MCA Syllabus

Program Core
Semester I

Course code: CA403
Course title: COMPUTER ORGANIZATION AND ARCHITECTURE
Pre-requisite(s):
Co- requisite(s):
Credits: 3 L:3 T:0 P: 0
Class schedule per week: 03
Class: MCA
Semester / Level: I/4
Branch: Master of Computer Applications
Name of Teacher:

Course Objectives

This course enables the students to:

1.	Provide knowledge of Computer Architecture
2.	Employ knowledge of various Digital Logic Circuits, Data Representation, Register and Processor level Design and Instruction Set architecture
3.	Develop the logical ability to Determine which hardware blocks and control lines are used for specific instructions
4.	Understand memory organization, I/O organization and its impact on computer cost/performance.
5.	Know merits and pitfalls in computer performance measurements.

Course Outcomes

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

CO1	Describe the merits and pitfalls in computer performance measurements and analyze the impact of instruction set architecture on cost-performance of computer design
CO2	Explain Digital Logic Circuits, Data Representation, Register and Processor level Design and Instruction Set architecture
CO3	Solve problems related to computer arithmetic and Determine which hardware blocks and control lines are used for specific instructions
CO4	Design a pipeline for consistent execution of instructions with minimum hazards
CO5	Explain memory organization, I/O organization and its impact on computer cost/performance.

SYLLABUS

Module I:
INTRODUCTION
Digital Logic Design: Axioms and laws of Boolean algebra, Reduction of Boolean expressions, conversion between canonical forms, Karnaugh map (4 variable), Half Adder, full adder, 4-bit parallel parity bit generator, checker circuit, Decoder, Encoder, Multiplexer, IC RAM, ROM, Memory Organization, Sequential Circuits, State transistors, Flip-flop, RS, JK, D-Latch, Master-slave.

(8L)

Module II:
INSTRUCTION SET ARCHITECTURE
Memory Locations and Addresses: Byte Addressability, Big-Endian and Little-Endian Assignments, Word Alignment, Instructions and Instruction Sequencing, Addressing Modes, Assembly Language, Subroutines, Additional Instructions, dealing with 32-Bit Immediate Values.

(8L)

Module III:
BASIC PROCESSING UNIT & PIPELINING
Basic Processing Unit: Some Fundamental Concepts, Instruction Execution, Hardware Components, Instruction Fetch and Execution Steps, Control Signals, Hardwired Control, CISC-Style Processors.
Pipelining: Basic Concept, Pipeline Organization, Pipelining Issues, Data Dependencies, Memory Delays, Branch Delays, Pipeline Performance Evaluation.

(8L)

Module IV:
MEMORY ORGANIZATION
Basic Concepts, Semiconductor RAM Memories, Read-only Memories, Direct Memory Access, Memory Hierarchy, Cache Memories, Performance Considerations, Virtual Memory, Memory Management Requirements, Secondary Storage

(8L)

Module V:
INPUT OUTPUT & PARALLEL PROCESSING
Basic Input Output: Accessing I/O Devices, Interrupts, Input Output Organization: Bus Structure, Bus Operation, Arbitration, Interface, Interconnection Standards.
Parallel Processing: Hardware Multithreading, Vector (SIMD) Processing, Shared-Memory Multiprocessors, Cache Coherence, Message-Passing Multicomputers, Parallel Programming for Multiprocessors, Performance Modeling.

(8L)

Books recommended:

TEXT BOOK

1. Hamacher Carl, et. al, "Computer Organization and Embedded Systems", 6th Edition, Tata McGraw Hill,New Delhi, 2011.**(T1)**
2. Patterson David A., "Computer Organization and Design: The Hardware Software / Interface", 5thEdition, 1994.**(T2)**
3. Mano M. Morris, "Computer System Architecture", Revised 3rdEdition, Pearson Education.**(T3)**

Course code: CA405
Course title: DATA STRUCTURES AND ALGORITHMS
Pre-requisite(s): High Level languages like C, C++, Java or Python
Co- requisite(s): Data Structures Lab
Credits: 3 L: 3 T: 0 P: 0
Class schedule per week: 4
Class: MCA
Semester / Level: I/4
Branch: MCA

Course Objectives
This course enables the students to:

1.	Provide knowledge of practical implementations and usage of Data Structures and Algorithms.
2.	Employ knowledge of various data structures during construction of a program.
3.	Develop the logical ability to store and retrieve data efficiently.
4.	Develop an appreciation of graph theory-based solutions for real life problems.
5.	Design and construct object-oriented software with an appreciation for data abstraction.

Course Outcomes
After the completion of this course, students are expected to

CO1	Identify various data structures and their usages.
CO2	Apply data structures in the modeling of computer-based systems in a way that demonstrates comprehension of the trade-offs involved in design.
CO3	Demonstrate the usage of optimal trees, heaps and priority queues.
CO4	Implement sorting algorithms.
CO5	Develop programs using algorithms in graph theory.

SYLLABUS

Module I:
Fundamental Data Structures: Using Arrays, Singly Linked Lists, Circularly Linked Lists, Doubly Linked Lists, Asymptotic Analysis.

(8L)

Module II:
Stacks, Queues, Dequeues: The Stack, Queue, Dequeue ADTs, Simple Array Based Stack, Queue, Dequeue Implementation, Implementing Stack, Queue with Singly Linked List, Reversing an Array using Stack, Matching Parenthesis and HTML tags, A Circular Queue.

(8L)

Module III:
Trees: General Trees, Binary Trees, Implementing Trees, Tree Traversal Algorithms, Binary Search Trees, AVL Trees, B Trees.

(8L)

Module IV:
Sorting: Merge sort, Quick sort, Studying sorting through algorithmic lens, Comparing Sorting Algorithms.
Heap: Priority Queues, Array Implementation of Heaps, Construction of Heaps, Heap Sort.

(8L)

Module V:
Graphs: Data Structures for graphs, Graph Traversals, Transitive Closure, Directed Acyclic Graphs, Shortest Paths, Minimum Spanning Trees. **(8L)**

Text books:

1. Goodrich Michael T., Tamassia Roberto, Goldwasser Michael H. "Data Structures and Algorithms in Java", Wiley, 6th Edition, 2014.
2. Klein Shmuel Tomi, Basic Concepts in Data Structures, Cambridge University Press, 1st Edition, 2016.

Reference books:

1. Yedidyah Langsam, Moshe Augenstein J., Tenenbaum Aaron M. "Data Structures using JAVA", Pearson Education, 2009.
2. Brass Peter "Advanced Data Structures", Cambridge University Press, 1st Edition.

Course code: CA407
Course title: Database Design Concepts
Pre-requisite(s):
Co-requisite(s):
Credits: 3 L:3 T:0 P:0
Class schedule per week: 4
Class: MCA
Semester / Level: I/4
Branch: MCA

Course Objectives

This course enables the students to:

1.	Observe that how the real world data is stored, retrieved, and communicate under the DBMS environment
2.	Design a logical model which having the unique relation between the Data.
3.	Apply the query for the modification of the system.
4.	Develop a conceptual design which allows as to avoid anomalies in superior's data.
5.	Discuss a system which allows to restrict the uncontrolled exaction and provide rigorous variation of the task.

Course Outcomes

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

CO1	Describe various data models and schemas used in database management systems.
CO2	Explain the fundamental concepts, data definitions and query processing tasks in relational query languages.
CO3	Recognize database design theory, and evaluate functional dependencies and normal forms in databases.
CO4	Formulate the operations of transaction and concurrent query processing tasks to obtain the correct results even under strict time constraints.
CO5	Interpret the foundational concepts of distributed databases. Illustrate several techniques related to transaction management and query processing in distributed database management systems.

SYLLABUS

MODULE I:

Introduction and Conceptual Modelling: Purpose of Database Systems, Data Models, Schemas and Instances, Three-Schema Architecture and Data Independence, Database languages, Database Architecture, Classification of DBMS, relational database, Database users and Administrators, Advantages of DBMS. Entities and Entity Sets, Relationships and Relationship Sets, Keys, Mapping, Constraints, ER Diagram, Reducing ER Diagram to tables, Generalization and Specialization, Aggregation**.** **(8L)**

MODULE II:

Relational Model: Concepts, Constraints, Languages, Design and Programming: Relational database Schemas, Relational Algebra, Relational Calculus (Tuple Relational calculus and Domain Relational calculus), Update operations, Transactions, Dealing with constraint violations. Binary Relational operation: JOIN and DIVISION, SQL, More complex SQL Queries, Security & Integrity violations, authorization and views, integrity constants, encryption, Statistical databases **(8L)**

MODULE III:

Database Design Theory and Methodology: Pitfalls in relational database design, Functional Dependencies, Decomposition Using Functional Dependencies. Normalization using functional Dependencies, General Definition of First, Second, Third and Forth Normal Form. Boyce-Codd Normal Form(BCNF), Multivalued and join dependencies, DKNF**.** **(8L)**

MODULE IV:

Transaction Processing Concepts and Concurrency Control Techniques: Transaction Processing, Desirable Properties of Transactions, Transaction State, Characterizing Schedules based on Recoverability and Serializability. Lock-Based Protocols, Timestamp-Based Protocols, Validation-Based Protocols, Multiple Granularity, Deadlock Handling, Recovery and Atomicity, Log-Based Recovery**.** **(8L)**

MODULE V:

Distributed Databases and Client-Server Architectures: Concepts and Types of Distributed databases, data fragmentation, Replication and Allocation Techniques for Distributed Database Design, Query Processing in Distributed Databases, Overview of Concurrency Control and Recovery in Distributed Databases, An Overview of 3-Tier Client-Server Architecture. **(8L)**

Text Book:

1. Elmasri Ramez, & Navathe S.B., "Fundamentals of Database Systems", 5th Edition, Pearson Education, 2006.

Reference Book:

1. Silberschatz A., &Korth H., "Database Systems Concepts", 5th Edition, McGraw Hill Higher Education, 2005.

Course code: CA409
Course title: OBJECT ORIENTED DESIGN USING JAVA
Pre-requisite(s):
Co- requisite(s):
Credits: 3 L:3 T:0 P: 0
Class schedule per week: 3
Class: MCA
Semester / Level: I/4
Branch: MCA

Course Objectives
This course enables the students:

1.	The course shall allow students to understand the basic tenets of OOP.
2.	The course will exemplify the basic syntax and constructs of JAVA
3.	The course will help students understand the application OOP principles and Improve their programming skills in core Java
4.	The course will explain basic JAVA characteristics and their working.
5.	The course aims to expose students to Use the Java packages, applets for software development

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

CO1	Identify the difference between procedural and OO programming.
CO2	Construct programs using various OOP principles.
CO3	Apply the knowledge gained for their project work as well as to develop some GUI applications using JAVA
CO4	Operate on files and strings in real life scenarios.
CO5	Analyze thread performance and inter thread communication issues

SYLLABUS

MODULE I:
Procedure-Oriented Programming, Object-Oriented programming, Benefits of OOP, Applications of OOP, Basics, Evolution of Java, Structure of JAVA Program, Simple Java Program, Tokens, Comments, Identifiers, Operators, Literals, Control Structures. Java Environment Setup, Compiling a Java Program, Java Virtual Machine, Philosophy of Java and Benefits**.** **(8L)**

MODULE II:
Data types and program statements: Primitive and reference data types, variables and constants, enumerated constants, labelled statement, expression and null statements, compound statement, control statement – decision and loops, jump statement, declaration statement, try-throw-catch-finally statement, declaring and creating arrays, accessing array elements, assigning values to array elements, multidimensional arrays. **(8L)**

MODULE III:

Functions, Data Abstraction and classes: Declaration, definition and call, main method arguments, reference variables, method overloading, parameter passing by value for primitive types, object references and arrays, scope of variables, return from methods.

Class and object, class members and initialization, access rights of members – public, private and protected access modifiers, constructor and copy constructor, mutability, finalization, dynamic memory management, garbage collection, this keyword, static members, scope of variables, interface – declaration, implementation and extending, package and package visibility.

(8L)

MODULE IV:

Inheritance and Collection classes: multi level and single inheritance, multiple inheritance of interfaces, Object class, access rights in subclasses and packages, constructor calling sequence, super keyword, dynamic binding of methods, abstract class, overriding, shadowing and hiding, finalize, association, aggregation and composition. String, StringBuffer, Date, Calendar, Math, Object, Class, Exception class. **(8L)**

MODULE V:

Input/Output and JAVA Applets: Stream clases – InputStream, OutputStream, Buffered Stream, file classes and handling, pushback streams, reader and writer classes, file reader and writer, serialization.

Applet code example, HTML tags for applet, applet lifecycle, color, font and basic GUI handling, basic graphics, animation. **(8L)**

Text books:

E. Balagurusamy - Programming in Java, 2nd Edition; Tata McGraw Hill Publication; New Delhi.

Reference books:

Patrick Naghton & H. Schildt – The Complete Reference Java 2, Tata McGraw Hill Publication, New Delhi.

Dietel,Dietel - Java How to program , 7th edition; Pearson Education , New Delhi.

Course code: CA411
Course title: MODERN OPERATING SYSTEMS
Pre-requisite(s): Data Structure, Computer System Architecture, Basic Course on Computer Programming
Co- requisite(s):
Credits: 3 L:3 T:0 P: 0
Class schedule per week: 03
Class: MCA
Semester / Level: I/4
Branch: MCA
Course Objectives
This course enables the students to:

1.	Present the main components of OS and their working
2.	Introduce the concepts of process and thread and their scheduling policies
3.	Introduce the various memory management techniques.
4.	Analyze the different techniques for managing memory, I/O, disk and files.
5.	Introduce the security and protection features of an Operating System.

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

CO1	Describe the main components of OS and their working
CO2	Explain the concepts of process and thread and their scheduling policies
CO3	Explain the various memory management techniques.
CO4	Compare the different techniques for managing memory, I/O, disk and files.
CO5	Explains the security and protection features of an Operating System.

SYLLABUS

MODULE I:
Overview of Operating Systems: OS and the Computer System, Efficiency, System Performance and User Convenience, Classes of Operating Systems, Batch Processing Systems, Multiprogramming Systems, Time Sharing Systems, Real Time Operating Systems, Distributed Operating Systems, Modern Operating Systems.

(8L)

MODULE II:

Processes and Threads: Processes and Programs, Programmer view of Processes, OS view of Processes, Threads, Case studies of Processes and Threads.

Scheduling: Preliminaries, Non-preemptive Scheduling Policies, Preemptive Scheduling Policies, Scheduling in Practice, Real Time Scheduling, Scheduling in Unix, Scheduling in Linux, Scheduling in Windows, Performance Analysis of Scheduling Policies. **(8L)**

MODULE III:

Memory Management: Managing the Memory Hierarchy, Static and Dynamic Memory Allocation, Memory Allocation to a Process, Reuse of Memory, Contiguous Memory Allocation, Noncontiguous Memory Allocation, Paging, Segmentation, Segmentation with Paging, Kernel Memory Allocation, A Review of Relocation, Linking and Program Forms.

Virtual Memory: Virtual Memory Basics, Demand Paging, Page Replacement Policies, Memory Allocation to a Process, Shared Pages, Memory Mapped Files, Unix Virtual Memory, Linux Virtual Memory, Virtual Memory using Segmentation.

(8L)

MODULE IV:

File Systems: File System and IOCS, Files and File Operations, Fundamental File Organizations, Directory Structures, File Protection, Interface between File System and IOCS, Allocation of Disk Space, Implementing File Access, File Sharing Semantics, File System Reliability, Virtual File System, Unix File System, Linux File System, Windows File System, Performance of File Systems.

(8L)

MODULE V:

Security and Protection: Overview of Security and Protection, Goals of Security and Protection, Security Attacks, Formal and Practical aspects of Security, Encryption, Authentication and Password Security, Access Descriptors and the Access Control Matrix, Protection Structures, Capabilities, Unix Security, Linux Security, Windows Security.

(8L)

Text Book:

1. Dhamdhere D.M., "Operating Systems: A Concept-Based Approach", 2nd Edition, TMH, New Delhi, 2006.

Reference Books:

1. SilberschatzA.,Galvin Peter B.,Greg Gagne, "Operating System Concepts", 6th Edition, John Wiley, Indian Reprint, 2003.
2. Crowley C., "Operating Systems: A Design-Oriented Approach", TMH, New Delhi, 2002.
3. Deitel H.M., "Operating Systems", 2nd Edition, Pearson Education, 2003.
4. Tanenbaum A.S., "Operating System: Design and Implementation", PHI, New Delhi, 2002.

