a .csv file. Compute the accuracy of the classifier, considering few test data sets.

Experiment 7:
Write a Python program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set.

Experiment 8:
Assuming a set of documents that need to be classified, use the naive Bayesian Classifier model to perform this task. Built-in Java classes/API can be used to write the program. Calculate the accuracy, precision and recall for your data set.

Experiment 9:
Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data set for clustering using k-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering using Python Programming.

Experiment 10:
Implement the non-parametric Locally Weighted Regression algorithm in order to fit data points. Select appropriate data set for your experiment and draw graphs.

III Semester	L	P	C
	0	3	1.5
20MC3L02 WEB TECHNOLOGIES LAB			

Course Objectives:

1. To implement the web pages using HTML and apply styles.
2. Able to develop a dynamic webpage by the use of java script.
3. Design to create structure of web page, to store the data in web document, and transport information through web.
4. Able to write a well formed / valid XML document.

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

Course Outcomes		Knowledge Level (K)#
CO1	Create dynamic and interactive web pages using HTML, CSS & Java Script	**K6**
CO2	Experiment with Learn and implement XML concepts	**K3**
CO3	Develop web applications using PHP	**K3**
CO4	Show the Install Tomcat Server and execute client-server programs	**K2**
CO5	Implement programs using Ruby programming	**K3**

Experiment 1:

Develop static pages (using HTML and CSS) of an online book store. The pages should resemble: www.flipkart.com The website should consist the following pages.

a) Home page
b) Registration and user Login
c) User Profile Page
d) Books catalog
e) Shopping Cart
f) Payment By credit card
g) Order Conformations

Experiment 2:

Create and save an XML document on the server, which contains 10 users information. Write a program, which takes User Id as an input and returns the user details by taking the user information from the XML document.

Experiment 3:

Write a PHP script to merge two arrays and sort them as numbers, in descending order.

Experiment 4:

Write a PHP script that reads data from one file and write into another file.

Experiment 5:
Write a PHP script to print prime numbers between 1-50.

Experiment 6:
Validate the Registration, user login, user profile and payment by credit card pages using JavaScript.

Experiment 7:
Write a PHP script to:

a. Find the length of a string.
b. Count no of words in a string.
c. Reverse a string.
d. Search for a specific string.

Experiment 8:
Install TOMCAT web server. Convert the static web pages of assignments 2 into dynamic web pages using servlets and cookies. Hint: Users information (user id, password, credit card number) would be stored in web.xml. Each user should have a separate Shopping Cart.

Experiment 9:
Redo the previous task using JSP by converting the static web pages of assignments 2 into dynamic web pages. Create a database with user information and books information. The books catalogue should be dynamically loaded from the database. Follow the MVC architecture while doing the website.

Experiment 10:
Install a database(Mysql or Oracle). Create a table which should contain at least the following fields: name, password, email-id, phone number(these should hold the data from the registration form). Practice 'JDBC' connectivity. Write a java program/servlet/JSP to connect to that database and extract data from the tables and display them. Experiment with various SQL queries. Insert the details of the users who register with the web site, whenever a new user clicks the submit button in the registration page.

Experiment 11:
Write a JSP which does the following job: Insert the details of the 3 or 4 users who register with the web site (week9) by using registration form. Authenticate the user when he submits the login form using the user name and password from the database.

Experiment 12:
Create a simple visual bean with a area filled with a color. The shape of the area depends on the property shape. If it is set to true then the shape of the area is Square and it is Circle, if it is false. The color of the area should be changed dynamically for every mouse click.

III Semester	L	P	C
	0	3	1.5
20MC3L03 NETWORKS AND SECURITY LAB			

Course Objectives:

- To learn basic understanding of cryptography, how it has evolved, and some key encryption techniques used today.
- To understand and implement encryption and decryption using Ceaser Cipher, Substitution Cipher, Hill Cipher.

