GENERAL COURSE STRUCTURE AND CREDIT DISTRIBUTION

CREDIT DISTRIBUTION

POSTGRADUATE PROGRAMME

Scheme of Programme Master of Computer Applications (MCA) Integrated

(Scheme PG A1: Postgraduate Programmes (Course work only))

Semester 1

Course Code	Course Title	Course ID	L	T	P	Credits	TI	TE	PI	PE	Total
Discipline Specific Courses (DSC)											
	Fundamentals of Computer Programming	242/MCA/CC101	3	-	2	4	25	50	5	20	100
	Computer Organization and Architecture	242/MCA/CC102	3	-	2	4	25	50	5	20	100
MIC/Vocational Courses (VOC)											
	One from pool	242/MCA/VO101	2	-	4	4	15	35	15	35	100
Multidisciplinary Courses (MDC)											
	One from pool	242/MCA/MD101	2	1	-	3	15	35	5	20	75
Ability Enhancement Course (AEC)											
	One from pool	242/MCA/AE101	2	0	0	2	15	35	-	-	50
Skill Enhancement Course (SEC)											
	One from pool	242/MCA/SE101	2	-	2	3	15	35	5	20	75
Value Addition Course (VAC)											
	One from pool	242/MCA/VA101	2	-	-	2	15	35	-	-	50
Total Credits						**22**	**Total Marks**				**550**

Semester 2

Course Code	Course Title	Course ID	L	T	P	Credits	TI	TE	PI	PE	Total
Discipline Specific Courses (DSC)											
	Data Structure and Algorithm	242/MCA/DSC201	3	-	2	4	25	50	5	20	100
	Operating System	242/MCA/DSC202	3	-	2	4	25	50	5	20	100
MIC/Vocational Courses (VOC)											
	One from pool	242/MCA/VO201	2	-	2	4	15	35	15	35	100
Multidisciplinary Courses (MDC)											
	One from pool	242/MCA/MD201	2	1	-	3	25	50	-	-	75
Ability Enhancement Course (AEC)											
	One from pool	242/MCA/AEC201	2	-	-	2	15	35	-	-	50
Skill Enhancement Course (SEC)											
	One from pool	242/MCA/SE201	2	-	1	3	15	35	5	20	75
Value Addition Course (VAC)											
	One from pool	242/MCA/VA201	2	-	-	2	15	35	-	-	50
Total Credits						**22**	**Total Marks**				**550**

Semester 3

Course Code	Course Title	Course ID	L	T	P	Credits	TI	TE	PI	PE	Total
Discipline Specific Courses (DSC)											
	Data Base Management System	242/MCA/DSC301	3		2	4	25	50	5	20	100
	Foundation of Web technology	242/MCA/DSC302	3		2	4	25	50	5	20	100
MIC/Vocational Courses (VOC)											
	One from pool	242/MCA/VO301	2	-	2	4	15	35	15	35	100
Multidisciplinary Courses (MDC)											
	One from pool	242/MCA/MD301	3	-	-	3	25	50	-	-	75
Ability Enhancement Course (AEC)											
	One from pool	242/MCA/AE301	2	-	-	2	15	35	-	-	50
Skill Enhancement Course (SEC)											
	One from pool	242/MCA/SE301	2	-	1	3	15	35	5	20	75
Value Addition Course (VAC)											
	One from pool	242/MCA/VA301	2	-	-	2	15	35	-	-	50
Total Credits						**22**	**Total Marks**				**550**

Semester 4

Course Code	Course Title	Course ID	L	T	P	Credits	TI	TE	PI	PE	Total
Discipline Specific Courses (DSC)											
	Object oriented programming using Python	242/MCA/DSC401	3		2	4	25	50	5	20	100
	Analysis and Design of Algorithms	242/MCA/DSC402	3		2	4	25	50	5	20	100
	Data Communication and Networking	242/MCA/DSC403	3		2	4	25	50	5	20	100
	Multi Media Technology	242/MCA/DSC404	3		2	4	25	50	5	20	100
Vocational Courses (VOC)											
	One from pool	242/MCA/VO401	2	-	4	4	15	35	15	35	100
Ability Enhancement Course (AEC)											
	One from pool	242/MCA/AEC401	2	-	-	2	15	35	-	-	50
Value Addition Course (VAC)											
	One from pool	242/MCA/VA401	2	-	-	2	15	35	-	-	50
Total Credits						**24**	**Total Marks**				**600**

Semester 5

Course Code	Course Title	Course ID	L	T	P	Credits	TI	TE	PI	PE	Total
Discipline Specific Courses (DSC)											
	Software Engineering	242/MCA/DSC501	3	-	2	4	25	50	5	20	100
	Artificial Intelligence	242/MCA/DSC502	3	-	2	4	25	50	5	20	100
	Mathematical foundation of Computer Science	242/MCA/DSC503	3	1	-	4	30	70	-	-	100
	Java programming	242/MCA/DSC504	3	-	2	4	25	50	5	20	100
Vocational Courses (VOC)											
	One from pool	242/MCA/VO501	2	-	4	4	15	35	15	35	100
Skill Enhancement Course (SEC) / Summer Internship Report											
	One from pool	242/MCA/SE501	-	-	8	4	-	-	30	70	100
Total Credits						**24**	**Total Marks**			**600**	

Semester 6

Course Code	Course Title	Course ID	L	T	P	Credits	TI	TE	PI	PE	Total
Discipline Specific Courses (DSC)											
	Compiler Design	242/MCA/DSC601	3		2	4	25	50	5	20	100
	Computer Graphics	242/MCA/DSC602	3		2	4	25	50	5	20	100
	Shell Programming using Linux	242/MCA/DSC603	3		2	4	25	50	5	20	100
	Information Retrieval Systems	242/MCA/DSC604	3		2	4	25	50	5	20	100
Vocational Courses (VOC)											
	One from pool	242/MCA/VO601	2	-	4	4	15	35	15	35	100
Skill Enhancement Course (SEC)											
	One from pool	242/MCA/SE601	1	-	2	2	5	20	5	20	50
Total Credits						**22**	**Total Marks**			**550**	

Semester 7

Course Code	Course Title	Course ID	L	T	P	Credits	TI	TE	PI	PE	Total
Discipline Specific Courses (DSC)											
	Cryptography and Network Security	242/MCA/DSC701	3	-	2	4	25	50	5	20	100
	Software testing and Quality Assurance	242/MCA/DSC702	3	-	2	4	25	50	5	20	100
	Digital Image processing	242/MCA/DSC703	3	-	2	4	25	50	5	20	100
	Fundamentals of Machine Learning	242/MCA/DSC704	3	-	2	4	25	50	5	20	100
	Big Data Analytics	242/MCA/DSC705	3	-	2	4	25	50	5	20	100
Vocational Course (VOC)											
	One from Pool	242/MCA/VO701	2	-	4	4	15	35	15	35	100
Total Credits						**24**	**Total Marks**			**600**	

Semester 8

Course Code	Course Title	Course ID	L	T	P	Credits	TI	TE	PI	PE	Total
Discipline Specific Courses (DSC)											
	Software Project Management	242/MCA/DSC801	3	-	2	4	25	50	5	20	100
	Mobile Computing	242/MCA/DSC802	3	1	-	4	30	70	-	-	100
	Pattern Recognition	242/MCA/DSC803	3	1	-	4	30	70	-	-	100
	Foundation of Data Science	242/MCA/DSC804	3	-	2	4	25	50	5	20	100
	Soft Computing	242/MCA/DSC805	3	-	2	4	25	50	5	20	100
Vocational Course (VOC)											
	One from Pool	242/MCA/VO801	3	1	-	4	30	70	-	-	100

Total Credits	24	Total Marks	600

Semester 9

Course Code	Course Title	Course ID	L	T	P	Credits	TI	TE	PI	PE	Total
Discipline Specific Courses (DSC)											
	Distributed Systems	242/MCA/DSC901	3	1	-	4	30	70	-	-	100
	Data Warehouse and Mining	242/MCA/DSC902	3	1	-	4	30	70	-	-	100
	Block Chain technology	242/MCA/DSC903	3	1	-	4	30	70	-	-	100
	Internet of Things and its application	242/MCA/DSC904	3	-	2	4	25	50	5	20	100
Skill Enhancement Course (SEC)											
	One from Pool	242/MCA/SE901	2	-	-	2	15	35	-	-	50
Value Addition Course (VAC)											
	Seminar	242/MCA/VA901	2	-	-	2	-	-	-	-	50
Total Credits						**20**	**Total Marks**				**500**

Semester 10

Course Code	Course Title	Course ID	L	T	P	Credits	TI	TE	PI	PE	Total
Discipline Specific Courses (DSC)											
	Social Network Analysis	242/MCA/DSC1001	3		2	4	25	50	5	20	100
Skill Enhancement Course (SEC)/ Project Report											
						12			90	210	300
Discipline Specific Elective Course(s)											
	Deep Learning	242/MCA/DSC1002	3		2	4	25	50	5	20	100
Total Credits						**20**	**Total Marks**			**500**	

#Four credits of internship earned by a student during summer internship after 2nd semester will be counted in 3rd semester of a student who pursue 2-year PG Programme without taking exit option.

Semester 1

Course code				
Category	Discipline Specific Courses (DSC)			
Course title	Fundamentals of Computer Programming			
Scheme and Credits	L	T	P	Credits
	3	-	2	4
Theory Internal	25			
Theory External	50			
Practical Internal	05			
Practical External	20			
Total	100			
Duration of Exam	3 hrs			

Note: *Examiner will be required to set NINE questions with all questions carrying equal marks. Question Number 1, covering the entire syllabus, will be compulsory. Examiner will set two questions from each Unit with internal choice. Student will be required to attempt FIVE questions in all, selection one question from every unit apart from the Question Number 1.*

Course outcomes:

CO1: Learn the functional units and classify types of computers, their applications and effects.
CO2: Understand system software's and their working.
CO3: Understand the logic building used in programming.
CO4: Design and develop algorithms for solving various real-life problems.
CO5: Design and develop programs using C.

UNIT-I

Computer Fundamentals: Concept of data and information. Components of Computer. Input and Output Device, Components of CPU, Memory and Storage Devices, Classification of Computers, Advantages and Limitations of Computer, Applications of Computer, Social concerns of Computer Technology: Positive and Negative Impacts, Computer Crimes, Viruses and their remedial solutions.

Computer Software: System and Application Software, Overview of Operating System Programming Languages Machine. Assembly. High Level Language, 4GL. Language Translator, Linker and Loader.

UNIT-II

Problem Solving: Problem Identification. Analysis, Algorithms, Flowcharts. Pseudo codes. Decision Tables. Program Coding. Program Testing and Execution.

C Programming Fundamentals: Keywords, Variables and Constants, Structure of a C program.

UNIT-Ill

Operators & Expressions: Arithmetic, Unary, Logical. Bit-wise, Assignment & Conditional Operators.

Decision Making: Decision making using if...else. Else If Ladder; Switch, break. Continue and Goto statements.

UNIT-IV

Loops: Looping using while, do...while, for statements. Nested loops.

Functions: Defining & Accessing User defined functions. Library Functions, Function Prototype, Passing Arguments, Passing array as argument. Recursion, Use of Library Functions. Macro vs. Functions, Pointers in C.

Textbooks & Reference Books:

1. E. Balaguruswamy: Programming in C. Tata McGraw Hill.
2. Rajender Singh Chhillar: Application of IT to Business, Ramesh Publishers, Jaipur.
3. Gill Nasib Singh: Computing Fundamentals and Programming in C, Khanna Books Publishing Co., New Delhi.

Fundamentals of Computer Programming
List of Experiments

1. Write a C program to compute roots of quadratic equation $ax^2+bx+c=0$, where a, b, and c are three coefficients of a quadratic equation are inputs.
2. Design and develop an algorithm to find the reverse of an integer number.
3. Design and develop an algorithm to check whether given number is PALINDROME or NOT. Implement a C program for the developed algorithm that takes an integer number as input and output the reverse of the same with suitable messages. Ex: Num: 2019, Reverse: 9102, Not a Palindrome.
4. Design and develop a c program to implement simple calculator using switch case statement.
5. Develop, implement and execute a C program to search a Number in a list using linear searching Technique.
6. Develop an algorithm, implement and execute a C program that reads N integer numbers and arrange them in ascending order using Bubble Sort.
7. Design and develop a C program to read and print a matrix and check whether a given Matrix is a sparse Matrix or not.
8. Write a C program to implements the following string manipulation functions till the use wishes to continue (infinite loop): (i) strcpy() (ii) srrlen() (iii) strrev () (iv) strcmp() (v) strcat(). b. Read a sentence and print frequency of vowels and total count of consonants.
9. Design and develop a C function RightRotate (x, n) that takes two integers x and n as input and returns value of the integer x rotated to the right by n positions. Assume the integers are unsigned.
10. Draw the flowchart and write a recursive C function to find the factorial of a number, n!, define by fact(n)=1, if n=0. Otherwise fact (n) =n*fact (n-1). Using this function, write a C program to compute the binomial coefficient nCr. Tabulate the results for different values of n and r with suitable messages
11. a. Write a C program to maintain a record of n student details using an array of structures with four fields (Roll number, Name, Marks, and Grade). Assume appropriate data type for each field. Input & Print the members of the structure

b. Write a C program to take 2 structures HH:MM: SS as T1 & T2 & display the time difference as structure as T3.

12. Write a C program using pointers to compute the sum, mean and standard deviation of all elements stored in an array of n real numbers.

Course code				
Category	Discipline Specific Courses (DSC)			
Course title	Computer Organization and Architecture			
Scheme and Credits	L	T	P	Credits
	3	-	2	4
Theory Internal	25			
Theory External	50			
Practical Internal	05			
Practical External	20			
Total	100			
Duration of Exam	3 hrs			

Note: The examiner will set nine questions in total. Question one will have seven parts from all units and the marks of first question will be of 20% of total marks of Question Paper and the remaining eight questions to be set by taking two questions from each unit and the marks of each question from Question no.2 to 9 will be of 20% of total marks of Question paper. The students have to attempt five questions in total, the first being compulsory and selecting one from each unit.

Course outcomes:

CO1. Understand the theory and architecture of central processing unit.
CO2. Analyse some of the design issues in terms of speed, technology, cost, performance.
CO3. Design a simple CPU with applying the theory concepts.
CO4. Use appropriate tools to design verify and test the CPU architecture.
CO5. Learn the concepts of parallel processing, pipelining and inter processor communication.

UNIT-1

Boolean algebra and logic gates

Boolean algebra and Logic gates, Combinational logic blocks(Adders, Multiplexers, Encoders, de-coder), Sequential logic blocks(Latches, Flip-Flops, Registers, Counters) Store program control concept, Flynn's classification of computers (SISD, MISD, MIMD); Multilevel viewpoint of a machine: digital logic, micro architecture, ISA, operating systems, high level language; structured organization; CPU, caches, main memory, secondary memory units & I/O; Performance metrics; MIPS, MFLOPS.

UNIT -2

Instruction set architecture

Instruction Set Architecture Instruction set based classification of processors (RISC, CISC, and their comparison); addressing modes: register, immediate, direct, indirect, indexed; Operations in the instruction set; Arithmetic and Logical, Data Transfer, Control Flow; Instruction set formats (fixed, variable, hybrid); Language of the machine: 8086 ; simulation using MSAM.

UNIT -3

Memory hierarchy

Basic non pipelined CPU Architecture and Memory Hierarchy & I/O Techniques CPU Architecture types (accumulator, register, stack, memory/ register) detailed data path of a typical register based CPU, Fetch-Decode-Execute cycle (typically 3 to 5 stage); microinstruction sequencing, implementation of control unit, Enhancing performance with pipelining. The need for a memory hierarchy (Locality of reference principle, Memory hierarchy in practice: Cache, main memory and secondary memory, Memory parameters: access/ cycle time, cost per bit); Main memory (Semiconductor RAM & ROM organization, memory expansion, Static & dynamic memory types); Cache memory (Associative & direct mapped cache organizations.

UNIT- 4

Parallelism and interrupts

Introduction to Parallelism and Computer Organization

[80x86] Goals of parallelism (Exploitation of concurrency, throughput enhancement); Amdahl's law; Instruction level parallelism (pipelining, super scaling –basic features); Processor level parallelism (Multiprocessor systems overview). Instruction codes, computer register, computer instructions, timing and control, instruction cycle, type of instructions, memory reference, register reference. I/O reference, Basics of Logic Design, accumulator logic, Control memory, address sequencing, micro-instruction formats, micro-program sequencer, Stack Organization, Instruction Formats, Types of interrupts; Memory Hierarchy.