Course code: MT123
Course title: BUSINESS COMMUNICATION
Pre-requisite(s):
Co- requisite(s):
Credits: 3 L: 2 T: 0 P: 2
Class schedule per week: 03
Class: MCA
Semester / Level: I/1
Branch: Master of Computer Applications
Name of Teacher:

Course Objectives
This course enables the students to:

1.	Analyze and demonstrate writing and speaking processes through invention, organization, drafting, revision, editing, and presentation.
2.	Understand the importance of specifying audience and purpose and to select appropriate communication choices.
3.	Interpret and appropriately apply modes of expression, i.e., descriptive, expositive, Narrative, scientific, and self-expressive, in written, visual, and oral communication
4.	Participate effectively in groups with emphasis on listening, critical and reflective thinking, and responding.
5.	Develop the ability to research and write a documented paper and/or to give an oral presentation.

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

CO1	Apply business communication strategies and principles to prepare effective communication for domestic and international business situations.
CO2	Utilize analytical and problem-solving skills appropriate to business communication.
CO3	Participate in team activities that lead to the development of collaborative work skills.
CO4	Select appropriate organizational formats and channels used in developing and presenting business messages
CO5	Communicate via electronic mail, Internet, and other technologies and deliver an effective oral business presentation.

SYLLABUS

Module I:
Introduction to Business Communication:
Importance and Objectives of Business communication, Process of communication, Barriers to effective communication, Techniques of effective communication. Forms of communication (Written, Oral, audio-visual communication).

(8L)

Module II:

Managing Business Communication:

Formal and Informal communication, Non- verbal communication (Body language, Gestures, Postures, Facial expressions). The cross-cultural dimensions of business communication. Techniques to effective listening, methods and styles of reading.

(8L)

Module III:

Other aspects of communication:

Vocabulary: Single word substitution, Idioms and phrases, Precis writing, Comprehension.

Group Discussions, Extempore, Principles of effective speech and presentations, Role-playing.

(8L)

Module IV:

Introduction to managerial writing:

Business letters: Inquiries, Circulars, Quotations, Orders, Acknowledgement, Claims & adjustments, Collection letters, Sales letters, Drafting of different resumes, Covering letters Applying for a job, Social correspondence, Invitation to speak.

Official Correspondence: Memorandum, Notice, Agenda, Minutes, Circular letters.

(8L)

Module V:

Report writing and Technical Proposals:

Business reports, Types, Characteristics, Importance, Elements of structure, Process of writing, Order of writing, the final draft, checklists for reports.

Technical proposals, Definitions, types and format.

(8L)

Books recommended:

TEXT BOOKS

1. "Communication Skills", Sanjay Kumar &PushpLata, Oxford University Press. **(T1)**
2. "Business Correspondence and Report Writing", R.C.Sharma, Krishna Mohan, McGraw Hill. **(T2)**
3. "Communication for Business", Shirley Taylor, V. Chandra, Pearson. **(T3)**

REFERENCE BOOKS

1. "Business Communication", HorySankar Mukherjee, Oxford University Press. **(R1)**
2. "Basic Business Communication", Lesikar I Flatley, McGraw Hill. **(R2)**
3. "Business Communication Today", Bovee, Thill and Chaterjee, Pearson. **(R3)**

Semester I
Laboratory

Course code: CA406
Course title: Data Structures and Algorithms Lab
Pre-requisite(s): High Level languages like C, C++, Java or Python
Co- requisite(s): Data Structures Lab
Credits:1.5 L: 0 T: 0 P:3
Class schedule per week: 3
Class: MCA
Semester / Level: I/4
Branch: MCA

Course Objectives
This course enables the students:

1.	To assess how the choice of data structures and algorithm design methods impact the performance of programs.
2.	To choose the appropriate data structure and algorithm design method for a specified application.
3.	To solve problems using data structures such as linear lists, stacks, queues, hash tables, binary trees, heaps, binary search trees, and graphs and writing programs for these solutions.
4.	Analyse and compare the different algorithms

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

CO1	Choose an appropriate data structure given a computational problem
CO2	Design and analyze the time and space efficiency of various data structures
CO3	Analyze run-time execution of previous learned sorting methods, including selection, merge sort, heap sort and quick sort
CO4	Have practical knowledge on the applications of data structures
CO5	Justify the choice of data structure for a given problem

SYLLABUS

1. Program to Find the Number of Elements in an Array
2. Develop and implement a menu driven program in C for the following Array operations
    a. Creating Array of N Integer elements.
    b. Display of Array elements with suitable headings.
    c. Inserting an element (ELEM) at a given valid position (POS).
    d. Deleting an element at a given valid position (POS).
    e. Exit
3. Programs for Stack, Queues and Circular Queues using Arrays
4. Program to convert an Infix Expression into Postfix and Postfix Evaluation
5. Program to implement stack using arrays
6. Program to implement stack using linked list
7. Program to implement multiple stack in a single array

8. Program to convert infix notation to postfix notation using stacks
9. Program to implement queue using arrays
10. Program to implement queue using pointers
11. Program to reverse elements in a queue
12. Program to implement circular queue using arrays
13. Program to create add remove & display element from single linked list
14. Program to create add remove & display element from double linked list
15. Program to count number of nodes in linear linked list
16. Program to create add remove & display element from circular linked list
17. Programs to implement stack & queues using linked representation
18. Program to concatenate two linear linked lists
19. Program to accept a singly linked list of integers & sort the list in ascending order.
20. Program to reverse linked list
21. Program to represent polynomial using linked list
22. Program to add two polynomials using linked list
23. Program for the creation of binary tree, provide insertion & deletion in c
24. Program for pre-order, post-order & in-order traversals of a binary tree using non reccursive.
25. Program to count no, of leaves of binary tree
26. Program for implementation of B-tree (insertion & deletion)
27. Program for implementation of multi-way tree in c
28. Program for implementation of AVL tree
29. Program to implement bubble sort program using arrays
30. Program to implement merge sort using arrays
31. Program to implement selection sort program using arrays
32. Program to implement insertion sort program using arrays
33. Program to implement topological sort using arrays
34. Program to implement heap sort using arrays
35. Program to implement heap sort using pointers
36. Program to implement bubble sort program using pointers
37. Program to implement linear search using pointers
38. Program to implement binary search using pointers
39. Program to implement linear search using arrays
40. Program to implement binary search using arrays

Text books:

1. Baluja G S, “Data Structure through C”, Ganpat Rai Publication, New Delhi, 2015.
2. Pai G A V, “Data Structures and Algorithms: Concepts, Techniques and Applications”, 2nd Edn, Tata McGraw-Hill, 2008.
3. Horowitz E., Sahni S., Susan A., “Fundamentals of Data Structures in C”, 2nd Edition, University Press, 2010.

Reference books:

1. Tremblay J. P., Sorenson P. G, “An Introduction to Data Structures with Applications”, 2nd Edn, McGraw-Hill, Inc. New York, NY, USA.
2. Lipschutz Seymour, “Data Structures”, 6th Edn, 9th Reprint 2008, Tata McGraw-Hill.
3. Drozdek Adam, “Data Structures and Algorithms in C++”, Thomson Learning, New Delhi – 2007.
4. Feller J., Fitzgerald B., “Understanding Open Source Software Development”, Pearson Education Ltd. New Delhi

Course code: CA408
Course title: DATABASE DESIGN CONCEPTS LAB
Pre-requisite(s):
Co-requisite(s):
Credits: 1.5 L: 0 T: 0 P:3
Class schedule per week: 3
Class: MCA
Semester/Level: I/4
Branch: MCA

Course Objectives

This course enables the students:

1.	To observe that how the real world data is stored, retrieved, and communicate under the DBMS environment
2.	To design a logical model which having the unique relation between the Data.
3.	To apply the query for the modification of the system.
4.	To develop a conceptual design which allows as to avoid anomalies in superior's data.
5.	To discuss a system which allows to restrict the uncontrolled exaction and provide rigorous variation of the task.

Course Outcomes

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

CO1	Describe various data models and schemas used in database management systems.
CO2	Explain the fundamental concepts, data definitions and query processing tasks in relational query languages.
CO3	Recognize database design theory, and evaluate functional dependencies and normal forms in databases.
CO4	Formulate the operations of transaction and concurrent query processing tasks to obtain the correct results even under strict time constraints.
CO5	Interpret the foundational concepts of distributed databases. Illustrate several techniques related to transaction management and query processing in distributed database management systems.

SYLLABUS

For the Tables given below: emp(empno,ename,job,mgr,hiredate,sal,comm,deptno,gr),

dept(deptno,dname,loc)

Write the following queries:

1. List all information about all department from emp table.
2. List all employee names along with their salaries from emp table.

3. List all department numbers, employee numbers and their managers numbers in descending order of deptno from emp table.
4. List department names and locations from the dept table.
5. List the employees belonging to the department 20.
6. List the name and salary of the employees whose salary is more than 1000.
7. List the names of the clerks working in the department 20.
8. List the names of analysts and salesmen.
9. List the details of the employees who have joined before the end of September 81.
10. List the names of employees who are not managers.
11. List the names of employees whose employee number are 7369, 7521, 7839, 7934, 7788.
12. List the employee details not belonging to the department 10, 30, and 40.
13. List the employee name and salary, whose salary is between 1000 and 2000.
14. List the employee names, who are not eligible for commission.(salary having >15,000 eligible for commission)
15. List the employees who are eligible for commission.
16. List the details of employees, whose salary is greater than 2000 and commission is NULL.
17. List the employees whose names start with an "S" (not"s").
18. List the name, salary and PF amount of all the employees(PF is calculated as 10% of salary).
19. List the empno, ename, sal in ascending order of salary.
20. List the employee name, salary, job and Department no descending order of Department No and salary.
21. List the employee details in ascending order of salary.
22. List the employee details in descending order of salary
23. Display name, and sal and commission of all employees whose monthly salary is greater than their commission.
24. Select SMITH HAS WORKED IN THE POSITION OF CLERK IN DEPT 20.Display result in this format.
25. Generate a statement which prompts the user at runtime. The intention is to display employees hired between 2 given dates.
26. Define a variable representing an expression used to calculate total annual remuneration. Use the variable in a statement which finds all employees who earn $30000 a year or more.
27. List all the employees name and salaries increased by 15% and expressed as a whole number of dollars.
28. Produce the following

EMPLOYEE AND	**JOB**
SMITH	CLERK
ALLEN	SALESMAN

29. Produce the following output:

SMITH (Clerk)

ALLEN (Salesman)

30. Do a case sensitive search for a list of employees with a job that the user enters.
31. It has been discovered that the sales people in dept. 30 are not all male. Please produce the following output.

ENAME	**DEPTNO**	**JOB**
ALLEN	30	Sales Person

32. Display each employees name and hiredate of dept 20.
33. Display each employees name, hiredate and salary review date. Assume salary review date is one year fromhiredate. Output should be in ascending review date.
34. Print list of employees displaying just salary, if more than 1500. If exactly 1500 display " On Target". If less than 1500 display " Below 1500".
35. Write a query which returns DAY of the week (i.e. MONDAY) for any date entered in the format DD/MM/YY.
36. Write a query to calculate length of service of each employee.
37. Find the minimum salary of all employees.
38. Find the maximum, minimum, and average salaries of all employees.
39. List the maximum and minimum salary of each job type.
40. Find how many managers are in each dept.
41. Find the average salary and average total remuneration of each job type. Remembers sales man earn commission.
42. Find out the difference between highest and lowest salary.
43. Find all department s which have more than three employees.
44. Check whether all employee nos are unique. (No Duplicate)
45. List lowest paid employee working for each Manager. Exclude any groups where the minimum salary is less than 1000. Sort the output by salary.
46. Produce a list showing employees 'salary grade'.(> 10000 A, >10000 &<20000 B, >20000 C)
47. Show only employee on Grade C.
48. .Show all employee in Dallas.
49. List the employees name, job, salary, grade and department for everyone in the company except clerks. Sort on salary, displaying the highest first.
50. List the following details of employees who earn $36000 a year or who are clerks.

Ename Job Annual Sal Dept no Dname Grade

51. Display all employees who earn less than their managers.
52. Display all employees by name and eno along with their managers name and number.
53. Modify above spoliation to display KING who has no MANAGER.
54. Find the job that was files in the first half of 1983 and the name job that was filled in the same period in 1984.
55. Find all employees who have joined before their manager.

EMPLOYEE HIREDATE MANAGER HIREDATE

56. Find the employees who earn the highest salary in each job, type, sort in descending order of salary.
57. Find the employees who earn the minimum salary for their job, Display the result in descending order of salary

58. Find the most recently hired employees in the department. Order by hiredate.
59. Show the details of any employee who earns a salary greater than the average for their department. Sort in department number order.
60. List all department where there are no employees.

Text book:

1. SQL, PL/SQL the programming Language of Oracle, Ivan Bayross, 4th edition

Course code: CA410
Course title: OBJECT ORIENTED DESIGN USING JAVA LAB
Pre-requisite(s):
Co- requisite(s):
Credits:1.5 L: 0 T: 0 P:3
Class schedule per week: 3
Class: MCA
Semester / Level: I/4
Branch: MCA

Course Objectives
This course enables the students to:

1.	Introduce the concepts ofobject-oriented programming and features of object-oriented programming languages.
2.	To learn advanced features of the JAVA programming language as a continuation of the previous course.
3.	To learn the characteristics of an object-oriented programming language: data abstraction and information hiding, inheritance, and dynamic binding of the messages to the methods
4.	To learn the basic principles of object-oriented design and software engineering in terms of software reuse and managing complexity.
5.	To enhance problem solving and programming skills in JAVA with extensive programming projects

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

CO1	Explain basic concepts of object-oriented programming.
CO2	Use the characteristics of an object-oriented programming language in a program.
CO3	Use the basic object-oriented design principles in computer problem solving
CO4	Develop their own Applications /Projects using JAVA
CO5	Simulate the problem in the subjects like Operating system, Computer networks and real world problems.

SYLLABUS

List of Programs as Assignments:

Objective: To Understand and Implement basic OOP features

1.Write a Program to design a class having static member function named showcount() which has the property of displaying the number of objects created of the class.
2. Write a Program which creates & uses array of object of a class.(for eg. implementing the list of Managers of a Company having details such as Name, Age, etc..).

Objective: To Understand and Implement special types of functions like friend function
3. Write a Program to swap private data members of classes named as class_1, class_2 using friend function.
4.Write an inline function to find largest of three number

Objective: To Understand and Implement the concept of constructors
5. Write a Program using copy constructor to copy data of an object to another object.
6. Write a program to perform addition of two complex numbers using constructor overloading. The first constructor which takes no argument is used to create objects which are not initialized, second which takes one argument is used to initialize real and imag parts to equal values and third which takes two argument is used to initialized real and imag to two different values.

Objective: To Understand and Implement the concept of Polymorphism
7. Write a program for overloading operator++ and operator—using friend functions
8. Write a program for developing a matrix class which can handle integer matrices of different dimensions. Also overload the operator for addition, multiplication & comparison of matrices.
9.Write a program to compute area of right angle triangle, equilateral triangle, isosceles triangle using function overloading concept.

Objective: To Understand and Implement the concept of Inheritance
10.Write a Program to design a student class representing student roll no. and a test class (derived class of student) representing the scores of the student in various subjects and sports class representing the score in sports. The sports and test class should be inherited by a result class having the functionality to add the scores and display the final result for a student.

11.Write a Program illustrating how the constructors are implemented and the order in which they are called when the classes are inherited. Use three classes named alpha, beta, gamma such that alpha, beta are base class and gamma is derived class inheriting alpha & beta.
Objective: To Understand and Implement exception handling
12. Write a program to raise an exception if any attempt is made to refer to an element whose index is beyond the array size.

Objective: To Understand and Implement File Operations
13.Write a program to read the class object of student info such as name , age ,sex ,height and weight from the keyboard and to store them on a specified file using read() and write() functions. Again the same file is opened for reading and displaying the contents of the file on the screen.

14.Write a program to perform the deletion of white spaces such as horizontal tab, vertical tab, space ,linefeed ,new line and carriage return from a text file and store the contents of the file without the white spaces on another file.

Books recommended:

Text books:

E. Balagurusamy - Programming in Java, 2nd Edition; Tata McGraw Hill Publication; New Delhi.

Reference books:

Patrick Naghton & H. Schildt – The Complete Reference Java 2, Tata McGraw Hill Publication, New Delhi.

Dietel,Dietel - Java How to program , 7th edition; Pearson Education , New Delhi.