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

Course Outcomes		Knowledge Level (K)#
CO1	Implement Data Link layers methods	**K3**
CO2	Demonstrate Networking programs	**K3**
CO3	Apply the knowledge of symmetric cryptography to implement encryption and decryption using Ceaser Cipher, Substitution Cipher, Hill Cipher	**K3**
CO4	Demonstrate the different algorithms like DES, BlowFish, and Rijndael, encrypt the text "Hello world" using Blowfish Algorithm.	**K2**
CO5	Analyze and implement public key algorithms like RSA, Diffie-Hellman Key Exchange mechanism, the message digest of a text using the SHA-1 algorithm	**K4**

List of Experiments:

1. Implement the data link layer farming methods such as character stuffing and bit stuffing.
2. Implement on a data set of characters the three CRC polynomials – CRC 12, CRC 16 andCRC CCIP.
3. Implement Dijkstra's algorithm to compute the Shortest path thru a graph.
4. Take an example subnet graph with weights indicating delay between nodes. Now obtain Routing table art each node using distance vector routing algorithm
5. Take an example subnet of hosts. Obtain broadcast tree for it.
6. Write a C program that contains a string (char pointer) with a value \Hello World'. The program should XOR each character in this string with 0 and displays the result.
7. Write a C program that contains a string (char pointer) with a value \Hello World'. The program should AND or and XOR each character in this string with 127 and display the result
8. Write a Java program to perform encryption and decryption using the following algorithms:
   a) Ceaser Cipher
   b) Substitution Cipher
   c) Hill Cipher

9. Write a Java program to implement the DES algorithm logic
10. Write a C/JAVA program to implement the BlowFish algorithm logic
11. Write a C/JAVA program to implement the Rijndael algorithm logic.
12. Using Java Cryptography, encrypt the text “Hello world” using BlowFish.
13. Create your own key using Java key tool.
    a) Write a Java program to implement RSA Algorithm
    b) Write a Java program to implement Public key Algorithm like El Gamal
    c) Implement the Diffie-Hellman Key Exchange mechanism using HTML and JavaScript. Consider the end user as one of the parties (Alice) and the JavaScript application as other party (bob).
    d) Calculate the message digest of a text using the SHA-1 algorithm in JAVA.
    e) Calculate the message digest of a text using the MD5 algorithm in JAVA.

IV Semester	L	P	C
	3	0	3
20MC4TE1 DIGITAL MARKETING			

Course Objectives:

Digital marketing aims at being SMART (Specific, Measurable, Achievable, Relevant and Time Related) so that people can withstand against competitors.

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

Course Outcomes		Knowledge Level (K)#
CO1	Explain about web pages with basic HTML5, DHTML tags using CSS and XML, the overview of W3C DOM	**K1**
CO2	Discuss the key elements of a digital Java Scripts	**K1**
CO3	Apply search engine optimization techniques to a website	**K4**
CO4	Illustrate how the effectiveness of a digital marketing campaign can be measured	**K2**
CO5	Demonstrate advanced practical skills in common digital marketing tools such as SEO, SEM, Social media and Blogs	**K4**

UNIT I:

HTML: Introduction, HTML5, Audio Elements, Video Elements, Organizing Elements. **Scripting Documents:** Dynamic Document content, Document properties, Legacy DOM, Document Collections, Overview of the W3C DOM, Traversing a Document, Finding Elements in a Document, Modifying a Document, Adding Content to a Document Example

UNIT II:

Cascading Style Sheets and Dynamic HTML: Overview of CSS, CSS for DHTML Scripting inline Styles, Scripting computed styles, Scripting CSS Classes, Scripting Style Sheets, **Java Script and XML:** Obtaining XML Documents, Manipulating XML with the DOM API, Transforming XML with XSLT querying XML with X path, Serializing XML, Example, XML and Web services.

UNIT III:

Search Engine Optimization (SEO): Searching Engine Marketing, Search Engine Optimization, Measuring SEO Success, Mapping with SEO Journey, **Search Advertising:** Online Advertising Payment Models, Search Advertising (Desktop & Mobile Devices), Planning &

Executing a search Advertising Camping, Strategic Implications of Advertising on the search Network.

UNIT IV:

Search Media Marketing: What is Social Media? Social Media Marketing, Social Media Marketing Strategy, Adopting Social Media in Organizations: Internal Learning, Paid-Owned-Earned Media, Social CRM, **Mobile Marketing:**

Mobile Internet in India, What is Mobile Marketing? Email Marketing Strategy, Forms of Mobile Marketing, Mobile Advertising, M-Commerce.