TEXT AND REFERENCE BOOKS:

1. Computer Organization and Design, 2nd Ed., by David A. Patterson and John L. Hennessy, Morgan 1997, Kauffmann.
2. Computer Architecture and Organization, 3rd Edi, by John P. Hayes, 1998, TMH.
3. Operating Systems Internals and Design Principles by William Stallings,4th edition, 2001, Prentice-Hall Upper Saddle River, New Jersey
4. Computer Organization, 5th Edi, by Carl Hamacher, Zvonko Vranesic,2002, Safwat Zaky.
5. Structured Computer Organisation by A.S. Tanenbaum, 4th edition, Prentice-Hall of India, 1999, Eastern Economic Edition.
6. Computer Organisation & Architecture: Designing for performance by W. Stallings, 4th edition, 1996, Prentice-Hall International edition.
7. Computer System Architecture by M. Mano, 2001, Prentice-Hall. 6.Computer Architecture- Nicholas Carter, 2002, T.M.H.
8. Gill, Nasib Singh and Dixit J.B.: Digital Design and Computer Organisation, University Science Press (Laxmi Publications), New Delhi

Semester 2

Course code				
Category	**Discipline Specific Courses**			
Course title	Data Structure and Algorithm			
Scheme and Credits	L	T	P	Credits
	3	-	2	4
TI	25			
TE	50			
PI	5			
PE	25			
Duration of Exam	3 HRS			

Note*: The examiner will set nine questions in total. Question one will have seven parts from all units and the marks of first question will be of 20% of total marks of Question Paper and the remaining eight questions to be set by taking two questions from each unit and the marks of each question from Question no.2 to 9 will be of 20% of total marks of Question paper . The students have to attempt five questions in total, the first being compulsory and selecting one from each unit.*

Course outcomes

CO1. List or describe types of data structures and operations that can be implemented on these data structures.

CO2. Demonstrate the use of various data structure and their related operations

CO3. Apply appropriate data structures with respect to effective storage of data and efficiency of the required operations on data for solving real world problems.

CO4. Analyse the time complexity of searching and algorithms.

CO5. formulate data structures and prescribe operations for given real world situations.

UNIT – I

Introduction: Elementary data organization, Data Structure definition, Data type vs. data structure, Categories of data structures, Data structure operations, Applications of data structures, Algorithms complexity and time-space tradeoff, Big-O notataion. Strings: Introduction, Storing strings, String operations, Pattern matching algorithms.

UNIT – II

Arrays: Introduction, Linear arrays, Representation of linear array in memory, address calculations, Traversal, Insertions, Deletion in an array, Multidimensional arrays, Parallel arrays, Sparse arrays.

Linked List: Introduction, Array vs. linked list, Representation of linked lists in memory, Traversal, Insertion, Deletion, Searching in a linked list, Header linked list, Circular linked list, Two-way linked list, Threaded lists, Garbage collection, Applications of linked lists.

UNIT – III

Stack: Introduction, Array and linked representation of stacks, Operations on stacks, Applications of stacks: Polish notation, Recursion. Queues: Introduction, Array and linked representation of queues, Operations on queues, Deques, Priority Queues, Applications of queues.

UNIT – IV

Tree: Introduction, Definition, Representing Binary tree in memory, Traversing binary trees, Traversal algorithms using stacks. Graph: Introduction, Graph theory terminology, Sequential and linked representation of graphs.

SUGGESTED READINGS

1. Seymour Lipschutz, "Data Structure", Tata-McGraw-Hill
2. Horowitz, Sahni & Anderson-Freed, "Fundamentals of Data Structures in C", Orient Longman.
3. Trembley, J.P. And Sorenson P.G., "An Introduction to Data Structures With Applications", Mcgrraw-Hill International Student Edition, New York.
4. Mark Allen Weiss Data Structures and Algorithm Analysis In C, Addison- Wesley, (An Imprint Of Pearson Education), Mexico City.Prentice- Hall Of India Pvt. Ltd., New Delhi.
5. Yedidyan Langsam, Moshe J. Augenstein, and Aaron M. Tenenbaum, "Data Structures Using C", Prentice- Hall of India Pvt. Ltd., New Delhi

Data Structure and Algorithm

List of Experiments

Note: At least 8 experiments are to be performed by the students.

List of Subject related Experiments:

1. Write a program to search an element in a two-dimensional array using linear search.
2. Using iteration & recursion concepts write programs for finding the element in the array using Binary Search Method
3. Write a program to perform following operations on tables using functions only

(a) Addition (b) Subtraction (c) Multiplication (d) Transpose

4. Using iteration & recursion concepts write the programs for Quick Sort Technique

5. Write a program to implement the various operations on string such as length of string concatenation, reverse of a string & copy of a string to another.

6. Write a program for swapping of two numbers using ‘call by value’ and ‘call by reference strategies.

7. Write a program to implement binary search tree.

8. (Insertion and Deletion in Binary search Tree)

9. Write a program to create a linked list & perform operations such as insert, delete, update, reverse in the link list

10. Write the program for implementation of a file and performing operations such as insert, delete, update a record in the file.

11. Create a linked list and perform the following operations on it

(a) add a node (b) Delete a node

Course code				
Category	**Discipline Specific Courses**			
Course title	**Operating System**			
Scheme and Credits	L	T	P	Credits
	3		2	4
TI	25			
TE	50			
PI	5			
PE	20			
Duration of Exam	3 HRS			

Note: The examiner will set nine questions in total. Question one will be compulsory. Question one will have seven parts of 2 marks each from all units, and the remaining eight questions of 14 marks each to be set by taking two questions from each unit. The students have to attempt five questions in total, the first being compulsory and selecting one from each unit.

Course outcomes

1. To understand the mechanisms of OS to handle processes and threads and their communication. 2. To understand the process management mechanisms and scheduling algorithms.

3. To understand the mechanisms involved in memory management in OS and virtual memory concepts.

4. To understand the file management and deadlocks handling techniques in OS.

UNIT – I

Fundamentals of Operating system: Introduction to Operating System, its need and operating System services, Early systems, Structures - Simple Batch, Multi programmed, timeshared, Personal Computer, Parallel, Distributed Systems, Real-Time Systems. Process Management: Process concept, Operation on processes, Cooperating Processes, Threads, and Inter-process Communication.

UNIT-II

CPU Scheduling: Basic concepts, Scheduling criteria, Scheduling algorithms : FCFS, SJF, Round Robin & Queue Algorithms. Deadlocks: Deadlock characterization, Methods for handling deadlocks, Banker'sAlgorithm.

UNIT-III

Memory Management: Logical versus Physical address space, Swapping, Contiguous allocation, Paging, Segmentation. Virtual Memory: Demand paging, Performance of demand paging, Page replacement, Page replacement algorithms, Thrashing.

UNIT-IV

File management: File system Structure, Allocation methods: Contiguous allocation, Linked allocation, Indexed allocation, Free space management: Bit vector, Linked list, Grouping, Counting. Device Management: Disk structure, Disk scheduling: FCFS, SSTF, SCAN, C-SCAN, LOOK, C-LOOK.

Suggested Readings

1. Abraham Silberschatz, Peter B. Galvin, " Operating System Concepts", Addison-Wesley publishing. Co., 7th. Ed., 2004.

2. Nutt Gary, "Operating Systems", Addison Wesley Publication, 2000.

3. Andrew S. Tannenbaum, "Modern Operating Systems", Pearson Education Asia, Second Edition, 2001.

4. William Stallings, "Operating Systems, "Internals and Design Principles", 4th Edition, PH, 2001.

5. Ekta Walia, "Operating Systems Concepts", Khanna Publishes, New Delhi, 2002

LAB: List of Experiments

CONTENTS

1. Basics of UNIX

2. UNIX commands

3. Implementation of FCFS and SJF CPU scheduling algorithms

4. Implementation of Round Robin and Priority CPU Scheduling

5. Implementation of Producer-Consumer problem using semaphores

6. Implementation of Dining Philosphers Problem

7. Implementation of FIFO Page Replacement Algorithms

8. Implementation of LRU Page Replacement Algorithms

9. Implementation of Sequential File Allocation Strategies

10. Implementation of Indexed File Allocation Strategies

COURSE OUTCOMES:

At the end of this course, students will demonstrate the ability to

CO1. Experiment with Unix commands and shell programming.

CO2. Able to build shell program for process and file system management with system calls.

CO3. Able to implement and analyse the performance of different algorithm of Operating Systems like CPU scheduling algorithm,

CO4. Able to implement and analyse the performance of different algorithm of page replacement algorithms, deadlock avoidance, detection algorithm and so on.

CO5. Able to design and develop a course project that can have positive impact on environment or society

or mankind.

Semester 3

Course code	MCA-201			
Category	Discipline Specific Courses			
Course title	**Data Base Management System**			
Scheme and Credits	L	T	P	Credits
	3		2	4
TI	25			
TE	50			
PI	5			
PE	20			
Duration of Exam				

Course Outcomes:

1. To understand the different issues involved in the design and implementation of a database system.

2. To study the physical and logical database designs, database modeling, relational, hierarchical, and network models

3. To understand and use data manipulation language to query, update, and manage a database

4. To develop an understanding of essential DBMS concepts such as: database security, integrity, concurrency, distributed database, and intelligent database, Client/Server (Database Server), Data Warehousing.

5. To design and build a simple database system and demonstrate competence with the fundamental tasks involved with modeling, designing, and implementing a DBMS.

UNIT – I

Basic Concepts – Data, Information, Records and files. Traditional file –based Systems-File Based Approach-Limitations of File Based Approach, Database Approach-Characteristics of Database Approach, advantages and disadvantages of database system, components of database system, Database Management System (DBMS), Components of DBMS Environment, DBMS Functions and Components, DBMS users, Advantages and Disadvantages of DBMS, DBMS languages. Roles in the Database Environment - Data and Database Administrator, Database Designers, Applications Developers and Users .

UNIT – II

Database System Architecture – Three Levels of Architecture, External, Conceptual and Internal Levels, Schemas, Mappings and Instances . Data Independence – Logical and Physical Data Independence . Classification of Database Management System, Centralized and Client Server architecture to DBMS . Data Models: Records- based Data Models, Object-based Data Models, Physical Data Models and Conceptual Modeling.

UNIT – III

Entity-Relationship Model – Entity Types, Entity Sets, Attributes Relationship Types, Relationship Instances and ER Diagrams, abstraction and integration. Basic Concepts of Hierarchical and Network Data Model, Relational Data Model :- Brief History, Relational Model Terminology-Relational Data Structure, Database Relations, Properties of Relations, Keys, Domains, Integrity Constraints over Relations, .

UNIT – IV

Relational algebra, Relational calculus, Relational database design: Functional dependencies, Modification anomalies, Ist to 3rd NFs, BCNF, 4th and 5th NFs, computing closures of set FDs, SQL: Data types, Basic Queries in SQL, Insert, Delete and Update Statements, Views, Query processing: General strategies of query processing, query optimization, query processor, concept of security, concurrency and recovery.

SUGGESTED READINGS

1. Elmasri & Navathe, “Fundamentals of Database Systems”, 5th edition, Pearson Education.

2. Thomas Connolly Carolyn Begg, “Database Systems”, 3/e, Pearson Education

3. C. J. Date, “An Introduction to Database Systems”, 8th edition, Addison Wesley N. Delhi

LAB: List of Experiments

CONTENTS

1. Creation of a database and writing SQL queries to retrieve information from the database.

2. Performing Insertion, Deletion, Modifying, Altering, Updating and Viewing records based on conditions.

3. Creation of Views, synonyms, Sequences, indexes, and save points.

4. Creating an employee database to set various constraints.

5. Creating relationships between the databases. iv. Study of PL/SQL block.

6. Write a PL/SQL block to satisfy some conditions by accepting input from the user.

7. Write a PL/SQL block that handles all types of exceptions.

8. Creation of Procedures

9. Creation of database triggers and functions

10. Mini project (Application Development using Oracle/ MySQL)

a. Inventory Control System

b. Material Requirement Processing.

c. Hospital Management System.

d. Railway Reservation System.

e. Personal Information System.

f. Web-Based User Identification System.

g. Time Table Management System.

h. Hotel Management

COURSE OUTCOMES:

At the end of this course, students will demonstrate the ability to

CO1: Understand basic concepts of database system and data models for relevant problems.

CO2: Understand the basic elements of a relational database management system.

CO3: Design entity relationship model and convert entity relationship diagrams into RDBMS and formulate SQL queries on the data.

CO4: Apply normalization for the development of application software.

Course code				
Category	Discipline Specific Courses			
Course title	**Foundation of Web technology**			
Scheme and Credits	L	T	P	Credits
	3		2	4
TI	25			
TE	50			
PI	5			
PE	20			
Duration of Exam	3 HRS			

NOTE: "The examiner will set nine questions in total. Question one will have seven parts from all units and the marks of first question will be of 20% of total marks of Question Paper and the remaining eight questions to be set by taking two questions from each unit and the marks of each question from Question no.2 to 9 will be of 20% of total marks of Question paper . The students have to attempt five questions in total, the first being compulsory and selecting one from each unit."

COURSE OUTCOMES

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

1 Design web pages using HTML5 and CSS.

.2 Understand objects and data validation in JavaScript.

.3 Build Dynamic web site using server-side PHP Programming and Database

connectivity.

4 Create web applications with Ajax

Unit I:

Internet Basics 25 25 Basic concepts, Communication on the Internet, Internet Domains, Internet Server Identities, Establishing Connectivity on the Internet, Client IP Address, Brief Overview of TCP/IP and its Services, Transmission Control Protocol, Web Server, Web Client, Domain Registration, Gateways. HTML: Introduction to Hypertext Markup Language, Common tags, Anchors, Backgrounds, Images, Webpage structure, Hyper linking, Lists, Character Formatting, Color Control, Images, Tables, Frames, Multimedia, Cascading style sheet, Application with layers.

Unit II:

JavaScript and XML Client-side Forms, JavaScript, Incorporating JavaScript in HTML, JavaScript expressions, Control flow and functions, String and Arrays, JavaScript objects, JavaScript Forms: Managing frames in JavaScript, Cookies, history, location. XML, XSL and other markup languages, CGI Scripting with Perl.

Unit III:

PHP Basic concepts strings, functions and OOPs, File handling, Working with database and AJAX.

Unit IV:

Servlets & JSP Introduction to Servlets and JSP basics. Implementation of Sessions, Cookies, Filter, Listener, and Wrapper, Expression language-EL, JSP Standard Tag Library-JSTL, MySQL, JDBC, Connection pooling.

References

1. Jackson, Web Technologies: A Computer Science Perspective, Pearson Education, 2007.
2. S. Holzner, Php: The Complete Reference, TMH, 2007.
3. Kriss Jamsa, Konrad King, HTML & Web Design, TMH Publications, 2002.
4. Jason Hunter, William Crawford, Servlet Programming, O'REILY, 2010.
5. Tom Negrino and Dori Smith, JavaScript for the World Wide Web, 3E, 2011.
6. Joel Murach, Andrea Steelman, Murach's Java Servlets and JSP, Murach's, 2E, 2008.
7. Robert Hoekman Jr., Java Servlet & JSP Cookbook, Schorr Pub, 2004.
8. Santosh Kumar K, JDBC, Servlets, And Jsp Black Book, Kogent Solutions Inc., 2008.

COURSE OUTCOMES:

1.Design and implement dynamic websites with good aesthetic sense of designing
and latest technical know-how's
2.Create web pages using HTML and Cascading Styles sheets
3 Analyze a web page and identify its elements and attributes
4 Create dynamic web pages using JavaScript
5 Build web applications using PHP

LIST OF EXPERIMENTS:

1.Design the following static web pages required for an online book store
web site.
1) HOME PAGE: The static home page must contain three frames.

2) LOGIN PAGE

3) CATOLOGUE PAGE: The catalogue page should contain the details of all the books available in the web site in a table.

4) REGISTRATION PAGE.

2. Write an HTML page that contains a selection box with a list of 5 countries. When the user selects a country, its capital should be printed next in the list. Add CSS to customize the properties of the font of the capital (color,bold and font size). 3. Write an HTML page including any required JavaScript that takes a number from text field in the range of 0 to 999 and shows it in words. It should not accept four and above digits, alphabets and special characters.

1. Develop and demonstrate PHP Script for the following problems: a) Write a PHP Script to find out the Sum of the Individual Digits. b) Write a PHP Script to check whether the given number is Palindrome or not
2. Create an XML document that contains 10 users information. Write a Java Program, which takes User Id as input and returns the user details by taking the user information from XML document using DOM parser or SAX parser
3. Implement the web applications with Database using (a) PHP, (b) Servlets and (c) JSP.
4. Modify the above PHP program to use an xml instead of database
5. Write a program to design a simple calculator using (a) JavaScript (b) PHP (c) Servlet and (d) JSP.