Program Core
Semester II

COURSE INFORMATION SHEET

Course Code: MT114
Course Title: FUNDAMENTALS OF MANAGEMENT AND ORGANIZATION BEHAVIOUR
Pre-requisite(s):
Co- requisite(s):
Credits: 3 L: 3 T: 0 P: 0
Class schedule per week: 03
Class: MCA
Semester / Level: I/1
Branch: Master of Computer Applications
Name of Teacher:

Course Objectives

This course enables the students to:

1.	To understand the concept of management principles and practices, as a discipline, as an art or a science, management and administration, managerial skills, roles of a manager and levels of management.
2.	To compare and contrast various development of management thought such as early classical approaches, administrative management, neo-classical approaches, behavioral approaches, modern approaches, business ethics and social responsibility.
3.	To classify the type of plans and to critically examine different types of planning and select the types of decisions for further growth of the organization.
4.	To create an organizational structure-formal and informal organization to point out span of control, authority, responsibility, accountability, delegation of authority, Departmentation, decentralization and can design a plan for manpower planning, job design, recruitment and selection, training and development and performance appraisal.
5.	To develop the core of leadership, directing function, motivational theories, communication process and different types of control system to facilitate change for the development of the organization.

Course Outcomes

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

CO1	To Debate management principles and practices as an art or a science, classify managerial skills and roles being played by a manager and recommend appropriate organisational structure.
CO2	To identify factors affecting Decision-making and Planning activity at all levels in an organization.
CO3	To Explain the key decisions related to the various Staffing functions in an organisation.
CO4	To analyse leadership styles, Communication and Motivation strategies adopted by managers and comment on their appropriateness. vis a vis nature of the organisation.

SYLLABUS

Module I:

Introduction: Concepts, Function or Process, Management Discipline, as an Arts or Science, Understanding Management and Administration, Managerial Skills, Roles of a Manager, Levels of Management.

Development of Management Thought: Classical Approaches- Scientific Management, Administrative Management: Bureaucracy, Behavioral Approach.

(8L)

Module II:

Planning: Nature and significance of Planning, Types of plans, Process of Planning, **Organizing:** Process of Organizing, Forms of Organizational Structure, Formal and informal organization

(8L)

Module III:

Staffing: Concept, Manpower Planning, Process of Manpower planning, Recruitment & Selection, Training & Development, Performance Appraisal.

Motivating: Significance of Motivation**,** Motivation process, Theories of Motivation and their application

(8L)

Module IV:

Leading: Concept of Leadership, Leadership Style, Theories of Leadership

Communication: Process, Importance of Communication, Communication Channels, Barriers to Communication. **(8L)**

Module V:

Controlling: Definition, Importance of controlling, Characteristics of control, Control process, Types of Control System, Introduction to CSR and Sustainable Development. **(8L)**

Books recommended:

TEXT BOOKS

1. "Management", Stoner and Freeman, Prentice Hall of India. **(T1)**
2. "Essentials of Management", Koontz and Heinz Weihrich, McGraw Hill. **(T2)**
3. "Management", Robbins &Coulter, Prentice Hall of India. **(T3)**

REFERENCE BOOKS

1. "Principles of Management", Gilbert, Mc Graw Hill. **(R1)**
2. "Principles and Practices", T. N. Chhabra, Dhanpat Rai and Sons Pvt. Ltd. **(R2)**
3. "Management: A Global and Entrepreneurial Perspective", Weihrich Heinz &Koontz Harold, Mc Graw Hill. **(R3)**
4. "Principles of Management", P.C.Tripathi and P.N.Reddy, Mc Graw Hill. **(R4)**

COURSE OUTCOME (CO) ATTAINMENT ASSESSMENT TOOLS & EVALUATION PROCEDURE

Direct Assessment

Assessment Tool	% Contribution during CO Assessment
Continuous Internal Assessment	50
Semester End Examination	50

Continuous Internal Assessment	% Distribution
3 Quizzes	30 % (3 × 10%)
Assignment (s)	10
Seminar before a committee	10

Assessment Components	CO1	CO2	CO3	CO4
Continuous Internal Assessment	✓	✓	✓	✓
Semester End Examination	✓	✓	✓	✓

Indirect Assessment –

1. Student Feedback on Faculty
2. Student Feedback on Course Outcome

MAPPING BETWEEN COURSE OUTCOMES AND PROGRAM OUTCOMES

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	1	2	1	2	2	1	2
CO2	3	1	2	1	1	2	1	1
CO3	3	3	1	2	1	1	1	1
CO4	2	2	1	2	1	1	2	1

If satisfying and < 34% = 1, 34-66% = 2, > 66% = 3

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

CD	Course Delivery methods	Course Outcome	Course Delivery Method
CD1	Lecture by use of boards/LCD projectors/OHP projectors	CO1	CD1
CD2	Tutorials/Assignments	CO2	CD1 and CD2
CD3	Seminars	CO3	CD1 and CD2
CD4	Mini projects/Projects	CO4	CD 1,CD2
CD5	Laboratory experiments/teaching aids		
CD6	Industrial/guest lectures		
CD7	Industrial visits/in-plant training		
CD8	Self- learning such as use of NPTEL materials and internets		

Course code: CA413
Course title: DATA COMMUNICATION AND COMPUTER NETWORK
Pre-requisite(s):
Co- requisite(s):
Credits: 3 L:3 T:0 P:0
Class schedule per week: 03
Class: MCA
Semester / Level: II/4
Branch: MCA

Course Objectives
This course enables the students:

1.	To build an understanding of the fundamental concepts of the data communication model and communications architecture.
2.	To study characteristics of communication mediums and the characteristics of signals propagated through different transmission media, including concepts of transmission impairments.
3.	To understand the basic principles of signal encoding techniques, error-detection, and error-correction techniques.
4.	To understand techniques for flow control and multiplexing for maximum utilization of bandwidths in the data communications process.
5.	To understand the various switching techniques and routing techniques for efficient transmission.

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

CO1	Understand and be able to explain the principles of a layered protocol architecture; be able to identify and describe the system functions in the correct protocol layer and further describe how the layers interact.
CO2	Understand, explain and calculate digital transmission over different types of communication media.
CO3	Understand, explain and solve mathematical problems for data-link and network protocols.
CO4	Describe the principles of access control to shared media and perform performance calculations.
CO5	Understand and explain the principles and protocols for route calculations and be able to perform such calculations.

SYLLABUS

MODULE - I
Data Communications and Networking Overview: A Communications Model, Data Communications, Data Communication Networking.
Protocol Architecture: The Need for a Protocol Architecture, A Simple Protocol Architecture, OSI, The TCP/IP Protocol Architecture.

(8L)

MODULE - II

Data Transmission: Concepts and Terminology, Analog and Digital Data Transmission, Transmission Impairments, Channel Capacity.

Guided and Wireless Transmission: Guided Transmission Media, Wireless Transmission, Wireless Propagation, Line-of-Sight Transmission.

(8L)

MODULE - III

Signal Encoding Techniques: Digital Data Digital Signals, Digital Data Analog Signals, Analog Data Digital Signals, Analog Data Analog Signals.

Digital Data Communication Techniques: Asynchronous and Synchronous Transmission, Types of Errors, Error Detection, Error Correction, Line Configurations, Interfacing.

(8L)

MODULE – IV

Data Link Control: Flow Control, Error Control, High-Level Data Link Control (HDLC).

Multiplexing: Frequency Division Multiplexing, Synchronous Time Division Multiplexing, Statistical Time Division Multiplexing.

Circuit Switching and Packet Switching: Switching Networks, Circuit-Switching Networks, Circuit-Switching Concepts, Control Signaling, Softswitch Architecture, Packet-Switching Principles, X.25, Frame Relay.

(8L)

MODULE -V

Asynchronous Transfer Model: Protocol Architecture, ATM Logical Connections, ATM Cells, Transmission of ATM Cells, ATM Service Categories, ATM Adaptation Layer.

Routing in Switched Networks: Routing in Circuit-Switching Networks, Routing in Packet-Switching Networks, Least-Cost Algorithms.

(8L)

Text Book:

1. Stallings W. “Data and Computer Communications”, 7thEdition., Pearson Education./ PHI, New Delhi, 2006.

Reference Books:

1. Forouzan B. A., “Data Communications and Networking”, 4th Edition. TMH, New Delhi, 2006.
2. Gupta P.C. “Data Communications and Computer Networks”, PHI, New Delhi 2006.

Course code: CA415
Course title: SOFTWARE ENGINEERING PRINCIPLES
Pre-requisite(s):
Co- requisite(s):
Credits: 3 L:3 T:0 P: 0
Class schedule per week: 03
Class: MCA
Semester / Level: II/4
Branch: MCA

Course Objectives

This course enables the students to:

1.	Students are effective team members, aware of cultural diversity, who conduct themselves ethically and professionally
2.	Students use effective communication skills and technical skills to assure production of quality software, on time and within budget.
3.	Students build upon and adapt knowledge of science, mathematics, and engineering to take on more expansive tasks.
4.	Able to increase level of self-reliance, technical expertise, and leadership.

Course Outcomes

After the completion of this course, students will be:

CO1	Explain the software engineering principles and techniques
CO2	Apply Software Project Management Practices
CO3	Apply the knowledge gained for their project work as well as to develop software following software engineering standards
CO4	Analyze various methods of software testing strategies
CO5	Develop self-reliance, technical expertise, and leadership.

SYLLABUS

MODULE: I

Introduction to Software Engineering: Evolving Role of Software, Changing Nature of Software, Legacy Software, Process Frame work, Process Patterns, Process Models, Waterfall Model, Incremental Process Models, Evolutionary Process Models, Specialized Process Models, Unified Process Model, Agile Process Model.

(8L)

MODULE: II

Requirement Engineering: A bridge to design and construction, Requirement Engineering Task, Initiating the Requirement Engineering Process, Eliciting Requirements, Developing Use case, Building the Analysis Model, Negotiating Requirements, Validating Requirements.

(8L)

MODULE: III

Design Engineering: Design Process and Design Quality, Design Concepts, Design Models, Pattern Based Software Design.

(8L)

MODULE: IV

Testing Strategies and Testing Tactics: Strategic Approach to software Testing, Test Strategies for conventional and Object Oriented Software, Validation Testing System Testing, White Box Testing, Basic Path Testing Control Structure Testing, Black Box Testing, Object Oriented Testing Methods. **(8L)**

MODULE: V

Metric for process and Estimation Techniques: Process metrics, Software Measurement, Software Project Estimation, Decomposition Techniques, Empirical Estimation Models, Estimation for Object Oriented Projects Specialized Estimation Techniques.

Software Quality and Configuration Management: Quality Concepts, Software Quality Assurance, Software Reliability, Software Configuration Management, SCM Repository, SCM Process. **(8L)**

Text Book:

1. Pressman Roger S., "Software Engineering – A Practitioner's Approach", 6thEdition., Tata McGraw Hill.

Reference Books:

1. Vliet Haus Van, "Software Engineering – Principles and Practice", Wiley John and Sons, 2nd Edition.
2. Sommerville Ian, "Software Engineering", 7th Edition., Pearson Education.

Course code: CA417
Course title: THEORY OF COMPUTATION
Pre-requisite(s):
Co- requisite(s):
Credits: 3 L:3 T:0 P: 0
Class schedule per week: 03
Class: MCA
Semester / Level: II/4
Branch: MCA

Course Objectives
This course enables the students to:

1.	Define a system and recognize the behavior of a system.
2.	Design finite state machines and the equivalent regular expressions.
3.	Construct pushdown automata and the equivalent context free grammars
4.	Design Turing machines and Post machines
5.	Learn about the issues in finite representations for languages and machines, as well as gain a more formal understanding of algorithms and procedures.

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

CO1	Relate formal languages and mathematical models of computation
CO2	Attain knowledge about different types of languages and the corresponding machines
CO3	Learn about the pushdown machine and its role in compiler construction
CO4	Understand the capability of real computers and learn examples of unsolvable problems.
CO5	Analyze classes of P, NP, NP-C and NP-Hard problems

SYLLABUS

MODULE: I
Basic Mathematical Objects and Mathematical Induction: Sets, logic, Functions, Relations, Alphabets, Strings, Languages, Principle of mathematical induction, Recursive definition. **(8L)**

MODULE: II
Regular Expressions and Finite Automata: Regular languages and Regular Expressions, Memory required to recognize a language, Finite Automata, capability & limitations of FSM, Deterministic Finite Automata , Non-Deterministic Finite Automata, NFA with e-moves, regular sets & regular expressions, Equivalence of DFA and NDFA, NFA from regular expressions, regular expressions from DFA, Moore versus Mealy m/c, two way finite automata equivalence with one way , Kleen's Theorem, applications of finite automata. **(8L)**

MODULE: III

Regular and Non-regular languages: Criterion for Regularity, Minimal Finite Automata, Pumping Lemma for Regular Languages, Decision problems, Regular Languages and Computers.

Context Free Grammars: Introduction, definition, Regular Grammar, Derivation trees, Ambiguity, Simplified forms and Normal Forms, Applications.

(8L)

MODULE: IV

Pushdown Automata: Definition, Moves, Instantaneous Descriptions, Language recognised by PDA, Deterministic PDA, Acceptance by final state & empty stack, Equivalence of PDA , Pumping lemma for CFL, Interaction and Complements of CFL,Decision algorithms.

Turing Machines: Definition and examples, Computing Partial Functions with Turing Machine(TM), Combining TMs, Variations of TMs, Multi-tape TMs, Non-deterministic TM, Universal TM, Church Thesis.

(8L)

MODULE: V

Recursively Enumerable Languages: Recursively Enumerable and Recursive, Enumerating Language, Context Sensitive and Chomosky Hierarchy.

Unsolvable Problems and Computable Functions: Nonrecursive Language and unsolvable Problems, Halting Problem, Rice Theorem, Post Correspondence Problem.

Computational Complexity: Discussion on P, NP, NPC and NP-Hard Problems.

(8L)

Text Books:

1. Martin John "Introduction to Languages and the Theory of Computation", 3rd Edition, TMH.

Reference Books:

1. Mishra K.L.P &Chandrasekharan N., "Theory of Computer Science", PHI.
2. Hopcroft John E. And Ullman Jeffrey D., "Introduction to Automata Theory, Languages & Computation", 3rd Edition, Narosa, 2008.
3. Lewis H. R. and Papadimitrou C. H, "Elements of the theory of Computation", PHI.

Course code: CA419
Course title: ANALYSIS OF ALGORITHMS
Pre-requisite(s):
Co- requisite(s):
Credits: 3 L:3 T:0 P: 0
Class schedule per week: 03
Class: MCA
Semester / Level: II/4
Branch: MCA

Course Objectives

This course enables the students to:

1.	Understand different notions of asymptotic complexity
2.	Find the time and space complexity of an algorithm
3.	Understand various algorithm design techniques like greedy, divide and conquer, and dynamic programming.
4.	Understand NP completeness.
5.	Implement, analyze, and compare algorithms.

Course Outcomes

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

CO1	Determine the asymptotic complexity of an algorithm including the solving of recurrence relations
CO2	Explain various algorithm design techniques like divide and conquer, greedy, dynamic programming, backtracking, and branch & bound.
CO3	Apply various algorithm design techniques to solve algorithmic problems.
CO4	Compare various algorithms of a given problem.
CO5	Design efficient algorithm of a given problem using deterministic or non-deterministic approach.

SYLLABUS

MODULE: I

Elementary Algorithmic: Introduction, Problems and instances, The efficiency of algorithms, Average and worst-case analyses, What is an elementary operation, why look for efficiency.
Asymptotic Notation: Introduction, A notation for "the order of", Other asymptotic notation, Conditional asymptotic notation, Conditional asymptotic notation, Asymptotic notation with several parameters, Operations on asymptotic notation.
Analysis of Algorithm: Introduction, Analyzing control structures, Using a barometer, Supplementary examples, Average-case analysis, Amortized analysis, Solving recurrences.

(8L)

MODULE: II

Greedy Algorithms: General characteristics of greedy algorithms; The knapsack problem; Minimum spanning trees: Prim, Kruskal algorithms, implementation issues and complexity analysis; Shortest path problem; scheduling.

(8L)

MODULE: III

Divide-and-conquer: Introduction; Large integer multiplication; Binary search; Merge Sort, Quick Sort; Finding the median and selection problem; Matrix Multiplication; Exponentiation.

(8L)

MODULE: IV

Dynamic Programming: Introduction and basic concepts; Calculation the binomial coefficient; The World Series; Making change;0-1 knapsack problem; All Pair Shortest paths and Transitive Closure; Chained matrix multiplication; Travelling Salesman Problem.