UNIT V:

E-Mail Marketing: E-Mail Marketing in India, What is E-Mail Marketing? E-Mail Marketing Strategy, Executing E-Mail Marketing, **Internet Marketing:**

Internet Marketing Strategy, Content Marketing, Content Marketing in India.

Text Books:

1. The Art of Digital Marketing: The Definitive Guide to Creating Strategic, Targeted, and Measurable Online Campaigns, Ian Dodson, Wiley, 2016
2. Programming the World Wide Web, Robet W Sebesta, Pearson, 8th edition, 2015

Reference Books:

1. Fundamentals of Digital Marketing, Second Edition, Pearson Paperback, 2019
2. Internet Marketing- A Practical approach in the India Context by Moutusy Maity, Oxford
3. Java Script: The Definite Guide David Flanagan, O' Reilly Publisher

IV Semester	L	P	C
	3	0	3
20MC4TE2 BLOCK CHAIN TECHNOLOGIES			

Course Objectives:

1. Impart strong technical understanding of Blockchain technologies
2. Develop familiarity of current technologies, tools, and implementation strategies
3. Introduce application areas, current practices, and research activity

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

Course Outcomes		Knowledge Level (K)#
CO1	Demonstrate the foundation of the Blockchain technology and understand the processes in payment and funding.	K2
CO2	Identify the risks involved in building Blockchain applications.	K5
CO3	Review of legal implications using smart contracts.	K4
CO4	Choose the present landscape of Blockchain implementations and Understand Cryptocurrency markets.	K3
CO5	Examine how to profit from trading cryptocurrencies.	K3

UNIT – I:

The consensus problem, Asynchronous Byzantine Agreement, AAP protocol and its analysis, Nakamoto Consensus on permission-less, nameless, peer-to-peer network, Abstract Models for BLOCKCHAIN, GARAY model, RLA Model, Proof of Work (PoW) as random oracle, formal treatment of consistency, liveness and fairness - Proof of Stake (PoS) based Chains, Hybrid models (PoW + PoS).

UNIT – II:

cryptographic basics for cryptocurrency, A short overview of Hashing, signature schemes, encryption schemes and elliptic curve cryptography

UNIT – III:

Bitcoin, Wallet, Blocks, Merkley Tree, hardness of mining, transaction verifiability, anonymity, forks, double spending, mathematical analysis of properties of Bitcoin.

UNIT – IV:

Ethereum: Ethereum Virtual Machine (EVM),Wallets for Ethereum, Solidity, Smart Contracts, some attacks on smart contracts

UNIT – V:

(Trends and Topics): Zero Knowledge proofs and protocols in Blockchain, Succinct non interactive argument for Knowledge (SNARK), pairing on Elliptic curves ,Zcash.

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. (Free download available)

Reference Books:

1. Joseph Bonneau et al, SoK: Research perspectives and challenges for Bitcoin and cryptocurrency, IEEE Symposium on security and Privacy, 2015 (article available for free download) {curtain raiser kind of generic article, written by seasoned experts and pioneers}.

2. J.A.Garay et al, The bitcoin backbone protocol - analysis and applications EUROCRYPT 2015 LNCS VOl 9057, (VOLII), pp 281-310. (Also available at eprint.iacr.org/2016/1048). (serious beginning of discussions related to formal models for bitcoin protocols).
3. R.Pass et al, Analysis of Blockchain protocol in Asynchronous networks , EUROCRYPT 2017, (eprint.iacr.org/2016/454) . A significant progress and consolidation of several principles).

IV Semester	L	P	C
	3	0	3
20MC4TE3 DEEP LEARNING			

Course Objectives:

- Demonstrate the major technology trends driving Deep Learning
- Build, train and apply fully connected deep neural networks
- Implement efficient (vectorized) neural networks
- Analyze the key parameters and hyper parameters in a neural network's architecture

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

Course Outcomes		Knowledge Level (K)#
CO1	Demonstrate the mathematical foundation of neural network	**K4**
CO2	Describe the machine learning basics	**K1**
CO3	Compare the different architectures of deep neural network	**K4**
CO4	Build a convolutional neural network	**K2**
CO5	Build and train RNN and LSTMs	**K4**

UNIT I:

Linear Algebra: Scalars, Vectors, Matrices and Tensors, Matrix operations, types of matrices, Norms, Eigen decomposition, Singular Value Decomposition, Principal Components Analysis.