Semester 4

Course code				
Category	Discipline Specific Courses (DSC)			
Course title	Object oriented programming using python			
Scheme and Credits	L	T	P	Credits
	3	-	2	4
Theory Internal	25			
Theory External	50			
Practical Internal	05			
Practical External	20			
Total	100			
Duration of Exam	3 hrs			

Note: *The examiner will set nine questions in total. Question one will have seven parts from all units and the marks of first question will be of 20% of total marks of Question Paper and the remaining eight questions to be set by taking two questions from each unit and the marks of each question from Question no.2 to 9 will be of 20% of total marks of Question paper. The students have to attempt five questions in total, the first being compulsory and selecting one from each unit.*

Course outcomes:

CO1: Write simple programs using built-in data types of Python.

CO2 Implement arrays and user defined functions in Python.

CO3 Solve problems spanning multiple disciplines using suitable programming constructs in Python.

CO4. Solve problems spanning multiple disciplines using the concepts of object oriented programming in Python.

Unit 1:

Introduction to Python: Python - a tool, not a reptile, Introduction to OOPS.

Unit 2:

Data Types, Variables, Basic Input-Output Operations, Basic Operators: Python literals, Operators - data manipulation tools, Variables - data-shaped boxes, Boolean Values, Conditional Execution, Loops, Lists and List Processing, Logical and Bitwise Operations: Making decisions in Python, Python's loops, Logic and bit operations in Python, Lists - collections of data, Sorting simple lists - the bubble sort algorithm, Lists in advanced applications.

Unit 3:

Functions, Tuples, Dictionaries, and Data Processing: writing functions in Python, how functions communicate with their environment, Returning a result from a function, Scopes in Python, functions, Tuples and dictionaries.

Unit 4 :

Modules, Packages, String and List Methods, handling Exceptions and Errors, The anatomy of exception, Characters and strings, Python's nature of strings, String methods, Strings in action.

The Object-Oriented Approach: Classes, Methods, Objects, and the Standard Objective Features, and Working with Files: Basic concepts of object programming, Properties, Methods, Inheritance, Generators and closures, processing files, Working with real files.

Reference books / web links:

1. Gowrishankar S, Veena A, “Introduction to Python Programming”, 1st Edition, CRC Press/Taylor & Francis, 2018. ISBN-13: 978-0815394372
2. Jake VanderPlas, “Python Data Science Handbook: Essential Tools for Working with Data”, 1st Edition, O'Reilly Media, 2016. ISBN-13: 978-1491912058
3. AurelienGeron, “Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems”, 2nd Edition, O’Reilly Media, 2019. ISBN – 13: 978-9352139057.
4. Wesley J Chun, “Core Python Applications Programming”, 3rd Edition, Pearson Education India, 2015. ISBN-

Object oriented programming using python

List of Experiments

1. Basic building blocks of a Python program (variables, conditional statements, loops, libraries, functions, errors).
2. Data structures (trees, dictionaries, tuples)
3. Object Oriented programming (classes, objects, inheritance, polymorphism, abstract classes).
4. PyQt for creating graphical user interfaces for interactive programs
5. NumPy (Matrices, vectors, linear algebra)
6. SciPy (Package for numerical computations)
7. Matplotlib (Plotting) 8. Interactive Python (IPython)

Course code				
Category	Discipline Specific Courses (DSC)			
Course title	Analysis and Design of Algorithms			
Scheme and Credits	L	T	P	Credits
	3	-	2	4
Theory Internal	25			
Theory External	50			
Practical Internal	05			
Practical External	20			
Total	100			
Duration of Exam	3 hrs			

Note: *The examiner will set nine questions in total. Question one will have seven parts from all units and the marks of first question will be of 20% of total marks of Question Paper and the remaining eight questions to be set by taking two questions from each unit and the marks of each question from Question no.2 to 9 will be of 20% of total marks of Question paper. The students have to attempt five questions in total, the first being compulsory and selecting one from each unit.*

Course outcomes:

CO1: Able to Argue the correctness of algorithms using inductive proofs and Analyze worst-case running times of algorithms using asymptotic analysis.

CO2: Able to explain important algorithmic design paradigms (divide-and-conquer, greedy method, dynamic-programming and Backtracking) and apply when an algorithmic design situation calls for it.

CO3: Able to Explain the major graph algorithms and Employ graphs to model engineering problems, when appropriate

CO4: Able to Compare between different data structures and pick an appropriate data structure for a design situation.

CO5: Able to Describe the classes P, NP, and NPComplete and be able to prove that a certain problem is NP-Complete.

CO6: Able to analyze String matching algorithms.

Unit I:

Introduction and basic concepts Complexity measures, worst-case and average-case complexity functions, problem complexity, quick review of basic data structures and algorithm design principles. Sorting and selection: Finding maximum and minimum, k largest elements in order; Sorting by selection, tournament

and heap sort methods, lower bound for sorting, other sorting algorithms - radix sort, quick sort, merge sort; Selection of k-th largest element.

Unit II:

Searching and set manipulation Searching in static table – binary search, path lengths in binary trees and applications, optimality of binary search in worst cast and average-case, binary search trees, construction of optimal weighted binary search trees; Searching in dynamic table – randomly grown binary search trees, AVL and (a,b) trees. Hashing: Basic ingredients, analysis of hashing with chaining and with open addressing. Union-Find problem: Tree representation of a set, weighted union and path compression-analysis and applications.

Unit III:

Graph problems Graph searching – BFS, DFS, shortest first search, topological sort; connected and biconnected components; Minimum spanning trees – Kruskal's and Prim's algorithms – Johnson's implementation of Prim's algorithm using priority queue data structures. Algebraic problems: Evaluation of polynomials with or without preprocessing. Winograd's and Strassen's matrix multiplication algorithms and applications to related problems, FFT, simple lower bound results.

Unit IV:

String processing String searching and Pattern matching, Knuth-Morris-Pratt algorithm and its analysis. NP-completeness: Informal concepts of deterministic and nondeterministic algorithms, P and NP, NP-completeness, statement of Cook's theorem, some standard NP-complete problems, approximation algorithms.

Reference books / web links:

1. T. H. Cormen, C. L. Leiserson, R. L. Rivest, and C. Stein, Introduction to Algorithms, MIT Press, 2003.

2. J. Kleinberg and E. Tardos, Algorithm Design, Addison-Wesley, 2006.

3. Harry R. Lewis and Larry Denenberg, Data Structures and Their Algorithms, Harper Collins, 1991.

4. A. Gibbons, Algorithmic Graph Theory, Cambridge University Press, 1985.

5. Michael T. Goodrich and Roberto Tamassia, Algorithm Design: Foundations, Analysis, and Internet Examples, John Wiley, 2006.

6. R. Sedgewick, Algorithms in C: Part 5, Addison Wesley, 2001.

7. M. H. Alsuwaiyel, Algorithm Design Techniques and Analysis, World Scientific, 1999. 8. Gilles Brassard and Paul Bratley, Algorithmics: theory and practice, Prentice-Hall, 1988.

Analysis and Design of Algorithms
List of Experiments

1. Write a program to implement different sorting techniques. • Bubble Sort • Insertion Sort • Selection Sort • Quick Sort • Merge Sort
2. Write a program to find minimum cost spanning tress.
3. Write a program to implement travelling sales person problem.
4. Write a program to find Longest Path in a Directed Acyclic Graph.
5. Write a program for Shortest path with exactly k edges in a directed and weighted graph.
6. Write a program find maximum number of edge disjoint paths between two vertices
7. Implement 0/1 Knapsack problem using Dynamic Programming.
8. Perform various tree traversal algorithms for a given tree.
9. Implement N-Queens Problem

Course code				
Category	Discipline Specific Courses (DSC)			
Course title	Data Communication and Networking			
Scheme and Credits	L	T	P	Credits
	3	-	2	4
Theory Internal	25			
Theory External	50			
Practical Internal	05			
Practical External	20			
Total	100			
Duration of Exam	3 hrs			

Note: *The examiner will set nine questions in total. Question one will have seven parts from all units and the marks of first question will be of 20% of total marks of Question Paper and the remaining eight questions to be set by taking two questions from each unit and the marks of each question from Question no.2 to 9 will be of 20% of total marks of Question paper. The students have to attempt five questions in total, the first being compulsory and selecting one from each unit.*

Course outcomes:

CO1: Student will be able to understand network communication using the layered concept, Open System Interconnect (OSI) and the Internet Model.

CO2: Student will be able to understand various types of transmission media, network devices; and parameters of evaluation of performance for each media and device.

CO3: Student will be able to understand the concept of flow control, error control and LAN protocols; to explain the design of, and algorithms used in, the physical, data link layers.

CO4: Student will understand the working principles of LAN and the concepts behind physical and logical addressing, subnetting and supernetting.

CO5: Student shall understand the functions performed by a Network Management System and to analyze connection establishment and congestion control with respect to TCP Protocol.

CO6: Student shall understand the functions performed by a Network Management System and to analyze connection establishment and congestion control with respect to TCP Protocol.

Unit I: Introduction Introduction of network, topology, Use of computer network, network hardware: LAN, WAN, MAN, Wireless Network, Reference Models: ISO-OSI model, TCP model. Physical layer:

Transmission media-Magnetic Media, Twisted Pair, Coaxial pair, Fiber 24 24 Optics, Line coding and multiplexing.

Unit II: Data link layer Data link layer design Issue, Error Detection and correction, Elementary Data link protocol, stop- and-wait ARQ, sliding window, Go-back-n, Selective Repeat ARQ. Related advanced algorithms to be studied. Mac sublayer: Multiple Access protocol: ALOHA, Slotted ALOHA, CSMA protocols, Introduction to MAC Protocols: 802.3, 802.4, 802.5, 802.11

Unit III: Network layer Network Design Issue, Routing algorithm-introduction, optimality Principle, Shortest Path, Flooding, Distance Vector Routing. Congestion Control Routing: General principle of Congestion control, leaky bucket algorithm, Token Bucket Algorithm. TCP/IP: The TCP/IP architecture, the Internet Protocol, ARP, DHCP and mobile IP, Internet routing protocols: RIP, OSPF, BGP. TCP/IP Implementation related case studies to be studied.

Unit IV: Transport layer Transport Services, Element of transport protocols, TCP connection management, TCP transmission policy, TCP congestion control & Timer management. Application layer: DNS, SMTP, POP3, FTP, TELNET, HTTPS. Related advanced protocols should be studied.

Reference books / web links:

1. William Stallings, Data and Computer Communication, Prentice Hall of India, 2007.

2. Behrouz A. Forouzan, Data Communication and Networking, McGraw-Hill, 2007.

3. Andrew S. Tanenbaum, Computer Networks, Prentice Hall, 2008.

4. Douglas Comer, Internetworking with TCP/IP, Volume 1, Prentice Hall of India, 2006.

5. W. Richard Stevens, TCP/IP Illustrated: The Protocol, Volume 1, Addison-Wesley, 2011.

6. William Stallings, Cryptography and Network Security: Principles and Practice, Prentice Hall of India, 2008. 7. Neal Koblitz, A course in number theory and cryptography, Springer, 2008.

8. R. C. Seacord, Secure Coding in C and C++, Addison-Wesley, 2005. 9. John Viega, Matt Messier), Pravir Chandra, Network Security with OpenSSL, O'Reilly, 2009.

Analysis and Design of Algorithms
List of Experiments

1. Study of Parallel data Communication between two computers.

2. Study of Network Topologies – Star, Bus & Ring

3. Implementation of stop and wait protocol using simulator.

4. Implementation of Sliding window protocol using simulator.

5. Implementation of Go-Back N protocol using simulator.

6. Implementation of Selective Repeat protocol using simulator.

7. Study the performance of the network with CSMA/ CD protocol.

8. Study the performance of the network with CSMA/ CA protocol.

9. Implementation of routing algorithm

a. Distance vector Routing Algorithm

b. Link State Routing Algorithm

10. Encryption and Decryption.

11. Study of Ethernets and Fast Ethernets

Course code				
Category	Discipline Specific Courses (DSC)			
Course title	Multi Media Technology			
Scheme and Credits	L	T	P	Credits
	3	-	2	4
Theory Internal	25			
Theory External	50			
Practical Internal	05			
Practical External	20			
Total	100			
Duration of Exam	3 hrs			

Note: *The examiner will set nine questions in total. Question one will have seven parts from all units and the marks of first question will be of 20% of total marks of Question Paper and the remaining eight questions to be set by taking two questions from each unit and the marks of each question from Question no.2 to 9 will be of 20% of total marks of Question paper. The students have to attempt five questions in total, the first being compulsory and selecting one from each unit.*

Course outcomes:

CO1: It will provide an understanding of the fundamental elements in multimedia.

CO2: It will emphasise on learning the representations, perceptions and applications of multimedia. CO3: To develop the skills for developing multimedia projects.

CO4: Understand the technologies behind multimedia applications

CO5: Summarize the key concepts in current multimedia technology.

CO6: Create quality multimedia software titles.

UNIT I

Multimedia System Overview Introduction to Multimedia Sytem, Multimedia Files:Image and sound file formats, features of software to read and write such files, Basics of digital audio, Basics of digital imaging ,Multimedia compression technologies and standards - VCD, DVD - MPEG-1/2/4/21.

UNIT II

Multimedia Authentication Multimedia Authentication: Pattern, Speaker and Behavior Recognition - Speaker Recognition – Face Recognition,

UNIT III

Multimedia Forensis Multimedia Forensics:Digital Forensics taxonomy, goals/requirements - Forensic Data Acquisition - Digital Forensics Tools -Forensics Analysis and Validation - File and Network Forensics – Techniques - Application forensics- Email, Graphics and Multimedia Forensics.

UNIT IV

Multimedia Fingerprinting: Multimedia Fingerprinting: Trace Traitors, Multimedia fingerprinting for tracing traitor, Steganalysis, Non-intrusive forensics, Standardization and Integration, Standardization on media security

Secured Multimedia and Multimedia Security Applications Secured Multimedia: Digital Rights Management Systems, and Technical Trends - Multimedia encryption - Digital Watermarking - Security Attacks. Multimedia Security Applications: Media Sensor Network - Voice over IP (VoIP) Security – DTH – Video Conference

Reference books / web links:

1. Wenjun Zeng, Heather Yu and Ching – Yung Lin, " Multimedia Security technologies for Digital rights Management", Elsevier Inc 2006.

2. Chun-Shien Lu, "Multimedia Security : Steganography and Digital Watermarking techniques for Protection of Intellectual Property" , Springer Inc 2007.

Multi Media Technology Lab
List of Experiments

1. Write a program to justify a text entered by the user on both left and right hand side. For example, the test "An architect may have a graphics program to draw anentire building but be interested in only ground floor", can be justified in 30 columns. An architect may have a graphics programs draw an entire building but interested in only ground floor.
2. Study the notes of a piano and stimulate them using the keyboard and store them in file.
3. Write a program to read a paragraph and store it to a file name suggested by the author.
4. Devise a routine to produce the animation effect of a square transforming to a triangle and then to a circle.

5. Write a program to show a bitmap image on your computer screen.
6. Create a web page for a clothing company which contains all the details of that company and at-least five links to other web pages.
7. Write a program by which we can split mpeg video into smaller pieces for the purpose of sending it over the web or by small capacity floppy diskettes and then joining them at the destination.
8. Write a program to simulate the game of pool table.
9. Write a program to simulate the game mine sweeper.
10. Write a program to play "wave" or "midi" format sound files

Semester 5

Course code				
Category	Discipline Specific Courses (DSC)			
Course title	Software Engineering			
Scheme and Credits	L	T	P	Credits
	3	-	2	4
Theory Internal	25			
Theory External	50			
Practical Internal	05			
Practical External	20			
Total	100			
Duration of Exam	3 hrs			

Note: *The examiner will set nine questions in total. Question one will have seven parts from all units and the marks of first question will be of 20% of total marks of Question Paper and the remaining eight questions to be set by taking two questions from each unit and the marks of each question from Question no.2 to 9 will be of 20% of total marks of Question paper. The students have to attempt five questions in total, the first being compulsory and selecting one from each unit.*

Course outcomes:

CO1: an ability to identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics

CO2: an ability to apply engineering design to produce solutions that meet specified needs with consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors

CO3: an ability to communicate effectively with a range of audiences

CO4: an ability to recognize ethical and professional responsibilities in engineering situations and make informed judgments, which must consider the impact of engineering solutions in global, economic, environmental, and societal contexts

CO5: an ability to function effectively on a team whose members together provide leadership, create a collaborative and inclusive environment, establish goals, plan tasks, and meet objectives

CO6: an ability to develop and conduct appropriate experimentation, analyze and interpret data, and use engineering judgment to draw conclusions

UNIT – I

Introduction: Software Crisis, Software Processes & Characteristics, Software life cycle models, Waterfall, Prototype, Evolutionary and Spiral Models. Software Requirements Analysis & Specifications: Requirement engineering, requirement elicitation techniques like FAST, QFD, requirements analysis using DFD, Data dictionaries & ER Diagrams, Requirements documentation, Nature of SRS, Characteristics & organization of SRS .