8L)

MODULE: V

Exploring Graphs: Traversing trees, Depth-first search, Breadth-first search; Backtracking; Branch-and-bound.

An Introduction to NP Completeness: Introduction, Concept of P and NP; Polynomial Reduction; NP Completeness Proof of some problems.

(8L)

Text Book:

1. Brassard G. & Bratley P., "Fundamentals of Algorithms", New Delhi, 2005.

Reference Books:

1. E.Horowitz. et.al., "Fundamentals of Computer Algorithms", Galgotia Publication Pvt. Ltd.,New Delhi, 2004.
2. Cormen T.H., Leiserson Charles E.,Rivest Ronald, Stein Clifford "Introduction to Algorithms" 3rd Edition, PHI, New Delhi, 2005.
3. Dasgupta S., Papadimitriou C.H., Vaziran U.V, " Algorithm" 3rd Edition , TMH, New Delhi, 2007.

PROGRAM ELECTIVE I

Course Code: CA414
Course Title: DATA COMMUNICATION AND COMPUTER NETWORK LAB
Pre-requisite(s):
Co- requisite(s):
Credits:1.5 L: 0 T: 0 P:3
Class schedule per week: 3
Class: MCA
Semester / Level: II/4
Branch: MCA

Course Objectives

This course enables the students to:

1.	To familiarize the student in introducing and exploring various Network topologies and networking protocols
2.	To understand the use of client/server architecture in application
3.	To enable the student on how to approach for networking problems using networking simulation tools.
4.	To Design reliable servers using both TCP and UDP sockets
5.	Familiar with network tools and network programming.

Course Outcomes

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

CO1	Express programming & simulation for networking problems.
CO2	Get a thorough understanding of various aspects of networking devices
CO3	Design and implement simulation of a simple LAN and a WAN that meet a specific set of criteria
CO4	Identify the elements of a communication network
CO5	Simulate various OSI layer protocols using C/C++/ Java

SYLLABUS

List of Programs as Assignments:

1. **Lab Assignment No: 1**

   Q1. To familiarize with the Lab Network Topology, Locating different interfaces, routers and switches. Studying different pools of IP addresses.

   Q2. Implement the data link layer framing methods such as character, character stuffing, and bit stuffing.

Q3. To learn and observe the usage of different networking commands e.g.PING, TRACEROUTE. Learning remote login using telnet session. Measuringtypical average delays between different locations of the network.

2. **Lab Assignment No: 2**

Q1. What is the IP of the machine you are using? Compare it with the IP of your neighbors. Are the IPs of your neighbors same? Why or Why not?

Q2. Ping" is a tool used to determine if a server is responding and to estimate the round trip time of a message sent to that server. Use the ping command for the following URLs and record the success or failure statistics along with the average round trip time.
a) google.com
b) facebook.com
c) bitmesra.ac.in

Q3. Trace the route that is taken when you try to access:
a) google.com b) facebook.com c) bitmesra.ac.in

Q4. Network Commands on Linux / Unix

3. **Lab Assignment No: 3**

Q1. Implement on a data set of characters the three CRC polynomials – CRC 12, CRC 16 and CRC 32.

Q2. Implementation of Sub-netting and Super-netting.

Q3. To study different types of transmission media, various topologies, and configure modem of computer HUB and Switches.

4. **Lab Assignment No: 4**

Q1. Write a C/C++ program to determine if the IP address is in Class A, B, C, D, or E.

Q2. Write a C/C++ program to determine if the IP address is in Class A, B, or C.

Q3. Write a C/C++ program to translate dotted decimal IP address into 32 bit address.

Q4. To implement a routing protocol and check its connectivity in a variable length subnet masked network

Q5. Write a C/C++ program to perform bit stuffing and de-stuffing.

5. **Lab Assignment No: 5**

Q1. Implement Dijkstra's algorithm to compute the Shortest path through a graph.

Q2. Take an example subnet graph with weights indicating delay between nodes. Now obtain Routing table art each node using distance vector routing algorithm

Q3. Take an example subnet of hosts. Obtain broadcast tree for it.

6. **Lab Assignment No: 6**
   Q1. Build implementations of the Internet protocols
   Q2. Implementation of Stop and Wait Protocol and Sliding Window Protocol.
   Q3. Write a code simulating ARP /RARP protocols.

7. **Lab Assignment No: 7**
   Q1. Create a socket for HTTP for web page upload and download
   Q2. Write a code simulating PING and TRACEROUTE commands.
8. **Lab Assignment No: 8**
   Q1. Study and implement model for Socket Programming and Client – Server model.
   Q2. Experiments with NS2(or any other simulator) to study behavior (especially performance of) link layer protocols such as Ethernet and 802.11 wireless LAN..
9. **Lab Assignment No: 9**
   Q1. Experimental study of application protocols such as HTTP, FTP,SMTP, using network packet sniffers and analyzers such as **Wireshark**. Small exercises in socket programming in C/C++/Java..

10. **Lab Assignment No: 10**

    Q1. Take a 64 bit playing text and encrypt the same using DES algorithm.

    Q2. Write a program to break the above DES coding

    Q3. Using RSA algorithm encrypts a text data and Decrypt the samebjective: To Understand and Implement Data Interpolation

11. **Lab Assignment No: 11**
    Q1. Applications using TCP and UDP Sockets like d. DNS e. SNMP f. File Transfer
    Q2. Study of Network simulator (NS).and Simulation of Congestion Control Algorithms using NS
    Q3. Echo client and echo server b. Chat c. File Transfer

Books recommended:
TEXT BOOKS

1. William Stallings, Data and Computer Communication, Prentice Hall of India.
2. Behrouz A. Forouzan, Data Communication and Networking, McGraw-Hill.
3. Andrew S. Tanenbaum, Computer Networks, Prentice Hall.

REFERENCE BOOKS

1. W. Richard Stevens, TCP/IP Illustrated, Volume 1, Addison-Wesley
2. Douglas Comer, Internetworking with TCP/IP, Volume 1, Prentice Hall of India.

Course code: CA416
Course title: SOFTWARE ENGINEERING LAB
Pre-requisite(s):
Co- requisite(s):
Credits:1.5 L: 0 T: 0 P:3
Class schedule per week: 3
Class: MCA
Semester / Level: II/4
Branch: MCA

Course Objectives

This course enables the students:

1.	To understand the concept of UML
2.	To gain knowledge of various diagrams.
3.	Learn about software requirement specification.
4.	To gain knowledge about software design specification.
5.	To learn about the relationships among different UML diagrams.

Course Outcomes

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

CO1	Identify the software requirement capturing process.
CO2	Elaborate knowledge about dynamic view of system.
CO3	Analyze about static view of software system.
CO4	Analysis the relationship among static and dynamic view of system.
CO5	Identify the process of deployment of software system.

SYLLABUS

List of Programs as Assignments:

1. **Lab Assignment No: 1**

Objective: To Understand and Implement Identification of Requirements from Problem Statements

Q1. To consider the problem statement for a project to be developed and list out the ambiguities, inconsistencies and incompleteness of the problem statement.

Q2. To identify different functionaries to be obtained from a system and characteristics that a system should have, but not possessed by the system itself

2. **Lab Assignment No: 2**

Objective: To Understand and Implement Estimation ofProject Metrics

Q1. To estimate the minimum size of the team one would require to develop a project through application of intermediate COCOMO.

Q2. To use Halstead's metrics to estimate the effort required to recreate a program in JAVA from C.

3. **Lab Assignment No: 3**

Objective: To Understand and Implement Modeling UML Use Case Diagrams and Capturing Use Case Scenarios

Q1. To draw a use case diagram for the given case study.

Q2. To identify the primary and secondary actors for the system and generalization of use cases and «include» stereotypes to prevent redundancy in the coding phase.

4. **Lab Assignment No: 4**

Objective: To Understand and Implement E-R Modeling from the Problem Statements

Q1. To identify the possible entity sets, their attributes, and relationships for the given case study.

Q2. To draw an ER diagram for the given case study.

5. **Lab Assignment No: 5**

Objective: To Understand and Implement Identification of Domain Classes from the Problem Statements

Q1. To identify potential classes and their attributes for the given case study.

Q2. To utilize expert knowledge on the subject matter to identify other relevant classes.

6. **Lab Assignment No: 6**

Objective: To Understand and Implement Identification of Components from the Problem Statements

Q1. To identify potential components for the given case study.

Q2. To draw component diagram for the given case study

7. **Lab Assignment No: 7**

Objective: To Understand and Implement State Chart and Activity Modeling

Q1. To draw a statechart diagram to graphically represent the given case study.

Q2. To draw an activity diagram to graphically represent the workflow of the given case study.

8. **Lab Assignment No: 8**

Objective: To Understand and Implement Modeling UML Class Diagrams and Sequence diagrams

Q1. To draw class diagram for the given case study.

Q2. To draw sequence diagram for the given case study.

9. **Lab Assignment No: 9**

Objective: To Understand and Implement Modeling Data Flow Diagrams

Q1. To draw data flow diagram (Level 0, 1 and 2) for the given case study.

10. **Lab Assignment No: 10**

Objective: To Understand and Implement Estimation of Test Coverage Metrics and Structural Complexity

Q1. To identify the basic blocks for a given program

Q2. To draw a CFG using the basic blocks

Q3. To determine McCabe's complexity from a CFG.

11. Lab Assignment No: 11

Objective: To Understand and Implement Designing Test Suites

Q1. To design a test suite for the given case study.

Q2. To verify implementation of functional requirements by writing test cases.

Q3. To analyze results of testing to ascertain the current state of the project.

12. Lab Assignment No: 12

Objective: To Understand and Implement Forward and Reverse Engineering

Q1. To obtain programs from UML diagrams.

Q2. To obtain UML diagrams from programs.

Books recommended:

TEXT BOOKS

1. Software Engineering, Ian Sommerville, Pearson, 10th Edition, 2016.**(T1)**
2. Software Engineering: A Practioner's Approach, Roger S. Pressman, McGraw Hills, 7th Edition, 2009.**(T2)**

REFERENCE BOOKS

1. Fundamentals of Software Engineering, Rajib Mall, Prentice-Hall of India, 3rd Edition, 2009.**(R1)**

Course code: CA422
Course title: IT TOOLS AND TECHNIQUES LAB
Pre-requisite(s):
Co- requisite(s):
Credits: 1.5 L: 0 T: 0 P:3
Class schedule per week: 3
Class: MCA
Semester / Level: IV/5
Branch: MCA
Course Objectives
This course enables the students

1.	To introduce the intricacies of working with large data.
2.	Present programming tricks and tools for handling large data.
3.	Highlight different specialized functions for working with large data
4.	Understand the challenges of working with unstructured data
5	Understand how to automate tasks related to large volumes of data

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

CO1	code language constructs specifically related to working with large datasets.
CO2	Recognize patterns in data amenable for application of specialized data handling functions
CO3	Scrape websites to collect data for data science purposes.
CO4	Design algorithms for different kinds of data like structured, unstructured, dates etc.
CO5	Automate data handling tasks.

SYLLABUS

Module 1: Working with large data – Introduction, Lists, List comprehensions, tuples, working with dictionaries, nested dictionaries, working with collections

Module 2: Functions, lambda functions, specialized functions e.g. map, filter, reduce, variable number of parameters, keyword arguments.

Module 3: Specialized string handling, working with files, regular expressions, working with data file formats e.g. csv, json, excel etc.

Module 4: Web scraping, collecting data from the internet, Sending emails,

Module 5: Working with dates and times, scheduling tasks, automatically launching programs

TEXT BOOKS

1. Introduction to Computation and Programming using Python, with Application to Computational Modeling and Understanding Data 3rd Edition, J. V. Guttag, MIT Press, 2021
2. Python Crash Course, A Hands on, Project based Introduction to Programming, 2nd Edition, Eric Matthes, No Starch Press, 2018

REFERENCE BOOKS

1. Fluent Python, Luciano Ramalho, Oreilly Press, 1st Edition, 2016
2. Data science from Scratch: First Principles with Python, Joel Grus, OReilly Press, 2015

SEMESTER III

Course code: CA511

Course title: BASICS OF MACHINE LEARNING

Pre-requisite(s):

Co- requisite(s): None

Credits: 3 L: 3 T: 0 P: 0

Class schedule per week: 03

Class: MCA

Semester / Level: III/5

Branch: MCA

Course Objectives

This course enables the students:

1.	To formulate machine learning problems corresponding to different applications.
2.	To understand various supervised, semi-supervised and unsupervised machine learning algorithms.
3.	To familiarize various machine learning software libraries and data sets publicly available.
4.	To develop machine learning based system for various real-world problems.
5.	To assess how the choice of a machine learning algorithm impacts the accuracy of a system.

Course Outcomes

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

CO1	Formulate machine learning problems corresponding to different applications: data, model selection, model complexity
CO2	Demonstrate understanding of a range of machine learning algorithms along with their strengths and weaknesses
CO3	Implement machine learning solutions to classification, regression, and clustering problems
CO4	Design and implement various machine learning algorithms in a range of real-world applications
CO5	Evaluate and analyse the performance of a machine learning algorithm or a system based on machine learning algorithm.

SYLLABUS

Module I

Introduction to Machine Learning

Machine Learning – what and why? Supervised learning and unsupervised learning. Basics of Linear Algebra - matrices and vectors, Eigen value decomposition, principal component analysis. **(8L)**

Module II

Supervised Learning

Linear Regression with one variable, cost function, gradient descent for linear regression. Linear regression with multiple variables, normal equation, gradient descent. Logistic regression, cost function, gradient descent. Regularization - the problem of overfitting, regularization in linear regression and logistic regression. **(8L)**

Module III

Dimensionality reduction- Principal components. Decision Tree, Overfitting and Pruning, Support Vector Machine and Kernel; Noise, bias-variance trade-off, under-fitting and over-fitting concepts. **(8L)**

Module IV

Neural Networks representations, forward propagation, multi class classification. neural networks cost function, backpropagation algorithm. Regularization and bias/ variance. Recurrent networks. **(8L)**

Module V

Unsupervised and Semi Supervised Learning

Clustering - K-means partitional clustering, choosing the number of clusters. Hierarchical Agglomerative Clustering. Expectation maximization (EM) for soft clustering. Semi-supervised learning with EM using labeled and unlabeled data. Brief introduction to ML applications in computer vision, speech and natural language processing, etc.

(8L)

Text Books:

1. Mitchell Tom, “Machine Learning”, Latest Edition, Mc-Graw Hill.

Reference Books:

1. Shwartz Shai Shalev, and David Shai Ben, “Understanding Machine Learning”, Cambridge University Press, 2017.
2. Bishop Christopher “Pattern Recognition and Machine Learning”, Springer, 2006.
3. A Course in Machine Learning by Hal Daumé III (freely available online)

Course code: CA513
Course title: COMPILER DESIGN
Pre-requisite(s): Automata theory
Co- requisite(s):
Credits: 4 L:3 T:1 P: 0
Class schedule per week: 04
Class: MCA
Semester / Level: III/5
Branch: MCA
Course Objectives
This course enables the students to:

1.	Understand the need of compiler
2.	Provide a thorough understanding of design, working, and implementation of programming languages
3.	Trace the major concept areas of language translation and compiler design
4.	Create an awareness of the function and complexity of modern compilers.
5.	Develop knowledge for developing tool for natural language processing

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

CO1	Understand the need of compiler for *interfaci*ng between users and machine
CO2	Perceive the role of several phases of compilation process
CO3	Trace the major concept areas of language translation and compiler design
CO4	Develop a comprehensive Compiler for a given language
CO5	Apply knowledge for developing tool for natural language processing

SYLLABUS

MODULE -I

Introduction to Compiling: Translators, Interpreters, Compiler, other language processors, Phases of a compiler, Passes of compiler, Back-end and Front-end of compiler, Basic idea on Symbol Table, Issues in Compiler construction, Concept on *l*-value and r-value, Programming Language basics, Compiler construction tools.

Lexical and Syntax Analysis: *Lexical analysis*: Role of a Lexical analyser, Input buffering, Specification and recognition of tokens, State-machine driven lexical analysers and their implementations, Lexical analyser generator tool: LEX/FLEX. **(8L)**

MODULE -II
Syntax analysis: Need and Role of Parser, Importance of Context Free Grammars in designing Parser, Parse trees, derivations and sentential forms, Ambiguity.
Top down parsing: Backtracking, Recursive descent and Predictive parsers (LL), Error-detection in LL parser

Bottom-up parsing: Simple Shift-Reduce parsing, LR Parsers: SLR, CLR and LALR parsers, Error detection in S-R parsing, Handling ambiguous grammar, Parser generator tool: YACC/BISON **(8L)**

MODULE -III

Syntax Directed Translation: Syntax directed definitions, Construction of syntax tree, Attribute grammars, Inherited and synthesized attributes, Dependency graphs, Evaluation orders of attributes, S-Attributed definitions, L-attributed definitions.