Probability and Information Theory: Random Variables, Probability Distributions, Marginal Probability, Conditional Probability, Expectation, Variance and Covariance, Bayes' Rule, Information Theory. Numerical Computation: Overflow and Underflow, Gradient-Based Optimization, Constrained Optimization, Linear Least Squares.

UNIT II:

Machine Learning: Basics and Underfitting, Hyper parameters and Validation Sets, Estimators, Bias and Variance, Maximum Likelihood, Bayesian Statistics, Supervised and Unsupervised Learning, Stochastic Gradient Descent, Challenges Motivating Deep Learning. Deep Feedforward Networks: Learning XOR, Gradient-Based Learning, Hidden Units, Architecture Design, Back-

Propagation and other Differentiation Algorithms.

UNIT III:

Regularization for Deep Learning: Parameter Norm Penalties, Norm Penalties as Constrained Optimization, Regularization and Under-Constrained Problems, Dataset Augmentation, Noise Robustness, Semi-Supervised Learning, Multi-Task Learning, Early Stopping, Parameter Tying and Parameter Sharing, Sparse Representations, Bagging and Other Ensemble Methods, Dropout, Adversarial Training, Tangent Distance, Tangent Prop and Manifold Tangent Classifier. Optimization for Training Deep Models: Pure Optimization, Challenges in Neural Network Optimization, Basic Algorithms, Parameter Initialization Strategies, Algorithms with Adaptive Learning Rates, Approximate Second-Order Methods, Optimization Strategies and Meta-Algorithms.

UNIT IV:

Convolutional Networks: The Convolution Operation, Pooling, Convolution, Basic Convolution Functions, Structured Outputs, Data Types, Efficient Convolution Algorithms, Random or Unsupervised Features, Basis for Convolutional Networks.

UNIT V:

Sequence Modeling: Recurrent and Recursive Nets: Unfolding Computational Graphs, Recurrent Neural Networks, Bidirectional RNNs, Encoder-Decoder Sequence-to-Sequence Architectures, Deep Recurrent Networks, Recursive Neural Networks, Echo State Networks, LSTM, Gated RNNs, Optimization for Long-Term Dependencies, Auto encoders, Deep Generative Models.

Text Books:

1) Ian Goodfellow, Yoshua Bengio, Aaron Courville, “Deep Learning”, MIT Press,2016.
2) Josh Patterson and Adam Gibson, “Deep learning: A practitioner's approach”, O'Reilly Media, First Edition, 2017.

Reference Books:

1) Fundamentals of Deep Learning, Designing next-generation machine intelligence algorithms, Nikhil Buduma, O’Reilly, Shroff Publishers, 2019.
2) Deep learning Cook Book, Practical recipes to get started Quickly, Douwe Osinga, O’Reilly, Shroff Publishers, 2019.

e-Resources:

1) https://keras.io/datasets/

2) http://deeplearning.net/tutorial/deeplearning.pdf
3) https://arxiv.org/pdf/1404.7828v4.pdf
4) https://github.com/lisa-lab/DeepLearningTutorials

IV Semester	L	P	C
	3	0	3
20MC4TE4 SOFTWARE TESTING METHODOLOGIES			

Course Objectives:

1. To study fundamental concepts in software testing and discuss various software testing issues and solutions in software unit, integration, regression and system testing
2. To learn how to plan a test project, design test cases and data, conduct testing, manage software problems and defects, generate a test report
3. To expose the advanced software testing concepts such as object-oriented software testing methods, web-based and component-based software testing
4. To understand software test automation problems and solutions
5. To learn how to write software test documents and communicate with engineers in various forms

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

	Course Outcomes	Knowledge Level (K)#
CO1	Identify and understand various software testing problems, apply software testing knowledge and engineering methods and solve these problems by designing and selecting software test models, criteria, strategies, and methods	**K4**
CO2	Design and conduct a software test process for a software project	**K3**
CO3	Analyze the needs of software test automation	**K3**
CO4	Use various communication methods and skills to communicate with their teammates to conduct their practice-oriented software testing projects	**K2**
CO5	Write test cases for given software to test it before delivery to the customer and write test scripts for both desktop and web based applications	**K4**

UNIT I:

Software Testing: Introduction, Evolution, Myths & Facts, Goals, Psychology, definition, Model for testing, Effective Vs Exhaustive Software Testing.