UNIT – II

Software Project Management Concepts: The Management spectrum, The People The Problem, The Process, The Project. Software Project Planning: Size Estimation like lines of Code & Function Count, Cost Estimation Models, COCOMO, Risk Management.

UNIT - III

Software Design: Cohesion & Coupling, Classification of Cohesiveness & Coupling, Function Oriented Design, Object Oriented Design, Software Metrics: Software measurements: What & Why, Token Count, Halstead Software Science Measures, Design Metrics, Data Structure Metrics Software Implementation: Relationship between design and implementation, Implementation issues and programming support environment, Coding the procedural design, Good coding style.

UNIT - IV

Software Testing: Testing Process, Design of Test Cases, Types of Testing, Functional Testing, Structural Testing, Test Activities, Unit Testing, Integration Testing and System Testing, Debugging Activities. Software Maintenance: Management of Maintenance, Maintenance Process, Reverse Engineering, Software Re-engineering, Configuration Management, Documentation.

Reference books / web links:

1. Gill, Nasib Singh : Software Engineering, Khanna Book Publishing Co. (P) Ltd. N. Delhi.

2. Pressman : Software Engineering, TMH.

3. Jalote, Pankaj : An Integrated Approach to Software Engineering, Narosa Publications.

4. Chhillar Rajender Singh : Software Engineering : Testing, Faults, Metrics, Excel Books, New Delhi.

5. Ghezzi, Carlo : Fundaments of Software Engineering, PHI.

6. Fairely, R.E. : Software Engineering Concepts, McGraw-Hill.

7. Lewis, T.G.: Software Egineering, McGraw-Hill.

8. Shere : Software Engineering & Management, Prentice Hall.

Software Engineering Lab
List of Experiments

1. An area for faculty posting of class materials such as course syllabus and handouts

2. An area for student posting of papers and other assignments

3. A grade book where faculty can record grades and each student can view his or her grades

4. An integrated email tool allowing participants to send announcement email messages to the entire class or to a subset of the entire class

5. A chat tool allowing synchronous communication among class participants

6. A threaded discussion board allowing asynchronous communication among participants

Course code				
Category	Discipline Specific Courses (DSC)			
Course title	Artificial Intelligence			
Scheme and Credits	L	T	P	Credits
	3	-	2	4
Theory Internal	25			
Theory External	50			
Practical Internal	05			
Practical External	20			
Total	100			
Duration of Exam	3 hrs			

Note: *The examiner will set nine questions in total. Question one will have seven parts from all units and the marks of first question will be of 20% of total marks of Question Paper and the remaining eight questions to be set by taking two questions from each unit and the marks of each question from Question no.2 to 9 will be of 20% of total marks of Question paper. The students have to attempt five questions in total, the first being compulsory and selecting one from each unit.*

Course outcomes:

CO1: Design user interfaces to improve human–AI interaction and real-time decision-making.

CO2: Evaluate the advantages, disadvantages, challenges, and ramifications of human–AI augmentation.

CO3: Design and develop symbiotic human–AI systems that balance the information processing power of computational systems with human intelligence and decision making.

CO4: Explain the benefits, limitations, and tradeoffs of designing engaging and ethical conversational user interactions, including those supported by chatbots, smart speakers, and other AI-driven, voice-based technologies.

CO5: Design and evaluate conversational interfaces for different users and contexts of use.

UNIT – I

Overview of A.I: Introduction to AI, Importance of AI, AI and its related field, AI techniques, Criteria for success. Problems, problem space and search: Defining the problem as a state space search, Production system and its characteristics, Issues in the design of the search problem Heuristic search techniques : Generate and test, hill climbing, best first search technique, problem reduction, constraint satisfaction

UNIT - II

Knowledge Representation: Definition and importance of knowledge, Knowledge representation, Various approaches used in knowledge representation, Issues in knowledge representation. Using Predicate Logic : Represent ting Simple Facts in logic, Representing instances and is_a relationship, Computable function and predicate.

UNIT - III

Natural language processing : Introduction syntactic processing, Semantic processing, Discourse and pragmatic processing. Learning: Introduction learning, Rote learning, Learning by taking advice, Learning in problem solving, Learning from example-induction, Explanation based learning.

UNIT - IV

Expert System: Introduction, Representing using domain specific knowledge, Expert system shells.

Reference books / web links:

1. David W. Rolston : Principles of Artificial Intelligence and Expert System Development, McGraw Hill Book Company.

2. Elaine Rich, Kevin Knight : Artificial Intelligence, Tata McGraw Hill.

3. D.W. Patterson, "Introduction to AI and Expert Systems", PHI, 1999 . 4. Nils J Nils

Artificial Intelligence Lab
List of Experiments

1. Write a Program to Implement Breadth First Search using Python.
2. Write a Program to Implement Depth First Search using Python.
3. Write a Program to Implement Tic-Tac-Toe game using Python.
4. Write a Program to Implement 8-Puzzle problem using Python.
5. Write a Program to Implement Water-Jug problem using Python.
6. Write a Program to Implement Travelling Salesman Problem using Python.
7. Write a Program to Implement Tower of Hanoi using Python.
8. Write a Program to Implement Monkey Banana Problem using Python.
9. Write a Program to Implement Alpha-Beta Pruning using Python.
10. Write a Program to Implement 8-Queens Problem using Python.

Course code				
Category	Discipline Specific Courses (DSC)			
Course title	Mathematical foundation of Computer Science			
Scheme and Credits	L	T	P	Credits
	3	-	2	4
Theory Internal	25			
Theory External	50			
Practical Internal	05			
Practical External	20			
Total	100			
Duration of Exam	3 hrs			

Note: *The examiner will set nine questions in total. Question one will have seven parts from all units and the marks of first question will be of 20% of total marks of Question Paper and the remaining eight questions to be set by taking two questions from each unit and the marks of each question from Question no.2 to 9 will be of 20% of total marks of Question paper. The students have to attempt five questions in total, the first being compulsory and selecting one from each unit.*

Course outcomes:

CO1: Ability to apply mathematical logic to solve problems.

CO2: Understand sets, relations, functions, and discrete structures.

CO3: Able to use logical notation to define and reason about fundamental mathematical concepts such as sets, relations, and functions.

CO4: Able to formulate problems and solve recurrence relations.

CO5: Able to model and solve real-world problems using graphs and trees.

UNIT - I

Mathematical logic: Introduction, Statements and Notation, Connectives, Normal Forms, Theory of Inference for the Statement Calculus, The Predicate Calculus, Inference Theory of the Predicate Calculus.

UNIT - II

Set theory: Introduction, Basic Concepts of Set Theory, Representation of Discrete Structures, Relations and Ordering, Functions. Algebraic Structures: Introduction, Algebraic Systems, Semi groups and Monoids, Groups, Lattices as Partially Ordered Sets, Boolean algebra.

UNIT – III

Elementary Combinatorics: Basics of Counting, Combinations and Permutations, Enumeration of Combinations and Permutations, Enumerating Combinations and Permutations with Repetitions, Enumerating Permutations with Constrained Repetitions, Binomial Coefficients, The Binomial and Multinomial Theorems, The Principle of Inclusion Exclusion.

UNIT – IV

Recurrence Relations: Generating Functions of Sequences, Calculating Coefficients of generating functions, Recurrence relations, Solving recurrence relations by substitution and Generating functions, The method of Characteristic roots, Solutions of Inhomogeneous Recurrence Relations. Graphs: Basic Concepts, Isomorphisms and Subgraphs, Trees and their Properties, Spanning Trees, Directed Trees, Binary Trees, Planar Graphs, Euler's Formula, Multigraphs and Euler Circuits, Hamiltonian Graphs, Chromatic Numbers, The Four-Color Problem

Course code				
Category	Discipline Specific Courses (DSC)			
Course title	Java programming			
Scheme and Credits	L	T	P	Credits
	3	-	2	4
Theory Internal	25			
Theory External	50			
Practical Internal	05			
Practical External	20			
Total	100			
Duration of Exam	3 hrs			

Note: *The examiner will set nine questions in total. Question one will have seven parts from all units and the marks of first question will be of 20% of total marks of Question Paper and the remaining eight questions to be set by taking two questions from each unit and the marks of each question from Question no.2 to 9 will be of 20% of total marks of Question paper. The students have to attempt five questions in total, the first being compulsory and selecting one from each unit.*

Course outcomes:

CO1: Able to solve real world problems using OOP techniques.

CO2: Able to understand the use of abstract classes.

CO3: Able to solve problems using java collection framework and I/o classes.

CO4: Able to develop multithreaded applications with synchronization.

CO5: Able to develop applets for web applications.

CO6: Able to design GUI based applications

UNIT-I

Object Oriented Methodology-1: Paradigms of Programming Languages, Evolution of OO Methodology, Basic Concepts of OO Approach, Comparison of Object Oriented and Procedure Oriented Approaches, Benefits of OOPs, Introduction to Common OO Language, Applications of OOPs . Object Oriented Methodology-2: Classes and Objects, Abstraction and Encapsulation, Inheritance, Method Overriding and Polymorphism.

UNIT-II

Java Language Basics: Introduction To Java, Basic Features, Java Virtual Machine Concepts, Primitive Data Type And Variables, Java Operators, Expressions, Statements and Arrays. Object Oriented Concepts: Class and Objects-- Class Fundamentals, Creating objects , Assigning object reference variables; Introducing Methods, Static methods, Constructors , Overloading constructors; This Keyword; Using Objects as Parameters, Argument passing, Returning objects , Method overloading, Garbage Collection, The Finalize ( ) Method. Inheritance and Polymorphism: Inheritance Basics, Access Control, Multilevel Inheritance, Method Overriding, Abstract Classes, Polymorphism, Final Keyword.

UNIT-III

Packages : Defining Package, CLASSPATH, Package naming, Accessibility of Packages , using Package Members. Interfaces: Implementing Interfaces, Interface and Abstract Classes, Extends and Implements together . Exceptions Handling : Exception , Handling of Exception, Using try-catch , Catching Multiple Exceptions , Using finally clause , Types of Exceptions, Throwing Exceptions, Writing Exception Subclasses.

UNIT-IV

Multithreading : Introduction , The Main Thread, Java Thread Model, Thread Priorities, Synchronization in Java, Inter thread Communication. I/O in Java : I/O Basics, Streams and Stream Classes ,The Predefined Streams, Reading from, and Writing to, Console, Reading and Writing Files , The Transient and Volatile Modifiers , Using Instance of Native Methods. Strings and Characters : Fundamentals of Characters and Strings, The String Class , String Operations , Data Conversion using Value Of ( ) Methods , String Buffer Class and Methods.

Reference books / web links:

1. Programming in Java, E Balagurusamy .
2. The Complete Reference JAVA, TMH Publication.
3. Begining JAVA, Ivor Horton, WROX Public.
4. JAVA 2 UNLEASHED, Tech Media Publications.
5. Patrick Naughton and Herbertz Schildt, “Java-2 The Complete Reference”, 1999,TMH.

SEMESTER 6

Course code				
Category	Discipline Specific Courses (DSC)			
Course title	Compiler Design			
Scheme and Credits	L	T	P	Credits
	3	-	2	4
Theory Internal	25			
Theory External	50			
Practical Internal	05			
Practical External	20			
Total	100			
Duration of Exam	3 hrs			

Note: *The examiner will set nine questions in total. Question one will have seven parts from all units and the marks of first question will be of 20% of total marks of Question Paper and the remaining eight questions to be set by taking two questions from each unit and the marks of each question from Question no.2 to 9 will be of 20% of total marks of Question paper. The students have to attempt five questions in total, the first being compulsory and selecting one from each unit.*

Course outcomes:

CO 1: Acquire knowledge of different phases and passes of the compiler and also able to use the compiler tools like LEX, YACC, etc. Students will also be able to design different types of compiler tools to meet the requirements of the realistic constraints of compilers.

CO 2: Understand the parser and its types i.e. Top-Down and Bottom-up parsers and construction of LL, SLR, CLR, and LALR parsing table.

CO 3: Implement the compiler using syntax-directed translation method and get knowledge about the synthesized and inherited attributes.

CO 4: Acquire knowledge about run time data structure like symbol table organization and different techniques used in that.

CO 5: Understand the target machine's run time environment, its instruction set for code generation and techniques used for code optimization.

Unit I:

Introduction Phases of compilation and overview Lexical Analysis (scanner): Regular language, finite automata, regular expression, from regular expression to finite automata, scanner generator (lex,flex).

Unit II:

Syntax Analysis (Parser) Context-free language and grammar, push-down automata, LL(1) grammar and top- down parsing, operator grammar, LR(0), SLR(1), LR(1), LALR(1) grammars and bottom-up parsing, ambiguity and LR parsing, LALR(1) parser generator (yacc, bison). Semantic Analysis: Attribute grammar, syntax directed definition, evaluation and flow of attribute in a syntax tree.

Unit III:

Symbol Table Its structure, symbol attributes and management. Run-time environment: Procedure activation, parameter passing, value return, memory allocation, and scope. Intermediate Code Generation: Translation of different language features, different types of intermediate forms.

Unit IV:

Code Improvement (optimization) Analysis: control-flow, data-flow dependence etc.; Code improvement local optimization, global optimization, loop optimization, peep-hole optimization etc. Architecture dependent code improvement: instruction scheduling (for pipeline), loop optimization (for cache memory) etc. Register allocation and target code generation Advanced topics: Type systems, data abstraction, compilation of object oriented features and non-imperative programming languages.

Reference books / web links:

1. A. V. Aho, R. Sethi, J. D. Ullman, Compilers: Principles, Techniques and Tools, Addison-Wesley, 2007.

2. M. L. Scott, Programming Language Pragmatics, Elsevier, 2009.

3. A. W. Appel, Modern Compiler Implementation in C/Java, Cambridge University Press, 2004.

4. K. D. Cooper and Linda Torczon, Engineering a Compiler, Elsevier, 2011.

5. A. I. Holob, Compiler Design in C, Prentice-Hall, 1994. 6. S. S. Muchnik, Advanced Compiler Design and Implementation, Elsevier, 1997.

Compiler Design Lab
List of Experiments

1. Practice of LEX/YACC of compiler writing.

2. Write a program to check whether a string belong to the grammar or not.

3. Write a program to generate a parse tree.

4. Write a program to find leading terminals.

5. Write a program to find trailing terminals.

6. Write a program to compute FIRST of non-terminal.

7. Write a program to compute FOLLOW of non-terminal.

8. Write a program to check whether a grammar is left Recursion and remove left Recursion.

9. Write a program to remove left factoring.

10. Write a program to check whether a grammar is operator precedent.

11. To show all the operations of a stack.

12. To show various operations i.e. red, write and modify in a text file

Course code				
Category	Discipline Specific Courses (DSC)			
Course title	Computer Graphics			
Scheme and Credits	L	T	P	Credits
	3	-	2	4
Theory Internal	25			
Theory External	50			
Practical Internal	05			
Practical External	20			
Total	100			
Duration of Exam	3 hrs			

Note: *The examiner will set nine questions in total. Question one will have seven parts from all units and the marks of first question will be of 20% of total marks of Question Paper and the remaining eight questions to be set by taking two questions from each unit and the marks of each question from Question no.2 to 9 will be of 20% of total marks of Question paper. The students have to attempt five questions in total, the first being compulsory and selecting one from each unit.*

Course outcomes:

CO1: Students will be able to describe the fundamental algorithms used in computer graphics and to some extent be able to compare and evaluate them

CO2: Students will be able to work and interact, through hands-on experiences, to design, develop, and modify electronically generated imaginary using a wide range of sophisticated graphical tools and techniques.

CO3: Students will be able to summarize different hidden surface elimination algorithms and shading techniques used in computer graphics and digital media production

CO4: Students will be able to explain about the technology necessary for creating multimedia content for the web, video, DVD, 2D and 3D graphics, Sound and programming.

CO5: Students can apply the knowledge, techniques, skills and modern tools to become successful professionals in communication and media industries

UNIT-I

Graphics Primitives: Introduction to computer graphics, Basics of Graphics systems, Application areas of Computer Graphics, overview of graphics systems, video-display devices, and raster-scan systems, random scan systems, graphics monitors and workstations and input devices. Output Primitives: Points

and lines, line drawing algorithms, mid-point circle and ellipse algorithms. Filled area primitives: Scan line polygon fill algorithm, boundary fill and floodfill algorithms .

UNIT-II

2-D Geometrical Transforms: Translation, scaling, rotation, reflection and shear transformations, matrix representations and homogeneous coordinates, composite transforms, transformations between coordinate systems. 2-D Viewing: The viewing pipeline, viewing coordinate reference frame, window to viewport coordinate transformation, viewing functions, Cohen-Sutherland and Cyrus-beck line clipping algorithms, Sutherland –Hodgeman polygon clipping algorithm.

UNIT-III

3-D Object Representation: Polygon surfaces, quadric surfaces, spline representation, Hermite curve, Bezier curve and B-Spline curves, Bezier and B-Spline surfaces. Basic illumination models, polygon-rendering methods.