Intermediate code generation: Variants of Syntax Trees, Three-address codes of different constructs, Translation of expressions, Type checking: Rules for type checking, Type conversion; **(8L)**

MODULE -IV

Machine independent code optimization: Sources of optimization, DAG, Peephole optimization and Basic Blocks, Loops in Flow Graphs, Data flow analysis and equations **(8L)**

MODULE -V

Runtime Environment and Code Generation:

Runtime environment: Storage organization: Static and Dynamic, Stack allocation and Heap allocation of memory;

Code generation: Issues in designing of a code generator, Register allocation and Assignment, Target machine (assembly code for 80- series) **(8L)**

Text Book:

1. Aho A.V., Sheth R. I. and Ullman J.D. "Compilers Principles Techniques and Tools", Pearson Education.

Reference Books:

1. Levine John R., Mason Tony, Brown Doug "Lex &Yacc", O'reilly.
2. Appel Andrew N., "Modern Compiler Implementation in C", Cambridge University Press.
3. Cooper & Linda "Engineering a Compiler", Elsevier theory.

Course code: CA515
Course title: SOFT COMPUTING
Pre-requisite(s):
Co- requisite(s):
Credits: 3 L: 3 T: 0 P: 0
Class schedule per week: 03
Class: MCA
Semester / Level: III/5
Branch: MCA

Course Objective:
This course enables the students:

1.	To know the basic functions of different AI branches.
2.	To understand the functionalities of neural networks .
3.	To know the application of fuzzy logic.
4.	To understand the basic functionalities of optimizations through soft computing.
5.	To find the basic functions of soft computing.

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

CO1	Solve numerical on Fuzzy sets and Fuzzy Reasoning.
CO2	Develop Fuzzy Inference System (FIS).
CO3	Solve problems on Genetic Algorithms
CO4	Explain concepts of neural networks
CO5	Develop neural networks models for various applications.

SYLLABUS

MODULE – I

Introduction to Artificial Intelligence System, Neural Network, Fuzzy Logic & Genetic Algorithm. Fuzzy Set Theory: Fuzzy Versus Crisp, Crisp Set, Fuzzy Set, Crip Relation, Fuzzy Relations. **(8L)**

MODULE – II

Fuzzy System: Crisp Logic, Predicate Logic, Fuzzy Logic, Fuzzy Rule Based System, Defuzzification Methods, and Applications. **(8L)**

MODULE – III

Genetic Algorithms, Basic Concepts, Creation Of Offspring, Working Principle, Encoding, Fitness Function, Reproduction.

Genetic Modeling, Inheritance Operations, Cross Over, Inversion And Deletion, Mutation Operator, Bit Wise Operators, Generation Cycle, Convergence Of Genetic Algorithm, Application, Multi-Level Optimization, Real Life Problems, Difference And Similarities Between GA And Other Traditional Methods, Advanced In GA**.** **(8L)**

MODULE – IV

Fundamentals Of Neural Networks, Basic Concepts Of Neural Network, Human Brain, Model Of An Artificial Neuron, Neural Network Architectures, Characteristic Of Neural Networks, Learning Method, Taxonomy Of Neural Network Architectures, History Of Neural Network Research, Early Neural Network Architectures, Some Application Domains.

(8L)

MODULE – V

Back Propagation Network Architecture Of Back Propagation Network, Back Propagation Learning, Illustration, Applications, Effect Of Tuning Parameters Of The Back Propagation Neural Network, Selection Of Various Parameters In BPN, Variations Of Standard Back Propagation Algorithm.

Associative Memory And Adaptive Resonance Theory, Autocorrelations, Hetrocorrelators , Multiple Training Encoding Strategy, Exponential BAM, Associative Memory For Real Coded Pattern Pairs, Applications, Introduction To Adaptive Resonance Theory, ARTI, Character Recognition Using ARI1 **(8L)**

Text Book:

1. Rajasekharan S. &Vijayalakshmi G. A. "Neural Network Fuzzy Logic and Gentic Algorithm Synthesis and Applications", Prentice Hall of India PLT, Pai, 2004.

Reference Book:

1. Jang JyhShing R, Sun C. T., Mizutani E. "Neuro Fuzzy and Soft Computing –A Computational Approach to Learning and Machine Intelligence", Prentice Hall of India, 1997.

PROGRAM ELECTIVE II

PROGRAM ELECTIVE III

COURSE INFORMATION SHEET

Course code: CA512
Course title: **BASICS OF MACHINE LEARNING LAB**
Pre-requisite(s):
Co- requisite(s):
Credits: **1.5** L: 0 T: 0 P: 3
Class schedule per week: 3
Class: MCA
Semester / Level: 3
Branch: MCA

Course Objectives:

This course enables the students:

A.	To learn how data is to be handled using appropriate data structures.
B.	Understand the basics of implementing attribute set reduction.
C.	Implement linear regression and compute relevant statistics.
D.	Learn the basics of classification using commonly used algorithms
E	Learn the basics of clustering using commonly used algorithms

Course Outcomes:

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

1.	The student should be able to manipulate large datasets and perform data preprocessing.
2.	Perform attribute reduction using Principal Component Analysis.
3.	Implement common techniques for performing Linear Regression and measure its performance.
4.	Implement standard techniques for classifying data and measure performance.
5	Implement standard techniques for clustering data and measure performance.

Syllabus:

1. Explore a dataset using Pandas. Compute attribute statistics, correlation, covariance, and other inferential statistics.
2. Perform PCA on a dataset to reduce attributes. Compare performance with available PCA modules in python.
3. Perform Linear regression on a dataset and compute the relevant parameters. Compute the error in the interpolation. Compare your results with implementations in Standard modules.
4. Classify a dataset with binary class distribution using logistic regression. Compute the values for accuracy, precision and recall and present the confusion matrix. Perform all necessary data preprocessing of the attributes involved.

5. Use logistic regression to perform OvA classification on a multi label dataset.
6. Classify a dataset using the K-NN algorithm. Perform a grid search to decide on the optimal value of K. Report the statistics of the results obtained.
7. Use the K-NN algorithm developed in Question 6 to perform K -fold cross validation. Compare your results with the basic implementation of the algorithm.
8. Perform binary K-Means on a dataset and compare its performance with the basic implementation of the K-Means.
9. Classify a dataset using the Naïve Bayes algorithm. Extend your algorithm to incorporate numerical attributes.
10. Cluster a dataset using the K-Means algorithm. Compute a suitable value of K using the grid search mechanism. Report the performance metrics of your algorithm e.g. homogeneity score, silhouette coefficient etc.
11. Write a program in python to cluster a dataset using the DBSCAN algorithm. The inputs to the algorithm would include epsilon and "p".
12. Perform agglomerative clustering of a dataset in python. The number of clusters would be an input to the algorithm and your algorithm should provide options to choose the distance metrics e.g. distance between centers, distance between nearest neighbours, distance between farthest points etc.
13. Write a program to train a single hidden layer neural network to classify a binary dataset.
14. Write a program use Information Gain to decide the splitting attribute for a dataset to be used in a decision tree classifier.
15. Use the function written in Question 14 to classify a dataset using a Decision tree.

Text Books:

1. Geron A., "Hands on Machine Learning with Scikit Learn and Tensorflow", 2nd edition, O' Reilly Press, 2020
2. Muller A. C., Guido S., "Hands Machine Learning with Python", O'Reilly Press, 2016

Reference Books:

1. Coelho L. P., Richert W., "Building Machine Learning Systems in Python", O' Reilly Press, 2nd edition, 2016

Gaps in the syllabus (to meet Industry/Profession requirements):N/A

POs met through Gaps in the Syllabus:N/A

Topics beyond syllabus/Advanced topics/Design:N/A

POs met through Topics beyond syllabus/Advanced topics/Design:N/A

Course Outcome (CO) Attainment Assessment Tools and Evaluation Procedure

Direct Assessment

Assessment Tools	% Contribution during CO Assessment
Continuous Internal Assessment	50
Semester End Examination	50

Continuous Internal Assessment	% Distribution
Mid semester examination	25
Two quizzes	20 (2×10)
Teacher's Assessment	5

Assessment Components	CO1	CO2	CO3	CO4	CO5
Continuous Internal Assessment	√	√	√	√	√
Semester End Examination	√	√	√	√	√

Indirect Assessment

1. Student Feedback on Faculty
2. Student Feedback on Course

Course Delivery Methods

CD1	Lecture by use of boards/LCD projectors/OHP projectors
CD2	Assignments
CD3	Laboratory experiments/Teaching aids/Seminars
CD4	Mini Projects
CD5	Industrial visits/in-plant training
CD6	Self- learning such as use of NPTEL materials and internets
CD7	Simulation

Mapping of Course Outcomes onto Program Outcomes

Course Outcome	Program Outcomes (POs)												Program Specific Outcomes (PSOs)		
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
CO1	3	3	3	3	2	3	2	1	1	2	1	2	3	3	2
CO2	3	3	2	3	3	3	2	1	1	1	1	2	3	2	2
CO3	3	3	3	3	3	2	1	1	1	2	1	2	3	3	2
CO4	3	3	3	3	2	2	1	1	1	2	1	2	3	3	3
CO5	3	3	3	3	3	3	3	1	1	2	1	2	3	3	3

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

Course Outcomes	Course Delivery Method
CO1	CD1,CD6
CO2	CD1, CD6,CD7
CO3	CD1, CD2, CD3,CD6,CD7
CO4	CD1, CD3,CD6,CD7
CO5	CD1,CD2,CD3,CD4,CD5,CD7

Course code: CA514
Course title: COMPILER DESIGN LAB
Pre-requisite(s): Automata theory.
Co- requisite(s): Compiler Design Th.
Credits:1.5 L: 0 T: 0 P:3
Class schedule per week: 3
Class: MCA
Semester / Level: II/4
Branch: MCA

Course Objectives

This course enables the students to:

1.	Perceive the role of several phases of compilation process
2.	Trace the major concept areas of language translation and compiler design
3.	Create an awareness of the function and complexity of modern compilers

Course Outcomes

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

CO1	Design lexical and parsing phases for R.Es. and Grammars of any language using any programming language.
CO2	Use Lex and Yacc tools for designing compiler.
CO3	Implement various parsing, conversion, optimization and code generation algorithms for the design of a compiler
CO4	Develop a comprehensive Compiler for a given language
CO5	Apply knowledge for developing tool for natural language processing

SYLLABUS – SYLLABUS OF CA513 COMPILER DESIGN (THEORY)

LAB. ASSIGNMENTS

LEXICAL ANALYSIS

1. Write a Lex program, which: Given an input C-program, outputs a stream of tokens (other than blank, tab, \n) on the screen. Output should be of the form: <lineno>: <token_name> <OPTIONAL token_attribute> As discussed, there are FIVE categories of tokens:

a) *Keywords* (e.g. for, while, if, char, int etc.):

- Each keyword should be given a separate token_name.
- No token_attribute
- There are 32 keywords in C-language.

b) *Operators* (e.g. +, ++, +=, etc.)

- Each operator should be given a separate token_name.

- No token_attribute

c) *Punctuation marks* (e.g. {, }, (,) etc.)

- Each keyword should be given a separate token_name.
- No token_attribute

d) *Identifiers* (e.g. name of functions, variables etc.)

- Common token_name = IDENTIFIER
- token_attribute: actual identifier string

e) *Constants*

1: Whole numbers (positive only, as negative sign should be classified as an operator)

- Common token_name = NUMBER
- token_attribute: actual number

2: Real numbers (*e.g*. 12.34)

- Common token_name = REAL_NUMBER
- token_attribute: actual number

3: Exponential numbers (*e.g.* 12.34e+56.78)

- Common token_name = EXP_NUMBER
- token_attribute: actual number

4: Character constants (*e.g*. 'A')

- Common token_name = CHAR_CONSTANT
- token_attribute: actual character

5: String constants (*e.g.* "ABC")

- Common token_name = STRING_CONSTANT
- token_attribute: actual string

2. Write Lex programs for the followings.

i) Checking the number of *a*'s in words generated over {*a*, *b*} is divisible by 2 or not

ii) Counting number of *vowels* in a text.

iii) Counting number of *characters*, *words*, *lines* in a text

3. Write a Lex program for: Given an input C-program (argv[1]), remove all comments, and output a program (argv[2]) without comments. You have to handle all FIVE types of comments:

(1) // Single line comment
(2) /* Multi line comment */
(3) /* Nested-1:
// Single line comment within a Multi line comment
*/
(4) /* Nested-2:
/* Multi line comment within */

another Multi line comment
*/

Comment Does not work!!

- If you mean comments of the form /* . . . */ then the answer is no. Logically, everything from **/*** is ignored except the first occurrence of ***/.** If a /* is encountered inside of the comment then that is also ignored so the matching ***/** will prematurely terminate the outside comment.

Logically, it is one kind of pattern matching problem where the concept of extra memory (Stack) is necessary which is **not** allowed in FA.

- Comments of the form // . . . are terminated by the end of the line so an additional // within that comment will not affect the comment.

(5) // Nested-3: /* Multi line comment with a single line comment
flowing to next line */

Step-1: Figure out the behaviour first. (1), (2) are straight forward. You need to discover how (3)-(5) behave. It may(?) happen that some lines are not deleted!
Step-2: Give the output file, which removes the comments according to rule you discover in Step-1.

PARSING

4. *Build parsers (using YACC OR BISON) for the following languages.*

i) $L=\{a^n b^n | n \geq 1\}$ over {a, b}.

ii) L(G) where rule set of G is { $S \rightarrow aSb$, $S \rightarrow bSa$, $S \rightarrow c$ } over {a, b, c}

5. *Build parser (using YACC/BISON) for the following*:

***i*)** Converting Infix expression to Postfix expression.

***ii*)** Designing a simple CALCULATOR

***iii*)** Verifying *declaration statement* of C-language

***iv*)** Verifying *multiple assignment* statement in C-language

***v*)** Verifying *for loop* construct in C- language

6. Design expression grammar for C-language and write code to generate **ABSTRACT SYNTAX TREE USING YACC.**

7. Using Lex and YACC tools, write a program to convert **infix** expression like, **(a+5)*b+(c-6)/d** into **postfix** expression.

8. Construct a Symbol Table for all the identifiers appearing in the input C-program.

Program Electives I

Semester II

Course code: CA431
Course title: DISTRIBUTED DATABASE CONCEPTS
Pre-requisite(s):
Co- requisite(s):
Credits: 3 L:3 T:0 P: 0
Class schedule per week: 03
Class: MCA
Semester / Level: II/4
Branch: MCA

Course Objectives
This course enables the students:

1.	To understand the structure of databases distributed over the network.
2.	To learn Query processing and decomposition.
3.	To understand how to create a distributed database using fragmentation.
4.	To learn transaction processing in a distributed environment.
5.	To understand how concurrency control is performed in a distributed environment.

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

CO1	Explain detailed architecture of distributed database.
CO2	Design a distributed database for any environment using horizontal and vertical fragmentation.
CO3	Describe transaction execution, rules and protocols used in concurrent access in a distributed environment.
CO4	Perform Query Processing and its decomposition a distributed database.
CO5	Design a reliable database.

SYLLABUS

Module I:

Introduction: Distributed Data Processing, What is a Distributed Database System? Promises of DDBSs, Problem Areas.

(8L)

Module II:

Distributed DBMS Architecture: DBMS Standardization, Architectural Models for Distributed DBMSs, Distributed DBMS Architecture.

Distributed Database Design: Alternative Design Strategies, Distribution Design Issues, Fragmentation, Allocation.

(8L)

Module III:

Overview of Query Processing: Query Processing Problem, Objectives of Query Processing, Complexity of Relational Algebra Operations, Layers of Query Processing.

Query Decomposition and Optimization: Query Decomposition, Query Optimization, Centralized Query Optimization, Distributed Query Optimization Algorithms.

(8L)

Module IV:

Transaction Management and Concurrency Control: Definition of a Transaction, properties of Transactions, Serializability Theory, Taxonomy of Concurrency Control Mechanisms, Locking-based Concurrency Control Algorithms, Timestamp-based Concurrency Control Algorithms, Deadlock Management.