Software Testing Terminology and Methodology: Software Testing Terminology, Software Testing Life Cycle, Software Testing Methodology. Verification and Validation: Verification & Validation Activities, Verification, Verification of Requirements, High level and low level designs, verifying code, Validation

UNIT II:

Dynamic Testing-Black Box testing techniques: Boundary Value Analysis, Equivalence class Testing, State Table based testing, Decision table based testing, Cause-Effect Graphing based testing, Error guessing

White-Box Testing: need, Logic Coverage criteria, Basis Path testing, Graph matrices, Loop testing, data flow testing, mutation testing

UNIT III:

Static Testing: Inspections, Structured Walkthroughs, Technical Reviews

Validation activities: Unit testing, Integration Testing, Function testing, system testing, acceptance testing

Regression testing: Progressives Vs regressive testing, Regression test ability, Objectives of regression testing, Regression testing types, Regression testing techniques

UNIT IV:

Efficient Test Suite Management: growing nature of test suite, Minimizing the test suite and its benefits, test suite prioritization, Types of test case prioritization, prioritization techniques, measuring the effectiveness of a prioritized test suite. Software Quality Management: Software Quality metrics, SQA models Debugging: process, techniques, correcting bugs.

UNIT V:

Automation and Testing Tools: need for automation, categorization of testing tools, selection of testing tools, Cost incurred, Guidelines for automated testing, overview of some commercial testing tools such as Win Runner, Load Runner, Jmeter and JUnit . Test Automation using Selenium tool.

Testing Object Oriented Software: Basics, Object oriented testing Web based Systems: Challenges in testing for web based software, quality aspects, web engineering, testing of web based systems, Testing mobile systems

Text Books:

1) Software Testing, Principles and Practices, Naresh Chauhan, Oxford.
2) Software Testing- Yogesh Singh, CAMBRIDGE.

Reference books:

1) Foundations of Software testing, Aditya P Mathur, 2ed, Pearson.
2) Software testing techniques – Baris Beizer, Dreamtech, second edition.
3) Software Testing, Principles, techniques and Tools, M G Limaye, TMH
4) Effective Methods for Software testing, Willian E Perry, 3ed, Wiley

e-Resources:

https://www.tutorialspoint.com/software_testing_dictionary/test_tools.html

IV Semester	L	P	C
	3	0	3
20MC4TE6 BIG DATA ANALYTICS			

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

Course Outcomes		Knowledge Level (K)#
CO1	Develop Java Structure for Big Data	**K4**
CO2	Analyzing GFS and HDFS Architecture	**K3**
CO3	Developing Map Reduce Programs	**K4**
CO4	Understand Writables	**K2**
CO5	Develop PIG and HIVE Data Models	**K4**

UNIT-I

Data structures in Java: Linked List, Stacks, Queues, Sets, Maps; Generics: Generic classes and Type parameters, Implementing Generic Types, Generic Methods, WrapperClasses, Concept of Serialization

UNIT-II

Working with Big Data: Google File System, Hadoop Distributed File System (HDFS) Building blocks of Hadoop (Name node, Data node, Secondary Name node, Job Tracker, Task Tracker), Introducing and Configuring Hadoop cluster (Local, Pseudo-distributed mode, Fully Distributed mode), Configuring XML files.

UNIT-III

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.

UNIT - IV

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

UNIT – V

Pig: Hadoop Programming Made Easier 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 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

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

Reference Books:

1. Hadoop in Practice by Alex Holmes, MANNING Publ.
2. Hadoop Map Reduce Cookbook, Srinath Perera, Thilina Gunarathne

Software Links:

1. Hadoop:http://hadoop.apache.org/
2. Hive: https://cwiki.apache.org/confluence/display/Hive/Home
3. Piglatin:http://pig.apache.org/docs/r0.7.0/tutorial.html

IV Semester	L	P	C
	3	0	3
20MC4TE7 SOFT COMPUTING			

Course Objectives:

- Develop the skills to gain a basic understanding of neural network theory and fuzzy logic theory.
- Introduce students to artificial neural networks and fuzzy theory from an engineering perspective