UNIT-IV

3-D Geometric Transformations: Translation, rotation, scaling, reflection and shear transformations, composite transformations. 3-D Viewing: Viewing pipeline, viewing coordinates, view volume and general projection transforms and clipping.

Reference books / web links:

1. Donald Hearn and M. Pauline Baker : Computer Graphics, PHI Publications.

2. Plastock : Theory & Problem of Computer Gaphics, Schaum Series.

3. Foley & Van Dam : Fundamentals of Interactive Computer Graphics, Addison-Wesley.

4. Newman : Principles of Interactive Computer Graphics, McGraw Hill.

5. Tosijasu, L.K. : Computer Graphics, Springer-Verleg

Computer graphics Lab
List of Experiments

1. Write a program for 2D line drawing as Raster Graphics Display.

2. Write a program for circle drawing as Raster Graphics Display.

3. Write a program for polygon filling as Raster Graphics Display

4. Write a program for line clipping.

5. Write a program for polygon clipping.

6. Write a program for displaying 3D objects as 2D display using perspective transformation.

7. Write a program for rotation of a 3D object about arbitrary axis.

8. Write a program for Hidden surface removal from a 3D object.

Course code				
Category	Discipline Specific Courses (DSC)			
Course title	Shell Programming using Linus			
Scheme and Credits	L	T	P	Credits
	3	-	2	4
Theory Internal	25			
Theory External	50			
Practical Internal	05			
Practical External	20			
Total	100			
Duration of Exam	3 hrs			

Note: *The examiner will set nine questions in total. Question one will have seven parts from all units and the marks of first question will be of 20% of total marks of Question Paper and the remaining eight questions to be set by taking two questions from each unit and the marks of each question from Question no.2 to 9 will be of 20% of total marks of Question paper. The students have to attempt five questions in total, the first being compulsory and selecting one from each unit.*

Course outcomes:

CO1: Describe the architecture and features of LINUX Operating System and distinguish it from other Operating System

CO2: Demonstrate LINUX commands for file handling and process control

CO3: Write Regular expressions for pattern matching and apply them to various filters for a specific task

CO4: Analyse a given problem and apply requisite facets of SHELL programming in order to devise a SHELL script to solve the problem

UNIT – I

INTRODUCTION TO LINUX AND LINUX UTILITIES: A brief history of LINUX, architecture of LINUX, features of LINUX, introduction to vi editor. Linux commands- PATH, man, echo, printf, script, passwd, uname, who, date, stty, pwd, cd, mkdir, rmdir, ls, cp, mv, rm, cat, more, wc, lp, od, tar, gzip, file handling utilities, security by file permissions, process utilities, disk utilities, networking commands, unlink, du, df, mount, umount, find, unmask, ulimit, ps, w, finger, arp, ftp, telnet, rlogin.Text Processing utilities and backup utilities , tail, head , sort, nl, uniq, grep, egrep, fgrep, cut, paste, join, tee, pg, comm, cmp, diff, tr, awk, cpio

UNIT - II

Introduction to Shells: Linux Session, Standard Streams, Redirection, Pipes, Tee Command, Command Execution, Command-Line Editing, Quotes, Command Substitution, Job Control, Aliases, Variables, Predefined Variables, Options, Shell/Environment Customization. Filters: Filters and Pipes, Concatenating files, Display Beginning and End of files, Cut and Paste, Sorting, Translating Characters, Files with Duplicate Lines, Count Characters, Words or Lines, Comparing Files

UNIT - III

Grep: Operation, grep Family, Searching for File Content. Sed :Scripts, Operation, Addresses, commands, Applications, grep and sed. UNIX FILE STRUCTURE: Introduction to UNIX file system, inode (Index Node), file descriptors, system calls and device drivers. File Management :File Structures, System Calls for File Management – create, open, close, read, write, lseek, link, symlink, unlink, stat, fstat, lstat, chmod, chown, Directory API – opendir, readdir, closedir, mkdir, rmdir, umask.

UNIT - IV

PROCESS AND SIGNALS: Process, process identifiers, process structure: process table, viewing processes, system processes, process scheduling, starting new processes: waiting for a process, zombie processes, orphan process, fork, vfork, exit, wait, waitpid, exec, signals functions, unreliable signals, interrupted system calls, kill, raise, alarm, pause, abort, system, sleep functions, signal sets. File locking: creating lock files, locking regions, use of read and write with locking, competing locks, other lock commands, deadlocks. INTER PROCESS COMMUNICATION: Pipe, process pipes, the pipe call, parent and child processes, and named pipes: fifos, semaphores: semget, semop, semctl, message queues: msgget, msgsnd, msgrcv, msgctl, shared memory: shmget, shmat, shmdt, shmctl, ipc status commands. INTRODUCTION TO SOCKETS: Socket, socket connections - socket attributes, socket addresses, socket, connect, bind, listen, accept, socket communications.

Reference books / web links:

1. W. Richard. Stevens (2005), Advanced Programming in the UNIX Environment, 3rd edition, Pearson Education, New Delhi, India. 2. Unix and shell Programming Behrouz A. Forouzan, Richard F. Gilberg.Thomson

2. Linux System Programming, Robert Love, O'Reilly, SPD.

3. Advanced Programming in the UNIX environment, 2nd Edition, W.R.Stevens, Pearson Education.

4. UNIX Network Programming, W.R. Stevens, PHI. UNIX for Programmers and Users, 3rd Edition, Graham Glass, King Ables, Pearson Education

Shell Programming using Linux Lab
List of Experiments

1. Study and Practice on various commands like man, passwd, tty, script, clear, date, cal, cp, mv, ln, rm, unlink, mkdir, rmdir, du, df, mount, umount, find, unmask, ulimit, ps, who, w.
2. 2. Study and Practice on various commands like cat, tail, head , sort, nl, uniq, grep, egrep,fgrep, cut, paste, join, tee, pg, comm, cmp, diff, tr, awk, tar, cpio.
3. a) Write a Shell Program to print all .txt files and .c files.

b) Write a Shell program to move a set of files to a specified directory.

c) Write a Shell program to display all the users who are currently logged in after a specified time.

d) Write a Shell Program to wish the user based on the login time.

4. a) Simulate cat command.

   b) Simulate cp command.

5. a) Simulate head command.

   b) Simulate tail command.

6. a) Simulate mv command.

   b) Simulate nl command.

7. Write a program to handle the signals like SIGINT, SIGQUIT, SIGFPE.
8. Implement the following IPC forms

   a) FIFO

   b) PIPE

9. Implement message queue form of IPC.
10. Implement shared memory form of IPC.
11. Write a Socket program to print system date and time (Using TCP/IP).

Course code				
Category	Discipline Specific Courses (DSC)			
Course title	Information Retrieval Systems			
Scheme and Credits	L	T	P	Credits
	3	-	2	4
Theory Internal	25			
Theory External	50			
Practical Internal	05			
Practical External	20			
Total	100			
Duration of Exam	3 hrs			

Note: *The examiner will set nine questions in total. Question one will have seven parts from all units and the marks of first question will be of 20% of total marks of Question Paper and the remaining eight questions to be set by taking two questions from each unit and the marks of each question from Question no.2 to 9 will be of 20% of total marks of Question paper. The students have to attempt five questions in total, the first being compulsory and selecting one from each unit.*

Course outcomes:

CO1: Learn to write code for text indexing and retrieval.

CO2: Learn to evaluate information retrieval systems

CO3: Learn to analyze textual and semi-structured data sets

CO4: Learn to evaluate information retrieval systems

CO5: Learn about text similarity measure

CO6: Understanding about search engine and Text Classification

UNIT I

Introduction to Information Retrieval Retrieval Strategies: Vector space model, Probabilistic retrieval strategies: Simple term weights, Non binary independence model, Language Models

UNIT II

Retrieval Utilities: Relevance feedback, Clustering, N-grams, Regression analysis, Thesauri.

UNIT III

Retrieval Utilities: Semantic networks, Parsing. Cross-Language Information Retrieval: Introduction, Crossing the language barrier.

UNIT IV

Efficiency: Inverted index, Query processing, Signature files, Duplicate document detection UNIT V Integrating Structured Data and Text: A Historical progression, Information retrieval as a relational application, Semi-structured search using a relational schema. Distributed Information Retrieval: A Theoretical model of distributed retrieval, Web search.

Reference books / web links:

1. Information Retrieval – Algorithms and Heuristics, David A. Grossman, Ophir Frieder, 2nd Edition, 2012, Springer, (Distributed by Universities Press)

2. Modern Information Retrieval Systems, Yates, Pearson Education

3. Information Storage and Retrieval Systems, Gerald J Kowalski, Mark T Maybury, Springer, 2000

4. Mining the Web : Discovering Knowledge from Hypertext Data, Soume

Information Retrieval Systems Lab
List of Experiments

1. Value of Visualization – What is Visualization and Why do it: External representation – Interactivity – Difficulty in Validation. Data Abstraction: Dataset types – Attribute types – Semantics. Task Abstraction – Analyze, Produce, Search, Query. Four levels of validation – Validation approaches – Validation examples. Marks and Channels

2. Rules of thumb – Arrange tables: Categorical regions – Spatial axis orientation – Spatial layout density. Arrange spatial data: Geometry – Scalar fields – Vector fields – Tensor fields. Arrange networks and trees: Connections, Matrix views – Containment. Map color: Color theory, Color maps and other channels.

3. Manipulate view: Change view over time – Select elements – Changing viewpoint – Reducing attributes.

4. Facet into multiple views: Juxtapose and Coordinate views – Partition into views – Static and Dynamic layers – Reduce items and attributes: Filter – Aggregate. Focus and context: Elide – Superimpose – Distort

Semester 7

Cryptography and Network Security

Course code				
Category	Discipline Specific Courses (DSC)			
Course title	Cryptography and Network Security			
Scheme and Credits	L	T	P	Credits
	3	-	2	4
Theory Internal	25			
Theory External	50			
Practical Internal	05			
Practical External	20			
Total	100			
Duration of Exam	03 Hours			

Note: Examiner will be required to set NINE questions with all questions carrying equal marks. Question Number 1, covering the entire syllabus, will be compulsory. Examiner will set two questions from each Unit with internal choice. Student will be required to attempt FIVE questions in all, selection one question from every unit apart from the Question Number 1.

COURSE OUTCOMES:

CO1: Discover all the Information Security Goals, Services & Mechanism for Network Security & necessity of Mathematics in designing crypto algorithms.

CO2: Develop and design various kinds of Symmetric Key crypto algorithms and to cryptanalyze them.

CO3: Construct and design various kinds of Asymmetric key crypto algorithms and Mathematics required for designing.

CO4: Analyze the integrity of information transmitted & to generate digital signatures.

CO5: Predict and apply the knowledge and skills obtained to design & understand latest cryptographic protocols used for securing information in networks or in storage.

UNIT-I

INTRODUCTION TO INFORMATION SECURITY & CRYPTOGRAPHY: What is Information Security, Need for security, Principles of Security, Threats, Types of Attacks, Services & Mechanisms, Mathematics of Cryptography: Integer Arithmetic, Modular Arithmetic, Matrices, Linear Congruence

UNIT-II

SYMMETRIC KEY CIPHERS: raditional Symmetric –Key Cipher design and Analysis-Different Substitution ciphers & Transposition ciphers, Basic Principles of designing Stream cipher & Block cipher.

Mathematics of Symmetric-Key Cryptography-Algebraic Structures, GF (2n) Fields.

Modern Symmetric Key Ciphers- Block Ciphers Design & Analysis - Data encryption Standard(DES), Advanced Encryption Standard (AES) Stream Ciphers Design & Analysis-LFSR based, RC4, A5/1.

UNIT-III

ASYMMETRIC KEY CIPHERS, HASH FUNCTIONS AND DIGITAL SIGNATURES: Mathematics of Asymmetric-Key Cryptography, Asymmetric key Ciphers, Hash Functions and MAC, Introduction to Digital Signatures.

UNIT-IV

NETWORK SECURITY: Network Security: Security at application layer – PGP & S/MIME, Key distribution Centre, Diffie-Hellman Key Exchange.

TEXT AND REFERENCE BOOKS:

1. Cryptography and Network Security 7th Edition, William Stallings
2. Cryptography and Network Security, 4rd edition, Forouzan & Mukopadhyay, TMH.
3. Information Security & Cryptography-Cryptography Made Simple, Nigel P Smart, Springer Verlag, 2016.
4. Cryptography & Network Security, 2nd Edition, Atul Kahate. McGrawHill

Cryptography and Network Security Lab

1. Program to find GCD of any two numbers a and b mod n by Generalized Euclidean Algorithm
2. Program to find Multiplicative Inverse of b in Zp, p prime by Generalized Euclidean Algorithm
3. Program to encrypt and decrypt by Ceaser Cipher
4. Program to Encrypt and Decrypt by Affine Cipher
5. Program to encrypt plaintext by Playfair Cipher
6. Program to generate addition and multiplication table for GF (22) Field
7. Program for key expansion algorithm of DES

8. Program to implement 4*4 S-box and perform two functionalities:

a.) Check given table of S-box and inverse S-box are invertible to each other.

b.) Ask the user to enter input to be given in S-box and return its output.

9. Program to implement encryption and decryption of RSA algorithm.

Software Testing and Quality Assurance

Course code				
Category	Discipline Specific Courses (DSC)			
Course title	Software testing and Quality Assurance			
Scheme and Credits	L	T	P	Credits
	3	-	2	4
Theory Internal	25			
Theory External	50			
Practical Internal	05			
Practical External	20			
Total	100			
Duration of Exam	03 Hours			

Note: Examiner will be required to set NINE questions with all questions carrying equal marks. Question Number 1, covering the entire syllabus, will be compulsory. Examiner will set two questions from each Unit with internal choice. Student will be required to attempt FIVE questions in all, selection one question from every unit apart from the Question Number 1.

Course Outcomes:

CO1. Understand testing concepts, techniques, and best practices in a systematic way that reflects an orderly evolution of testing process growth on both an individual and organizational level.

CO2. Understand and apply various functional testing.

CO3. Apply software quality concepts, standards, measurements, and practices that support the production of quality software.

CO4. Evaluate various test case.

CO5. Understand various test generation activities.

UNIT I

PRELIMINARIES OF SOFTWARE TESTING: Overview of testing- Need for software testing – Testing principles – STLC models. Testing in STLC models: Unit Testing, Integration Testing, System Testing, Acceptance Testing. Testing of software attributes: Smoke test, functional testing, usability testing, security, compliance testing.

UNIT II

TESTING METHODOLOGIES: Test Design techniques: Black Box testing- White Box testing – Experience- Based testing. Test Management Process: Formation of testing team- roles and

responsibilities Test planning and control –Test analysis and design- Test implementation and execution – Test evaluation and reporting- Test closure activities.

TOOLS FOR TESTING: Test tool classification- Tools for management and control- Tools for specification- Tools for static and dynamic testing- Tools for non- functional tests. Manual testing versus automated testing- automated testing tools.

UNIT III

OVERVIEW OF QUALITY ASSURANCE: Definition of software quality and quality assurance- Quality assurance versus Quality control- Quality factors- Quality components – Quality plans- Software quality metrics Costs of software quality- Quality Management Framework- Commercial and government Standards in SQA - Pareto principal in SQA.

UNIT IV

QUALITY MANAGEMENT AND AUDIT: Requirements for SQA- Software QA versus Traditional QA- Defect prevention and process improvement- Software inspection- Software audit methods- Comparison of Quality Assurance techniques and activities- Quality improvement methods- Management and its role in SQA - Quality management in IT.

TEXT AND REFERENCE BOOKS:

1. Andreas Spillner, Tilo Linz , Hans Schaefer “Software Testing Foundations - A Study Guide for the Certified Tester Exam” , Foundation Level ISTQB compliant, 4th Edition, Santa Barbara, CA :Rocky Nook, Inc, 2014.
2. Anne MetteJonassen Hass, “Guide to Advanced Software Testing”, Artech House Publishers, 2008.

Software testing and Quality Assurance Lab

1. Write a test case to test login window using manual testing
2. Write a test case to test triangle using manual testing
3. Write a test case to test valid mobile no using manual testing Write a test case to test ATM machine no using manual testing
4. Write the script to test the “save” functionality of notepad using rational robot
5. Write the script to test “find” functionality of notepad using rational robot

Digital Image Processing

Course code	7.3			
Category	Discipline Specific Courses (DSC)			
Course title	Digital Image processing			
Scheme and Credits	L	T	P	Credits
	3	-	2	4
Theory Internal	25			
Theory External	50			
Practical Internal	05			
Practical External	20			
Total	100			
Duration of Exam	03 Hours			

Note: Examiner will be required to set NINE questions with all questions carrying equal marks. Question Number 1, covering the entire syllabus, will be compulsory. Examiner will set two questions from each Unit with internal choice. Student will be required to attempt FIVE questions in all, selection one question from every unit apart from the Question Number 1.