(8L)

Module V:

Distributed DBMS Reliability: Reliability Concepts and Measures, Failures and Fault Tolerance in Distributed Systems, Failures in Distributed DBMS, Local Reliability Protocols, Distributed Reliability Protocols.

(8L)

Text Books:

1. M. Tamer Ozsu, Patrick Valduriez, "Distributed Database Systems", 2nd Edition, Pearson, 2011.

Reference Books:

1. Elmasri Navathe, "Fundamental of Database Systems", 5th Edition, Pearson Education, 2008.
2. Thomas Connolly, Carolyn Begg, "Database Systems – A Practical Approach to Design, implementation and Management", 4th Edition, Pearson Education, 2008.
3. Silberschatz, Korth, Sudarshan, "Database System Concepts", 4th Edition, McGraw Hill, 2002.

Course code: CA433
Course title: INTRUSION DETECTION SYSTEM
Pre-requisite(s):
Co- requisite(s):
Credits: 3 L: 3 T: 0 P: 0
Class schedule per week: 03
Class: MCA
Semester / Level: II/4
Branch: MCA

Course Objectives

This course enables the students:

1.	To Understand Model of Intrusion Analysis
2.	To provide a brief description of security design principles.
3.	To evaluate physical solutions for preventing intrusion.
4.	To acquire knowledge on requirements of responses, types of responses and methodology of mapping responses policy.

Course Outcomes

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

CO1	Classify and Explain a network intrusion detection system.
CO2	Develop predictive measures to assess and prevent intrusion.
CO3	Assess implications of privacy, security and ethical issues as they pertain to organizations IT infrastructure.
CO4	Diagnosis possible hacks and purpose polices to outline what do when an intrusion occurs.
CO5	Integrate techniques to provide solutions for preventing intrusion.

SYLLABUS

Module I:

Defining Intrusion Detection, The state of threats against computers, networked systems-Overview of computer security solutions Audit: setting, firewalls, VPN's Overview of Intrusion Detection and Intrusion Prevention-Network and Host-based IDS.

(8L)

Module-II:

Classes of attacks - Network layer: scans, denial of service, penetration-Application layer: software exploits, code injection-Human layer: identity theft, root access-Classes of attackers-Kids/hackers/sop Hesitated groups-Automated: Drones, Worms, Viruses.

(8L)

Module III:

A General IDS model and taxonomy, Signature-based Solutions, Snort, Snort rules, Evaluation of IDS, Cost sensitive IDS.

Anomaly Detection Systems and Algorithms-Network Behavior Based Anomaly Detectors (rate based)-Host-based Anomaly Detectors-Software Vulnerabilities- State transition, Immunology, Payload Anomaly Detection**.** **(8L)**

Module IV:

Attack trees and Correlation of alerts-Autopsy of Worms and Botnets-Malware detection- Obfuscation, polymorphism-Document vectors.

(8L)

Module V:

Email/IM security issues-Viruses/Spam-From signatures to thumbprints to zero day, detection-Insider Threat issues-Taxonomy-Masquerade and Impersonation-Traitors, Decoys and Deception-Future: Collaborative Security.

(8L)

TEXT BOOKS:

1. The Art of Computer Virus Research and Defense, Peter Szor, Symantec Press ISBN 0-321-30545-3

2. Crimeware, Understanding New Attacks and Defenses, Markus Jakobsson and Zulfikar Ramzan, Symantec Press, ISBN: 978-0-321-50195-0 2008

3. Intrusion Detection by Rebecca Gurley Bace Macmillan Technical Publishing, 2000

REFERENCE BOOKS:

1. Intrusion Detection System by Robert D Petro Springer 2015

Course code: CA435
Course title: MODERN ARTIFICIAL INTELLIGENCE
Pre-requisite(s):
Co- requisite(s):
Credits: 3 L: 3 T: 0 P: 0
Class schedule per week: 03
Class: MCA
Semester / Level: II/4
Branch: MCA

Course Objectives

This course enables the students to:

1.	Understand the importance of AI based systems.
2.	UseAI based techniques in real world problems.
3.	Design an intelligent system, component or process to meet desired needs with constraints.
4.	Create artificial intelligence systems for multidisciplinary domains.
5.	Work collaborate to formulate and solve engineering problemsbased onAI principles.

Course Outcomes

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

CO1	Understand the principles and approaches of artificial intelligence and different aspects of Intelligent agent.
CO2	Apply different search techniques for solving real world complex problems and select the most appropriate solution by comparative evaluation.
CO3	Design AI based systems and their componentswith reasoning even in the presence of incomplete and/or uncertain information.
CO4	Develop knowledge-based systems with proper representation schemes.
CO5	Analyze the pros and cons of different AI systems and their design.

SYLLABUS

MODULE-I

Introduction: Overview of Artificial Intelligence- Problems of AI, AI Technique, Tic - Tac - Toe Problem.

Intelligent Agents: Agents & Environment, Nature Of Environment, Structure Of Agents, Goal Based Agents, Utility Based Agents, Learning Agents.

Problem Solving: Problems, Problem Space & Search: Defining The Problem As State Space Search, Production System, Problem Characteristics, Issues In The Design Of Search Programs. **(8L)**

MODULE-II

Search Techniques: Solving Problems by Searching, Problem Solving Agents, Searching For Solutions; Uniform Search Strategies: Breadth First Search, Depth First Search, Depth Limited Search, Bi-directional Search, Comparing Uniform Search Strategies.

Heuristic Search Strategies: Greedy Best-First Search, A* Search, Memory Bounded Heuristic Search: Local Search Algorithms & Optimization Problems: Hill Climbing Search, Simulated Annealing Search, Local Beam Search, Genetic Algorithms; Constraint Satisfaction Problems, Local Search For Constraint Satisfaction Problems.

Adversarial Search: Games, Optimal Decisions & Strategies in Games, The Mini Max Search Procedure, Alpha-Beta Pruning, Additional Refinements, Iterative Deepening.

(8L)

MODULE-III

Knowledge & Reasoning: Knowledge Representation Issues, Representation & Mapping, Approaches to Knowledge Representation, Issues in Knowledge Representation.

Using Predicate Logic: Representing Simple Fact in Logic, Representing Instant & ISA Relationship, Computable Functions & Predicates, Resolution, and Natural Deduction. **Representing Knowledge Using Rules:** Procedural Verses Declarative Knowledge, Logic Programming, Forward Verses Backward Reasoning, Matching, Control Knowledge

(8L)

MODULE-IV

Probabilistic Reasoning: Representing Knowledge in an Uncertain Domain, Bayesian Networks, Dempster-Shafer Theory.

Planning: Overview, Components of A Planning System, Goal Stack Planning, Hierarchical Planning.

Learning: Forms of Learning, Inductive Learning, Explanation Based Learning, Neural Net Learning & Genetic Learning. **(8L)**

MODULE-V

Natural Language Processing: Brief introduction to Syntactic Processing, Semantic Analysis, Discourse & Pragmatic Processing.

Robotics: Introduction, Robot hardware, robotic perception, planning to move, planning uncertain movements, robotic software architecture, application domains.

(8L)

Text books:

1. Russel S. and Norvig P. "Artificial Intelligence a Modern Approach", 3rd Edition, Pearson Education.
2. Rich E. & Knight K. "Artificial Intelligence", 2nd Edition, TMH, New Delhi.

Course code: CA437
Course title: INFORMATION RETRIEVAL
Pre-requisite(s):
Co- requisite(s):
Credits: 3 L: 3 T: 0 P: 0
Class schedule per week: 03
Class: MCA
Semester / Level: II/4
Branch: MCA

Course Objectives
This course enables the students to:

1.	To understand the basic component of information retrieval.
2.	To explore the application areas of information retrieval.
3.	To understand the idea of indexing and pre-processing of data.
4.	To explore the different IR evolution techniques.
5.	To understand the concepts of Query Expansion techniques.

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

CO1	Explain the working of a search engine and details of the individual components.
CO2	Apply efficient techniques for the indexing of documents
CO3	Implement various indexing, scoring, ranking and relevance feedback models and techniques for information retrieval
CO4	Develop a complete IR system from scratch
CO5	Evaluate and analyse the performance of a retrieval systems using a suitable test collection

SYLLABUS

Module I
Introduction
Introduction; Search Engine Architecture; An overview of crawling, text transformation, index creation, user interaction, ranking, link analysis, evaluation and deep web.

(8L)

Module II

Pre-processing and Indexing

Pre-processing: tokenization, stop word, normalization, stemming, wildcard queries, spelling correction – edit distance and k-gram; Indexing: Index construction; Index compression.

(8L)

Module III

Scoring

Parametric and zone indexes; term frequency and weighting; vector space model; efficient scoring and ranking; vector space scoring.

(8L)

Module IV

IR Evaluation

Evaluation; Standard test collection; Evaluation of unranked and ranked retrieval; Assessing relevance; System quality and user utility.

(8L)

Module V

Relevance Feedback and Query Expansion

Relevance feedback and pseudo relevance feedback; query reformulation.

(8L)

Text book:

Manning, Christopher D., Raghavan Prabhakar, and SchützeHinrich, "Introduction to Information Retrieval", Cambridge: Cambridge University Press, 2008.(T1)

Reference books:

Grossman David A., Frieder Ophir "Information Retrieval: Algorithms and Heuristics", Springer.(R1)

Croft Bruce, Metzler Donald, and Strohman Trevor "Search Engines: Information Retrieval in Practice", Pearson Education, 2009.(R2)

Ricardo Baeza-Yates and Neto Berthier Ribeiro "Modern Information Retrieval", 2nd Edition, Addison-Wesley, 2011.(R3)

Course code: CA439
Course title: IMAGE PROCESSING
Pre-requisite(s):
Co- requisite(s):
Credits: 3 L: 3 T: 0 P: 0
Class schedule per week: 03
Class: MCA
Semester / Level: II/4
Branch: MCA

Course Objective:
This course enables the students:

1.	Understand the fundamentals of digital image processing.
2.	Develop a Broad knowledge of Spatial and Frequency image transforms used for enhancing an image.
3.	Learn Image restoration techniques and noise models used for restoring an image.
4.	Understand Lossless and lossy image compression techniques.
5.	Know Morphological processing algorithms for various operations on an image.

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

CO1	Understand the concept of image formation, digitization and the role human visual system plays in perception of image data.
CO2	Acquire an appreciation for spatial and frequency based techniques for enhancing the appearance of an image duly applying them in different applications.
CO3	Discern the difference between noise models, gain an insight into assessing the degradation function and realize different spatial and frequency based filtering techniques for reduction and removal of noise.
CO4	Synthesize a solution to image compression using the concept of information theory and lossless and lossy compression techniques.
CO5	Design and create practical solutions using morphological operators for common image processing problems and assess the results.

SYLLABUS

MODULE -I
What Is Digital Image Processing, Fundamental Steps in Digital Image Processing , Components of an Image Processing System, Elements of Visual Perception, Light and the Electromagnetic Spectrum, Image Sensing and Acquisition, Image Sampling and Quantization, Some Basic Relationships Between Pixels, Linear and Nonlinear Operations.

(8L)

MODULE -II

Enhancements in Spatial Domain: Some Basic Gray Level Transformations, Histogram Processing, Enhancement Using Arithmetic/Logic Operations, Basics of Spatial Filtering, Smoothing Spatial Filters, Sharpening Spatial Filters, Combining Spatial Enhancement Methods.
Enhancements in Frequency Domain: Introduction to the Fourier Transform and the Frequency Domain, Smoothing Frequency-Domain Filters, Sharpening Frequency Domain Filters, Homomorphism Filtering **(8L)**

MODULE -III

Image Restoration: A Model of the Image Degradation/Restoration Process, Noise Models. Restoration in the Presence of Noise Only-Spatial Filtering, Periodic Noise Reduction by Frequency Domain Filtering, Linear, Position-Invariant Degradations, Estimating the Degradation Function, Inverse Filtering, Mean Square Error (Wiener) Filtering, Constrained Least Squares Filtering, Geometric Mean Filter, Geometric Transformations.

(8L)

MODULE -IV

Image Compression: Fundamentals, Image Compression Models, Elements of Information Theory, Error-Free Compression, Lossy Compression.

(8L)

MODULE -V

Morphological Image Processing and Segmentation: Preliminaries, Dilation and Erosion, Opening and Closing, The Hit-or-Miss Transformation. Some Basic Morphological Algorithms, Detection of Discontinuities, Edge Linking and Boundary Detection, Thresholding, Region-Based Segmentation. **(8L)**

Text books:

1. Rafael. C. & Woods Richard E. "Digital Image Processing", 3rd Edition, Pearson Education, New Delhi, 2009.

Reference books:

1. Pratt W.K. "Digital Image Processing", 4th Edition, John Wiley & sons Inc., 2006.
2. Sonka M., Hlavac Vaclav, Boyle Roger "Image Processing, Analysis and Machine Vision", 2nd Edition, Thomson Learning, India Edition, 2007.
3. Jayaraman "Digital Image Processing", Tata McGraw. Hill Education, 2011.

Course code: CA441
Course title: DATA MINING TECHNIQUES
Pre-requisite(s):
Co- requisite(s):
Credits: 3 L:3 T:0 P: 0
Class schedule per week: 03
Class: MCA
Semester / Level: II/4
Branch: MCA
Course Objectives

This course enables the students:

1.	Examine the types of the data to be mined and apply pre-processing methods on raw data.
2.	To introduce the basic concepts of Data Warehouse and Data Mining techniques
3.	Apply the techniques of clustering, classification, association finding, feature selection and visualization to real world data
4.	Prepare students for research in the area of data mining and related applications and Enhance students communication and problem solving skills
5.	Provide the students with practice on applying data mining solutions using common data mining software tool /programming languages.

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

CO1	Describe the fundamentals of data mining systems as well as issues related to access and retrieval of data at scale.
CO2	Explain the various data mining functionalities and data warehousing techniques.
CO3	pply the various data mining techniques to solve classification, clustering and association rule mining problems.
CO4	Analyze and choose among different approaches of a data mining task.
CO5	Design and evaluate data mining models to be used in solving real life problems, keeping in view social impacts of data mining.

SYLLABUS

MODULE – I
Data Mining: Introduction, Relational Databases, Data Warehouses, Transactional databases, Advanced database Systems and Application, Data Mining Functionalities, Classification of Data Mining Systems, Major Issues in Data Mining. Data Processing: Data Cleaning, Data Integration and Transformation, Data Reduction. **(8L)**

MODULE – II
Data Warehouse: Introduction, A Multidimensional data Model, Data Warehouse Architecture, Data Warehouse Implementation, Data Cube Technology, From Data Warehousing to Data Mining. Data Cube Computation and Data Generalization

(8L)

MODULE – III

Mining Association Rules in Large Databases: Association Rule Mining, Single – Dimensional Boolean Association Rules, Multilevel Association Rules from Transaction Databases, Multi Dimensional Association Rules from Relational Databases, From Association Mining to Correlation Analysis. **(8L)**

MODULE – IV

Classification and Prediction: Classification & Prediction, Issues Regarding Classification & Prediction, Classification by decision Tree Induction, Bayesian Classification, Classification by Back propagation, Classification based on concepts & Association Rule Analysis, Other Classification Methods, Prediction, Classification Accuracy.

(8L)

MODULE – V

Cluster Analysis: Introduction, Types of Data in Cluster Analysis, A Categorization of Major Clustering Methods, Partitioning Method - k- Medoids Algorithm, CLARANS, Hierarchical Methods - BIRCH, ROCK Density-Based Methods - DBSCAN, Outlier Analysis.

(8L)

Text books:

1. Jiawei Han & Micheline Kamber "Data Mining Concepts & Techniques", Publisher Harcourt India. Private Limited, 3nd Edition.

Reference books:

1. Gupta G.K. "Introduction to Data Mining with case Studies", PHI, New Delhi, 2006.
2. Berson A. & Smith S.J. "Data Warehousing Data Mining", COLAP, TMH, New Delhi, 2004.
3. Dunham H.M. & Sridhar S. "Data Mining", Pearson Education, New Delhi, 2006.