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

	Course Outcomes	Knowledge Level (K)#
CO1	Comprehend the fuzzy logic and the concept of fuzziness involved in various systems and fuzzy set theory	K5
CO2	Understand the concepts of fuzzy sets, knowledge representation using fuzzy rules, approximate reasoning, fuzzy inference systems, and fuzzy logic	K3
CO3	To understand the fundamental theory and concepts of neural networks, Identify different neural network architectures, algorithms, applications and their limitations	K3
CO4	Understand appropriate learning rules for each of the architectures and learn several neural network paradigms and its applications	K4
CO5	Reveal different applications of these models to solve engineering and other problems.	K6

UNIT I: Fuzzy Set Theory: Introduction to Neuro, Fuzzy and Soft Computing, Fuzzy Sets, Basic function and Terminology, Set-theoretic Operations, Member Function Formulation and Parameterization, Fuzzy Rules and Fuzzy Reasoning, Extension Principle and Fuzzy Relations, Fuzzy If-Then Rules, Fuzzy Reasoning, Fuzzy Inference Systems, Mamdani Fuzzy Models, Sugeno Fuzzy Models, Tsukamoto Fuzzy Models, Input Space Partitioning and Fuzzy Modeling.

UNIT II: Optimization: Derivative based Optimization, Descent Methods, and The Method of Steepest Descent, Classical Newton's Method, Step Size Determination, Derivative-free Optimization, Genetic Algorithms, Simulated Annealing, and Random Search, Downhill Simplex Search.

UNIT III: Artificial Neural Networks: Introduction and ANN Structure, Biological neurons and artificial neurons. Model of an ANN, Activation functions used in ANNs, Typical classes of network architectures, Single layer perceptrons, Structure and learning of perceptrons. Feed forward ANN, Structures of Multi-layer feed forward networks, back propagation algorithm, Back propagation - training and convergence.

UNIT IV: Neuro Fuzzy Modeling: Adaptive Neuro-Fuzzy Inference Systems, Architecture Hybrid Learning Algorithm, Learning Methods that Cross-fertilize ANFIS and RBFN Coactive Neuro Fuzzy Modeling, Framework Neuron Functions for Adaptive Networks Neuro Fuzzy Spectrum.

UNIT V: Applications Of Computational Intelligence: Printed Character Recognition, Inverse Kinematics Problems, Automobile Fuel Efficiency Prediction, Soft Computing for Coloripe Prediction.

Text Books:

1) "Neuro-Fuzzy and Soft Computing", J.S.R.Jang, C.T.Sun and E.Mizutani, PHI, 2004, Pearson Education 2004
2) Satish Kumar, "Neural Networks: A classroom approach", Tata McGraw Hill, 2004.
3) "Artificial Intelligence and Intelligent Systems", N.P.Padhy, Oxford University Press, 2006

Reference Books:

1) Artificial Intelligence, Second Edition, Elaine Rich & Kevin Knight, Tata McGraw Hill Publishing Comp., New Delhi, , 2nd edition-2006
2) "Fuzzy Logic with Engineering Applications", Timothy J.Ross, McGraw-Hill, 3rd edition-1997

IV Semester	L	P	C
	3	0	3
20MC4TE8 SOFTWARE DEFINED NETWORKS			

Course Objectives:

1. To learn threats and risks within context of the cyber security architecture.
2. Student should learn and Identify security tools and hardening techniques.
3. To learn types of incidents including categories, responses and timelines for response.

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

Course Outcomes		Knowledge Level (K)#
CO1	Explain the key benefits of SDN by the separation of data and control planes	K5
CO2	Interpret the SDN data plane devices and Openflow Protocols	K3
CO3	Apply the operation of SDN control plane with different controllers	K3
CO4	Apply techniques that enable applications to control the underlying network using SDN	K4
CO5	Design Network Functions Virtualization components and their roles in SDN	K6

UNIT-I:

SDN Background and Motivation-Evolving network requirements-The SDN Approach: Requirements, SDN Architecture, Characteristics of Software-Defined Networking, SDN and NFV-Related Standards: Standards-Developing Organizations, Industry Consortia, Open Development Initiatives.

UNIT-II:

SDN Data plane and OpenFlow-SDN data plane: Data plane Functions, Data plane protocols, Openflow logical network Device: Flow table Structure, Flow Table Pipeline, The Use of Multiple Tables, Group Table- OpenFlow Protocol.