COURSE OUTCOMES:

CO1: Know and understand the basics and fundamentals of digital image processing, such as digitization, sampling, quantization, and 2D-transforms.

CO2: Operate on images using the techniques of smoothing, sharpening and enhancement.

CO3: Understand the restoration concepts and filtering techniques.

CO4: Learn the basics of segmentation, features extraction, compression and recognition methods for color models.

Unit I

Fundamentals Need for DIP- Fundamental steps in DIP – Elements of visual perception -Image sensing and Acquisition – Image Sampling and Quantization – Imaging geometry, discrete image mathematical characterization.

Unit II

Image Transforms Two dimensional Fourier Transform- Properties – Fast Fourier Transform – Inverse FFT,Discrete cosine transform and KL transform.-Discrete Short time Fourier Transform- Wavelet Transform- Discrete wavelet Transform- and its application in Compression. Image Enhancement

Unit III

Spatial Domain: Basic relationship between pixels- Basic Gray Level Transformations – Histogram Processing – Smoothing spatial filters- Sharpening spatial filters. Frequency Domain: Smoothing frequency domain filters- sharpening frequency domain filters Homomorphic filtering. Image Restoration: Overview of Degradation models –Unconstrained and constrained Restorations-Inverse Filtering, WienerFilter.

Unit IV

Feature Extraction: Detection of discontinuities – Edge linking and Boundary detection- Thresholding--Edge based Segmentation-Region based Segmentation- Matching-Advanced optimal border and surface detection- Use of motion in segmentation. Image Morphology – Boundary descriptors- Regional descriptors.

TEXT AND REFERENCE BOOKS:

1. Rafael C.Gonzalez & Richard E.Woods – Digital Image Processing – Pearson Education- 2/e – 2004.
2. Anil.K.Jain – Fundamentals of Digital Image Processing- Pearson Education-2003.

Reference Books:

1. B.Chanda & D.Dutta Majumder – Digital Image Processing and Analysis – Prentice Hall of India – 2002
2. William K. Pratt – Digital Image Processing – John Wiley & Sons-2/e, 2004

Digital Image Processing Lab

1. Display of Gray scale Images.
2. Histogram Equalization.
3. Design of Non-linear Filtering.
4. Determination of Edge detection using Operators.
5. Filtering in frequency domain.
6. Display of colour images.
7. Conversion between colour spaces.
8. DWT of images.
9. Segmentation using watershed transform.

Fundamentals of Machine Learning

Course code	7.4			
Category	Discipline Specific Courses (DSC)			
Course title	Fundamentals of Machine Learning			
Scheme and Credits	L	T	P	Credits
	3	-	2	4
Theory Internal	25			
Theory External	50			
Practical Internal	05			
Practical External	20			
Total	100			
Duration of Exam	03 Hours			

Note: Examiner will be required to set NINE questions with all questions carrying equal marks. Question Number 1, covering the entire syllabus, will be compulsory. Examiner will set two questions from each Unit with internal choice. Student will be required to attempt FIVE questions in all, selection one question from every unit apart from the Question Number 1.

Course Outcome

CO1.Understand, visualize, analyse and preprocess the data from a real-time source.

CO2.Apply appropriate algorithm to the data.

CO3.Analyze the results of algorithm and convert to appropriate information required for the real – time application.

CO4.Evaluate the performance of various algorithms that could be applied to the data and to suggest most relevant algorithm according to the environment.

UNIT-I

Introduction to Machine Learning and Pre-requisites Introduction to Machine Learning – Learning Paradigms – PAC learning – Version Spaces – Role of Machine Learning in Artificial Intelligence applications.

UNIT-II

Supervised Learning – I Linear and Non-Linear examples – Multi–Class & Multi-Label classification – Linear Regression – Multiple Linear Regression – Naïve Bayes Classifier – Decision Trees – ID3 – CART – Error bounds.

Supervised Learning – II

K-NN classifier – Logistic regression – Perceptron – Single layer & Multi-layer – Support Vector Machines – Linear & Non-linear – Metrics & Error Correction

UNIT-III

Unsupervised Learning & Ensemble Learning Clustering basics (Partitioned, Hierarchical and Density based) - K-Means clustering – K- Mode clustering – Self organizing maps – Expectation maximization – Principal Component Analysis – Kernel PCA – tSNE (t-distributed stochastic neighbor embedding) - Metrics & Error Correction.

Ensemble Learning Bias – Variance Tradeoff – Bagging and Boosting (Random forests, Adaboost, XG boost inclusive) – Metrics & Error Correction.

UNIT-IV

Machine Learning in Practice Class Imbalance – SMOTE – One Class SVM – Optimization of hyper parameters.

Reinforcement Learning (RL) Basics of RL – RL Framework – Markov Decision Process – Exploration Vs Exploitation - Polices, Value Functions and Bellman Equations – Solution Methods – Q-learning.

TEXT AND REFERENCE BOOKS:

1. Ethem Alpaydin, Introduction to Machine Learning, MIT Press, Prentice Hall of India, Third Edition 2014.
2. Richard S. Sutton and Andrew G. Barto, Reinforcement Learning: An Introduction (Adaptive Computation and Machine Learning series) 2nd edition, A Bradford Book; 2018.
3. Mehryar Mohri, Afshin Rostamizadeh, Ameet Talwalkar, Foundations of Machine Learning, MIT Press, 2012.
4. Tom Mitchell, Machine Learning, McGraw Hill, 3rd Edition, 1997.
5.Charu C. Aggarwal, Data Classification Algorithms and Applications, CRC Press, 2014

List of Experiments

1. Linear & Multiple Linear Regression
2. Naïve Bayes classifier
3. Decision trees – ID3 & CART
4. Logistic regression
5. Support Vector Machines – Linear & Non-linear

6. Single & Multilayer Perceptron
7. K-NN, K-Means & K-mode clustering
8. Random – forest
9. Adaboost, XGboost
10. Principal component analysis
11. Self – Organizing maps
12. Q-Learning

Big Data Analytics

Course code	7.5			
Category	Discipline Specific Courses (DSC)			
Course title	Big Data Analytics			
Scheme and Credits	L	T	P	Credits
	3	-	2	4
Theory Internal	25			
Theory External	50			
Practical Internal	05			
Practical External	20			
Total	100			
Duration of Exam	03 Hours			

Note: Examiner will be required to set NINE questions with all questions carrying equal marks. Question Number 1, covering the entire syllabus, will be compulsory. Examiner will set two questions from each Unit with internal choice. Student will be required to attempt FIVE questions in all, selection one question from every unit apart from the Question Number 1.

Course Outcomes

CO1. Explore the fundamental concepts of Big Data analysis

CO2. Identify and successfully apply appropriate techniques and tools to solve actual Big Data problems (derive value from vast data sets)

CO3. Examine distributed and parallel computing and its application for big data analysis

CO4. Analyse how to deal with huge amount of data and propose scalable solutions

CO5. Evaluate statistical packages and deriving intelligence from unstructured information

Unit I

Introduction to Big Data, challenges of conventional systems, Evolution of analytic scalability, Modern data analytic tools

Modelling techniques: Mining frequent itemsets, Apriori algorithm, Handling large data sets in main memory, Clustering techniques, clustering for parallelism, Classification and Prediction: Decision Tree induction, Developing models using Decision Tree Algorithms

Unit II

Frameworks: Overview of Hadoop, Hadoop Distributed File System, HDFS design and architecture Hadoop Map reduce Framework, HBASE, Interacting HDFS using HIVE, sample programs in HIVE-PIG

Unit III

Data Analysis and mining data streams: Regression modelling, Rule Induction Fuzzy decision trees and neural networks Introduction to streams concepts, Real time analytics platform, case studies

Unit IV

Visualization: Visual data analysis techniques, Interaction techniques Analytics using statistical packages, association intelligence from unstructured information Text analytics, industry challenges and application of analytics

TEXT AND REFERENCE BOOKS:

1. Bill Franks, "Taming the big data tidal wave: finding opportunities in huge data streams with advanced analytics", John Wiley & Sons,2012
2. Anand Rajaraman and Jeffrey David Ullman, "Mining of Massive Datasets", Cambridge University Press,2012
3. Michael Berthold, David J. Hand, "Intelligent Data Analysis", Springer 2007
4. Jiwaei Han, Micheline Kamber, "Data Mining Concepts and Techniques", Second Edition, Elsevier, Reprinted 2008

Big Data Analytics Lab

1. Install Apache Hadoop

2. Develop a Map Reduce program to calculate the frequency of a given word in a given file. 3. Develop a Map Reduce program to find the maximum temperature in each year. 4. Develop a Map Reduce program to find the grades of student's.

5. Develop a Map Reduce program to implement Matrix Multiplication.

6. Develop a Map Reduce to find the maximum electrical consumption in each year given electrical consumption for each month in each year.

7. Develop a Map Reduce to analyze weather data set and print whether the day is shinny or cool day.

8. Develop a Map Reduce program to find the number of products sold in each country by considering sales data containing fields like

Tranction _Date	Prod uct	Pri ce	Payment _Type	Na me	Ci ty	St ate	Cou ntry	Account_ Created	Last_L ogin	Latit ude	Longi tude

9. Develop a Map Reduce program to find the tags associated with each movie by analyzing movie lens data.

10. XYZ.com is an online music website where users listen to various tracks, the data gets collected which is given below. The data is coming in log files and looks like as shown below.

UserId	TrackId	Shared	Radio	Skip
111115	222	0	1	0
111113	225	1	0	0
111117	223	0	1	1
111115	225	1	0	0

Write a Map Reduce program to get the following

- Number of unique listeners
- Number of times the track was shared with others
- Number of times the track was listened to on the radio
- Number of times the track was listened to in total
- Number of times the track was skipped on the radio

11. Develop a Map Reduce program to find the frequency of books published each year and find in which year maximum number of books were published using the following data.

Title	Author	Published year	Author country	Language	No of pages

12. Develop a Map Reduce program to analyze Titanic ship data and to find the average age of the people (both male and female) who died in the tragedy. How many persons are survived in each class.

The titanic data will be..

Column 1 :PassengerI d
Column 2 : Survived (survived=0 &died=1)
Column 3 :Pclass
Column 4 : Name
Column 5 : Sex
Column 6 : Age
Column 7 :SibSp
Column 8 :Parch
Column 9 : Ticket
Column 10 : Fare
Column 11 :Cabin
Column 12 : Embarked

13. Develop a Map Reduce program to analyze Uber data set to find the days on which each basement has more trips using the following dataset. The Uber dataset consists of four columns they are

dispatching_base_number	date	active_vehicles	trips

14. Develop a program to calculate the maximum recorded temperature by year wise for the weather dataset in Pig Latin

15. Write queries to sort and aggregate the data in a table using HiveQL.

16. Develop a Java application to find the maximum temperature using Spark.

Semester 8

Software Project Management

Course code	8.1			
Category	Discipline Specific Courses (DSC)			
Course title	Software Project Management			
Scheme and Credits	L	T	P	Credits
	3	-	2	4
Theory Internal	25			
Theory External	50			
Practical Internal	05			
Practical External	20			
Total	100			
Duration of Exam	3 hrs.			

Note: Examiner will be required to set NINE questions with all questions carrying equal marks. Question Number 1, covering the entire syllabus, will be compulsory. Examiner will set two questions from each Unit with internal choice. Student will be required to attempt FIVE questions in all, selection one question from every unit apart from the Question Number 1.

Course Outcomes:

CO1: Ability to apply knowledge of engineering and computing appropriate to the discipline.

CO2: Ability to analyze a problem, and identify and define the computing requirements appropriate to its solution.

CO3: Ability to design, implement, and evaluate software project.

CO4: Ability to use current techniques, skills, and tools necessary for computing practice.

CO5: Determine an appropriate project management approach through an evaluation of the business context and scope of the project.

UNIT- I

Introduction to Software Project Management The nature of software production; Key objectives of effective management: quality, productivity, risk reduction; role of the software project manager.

UNIT- II

Business Planning: determining objectives, forecasting demand for product, proposal writing, requirement analysis, legal issues (patent, copyright, liability, warranty);

Life cycle models, types of plans, plan documentation methods: PERT and CPM, Gantt charts, work breakdown structures, standards

UNIT- III

Entry and exit criteria, intermediate checkpoints, performance prediction and analysis people, prototyping and modeling, inspections and reviews, process and process assessment, development methods, metrics, configuration management, testing and quality assurance, capacity planning, estimating - what it takes to do the job: cost (direct and indirect), resources, time, size and complexity of product risk determination, role of requirements and design in estimating, financial planning-budgeting, resource allocation, organizational considerations (teams, hierarchies, etc), technology, human factors and usability, tools and environments, transition of product to the user.

UNIT- IV

Managing the task: project control, managing the plan, reviews, feedback and reporting mechanisms, configuration management, quality control and quality assurance, managing change, readjusting goals and milestones, risk management, testing phases, formalized support activities; Managing the team: Team organizations, recruiting and staffing-picking the right people, technical leadership, avoiding obsolescence-training etc.; Managing the context: Communication skill, decision theory, business management, assessing the organizations ability to perform the process, probability and statistics; Managing product support and maintenance, Evaluation of the project.

Text Books and References:

1. Tom Demarco, Controlling Software Project Management, Measurement, Prentice Hall, New Jersey.
2. Tom Glib, Finzi Susannah, Principles of Software Engineering Management, Addison Wesley, England.
3. Bob Hughes and Mike Cotterell; Software Project Management, third edition, Tata McGraw Hill Publishing Company Ltd., New Delhi.
4. PankajJalote; Software Project Management in Practice, Pearson Education Asia.
5. Watts S. Humphrey; Winning with Software? An Executive Strategy, Pearson Education Asia.
6. Philip Metzger, Managing a Programming Project, Prentice Hall, New Jersey.

Software Project Management Lab

List of Experiments:

1. Create a MS Project application, set the file properties, and set the Project Calender.
2. Using project planning activities, draw the PERT for the project.

3. Draw the Gantt charts for the software project.
4. Using the SPM – manage, plan and organize the project.
5. Using MS project, plan and organize the software and split the task.
6. Using MS Project Link, Move and copy tasks in Software Project
7. Draw the checkpoints and milestones of a project
8. Using MS Project do the time estimation of tasks and Set task dependencies &constraints.
9. Using MS Project assign the resources and set the notes for resources.
10. Using MS Project workspace base line the project and review the critical path

Mobile Computing

Course code	8.2			
Category	Discipline Specific Courses (DSC)			
Course title	Mobile Computing			
Scheme and Credits	L	T	P	Credits
	3	1	-	4
Theory Internal	30			
Theory External	70			
Practical Internal	-			
Practical External	-			
Total	100			
Duration of Exam	3 hrs.			

Note: Examiner will be required to set NINE questions with all questions carrying equal marks. Question Number 1, covering the entire syllabus, will be compulsory. Examiner will set two questions from each Unit with internal choice. Student will be required to attempt FIVE questions in all, selection one question from every unit apart from the Question Number 1.

Course Outcomes:

CO1: Characterize and categorize Mobile Computing Environments along with the knowledge of their deployment and applications.
CO2: Learn the operational and architectural aspects of Mobile telephony and its generations.
CO3: Gain knowledge of the architectural issues of contemporary wireless LANs and the reformations accomplished in the Network and Transport layers of TCP/IP model for adaptation to Mobile Networks.
CO4: Get familiar with the application level support for mobility in Networks along with pertinent design issues related with Mobile Adhoc Networks and Wireless Sensor Networks.

Unit – I

Mobile Communications and Mobile Computing: Overview & Classification; Mobile Computing Applications; Characteristics of Mobile Computing; Data Dissemination; Wireless Transmission & Cellular Systems; Mobile Devices; Mobile Computing Architecture; Mobile Computing through Telephony; Multiple Access Procedures; Emerging Technologies; Generations of Mobile Communication Technologies. Satellite Communication: Basics, Applications, Personal Satellite Communications; Routing, Localization, Handover, Future Wideband Access Systems. The Cellular Concept: Components

& Architecture; Call Setup, Frequency reuse, Cell Design; Interference in Cellular System; Channel Assignment & Handoff; Mobility Management; Application of Smart Antennas in Cellular Telephony.

Unit – II

Mobile Telecommunication System: GSM: Services & Architecture; Protocols; Localization & Calling; GSM Channels; GSM Addresses and Identifiers; Frequency Allocation, Call Routing; Mobility Management; Security; New Data Services; EDGE; General Packet radio Service (GPRS): Architecture, Operations, Data Services, Applications; IS95; Universal Mobile Telecommunication System (UMTS): Architecture, Handover, Security; DECT; TETRA. Wireless Media Access Control: Multiplexing & Modulation; Frequency Hopping Spread Spectrum (FHSS); Direct Sequence Spread Spectrum; Code Division Multiple Access (CDMA) in Mobile Communication Systems; 3G Wireless Communication Standards; WCDMA, OFDM, High Speed Packet Access (HSPA); Long Term Evolution (LTE); Broadband Wireless Access Standards. Introduction to 4G & 5G Networks: Introduction, Applications & Architecture of 4G & 5G Networks.