PROGRAM ELECTIVE - II

Course code: CA519
Course title: MOBILE COMPUTING
Pre-requisite(s):
Co-requisite(s):
Credits: 3 L:3 T:0 P:0
Class schedule per week: 03
Class: MCA
Semester / Level: III/5
Branch: MCA
Name of Teacher:

Course Objectives

This course enables the students:

1.	Understand basic mobile network concepts and its architectures.
2.	Know Protocols like mobile telephony and introduce to the concepts of blue tooth
3.	Comprehend the GSM architectures and its features that support mobile communications.
4.	Understand the network management and Middleware services used in Ip and Mobile telephony
5.	Get accustomed to the concepts like GPRS, 3G, 4G networks

Course Outcomes

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

CO1	Identify the role of cellular networks in Mobile and Pervasive Computing
CO2	Analyse about the basic architecture for a pervasive computing environment
CO3	Assess the principles for routing and allocating the resources on the 3G-4G wireless network
CO4	Evaluate mobile computing applications based on the paradigm of context aware computing
CO5	Design and develop applications in mobile and pervasive computing environment

SYLLABUS

Module –I

Introduction: Basics of mobile networks, middleware and gateways, application and services, Mobile Computing Architecture: architecture for mobile computing, three tier architecture.

(8L)

Module –II

Mobile Computing through Telephony: evolution of telephony, multiple access procedures, mobile computing through telephone. Emerging Technologies: introduction, Bluetooth, radio frequency identification, wireless broadband, mobile IP, IPV6.

(8L)

Module –III

Global System for Mobile Communications GSM: introduction, GSM architecture , call routing in GSM, GSM address and identifiers, network aspect in GSM, GSM frequency allocation, authenticity and security. Mobile computing over SMS.

(8L)

Module –IV

General Packet Radio Service GPRS:GPRS and packet data network, GPRS network architecture, GPRS network operation, data services in GPRS, applications for GPRS, limitations for GPRS, Wireless Application Protocol

Client Programming: introduction, moving beyond the desktop, a peak under the hood: hardware overview, mobile phone, PDA, design constraints in application for handheld devices.

(8L)

Module –V

CDMA and 3G, VoIP, call routing, voice over IP applications, IP multimedia subsystem, Mobile VoIP. **(8L)**

Text Book:

1. Talukedar Ashok, Ahmed Hasan, YavagalRoopa R "Mobile Computing Technology, Applications and Service Creation",Tata McGraw -Hill Education ,2010.

Reference Books:

1. Schiller Jochen H. "Mobile Communications", 2nd Edition, Addison wesley.
2. Kamal Raj "MobileComputing", 2nd Edition, Oxford University Press.
3. Behravanfar Reza "Mobile Computing Principles: Designing and Developing Mobile Applications with UML and XML", ISBN: 0521817331, Cambridge University Press, October 2004.
4. Adelstein Frank, Gupta Sandeep K.S., Richard III Golden G., Schwiebert, Loren, "Fundamentals of Mobile and Pervasive Computing", ISBN: 0071412379, McGraw-Hill Professional, 2005.
5. Hansmann Uwe, MerkLothar, Nicklous Martin S., Stober Thomas "Principles of Mobile Computing", 2nd Edition., Springer, 2003.

Course code: CA521
Course title: CYBER SECURITY
Pre-requisite(s):
Co-requisite(s):
Credits:3 L:3 T:0 P:0
Class schedule per week: 03
Class: MCA
Semester / Level:III/5
Branch: MCA

Course Objectives
This course enables the students:

1.	Provides an in-depth study of the rapidly changing and fascinating field of computer forensics.
2.	Combines both the technical expertise and the knowledge required to investigate, detect and prevent digital crimes
3.	Knowledge on digital forensics legislations, digital crime, forensics processes and procedures, data acquisition and validation, e-discovery tools.
4.	E-evidence collection and preservation, investigating operating systems and file systems, network forensics, art of steganography and mobile device forensics.

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

CO1	Understand relevant legislation and codes of ethics.
CO2	Apply Computer forensics and digital detective and various processes, policies and procedures.
CO3	Understand E-discovery, guidelines and standards, E-evidence, tools and environment.
CO4	Learn the techniques ofEmail and web forensics and network forensics tools.
CO5	Integrate techniques to recover data from computer and hand held devices.

SYLLABUS

Module – I

Introduction to Cybercrime, Classifications of Cyber Crimes, Local and Global perspectives on Cybercrime, Cyber offences, Cyberstalking, Cyber crime and cloud computing, cyber crimes through hand held devices**.** **(8L)**

Module-II

Cyber Security Vulnerabilities and Cyber Security Safeguards , Cyber Security Vulnerabilities- Overview, vulnerabilities in software, System administration, Complex Network Architectures, Open Access to Organizational Data, Weak Authentication, Unprotected Broadband communications, Poor Cyber Security Awareness. Cyber Security Safeguards- Overview, Access control, Audit, Authentication, Biometrics, Cryptography, Deception, Denial of Service Filters, Ethical Hacking, Firewalls, Intrusion Detection Systems, Response, Scanning, Security policy, Threat Management **.** **(8L)**

Module- III

Securing Web Application, Services and Servers

Introduction, Basic security for HTTP Applications and Services, Basic Security for SOAP Services, Identity Management and Web Services, Authorization Patterns, Security Considerations, Challenges.

(8L)

Module- IV

Intrusion Detection and Prevention

Intrusion, Physical Theft, Abuse of Privileges, Unauthorized Access by Outsider, Malware infection, Intrusion detection and Prevention Techniques, Anti-Malware software, Network based Intrusion detection Systems, Network based Intrusion Prevention Systems, Host based Intrusion prevention Systems, Security Information Management, Network Session Analysis, System Integrity Validation.

(8L)

Module-V

Cyberspace and the Law

Introduction: Cyber Security Regulations, Roles of International Law, the state and Private Sector in Cyberspace, Cyber Security Standards. The INDIAN Cyberspace, National Cyber Security Policy 2013. **(8L)**

TEXT BOOKS:

1. Cyber Security Understanding Cyber Crimes, Computer Forensics and Legal Perspectives by Nina Godbole and SunitBelpure, Publication WileyIndian Print 2014.
2. Anti-Hacker Tool Kit (Indian Edition) by Mike Shema, Publication Mc Graw Hill

Course code: CA523
Course title: CLOUD COMPUTING
Pre-requisite(s):
Co-requisite(s):
Credits:3 L:3 T:0 P:0
Class schedule per week: 03
Class: MCA
Semester / Level: III/5
Branch: MCA

Course Objectives
This course enables the students to:

1.	Understand about security requirements in cloud.
2.	Learn about infrastructure security at different levels
3.	Know about management standards of cloud security
4.	Develop and Apply trust-based security model to different layers

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

CO1	Identify security aspects of each cloud model
CO2	Implement a public cloud instance using a public cloud service provider
CO3	Apply trust-based security model to different layer
CO4	Develop a risk-management strategy for moving to the Cloud
CO5	Identify various research domain of cloud computing

SYLLABUS

Module I
Introduction: Essentials, Benefits and need for Cloud Computing - Business and IT Perspective - Cloud and Virtualization - Cloud Services Requirements - Cloud and Dynamic Infrastructure - Cloud Computing Characteristics Cloud Adoption. (8L)

Module II
Principles of Parallel and Distributed Computing: Eras of computing, Parallel vs. Distributed computing, Elements of parallel computing, Elements of distributed computing, Technologies for distributed computing.
(8L)

Module III
Virtualization: Introduction, Characteristics of virtualized environments, Taxonomy of virtualization techniques, Virtualization and cloud computing, Pros and cons of virtualization, Technology examples.
Storage virtualization:Storage Area Networks - Network-Attached storage - Cloud Server Virtualization - Virtualized Data Centre. (8L)

Module IV
Cloud computing architecture: Introduction, Cloud reference model, Types of clouds, Economics of the cloud, Open challenges. (8L)

Module V

Cloud platforms in industry and Cloud applications : Amazon web services, Google app engine, Microsoft azure, Observations, Scientific applications, Scientific, Business and Consumer applications.

(8L)

Text Book:

Buyya Raj Kumar, Vecchiola Christian &Thamarai S. Selvi, "Mastering Cloud Computing", McGraw Hill Publication, New Delhi, 2013.(T1)

Reference Books:

Velte T., Velte A. and Elsenpeter R., "Cloud Computing: A Practical Approach", McGraw Hill, India.(R1)

Buyya R., Broberg J., "Cloud Computing: Principles and Paradigms", Wiley.(R2)

Hwang K., Fox G. and Dongarra J., "Distributed and Cloud Computing, From Parallel Processing to the Internet of Things", Morgan Kaufmann, 2012.(R3)

Course code: CA525
Course title: DEEP LEARNING
Pre-requisite(s):
Co- requisite(s):
Credits:3 L:3 T:0 P:0
Class schedule per week: 03
Class: MCA
Semester / Level:III/5
Branch: MCA
Course Objectives
This course enables the students:

1.	To understand the basic component of Machine Learning.
2.	To explore the application areas of Neural Networks.
3.	To understand the idea of Recurrent Neural Networks.
4.	To explore the basic concepts of Feed forward Neural Networks.
5.	To understand the concepts of mathematical modelling.

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

CO1	Able to differentiate between machine learning and deep learning
CO2	Identify problems suitable for application of deep learning.
CO3	Illustrate the working of FF Neural Networks and their modifications.
CO4	Apply Convolutional & Recurrent Neural Networks to solve problems
CO5	Analyse the efficiency of deep learning systems.

Syllabus

Module I

Introduction and Basics of Machine Learning

Beginnings of ANN, XOR Problem, From Cognitive Science to Deep Learning, NNs and their importance. Elementary classification problem, evaluating classification results, Simple Classifier – Naïve Bayesian Classifier, Simple NN: Logistic Regression, Learning without Labels, Learning alternative representation of data – PCA.

(8L)

Module II

Feed forward Neural Networks:

Basic concept and terminology, Representing networks, Perceptron rule, Delta rule, From logistic regression to Backpropagation, Backpropagation, Complete Feedforward NNs.

(8L)

Module III

Modifications & Extensions of FF Neural Nets

Regularization, L1 & L2 regularization, Learning Rate, Momentum and Dropout, Stochastic Gradient Descent and Online Learning, Problems with multiple hidden layers, Vanishing and exploding gradients.

(8L)

Module IV

Convolution & Recurrent Neural Networks

Introduction, Feature maps and Pooling, Building a complete convolutional neural network. Recurrent Neural Networks – Sequences of unequal length, Settings for learning with recurrent neural networks, Adding feedback loops and Unfolding neural networks, Elman Networks, LSTM

(8L)

Module V

Auto encoders

Learning Representations, Different Autoencoder Architectures, Stacking Autoencoders.

(8L)

Text book:

1. Skansi S., Introduction to Deep Learning - From Logical Calculus to Artificial Intelligence, 1st Edition, Springer International Publishing, 2018.

Reference book:

1. Buduma N., Fundamentals of Deep Learning, 1st Edition, O Reilly Media, 2016.

Course code: CA527
Course title: COMPUTER VISION
Pre-requisite(s):
Co- requisite(s):
Credits:3 L:3 T:0 P:0
Class schedule per week: 03
Class: MCA
Semester / Level: III/5
Branch: MCA
Course Objectives
This course enables the students:

1.	Be familiar with both the theoretical and practical aspects of computing with images.
2.	Have described the foundation of image formation, measurement, and analysis.
3.	Understand the geometric relationships between 2D images and the 3D world.
4.	Grasp the principles of state-of-the-art deep neural networks

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

CO1	Developed the practical skills necessary to build computer vision applications.
CO2	To have gained exposure to object and scene recognition and categorization from images.
CO3	Develop algorithm for classification and clustering.
CO4	Illustrate the techniques of feature extraction and analysis.
CO5	Apply in different engineering application such activity recognition, computational photography, biometrics.

SYLLABUS

Module I

Overview, computer imaging systems, lenses, Image formation and sensing,Image analysis, pre-processing and Binary image analysis.

(8L)

Module II

Edge detection, Edge detection performance, Hough transform, corner detection.

(8L)

Module III

Segmentation, Morphological filtering, Fourier transformsFeature extraction, shape, histogram, color, spectral, texture, using CVIPtools,Feature analysis, feature vectors, distance /similarity measures, data preprocessing.

(8L)

Module IV

Pattern Analysis:

Clustering: K-Means, K-Medoids, Mixture of GaussiansClassification: Discriminate Function, Supervised, Un-supervised, Semi supervisedClassifiers: Bayes, KNN, ANN models; Dimensionality Reduction: PCA, LDA,ICA, and Non-parametric methods.

(8L)

Module V

Recent trends inActivity Recognition, computational photography, Biometrics.

(8L)

Text Books:

1. Szeliski,R.," Computer Vision: Algorithms and Applications,"Springer,2011.
2. Goodfellow, Bengio, and Courville,"Deep Learning," First Edison.MIT Press,2016.
3. Fisher,R.B.,Breckon,T. P. , Dawson-Howe,K,Fitzgibbon,A , Robertson,C., Trucco,E. , Williams,C. K. I., "Dictionary of Computer Vision and Image Processing,"Second Edison,Wiley,2014.

Reference Book:

1. Forsyth,D.A.,Ponce,J.,"Computer Vision A Modern Approach," Second Edison, Pearson Education,2015.

Course code: CA529
Course title: NETWORK SECURITY AND CRYPTOGRAPHY
Pre-requisite(s):
Co- requisite(s):
Credits:3 L:3 T:0 P:0
Class schedule per week: 03
Class: MCA
Semester / Level: III/5
Branch: MCA

Course Objectives
This course enables the students:

1.	To understand the foundations of cryptographic attacks.
2.	To gain knowledge of encrypting data, and to choose between different algorithms.
3.	Prepare students for research in the area of cryptography and enhance students communication and problem solving skills
4.	To differentiate between the encryption techniques and know their suitability to an application.
5.	To effectively apply their knowledge to the construction of secure cryptosystems.

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

CO1	Understand the various types of cryptographic attacks and the mathematics behind cryptography.
CO2	Describe the various types of ciphers and hash functions.
CO3	Apply the different cryptographic techniques to solve real life problems.
CO4	Evaluate different techniques as to their suitability to various applications.
CO5	Develop a cryptosystem keeping in view social issues and societal impacts.

SYLLABUS

Module I
Foundations – Protocol Building Blocks - Basic Protocols - Intermediate Protocols - Advanced Protocols - Zero-Knowledge Proofs - Zero-Knowledge Proofs of Identity -Blind Signatures - Identity-Based Public-Key Cryptography.

(8L)

Module II
Key Length - Key Management – Public Key Cryptography versus Symmetric Cryptography - Encrypting Communications Channels - Encrypting Data for Storage - Hardware Encryption versus Software Encryption - Compression, Encoding, and Encryption - Detecting Encryption – Hiding and Destroying Information**.** **(8L)**

Module III

Information Theory - Complexity Theory - Number Theory - Factoring - Prime Number Generation - Discrete Logarithms in a Finite Field - Data Encryption Standard (DES) – Lucifer - Madryga - NewDES - GOST – 3 Way – Crab – RC5 - Double Encryption - Triple Encryption - CDMF Key Shortening - Whitening.

(8L)

Module IV

Pseudo-Random-Sequence Generators and Stream Ciphers – RC4 - SEAL - Feedback with Carry Shift Registers - Stream Ciphers Using FCSRs - Nonlinear-Feedback Shift Registers - System-Theoretic Approach to Stream-Cipher Design - Complexity-Theoretic Approach to Stream-Cipher Design - N- Hash - MD4 - MD5 - MD2 - Secure Hash Algorithm (SHA) - OneWay Hash Functions Using Symmetric Block Algorithms - Using Public-Key Algorithms - Message Authentication Codes

(8L)

Module V

RSA - Pohlig-Hellman - McEliece - Elliptic Curve Cryptosystems -Digital Signature Algorithm (DSA) - Gost Digital Signature Algorithm - Discrete Logarithm Signature Schemes - Ongchnorr-Shamir -Cellular Automata - Feige-Fiat-Shamir -Guillou-Quisquater - Diffie-Hellman - Station-to-Station Protocol -Shamir's Three-Pass Protocol - IBM Secret-Key Management Protocol - MITRENET - Kerberos - IBM Common Cryptographic Architecture.

(8L)

Text Books:

1. Schneier Bruce, "Applied Cryptography: Protocols, Algorithms, and Source Code in C", 2nd Edition, John Wiley & Sons, Inc, 1996.
2. Mao Wenbo, "Modern Cryptography Theory and Practice", Pearson Education, 2004.
3. KahateAtul, "Cryptography and Network Security", Tata McGrew Hill, 2003.