UNIT-III:

SDN Control Plane-SDN Control Plane Architecture: Control Plane Functions, Southbound Interface, Northbound Interface, Routing, ITU-T Model- Open Daylight-REST- Cooperation and Coordination among Controllers.

UNIT-IV:

SDN Application Plane-SDN Application Plane Architecture: Northbound Interface, Network Applications, User Interface- Network

Services Abstraction Layer :Abstractions in SDN, Frenetic-Traffic Engineering Measurement and Monitoring- Security- Data Center Networking- Mobility and Wireless.

UNIT-V:

Network Functions Virtualization- Background and Motivation for NFV- Virtual Machines- NFV Concepts: Simple Example of the Use of NFV,NFV Principles, High-Level NFV Framework, NFV Benefits and Requirements- NFV Reference Architecture: NFV Management and Orchestration

Text Books:

1. William Stallings, "Foundations of Modern Networking",Pearson Ltd. ,2016.
2. Software Defined Networks: A Comprehensive Approach by Paul Goransson and Chuck Black,Morgan Kaufmann Publications, 2014
3. SDN - Software Defined Networks by Thomas D. Nadeau & Ken Gray, O'Reilly, 2013

Reference Books:

1. Feamster, Nick, Jennifer Rexford, and Ellen Zegura. "The road to SDN: an intellectual historyof programmable networks." ACM SIGCOMM Computer Communication Review 44.2 (2014): 87-98.
2. Kreutz, Diego, et al. "Software-defined networking: A comprehensive survey." Proceedingsof the IEEE 103.1 (2015): 14-76.

Web Reference:

1. https://www.coursera.org/learn/sdn

IV Semester	L	P	C
	3	0	3

20MC4TE9 NETWORK PROGRAMMING

Course Objectives:

1. Student able to learn about the protocols which are using in the current scenario.
2. To learn and understand client server relations and OSI programming Implementation of the socket and IPC.

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

	Course Outcomes	Knowledge Level (K)#
CO1	Explain OSI Model and Standard Internet Protocols	K5
CO2	How to handle server process termination	K3
CO3	Acquire the knowledge of Elementary TCP sockets and I/O Multiplexing and socket	K3
CO4	Demonstrate UDP Sockets	K4
CO5	Demonstrate the concepts of FIFOs streams messages and Remote logins	K6

UNITI:

Introduction to Network Programming: OSI model, Unix standards, TCP and UDP & TCP connection establishment and Format, Buffer sizes and limitation, standard internet services, Protocol usage by common internet application.

UNIT II:

TCP client server: Introduction, TCP Echo server functions, Normal startup, terminate and signal handling server process termination, Crashing and Rebooting of server host shutdown of server host.

UNIT III:

Sockets: Address structures, value – result arguments, Byte ordering and manipulation function and related functions Elementary TCP sockets – Socket, connect, bind, listen, accept, fork and exec function, concurrent servers. Close function and related function.

I/O Multiplexing and socket options: I/O Models, select function, Batch input, shutdown function, poll function, TCP Echo server, getsockopt and setsockopt functions. Socket states, Generic socket option IPV6 socket option ICMPV6 socket option IPV6 socket option and

TCP socket options.

UNIT IV:

Elementary UDP sockets: Introduction UDP Echo server function, lost datagram, summary of UDP example, Lack of flow control with UDP, determining outgoing interface with UDP.

Elementary name and Address conversions: DNS, gethost by Name function, Resolver option, Function and IPV6 support, uname function, other networking information.

UNIT V:

IPC: Introduction, File and record locking, Pipes, FIFOs streams and messages, Name spaces, system IPC, Message queues, Semaphores. **Remote Login:** Terminal line disciplines, Pseudo-Terminals, Terminal modes, Control Terminals, rlogin Overview, RPC Transparency Issues.

Textbooks:

1) UNIX Network Programming, Vol. I, SocketsAPI, 2nd Edition. - W.Richard Stevens, Pearson Edn. Asia.
2) UNIX Network Programming, 1st Edition, - W.Richard Stevens. PHI.

References Books:

1) UNIX Systems Programming using C++ T CHAN, PHI.
2) UNIX for Programmers and Users, 3rd Edition Graham GLASS, King abls, Pearson Education
3) Advanced UNIX Programming 2nd Edition M. J. ROCHKIND, Pearson Education