Unit–III

Wireless LANs: Infrastructure & Ad hoc Networks; IEEE 802.11: Architecture & Services; Physical & MAC layer; HIPERLAN1/2; Bluetooth: User Scenario, Architecture & Security. Mobile Network Layer: Mobile IP: Packet Delivery and Handover Management, Location Management, Registration, Tunneling and Encapsulation, Route Optimization, DHCP, VoIP. Mobile Transport Layer: Conventional TCP/IP Protocols, Indirect TCP, Snooping TCP, Mobile TCP, TCP over 2.5/3G Wireless networks.

Unit–IV

Support for Mobility: File Systems; World Wide Web; Wireless Application Protocol (WAP). Application Servers and Management: Mobile Agent; Application Framework; Application Server; gateways; Device Management; Overview of Mobile Application Development Platforms. Mobile Ad-hoc and Wireless Sensor Networks: Introduction to MANETs and their Applications; Routing & Routing Algorithms; Security in Ad-hoc Networks; Wireless Sensor Networks: Overview and Data Dissemination; Applications; Vehicular Ad Hoc networks (VANET); MANET vs VANET.

TEXT AND REFERENCE BOOKS:

1. Jochen Schiller, Mobile Communications, Addison-Wesley.

2. Asoke K Talukder, Hasan Ahmed, Roopa R Yavagal, Mobile Computing, Technology Applications and Service Creation, McGraw Hill.

3. Raj Kamal, Mobile Computing, Oxford University Press

4. Krzysztof Wesolowski, Mobile Communication Systems, Wiley

5. William Stallings ,Wireless Communications & Networking, Second Edition, Pearson

6. UweHansmann, LotherMerk, Martin S. Nicklaus, Thomas Stober, Principles of Mobile Computing, Springer.

7. William.C.Lee, Mobile Cellular Telecommunications-Analog and Digital Systems, Tata McGraw Hill Edition

Pattern Recognition

Course code	8.3			
Category	Discipline Specific Courses (DSC)			
Course title	Pattern Recognition			
Scheme and Credits	L	T	P	Credits
	3	1	-	4
Theory Internal	30			
Theory External	70			
Practical Internal	-			
Practical External	-			
Total	100			
Duration of Exam	3 hrs.			

Note: Examiner will be required to set NINE questions with all questions carrying equal marks. Question Number 1, covering the entire syllabus, will be compulsory. Examiner will set two questions from each Unit with internal choice. Student will be required to attempt FIVE questions in all, selection one question from every unit apart from the Question Number 1.

COURSE OUTCOMES:

CO1. Understand basic mathematical and statistical techniques commonly used in pattern recognition.

CO2. Apply a variety of pattern recognition algorithms.

CO3. Understand and apply various pre-processing algorithms.

CO4. Apply various algorithms for image classification

Unit I

Introduction and mathematical Preliminaries

Principles of pattern recognition: Uses, mathematics, Classification and Bayesian rules, Clustering vs classification, Basics of linear algebra and vector spaces, Eigen values and eigen vectors, Rank of matrix and SVD

Unit II

Pattern Recognition basics

Bayesian decision theory, Classifiers, Discriminant functions, Decision surfaces, Parameter estimation methods, Hidden Markov models, dimension reduction methods, Fisher discriminant analysis, Principal component analysis, non-parametric techniques for density estimation, non- metric methods for pattern classification, unsupervised learning, algorithms for clustering: K- means, Hierarchical and other methods

Unit III

Feature Selection and extraction

Problem statement and uses, Branch and bound algorithm, Sequential forward and backward selection, Cauchy Schwartz inequality, Feature selection criteria function: Probabilistic separability based and Interclass distance based, Feature Extraction: principles

Visual Recognition

Human visual recognition system, Recognition methods: Low-level modelling (e.g. features), Mid- level abstraction (e.g. segmentation), High-level reasoning (e.g. scene understanding); Detection/Segmentation methods; Context and scenes, Importance and saliency, Large-scale search and recognition, Egocentric vision, systems, Human-in-the-loop interactive systems, 3D scene understanding.

Unit IV

Recent advancements in Pattern Recognition

Comparison between performance of classifiers, Basics of statistics, covariance and their properties, Data condensation, feature clustering, Data visualization, Probability density estimation, Visualization and Aggregation, FCM and soft-computing techniques, Examples of real-life datasets.

TEXT AND REFERENCE BOOKS:

1. Pattern Recognition and Machine Learning by Christopher M. Bishop, Springer, 2006.

2. Pattern Classification by Richard O. Duda, Peter E. Hart, David G. Stork, Wiley, 1973.

Foundation of Data Science

Course code	8.4			
Category	Discipline Specific Courses (DSC)			
Course title	Foundation of Data Science			
Scheme and Credits	L	T	P	Credits
	3	-	2	4
Theory Internal	25			
Theory External	50			
Practical Internal	05			
Practical External	20			
Total	100			
Duration of Exam	3 hrs.			

Note: Examiner will be required to set NINE questions with all questions carrying equal marks. Question Number 1, covering the entire syllabus, will be compulsory. Examiner will set two questions from each Unit with internal choice. Student will be required to attempt FIVE questions in all, selection one question from every unit apart from the Question Number 1.

COURSE OUTCOMES:

CO1: Understand and implement the basics of programming in Python

CO2: Apply the Numpy package for numerical calculations in Python

CO3: Apply Pandas package for loading and preprocessing data in Python. Implement various data visualization tools of Python on real world data.

CO4: Understand and implement the Machine Learning Concepts in Python.

UNIT - I

Overview of Python Programming Concepts: The concept of data types; variables, assignments; numerical types; operators and expressions; Control Structures; String manipulations; File Handling – creating, reading/writing text/number files; Dictionaries; Functions; OOPs Concepts.

UNIT - II

Introduction to Numpy - Creation on Array ,Array generation from Uniform distribution, Random array generation, reshaping, maximum and minimum, reshaping, Arithmetic operations, Mathematical functions, Bracket Indexing and Selection, Broadcasting, Indexing a 2D array (matrices); Data Manipulation with Pandas -Creating a Series - from lists, arrays and dictionaries, Storing data in series

from intrinsic sources, Creating Data Frames, Imputation, Grouping and aggregation, Merging, Joining, Concatenation, Find Null Values or Check for Null Values, Reading data from csv, txt, excel, web.

UNIT - III

Introduction to Visualization - Installing and setting up visualization libraries, Canvas and Axes, Subplots, Common plots – scatter, histogram, boxplot, Logarithmic scale, Placement of ticks and custom tick labels, Pandas Viz, Style Sheets, Plot type, Area, Barplots, Histograms, Line Plots, Scatter Plots, BoxPlots, Hexagonal Bin Plot, Kernel Density Estimation plot (KDE), Distribution Plots, Categorical Data Plots, Combining Categorical Plots, Matrix Plots, Regression Plots, Grids; Python Visualizations toolkits/libraries.

UNIT - IV

Introduction to Machine Learning with SciKit-Learn & PyTorch– Data Representation and basic functions Estimator, parameters & model validation, Model Selection, Curve, Grid search, Feature engineering, Naive Bayes Classification, Linear regression, SVM etc; Overview of other Python ML/Deep Learning toolkits/Libraries. Introduction to NLP with NLTK and its functions, modules like speech tagging, tokenization, parsing, segmentation, recognition, cleaning & normalization of text etc; Overview of other Python NLP toolkits/Libraries.

TEXT AND REFERENCE BOOKS:

1. Charles Dierbach., Introduction to Python using Computer Science, Wiley Publications, Second Edition, 2015
2. Mark Lutz , Learning Python, O'Reilly publications , Fifth Edition, 2015
3. Jake Vandar Plas, Python Data Science Handbook, O'Reilly , 2016
4. Paul Barry, Head First Python, Orielly Publications, Second Edition, 2010

Fundamentals of Data Science Lab

1. Python program to display details about the operating system, working directory, files And directories in the current directory, lists the files and all directories, scan and classify them as directories and files.
2. Python program to convert an array to an array of machine values and vice versa
3. Python program to get information about the file pertaining to the file mode and to get time values with components using local time and gm time.
4. Python program to connect to Google using socket programming
5. Python program to perform Array operations using Numpy package
6. Python program to perform Data Manipulation operations using Pandas package.

7. Python program to display multiple types of charts using Matplotlib package
8. Python program to perform File Operation on Excel Data Set
9. Python program to implement with Python Sci Kit-Learn & NLTK.
10. Python program to implement with Python NLTK/Spicy/Py NLPI.

Soft Computing

Course code	8.5			
Category	Discipline Specific Courses (DSC)			
Course title	Soft Computing			
Scheme and Credits	L	T	P	Credits
	3	-	2	4
Theory Internal	25			
Theory External	50			
Practical Internal	05			
Practical External	20			
Total	100			
Duration of Exam	3 hrs.			

Note: Examiner will be required to set NINE questions with all questions carrying equal marks. Question Number 1, covering the entire syllabus, will be compulsory. Examiner will set two questions from each Unit with internal choice. Student will be required to attempt FIVE questions in all, selection one question from every unit apart from the Question Number 1.

Course Outcomes:

CO1: Understand soft computing paradigms including Artificial Intelligence Systems, Neural Networks, Fuzzy Logic, and Genetic Algorithms.

CO2: Develop proficiency in genetic algorithms, including concepts like encoding, fitness functions, and selection methods.

CO3: Study various models of ANNs and their learning algorithms (supervised, unsupervised, reinforcement learning).

CO4: Learn the principles of fuzzy logic, including membership functions, fuzzy sets, and fuzzy inference systems.

UNIT-I

Introduction: Introduction to soft computing, Soft Vs Hard Computing, Different Components of Soft Computing: Artificial Intelligence Systems, Neural Networks, Fuzzy Logic, Genetic Algorithms.

Genetic algorithms: Basic concepts; Encoding; Fitness Function; Reproduction-Roulette wheel, Boltzmann, tournament, rank, and steady state selections; Convergence of GA, Problem Solving using GA.

UNIT-II

Artificial Neural Networks: Introduction to biological and artificial neural network; Different artificial neural network models; Supervised, Unsupervised and Reinforcement Learning; Hebbian Learning, Generalized Hebbian learning algorithm.

Artificial Neural Networks Architecture: Basic building block of an artificial neuron, Activation functions, Introduction to Early ANN architectures: McCulloch & Pitts model; Single Perceptron, Backpropagation networks; Multi-Layer Perceptron; Hopfield Network; Applications of Neural Network**.**

UNIT-Ill

Fuzzy systems and applications: Notion of Fuzziness, Membership Functions, Fuzzification and Defuzzification; Operations on Fuzzy Sets, Fuzzy Functions and Linguistic Variables; Fuzzy Relations, Fuzzy Rules and Fuzzy Inference; Fuzzy Control System and Fuzzy Rule Based Systems.

UNIT-IV

Applications: Pattern Recognitions, Image Processing, Biological Sequence Alignment and Drug Design, Robotics and Sensors, Information Retrieval Systems, Share Market Analysis, Natural Language Processing.

TEXT AND REFERENCE BOOKS:

1. M. Mitchell: An Introduction to Genetic Algorithms, Prentice-Hall.
2. J.5.R.Jang, C.T.Sun and E.Mizulani; Neuro-Fuzzy and Soft Computing, PHI, Pearson Education.
3. Timothy J.Ross: Fuzzy Logic with Engineering Applications, McGraw- ill.
4. Davis E. Goldberg: Genetic Algorithms: Search, Optimization and Machine Learning. Addison Wesley.
5. S. Rajasekaran and G.A.V.Pai: Neural Networks, Fuzzy Logic and Genetic Algorithms, PHI.
6. D. E. Goldberg: Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley.

Soft Computing Lab

1. WAP to implement Artificial Neural Network
2. WAP to implement Activation Functions
3. WAP to implement Adaptive prediction in ADALINE NN
4. WAP to implement LMS and Perceptron Learning Rule
5. WAP to implement ART NN
6. WAP to implement BAM Network
7. WAP to implement Full CPN with input pair
8. WAP to implement discrete Hopfield Network
9. WAP to implement Hebb Network
10. WAP to implement Hetro associate neural net for mapping input vectors to output vectors

Semester 9

Distributed Systems

Course code	9.1			
Category	Discipline Specific Courses (DSC)			
Course title	Distributed Systems			
Scheme and Credits	L	T	P	Credits
	3	1		4
Theory Internal	30			
Theory External	70			
Practical Internal	-			
Practical External	-			
Total	100			
Duration of Exam	3 hrs.			

Note: Examiner will be required to set NINE questions with all questions carrying equal marks. Question Number 1, covering the entire syllabus, will be compulsory. Examiner will set two questions from each Unit with internal choice. Student will be required to attempt FIVE questions in all, selection one question from every unit apart from the Question Number 1.

COURSE OUTCOMES:

CO1: List the principles of distributed systems and describe the problems and challenges associated with these principles.

CO2: Understand Distributed Computing techniques, Synchronous and Processes. CO3: Apply Shared Data access and Files concepts.

CO4: Design distributed system that fulfills requirements with regards to key distributed systems properties.

CO5: Understand Distributed File Systems and Distributed Shared Memory.

UNIT - I

Introduction: Distributed Operating Systems Definition and goals, Hardware and Software concepts, Design issues. Communication in Distributed System: Computer Network and Layered protocols, Message passing and related issues, synchronization, Client Server model & its implementation, remote procedure call and implementation issues, Case Studies: SUN RPC, DEC RPC

UNIT - II

Synchronization in Distributed System: Clock synchronization and related algorithms, mutual exclusion, Deadlock in distributed systems Processes and processors in Distributed systems: Threads, system model, processor allocation, scheduling in distributed systems: Load balancing and sharing approach, fault tolerance, real time distributed systems, Process migration and related issues

UNIT - III

Distributed File systems: Introduction, features & goal of distributed file system, file models, file accessing models, file sharing semantics, file caching scheme, file replication, fault tolerance, trends in distributed file system, case study. Distributed Shared Memory: Introduction, general architecture of DSM systems, design and implementation issues of DSM, granularity, structure of shared memory space, consistency models, replacement strategy, thrashing

UNIT - IV

Security Issues: Introduction of Security in Distributed OS, Overview of security techniques, features, Need, Access Control, Security Management Distributed Web-based Systems: Architecture, Processes, Communication, Naming, Synchronization Case Studies: JAVA RMI, Sun Network File System, Google Case Study

TEXT AND REFERENCE BOOKS:

1. Distributed Operating Systems by Andrew S Tannebaum, Pearson

2. Distributed Operating Systems Concepts and Design, Pradeep K. Sinha, PHI

3. Distributed Systems: Concepts and Design by George Coulouris, Jean Dollimore, TimKindberg, Pearson

4. Distributed Computing by Sunita Mahajan & Seema Shah OXFORD

5. Distributed Systems: Principles and Paradigms by Andrew S Tanebaum, Maarten Van Steen, PHI 6. Distributed Computing, Fundamentals, Simulations and Advanced topics, 2nd Edition, HagitAttiya and Jennifer Welch, Wiley India

Data Warehouse and Mining

Course code	9.2			
Category	Discipline Specific Courses (DSC)			
Course title	Data Warehouse and Mining			
Scheme and Credits	L	T	P	Credits
	3	1	-	4
Theory Internal	30			
Theory External	70			
Practical Internal	-			
Practical External	-			
Total	100			
Duration of Exam	3 hrs.			

Note: Examiner will be required to set NINE questions with all questions carrying equal marks. Question Number 1, covering the entire syllabus, will be compulsory. Examiner will set two questions from each Unit with internal choice. Student will be required to attempt FIVE questions in all, selection one question from every unit apart from the Question Number 1.

COURSE OUTCOMES:

CO1. To understand Concept of Data Mining, Data warehousing and schemas for multidimensional Databases.

CO2. To understand Basic Statistics in order to apply data mining techniques.

CO3. To analyse the data, identify the problems, and choose the relevant models and algorithms to apply.

CO4. To combine and consolidate data from various databases scattered throughout a company into a Datawarehouse.

CO5. To characterize the kinds of patterns that can be discovered by association rule mining, classification and clustering.

UNIT-I

BASIC STATISTICS: Statistical descriptions of data: mean, median, mode, Measuring dispersion of data: range, quartiles, variance, standard deviation, chi-square test, Correlation coefficient and covariance, Regression analysis.

UNIT-II

DATA PREPROCESSING: Introduction to Data pre-processing, Need to preprocess the data, Data cleaning: missing values, Data integration: Redundancy, Data reduction and its strategies, Data transformation and discretization, Strategies of data transformation.