Reference Book:

1. Stallings William, "Cryptography & Network Security Principles and Practice", Pearson Education.

PROGRAM ELECTIVE - III

Course code: CA539
Course title: PARALLEL COMPUTING
Pre-requisite(s):
Co- requisite(s):
Credits: 3 L:3 T:0 P:0
Class schedule per week: 03
Class: MCA
Semester / Level:III/5
Branch: MCA
Course Objectives

This course enables the students to:

1.	Learn different types of parallelisms achieved over different computer models
2.	Write parallel algorithms (and programs) for computer problems
3.	Map parallel algorithms from architecture to architecture
4.	Identify the issues in concurrency control

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

CO1	Analyze the need of concurrent execution of problems
CO2	Summarize the issues of concurrency control
CO3	Relate the parallel algorithm from organization to organization
CO4	Measure a range of parallel algorithms on different architectures.
CO5	Apply the concept parallelism in solving the problems of different domains

SYLLABUS

Module I
Introduction: Parallel Processing Environment- Pipelining and Data Parallelism, Flynn's Taxonomy, Speedup, Scaled Speedup, Analysing parallel algorithms, P-RAM Algorithms.

(8L)

Module II
Processor Array, MIMD: Multiprocessors (shared) and Multi-computers (distributed), Networks(Processor organizations):Static and dynamic Interconnection Networks, Message Transferring procedures.

(8L)

Module III
Mapping and Scheduling, Dynamic Load Balancing on Multi-computers, Static Scheduling on UMA Multiprocessors, Parallel Programming model using process and thread, Deadlock and Synchronization issues.

(8L)

Module IV

Elementary Parallel Algorithm: Matrix Multiplication: Sequential Matrix Multiplication, Algorithms for Processor Array, Algorithms for Multiprocessors, Algorithms for Multi-computers.

(8L)

Module V

Solving set of linear equations: Gaussian Elimination, The Jacobi Algorithm, Finding roots of non-linear equations, Sorting algorithms: Enumeration Sort, ODD-EVEN Transposition sort, BITONIC Merge, Quicksort Based Algorithms, Graph Algorithms.

(8L)

Text books:

1. Quin M. J., Parallel Computing: Theory and Practice, McGraw Hill, New York, 1994.

Reference books:

1. Akl Selim G., The Design and Analysis of Parallel algorithms, Prentice Hall International.
2. Sasikumar M., Shikhare D. and Prakash P. Ravi, Introduction to Parallel Processing, PHI, 2006.

Course code: CA541
Course title: DIGITAL FORENSIC
Pre-requisite(s):
Co- requisite(s):
Credits:3 L:3 T:0 P:0
Class schedule per week: 03
Class: MCA
Semester / Level: III/5
Branch: MCA
Course Objectives
This course enables the students:

1.	Provides an in-depth study of the rapidly changing and fascinating field of computer forensics.
2.	Combines both the technical expertise and the knowledge required to investigate, detect and prevent digital crimes
3.	Knowledge on digital forensics legislations, digital crime, forensics processes and procedures, data acquisition and validation, e-discovery tools.
4.	E-evidence collection and preservation, investigating operating systems and file systems, network forensics, art of steganography and mobile device forensics.

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

CO1	Understand relevant legislation and codes of ethics.
CO2	Apply Computer forensics and digital detective and various processes, policies and procedures.
CO3	To understand the basic digital forensics and techniques for conducting the forensic examination on different digital devices.
CO4	Evaluate the techniques ofEmail and web forensics and network forensics tools.
CO5	examine digital evidences such as the data acquisition, identification analysis

SYLLABUS

Module – I

Computer forensics fundamentals, Benefits of forensics, computer crimes, computer forensics evidence and courts, legal concerns and private issues.

(8L)

Module-II

Understanding Computing Investigations – Procedure for corporate High-Tech investigations, understanding data recovery work station and software, conducting and investigations.

(8L)

Module- III

Data acquisition- understanding storage formats and digital evidence, determining the best acquisition method, acquisition tools, validating data acquisitions, performing RAID data acquisitions, remote network acquisition tools, other forensics acquisitions tools. **(8L)**

Module- IV

Case studies Discuss the various court orders etc., methods tosearch and seizure electronic evidence, retrieved and un-retrieved communications, Discuss the importance of understanding what court documents would be required for a criminal investigation., specific tools and techniques, Forensic auditing.

(8L)

Module-V

Current computer forensics tools- software, hardware tools, validating and testing forensic software, addressing data-hiding techniques, performing remote acquisitions, E-Mail investigations- investigating email crime and violations, understanding E-Mail servers, specialized E-Mail forensics tool.

(8L)

TEXT BOOKS:

1. Warren G. Kruse II and Jay G. Heiser, "Computer Forensics: Incident Response Essentials", Addison Wesley, 2002.
2. Nelson, B, Phillips, A, Enfinger, F, Stuart, C., "Guide to Computer Forensics and Investigations, 2nd ed., Thomson Course Technology, 2006, ISBN: 0-619-21706-5.

Course code: CA543
Course title: INTERNET OF THINGS (IoT)
Pre-requisite(s):
Co-requisite(s):
Credits: 3 L:3 T:0 P:0
Class schedule per week: 03
Class: MCA
Semester / Level: III/5
Branch: MCA

Course Objectives
This course enables the students to:

1.	Understand the basic concept and the IoT Paradigm
2.	Know the state of art architecture for IoT applications
3.	Learn the available protocols used for IoT
4.	Design basic IoT Applications.
5.	Evaluate optimal IoT applications.

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

CO1	Identify the IoT Components and its capabilities
CO2	Explain the architectural view of IoT under real world constraints
CO3	Analyse the different Network and link layer protocols
CO4	Evaluate and choose among the transport layer protocols
CO5	Design an IoT application

SYLLABUS

Module I
IoT - An Architectural Overview
Building an architecture, Main design principles and needed capabilities, An IoT architecture outline, standards considerations. M2M and IoT Technology Fundamentals- Devices and gateways, Local and wide area networking, Data management, Everything as a Service(XaaS), M2M and IoT Analytics, Knowledge Management.

(8L)

Module II
IoT Architecture - State of the Art
Introduction, State of the art, Reference Model and architecture, IoT reference Model - IoT Reference Architecture **(8L)**

Module III
IoT Data Link Layer & Network Layer Protocols
PHY/MAC Layer (3GPP MTC, IEEE 802.11, IEEE 802.15), WirelessHART,BluetoothLow Energy, Zigbee Smart Energy
Network Layer-IPv4, IPv6, 6LoWPAN

(8L)

Module IV
Transport & Session Layer Protocols
Transport Layer (TCP, MPTCP, UDP,)
Session Layer-HTTP, CoAP, XMPP, AMQP, MQTT. **(8L)**

Module V
Layer Protocols & Security
Service Layer -oneM2M, ETSI M2M, security in IoT and M2M applications **(8L)**

Text Books:

1. Holler Jan, TsiatsisVlasios, Mulligan Catherine, Avesand Stefan, Karnouskos Stamatis, Boyle David, "From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence", 1st Edition, Academic Press, 2014.
2. Waher Peter, "Learning Internet of Things", PACKT publishing, BIRMINGHAM-MUMBAI

Reference Books:

1. Reiter Bernd Scholz, Michahelles Florian, "Architecting the Internet of Things", Springer, ISBN 978-3-642-19156-5 e-ISBN 978-3-642-19157-2.
2. Minoli Daniel, "Building the Internet of Things with IPv6 and MIPv6:".

Course code: CA545
Course title: NATURAL LANGUAGE PROCESSING
Pre-requisite(s):
Co- requisite(s):
Credits: 3 L: 3 T: 0 P: 0
Class schedule per week: 03
Class: MCA
Semester / Level: III/5
Branch: MCA

Course Objectives

This course enables the students:

1.	To understand the algorithms available for the processing of linguistic information and computational properties of natural languages.
2.	To conceive basic knowledge on various morphological, syntactic and semantic NLP tasks.
3.	To familiarize various NLP software libraries and data sets publicly available.
4.	To develop systems for various NLP problems with moderate complexity.
5.	To learn various strategies for NLP system evaluation and error analysis.

Course Outcomes

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

CO1	Describe the concepts of morphology, syntax, semantics, discourse & pragmatics of natural language.
CO2	Demonstrate understanding of the relationship between NLP and statistics & machine learning.
CO3	Discover various linguistic and statistical features relevant to the basic NLP task, namely, spelling correction, morphological analysis, parts-of-speech tagging, parsing and semantic analysis.
CO4	Develop systems for various NLP problems with moderate complexity.
CO5	Evaluate NLP systems, identify shortcomings and suggest solutions for these shortcomings.

SYLLABUS

MODULE-I

Introduction to NLP

NLP – introduction and applications, NLP phases, Difficulty of NLP including ambiguity; Spelling error and Noisy Channel Model; Concepts of Parts-of-speech and Formal Grammar of English.

(8L)

MODULE-II

Language Modelling: N-gram and Neural Language Models

Language Modelling with N-gram, Simple N-gram models, Smoothing (basic techniques), Evaluating language models; Neural Network basics, Training; Neural Language Model, Case study: application of neural language model in NLP system development

(8L)

MODULE-III

Parts-of-speech Tagging

Parts-of-speech Tagging: basic concepts; Tagset; Early approaches: Rule based and TBL; POS tagging using HMM, Introduction to POS Tagging using Neural Model.

(8L)

MODULE-IV

Parsing

Basic concepts: top down and bottom up parsing, treebank; Syntactic parsing: CKY parsing; Statistical Parsing basics: Probabilistic Context Free Grammar (PCFG); Probabilistic CKY Parsing of PCFGs.

(8L)

MODULE-V

Semantics

Vector Semantics; Words and Vector; Measuring Similarity; Semantics with dense vectors; SVD and Latent Semantic Analysis; Embeddings from prediction: Skip-gram and CBOW; Concept of Word Sense; Introduction to WordNet

. **(8L)**

Text books:

1. Jurafsky Dan and Martin James H. "Speech and Language Processing" ,3rd Edition, 2018.

Reference books:

1. Jurafsky D. and Martin J. H., "Speech and language processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition", 2nd Edition, Upper Saddle River, NJ: Prentice-Hall, 2008.
2. Goldberg Yoav "A Primer on Neural Network Models for Natural Language Processing".

Course code: CA547
Course title: BIG DATA ANALYTICS
Pre-requisite(s):
Co- requisite(s):
Credits:3 L:3 T:0 P:0
Class schedule per week: 03
Class: MCA
Semester / Level: III/5
Branch: MCA

Course Objectives
This course enables the students:

1.	To provide an overview of approaches facilitating data analytics on huge datasets in different domain.
2.	To provide the knowledge on NoSQL and different partitioning method to handle large datasets.
3.	To provide an overview of Apache Hadoop and HDFS Concepts and Interfacing with HDFS
4.	To understand Map Reduce Jobs in Hadoop framework
5.	To provide the knowledge of various Hadoop based tool for processing large datasets.

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

CO1	Describe big data and use cases from selected business domains
CO2	Explain NoSQL big data management
CO3	Install, configure, and run Hadoop and HDFS
CO4	Perform map-reduce analytics using Hadoop
CO5	Use Hadoop related tools such as HBase, Cassandra, Pig, and Hive for big data analytics

SYLLABUS

Module I
Introduction
What is big data, why big data, convergence of key trends, unstructured data, industry examples of big data, web analytics, big data and marketing, fraud and big data, risk and big data, credit risk management, big data and algorithmic trading, big data and healthcare, big data in medicine, advertising and big data, big data technologies, introduction to Hadoop, open source technologies, cloud and big data, mobile business intelligence, Crowd sourcing analytics, inter and trans firewall analytics.

(8L)

Module II

Introduction to NoSQL, aggregate data models, aggregates, key-value and document data models, relationships, graph databases, schemaless databases, materialized views, distribution models, sharding, master-slave replication, peer peer replication, sharding and replication, consistency, relaxing consistency, version stamps, map-reduce, partitioning and combining, composing map-reduce calculations.

(8L)

Module III

Data format, analyzing data with Hadoop, scaling out, Hadoop streaming, Hadoop pipes, design of Hadoop distributed file system (HDFS), HDFS concepts, Java interface, data flow, Hadoop I/O, data integrity, compression, serialization, Avro, file-based data structures.

(8L)

Module IV

MapReduce workflows, unit tests with MRUnit, test data and local tests, anatomy of MapReduce job run, classic Map-reduce, YARN, failures in classic Map-reduce and YARN, job scheduling, shuffle and sort, task execution, MapReduce types, input formats, output formats.

(8L)

Module V

Hbase, data model and implementations, Hbase clients, Hbase examples, praxis.Cassandra, Cassandra data model, Cassandra examples, Cassandra clients, Hadoop integration, Pig, Grunt, pig data model, Pig Latin, developing and testing Pig Latin scripts. Hive, data types and file formats, HiveQL data definition, HiveQL data manipulation, HiveQL queries.

(8L)

Text Books:

1. Michael Minelli, Michelle Chambers, and AmbigaDhiraj, "Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses", Wiley, 2013.
2. P. J. Sadalage and M. Fowler, "NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence", Addison-Wesley Professional, 2012.
3. Tom White, "Hadoop: The Definitive Guide", Third Edition, O'Reilley, 2012.

Reference Books:

1. Sammer ,E., "Hadoop Operations," O'Reilley, 2012
2. Capriolo ,E., Wampler ,D., and Rutherglen ,J., "Programming Hive," O'Reilley, 2012
3. George ,L., "HBase: The Definitive Guide," O'Reilley, 2011
4. Gates ,A., "Programming Pig," O'Reilley, 2011

Course code: CA549
Course title: BLOCK CHAIN TECHNOLOGY
Pre-requisite(s):
Co- requisite(s):
Credits:3 L:3 T:0 P:0
Class schedule per week: 03
Class: MCA
Semester / Level: III/5
Branch: MCA

Course Objectives
This course enables the students:

1.	To provide an overview of the different blockchain technologies.
2.	To provide the knowledge on the need of blockchain and its applicability in real world problem.
3.	To provide the knowledge of cryptocurrency design and its security against scam ,fraud, hacking.
4.	To provide the ability to design and implement new ways of using blockchain for applications other than cryptocurrency.
5.	To be able to apply the knowledge gained through the course in actual blockchain development or blockchain contract developer

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

CO1	Learn and explain the difference between centralized, decentralized network and blockchain.
CO2	Explain fundamental concepts of blockchain using hashes and consensus.
CO3	Understand the concept of mining in blockchains.
CO4	Understand the working of Bitcoin and its security.
CO5	Know about the different platforms for implementing blockchain and its varied application.

SYLLABUS

Module I
Introduction to Blockchain Technology

Introduction to Blockchain, Trusted Third party for transactions, Difference between centralized, decentralized and distributed peer to peer networks, Types of Blockchain (Permission Blockchain vs. Permissionless Blockchain), History of Bitcoins.

(8L)

Module II

Fundamental concepts of Blockchain

Concepts of Block, Transactions, Hashes, Consensus. Hashes: Hash cryptography, Encryption vs. hashing, Transactions: Recording transactions, Digital Signature, Verifying and confirming transactions, Blocks and blockchain: Hash pointers, Blocks, Consensus building. Distributed consensus, Byzantine generals problem, Consensus mechanism: POW, POS, POB, POA, etc. Blockchain Architecture, Markle Root Tree.

(8L)

Module III

Mining and simulating blockchain

Mining and simulating blockchain: Game theory behind competitive mining. Incentives: mining and transaction fees, Energy expended in mining.

(8L)

Module IV

Bitcoin ad Security

Bitcoin: Bitcoin creation, exchanges. Wallets, security. Protecting blockchain from attackers. Forks – soft and hard, Blockchain security, Key Management in Bitcoin, Case studies.

(8L)

Module V

Platforms and Applications

Introduction to Blockchain platform: Ethereum, Hyperledger, IOTA, EOS, Multichain, Bigchain, CORDA, SOLIDITY, Designing a new blockchain, Distributed Application (DAPP). Applications: E-Governance, Elections, File sharing, Micropayments
Challenges and Research Issues in blockchain.

(8L)

Text Books:

1. Bitcoin and Cryptocurrency technologies: a comprehensive introduction. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, and Steven Goldfeder. Princeton University Press, First edition,2016
2. Blockchain Applications: A Hands-On Approach. ArshdeepBahga, Vijay Madisetti. VPT Publisher. First edition,2018.
3. Blockchain: Step – by – Step Guide to Understand by Paul Laurence, Createspace Independent Pub.

Reference Books:

1. Introducing Ethereum and Solidity Foundations of Cryptocurrency and Blockchain Programming for Beginners by Chris Dannen, Apress
2. Blockchain: The comprehensive beginner's guide by Frank Walrtin

Web References:

3. https://bitcoin.org/bitcoin.pdf
4. https://blockchain.mit.edu/how-blockchain-works