UNIT-III

INTRODUCTION TO DATA MINING AND ITS ALGORITHMS: Introduction to Data Mining and algorithms, Processes, Market Basket Analysis, The Apriori Algorithm, Decision Tree.

INTRODUCTION TO DATA WAREHOUSING AND DATA WRANGLING: Data Warehouse, Difference between Operational Database systems and Data Warehouse, From Table and Spreadsheets to Data Cubes, Schemas for Multidimensional Databases: Star, Snowflakes and Fact Constellations, Data Wrangling, Combining and Merging DataSets, Reshaping and Pivoting, Data Transformation, String Manipulation, Regular Expressions (Regex)

UNIT-IV

ETL: ETL Phase 1 Data Wrangling before the Load, ETL Phase 2 Step-by-step guide to uploading data using SSIS, Handling errors during ETL Phases 1,2, ETL Phase 3 Data Wrangling after the load, Handling errors during ETL Phase 3, Different types of ETL tools.

TEXT AND REFERENCE BOOKS:

1. Jiawei Han, Micheline Kamber and Jian Pei, “Data Mining Concepts and Techniques”, Third Edition, Elsevier, 2011.
2. Alex Berson and Stephen J. Smith “Data Warehousing, Data Mining & OLAP”, Tata McGraw – Hill Edition, Tenth Reprint 2007
3. K.P. Soman, Shyam Diwakar and V. Ajay “Insight into Data mining Theory and Practice”, Easter Economy Edition, Prentice Hall of India, 2006
4. G. K. Gupta “Introduction to Data Mining with Case Studies”, Easter Economy Edition, Prentice Hall of India, 2006.
5. Pang-Ning Tan, Michael Steinbach and Vipin Kumar “Introduction to Data Mining”, Pearson Education, 2007.

Block Chain Technology

Course code	9.3			
Category	Discipline Specific Courses (DSC)			
Course title	Block Chain Technology			
Scheme and Credits	L	T	P	Credits
	3	1	-	4
Theory Internal	30			
Theory External	70			
Practical Internal	-			
Practical External	-			
Total	100			
Duration of Exam	3 hrs.			

Note: Examiner will be required to set NINE questions with all questions carrying equal marks. Question Number 1, covering the entire syllabus, will be compulsory. Examiner will set two questions from each Unit with internal choice. Student will be required to attempt FIVE questions in all, selection one question from every unit apart from the Question Number 1.

COURSE OUTCOMES:

CO1: Student will be able to understand the fundamentals of blockchain technology.

CO2: Apply knowledge of implementations of Bitcoin to develop solutions in the appropriate domains.

CO3: Apply knowledge of implementations of Ethereum to develop solutions in the appropriate domains.

CO4: Apply knowledge of implementations of Hyper ledger to develop solutions in the appropriate domain.

CO5: Explore the blockchain technology to apply in real world problems of cryptography domains

UNIT-I

INTRODUCTION TO BLOCKCHAIN TECHNOLOGY: Distributed systems – The history of blockchain – Introduction to blockchain – CAP theorem and block chain – Benefits and limitations of blockchain – Decentralization using block chain - Methods of decentralization – Routes to decentralization

UNIT-II

CRYPTOGRAPHY IN BLOCKCHAIN: Introduction – cryptographic primitives – Assymetric cryptography – public and private keys -line interface – Bitcoin improvement proposals (BIPs) – Consensus Algorithms.

UNIT-III

BITCOIN: Introduction – Transactions – Structure - Transactions types – The structure of a block– The genesis block – The bitcoin network– Wallets and its types– Bitcoin payments– Bitcoin investment and buying and selling bitcoins – Bitcoin installation – Bitcoin programming and the command-line interface – Bitcoin improvement proposals (BIPs).

UNIT-IV

ETHEREUM: Ethereum block chain- Elements of the Ethereum block chain– Precompiled contracts – Accounts and its types – Block header- Ether – Messages – Mining - Clients and wallets – Trading and investment – The yellow paper - The Ethereum network - Applications developed on Ethereum - Scalability and security issue

TEXT AND REFERENCE BOOKS:

1. Bashir, Mastering Blockchain: Distributed ledger technology, decentralization, and smart contracts explained, 2nd Edition, 2nd Revised edition edition. Birmingham: Packt Publishing, 2018.
2. M. Antonopoulos, Mastering bitcoin, First edition. Sebastopol CA: O‘Reilly,2015.
3. Z. Zheng, S. Xie, H. Dai, X. Chen, andH. Wang, ―An Overviewof Blockchain Technology: Architecture, Consensus, and Future Trends,‖ in 2017 IEEE International Congress on Big Data (BigData Congress), 2017, pp.557–564.

Internet of Things and its application

Course code	9.4			
Category	Discipline Specific Courses (DSC)			
Course title	Internet of Things and its application			
Scheme and Credits	L	T	P	Credits
	3	-	2	4
Theory Internal	25			
Theory External	50			
Practical Internal	05			
Practical External	20			
Total	100			
Duration of Exam	3 hrs.			

Note: Examiner will be required to set NINE questions with all questions carrying equal marks. Question Number 1, covering the entire syllabus, will be compulsory. Examiner will set two questions from each Unit with internal choice. Student will be required to attempt FIVE questions in all, selection one question from every unit apart from the Question Number 1.

Course Outcomes:

CO1. Understand IoT value chain structure (device, data cloud), application areas and technologies involved.

CO2. Understand IoT sensors and technological challenges faced by IoT devices, with a focus on wireless, energy, power, and sensing modules

CO3. Market forecast for IoT devices with a focus on sensors

CO4. Explore and learn about Internet of Things with the help of preparing projects designed for Raspberry Pi

UNIT-I:

Introduction to Signals and systems - Brief Introduction Introduction to Internet of Things- Definition and Characteristics of IoT, Sensors, Actuators, Physical Design of IoT – IoT Protocols, IoT communication models, IoT Communication APIs, IoT enabled Technologies – Wireless Sensor Networks, Cloud Computing, Embedded Systems, IoT Levels and Templates, Domain Specific IoTs – Home, City, Environment, Energy, Agriculture and Industry.

UNIT-II:

IoT and M2M- Software defined networks, network function virtualization, difference between SDN and NFV for IoT, Basics of IoT System Management with NETCONF, YANG- NETCONF, YANG, SNMP NETOPEER

UNIT-III:

IoT Physical Devices and Endpoints- Introduction to Arduino and Raspberry Pi- Installation, Interfaces (serial, SPI, I2C) Controlling Hardware- Connecting LED, Buzzer, Switching High Power devices with transistors, Controlling AC Power devices with Relays, Controlling servo motor, speed control of DC Motor, unipolar and bipolar Stepper motors

UNIT-IV:

Sensors- Light sensor, temperature sensor with thermistor, voltage sensor, ADC and DAC, Temperature and Humidity Sensor DHT11, Motion Detection Sensors, Wireless Bluetooth Sensors, Level Sensors, USB Sensors, Embedded Sensors, Distance Measurement with ultrasound sensor

UNIT-V: IoT Physical Servers and Cloud Offerings– Introduction to Cloud Storage models and communication APIs Web Server – Web server for IoT, Cloud for IoT, Python web application framework Designing a RESTful web API.

TEXT AND REFERENCE BOOKS:

1. Internet of Things - A Hands-on Approach, Arshdeep Bahga and Vijay Madisetti, Universities Press, 2015, ISBN: 9788173719547 R22 B.Tech. CSE (IOT) Syllabus JNTU Hyderabad

2. Getting Started with Raspberry Pi, Matt Richardson & Shawn Wallace, O'Reilly (SPD), 2014, ISBN: 9789350239759

3. Raspberry Pi Cookbook, Software and Hardware Problems and solutions, Simon Monk, O'Reilly (SPD), 2016, ISBN 7989352133895

4. Peter Waher, 'Learning Internet of Things', Packt Publishing, 2015 Editors Ovidiu Vermesan

5. Peter Friess,'Internet of Things – From Research and Innovation to Market Deployment', River Publishers, 2014

6. N. Ida, Sensors, Actuators and Their Interfaces, SciTech Publishers, 2014.

Internet of Things and its application Lab

1. Familiarization with Arduino/Raspberry Pi and perform necessary software installation.

2. To interface LED/Buzzer with Arduino/Raspberry Pi and write a program to turn ON LED for 1 sec after every 2 seconds.

3. To interface Push button/Digital sensor (IR/LDR) with Arduino/Raspberry Pi and write a program to turn ON LED when push button is pressed or at sensor detection.

4. To interface DHT11 sensor with Arduino/Raspberry Pi and write a program to print temperature and humidity readings.

5. To interface motor using relay with Arduino/Raspberry Pi and write a program to turn ON motor when push button is pressed.

6. To interface OLED with Arduino/Raspberry Pi and write a program to print temperature and humidity readings on it.

7. To interface Bluetooth with Arduino/Raspberry Pi and write a program to send sensor data to smartphone using Bluetooth.

8. To interface Bluetooth with Arduino/Raspberry Pi and write a program to turn LED ON/OFF when '1'/'0' is received from smartphone using Bluetooth.

9. Write a program on Arduino/Raspberry Pi to upload temperature and humidity data to thingspeak cloud.

10. Write a program on Arduino/Raspberry Pi to retrieve temperature and humidity data from thingspeak cloud.

Semester 10

Social Network Analysis

Course code	10.1			
Category	Discipline Specific Courses (DSC)			
Course title	Social Network Analysis			
Scheme and Credits	L	T	P	Credits
	3	1	-	4
Theory Internal	30 Marks			
Theory External	70 Marks			
Practical Internal	-			
Practical External	-			
Total	100 Marks			
Duration of Exam	03 Hours			

Note: Examiner will be required to set NINE questions with all questions carrying equal marks. Question Number 1, covering the entire syllabus, will be compulsory. Examiner will set two questions from each Unit with internal choice. Student will be required to attempt FIVE questions in all, selection one question from every unit apart from the Question Number 1.

Unit-I: INTRODUCTION Introduction to Semantic Web: Limitations of current Web – Development of Semantic Web – Emergence of the Social Web – Social Network analysis: Development of Social Network Analysis – Key concepts and measures in network analysis – Electronic sources for network analysis: Electronic discussion networks, Blogs and online communities – Web-based networks – Applications of Social Network Analysis.

Unit-II: MODELLING, AGGREGATING AND KNOWLEDGE REPRESENTATION Ontology and their role in the Semantic Web: Ontology-based knowledge Representation – Ontology languages for the Semantic Web: Resource Description Framework – Web Ontology Language – Modelling and aggregating social network data: State-of-the-art in network data representation – Ontological representation of social individuals – Ontological representation of social relationships – Aggregating and reasoning with social network data – Advanced representations.

Unit-III: EXTRACTION AND MINING COMMUNITIES IN WEB SOCIAL NETWORKS Extracting evolution of Web Community from a Series of Web Archive – Detecting NATIONAL INSTITUTE OF TECHNOLOGY, TIRUCHIRAPPALLI communities in social networks – Definition of community – Evaluating communities – Methods for community detection and mining – Applications of community mining algorithms – Tools for detecting communities social network infrastructures and

communities – Decentralized online social networks – Multi-Relational characterization of dynamic social network communities.

Unit-IV: PREDICTING HUMAN BEHAVIOR AND PRIVACY ISSUES Understanding and predicting human behaviour for social communities – User data management – Inference and Distribution – Enabling new human experiences – Reality mining – Context – Awareness – Privacy in online social networks – Trust in online environment – Trust models based on subjective logic – Trust network analysis – Trust transitivity analysis – Combining trust and reputation – Trust derivation based on trust comparisons – Attack spectrum and counter measures.

Graph theory – Centrality – Clustering – Node-Edge Diagrams – Matrix representation – Visualizing online social networks, visualizing social networks with matrix-based representations – Matrix and Node-Link Diagrams – Hybrid representations – Applications – Cover networks – Community welfare – Collaboration networks – Co-Citation networks.

TEXT AND REFERENCE BOOKS:

1. Peter Mika, ―Social Networks and the Semantic Web, First Edition, Springer 2007.

2. Borko Furht, ―Handbook of Social Network Technologies and Applications, 1st Edition, Springer, 2010.

3. Guandong Xu ,Yanchun Zhang and Lin Li,-Web Mining and Social Networking – Techniques and applications, First Edition, Springer, 2011.

4. Dion Goh and Schubert Foo,-Social information Retrieval Systems: Emerging Technologies and Applications for Searching the Web Effectively, IGI Global Snippet, 2008.

5. Max Chevalier, Christine Julien and Chantal Soulé-Dupuy, Collaborative and Social Information Retrieval and Access: Techniques for Improved user Modelling, IGI Global Snippet, 2009.

6. John G. Breslin, Alexander Passant and Stefan Decker, -The Social Semantic Web, Springer, 2009

Deep Learning

Course code	10.2			
Category	Discipline Specific Elective Course(s)			
Course title	Deep Learning			
Scheme and Credits	L	T	P	Credits
	3	-	2	4
Theory Internal	25			
Theory External	50			
Practical Internal	05			
Practical External	20			
Total	100			
Duration of Exam	3 hrs.			

Note: Examiner will be required to set NINE questions with all questions carrying equal marks. Question Number 1, covering the entire syllabus, will be compulsory. Examiner will set two questions from each Unit with internal choice. Student will be required to attempt FIVE questions in all, selection one question from every unit apart from the Question Number 1.

Course Outcomes (COs):

CO1: Define and Apply concepts of Artificial Neural Networks on real world data. Students will also be able to differentiate deep learning from shallow learning.

CO2: Describe, Implement and Analyze Convolutional Neural Network for image datasets. Students will be able to describe the concepts of Convolutional Neural Network and its architecture and implement the model for predictive analysis and analyze its performance on real world datasets.

CO3: Identify, Describe, Apply and Determine Natural Language Processing techniques for textual datasets. Students will be able to identify the applications for natural language processing, describe the various steps involved in natural language processing process and determine the best process for handling textual data for real world applications.

CO4: Explain, Apply and Compare various sequential models for time series data. Students will be able to explain the requirement of sequential models for handling time series data, apply the models for prediction and compare their performance on various applications.

Unit: 1

Introduction to ANN and Deep Learning Overview of Machine Learning and Neural Networks. Building an ANN, Activation Functions, Evaluating, Improving and Tuning the ANN. Loss functions, Gradient Descent, Back propagation, Hyper parameter tuning. Introduction to Deep Learning, Optimisers, Momentum.

Unit: 2

Deep Learning for Image Processing Basics of Image Processing, Introduction to Tensor flow and Keras, Introduction to CNN, Building a CNN: Convolution layers, Activation functions, Pooling, Flattening, Full Connection, Evaluating, Tuning the CNN, Dropout to prevent Over fitting, CNN applications, Transfer Learning models.

Unit: 3

Natural Language Processing Introduction to NLP (Natural Language Processing), NLTK and Spacy basics, Tokenization, Stemming, Lemmatization, Stop Words, Bag of Words and Bag of N grams, Word Embeddings.

Unit: 4

Models for Sequential Analysis Introduction – Recurrent Neural Network (RNN), Vanishing Gradient, RNN limitations, Introduction to Long Short-Term Memory (LSTM), LSTM Variations, Gated Recurrent Neural Networks (GRU), Application of these architectures to natural language processing and time series.

TEXT AND REFERENCE BOOKS:

1. Francois Chollet, Deep Learning with Python, Manning Publications, First Edition, 2018
2. Ian Goodfellow,Yoshua Bengio, Aaron Courville, Deep Learning , MIT Press, First Edition, 2016
3. Stephen Boyd, Convex Optimization, Cambridge University Press, First Edition, 2015
4. Generative Deep Learning: Teaching Machines to Dream" by David Foster

DEEP LEARNING LAB

1. To explore the basic features of Tensor flow and Keras packages.
2. To build an ANN Model to convert temperature in degree Celsius to Fahrenheit.
3. To build an ANN model for regression problem on house predication dataset.
4. To build an ANN model for classification problem on breast cancer classification to see the effect of:
    i. Early Stopping
    ii. Dropouts

5. To build an advance ANN classification model for churn modelling data with:
    a. Cross Validation
    b. Grid Search
6. To perform Convolutional Neural Networks for Image Classification on MNSIT Dataset.
7. To create CNN model with dataset containing images of cats and dogs for image classification.
8. To build an image classifier with Keras and Convolutional Neural Networks for the Fashion MNIST dataset.
9. To train a CNN model to classify images from the CIFAR-10 database.
10. To implement transfer learning using the pre-trained model (VGG16) on image dataset.

11. a) To perform tokenization and stemming on text data using NLTK

b) To perform lemmatization and remove stop words on text data using NLTK

c) To perform lemmatization and remove stop words on text data using Spacy

12. a) To perform tokenization and stemming on text data using NLTK

b) To perform lemmatization and remove stop words on text data using NLTK

c) To perform lemmatization and remove stop words on text data using Spacy