छत्रपति शाहू जी महाराज विश्वविद्यालय, कानपुर

CHHATRAPATI SHAHU JI MAHRAJ UNIVERSITY, KANPUR

(पूर्ववर्ती कानपुर विश्वविद्यालय कानपुर)

Formerly Kanpur University, Kanpur – 208024

A Documentary Support

For

Metric No. – 1.1.1

Programme Outcomes & Course Outcomes

Under the

Criteria - I

(Curriculum Design and Development)

Key Indicator - 1.1

In

Metric No. – 1.1.1

M.Sc. Computer Science

Co-ordinator
Internal Quality Assurance Cell
CSJM University, Kanpur

(Registrar)
C.S.J.M.University
Kanpur

REGISTRAR
C.S.J.M. UNIVERSITY
KANPUR

CHHATRAPATI SHAHU JI MAHARAJ UNIVERSITY
KANPUR

SYLLABUS

(M.Sc.)

COMPUTER SCIENCE

UNIVERSITY INSTITUTE OF ENGINEERING & TECHNOLOGY

SCHOOL OF ENGINEERING & TECHNOLOGY

UNIVERSITY INSTITUTE OF ENGINEERING & TECHNOLOGY

SCHOOL OF ENGINEERING & TECHNOLOGY

Vision

To achieve excellence in engineering education, empower students to be technically competent professionals and entrepreneurs with strong ethical values so as to significantly contribute as agents for universal development and societal transformation

Mission

To provide affordable quality education at par with global standards of academia and serve society with harmonious social diversity

To encourage new ideas and inculcate an entrepreneurial attitude amongst the students, and provide a robust research ecosystem

To practice and encourage high standards of professional ethics and accountability among students

Master in Computer Science

Program Outcomes (POs)

PO1	**Computer science knowledge**: Acquire strong fundamental knowledge of computer science and engineering along with mathematics.
PO2	**Programming languages and tools**: Possess programming skills in different contemporary programming languages and use different development tools. Be able to select the appropriate tool/programing language/platform and understand the limitations of the same while implementing the solution.
PO3	**Problem analysis**: Ability to identify, formulate & analyse requirements of a problem to provide sustainable solution which are in coherence with the local/regional/national or global needs and feasibility
PO4	**Handle complex problems**: Design solution for complex problems, which incorporate components and processes, which are sustainable and reusable.
PO5	**Teamwork**: Posses Flexibility to adapt to a team environment. To be able to work as an individual or as a member or a team leader in multidisciplinary team organizations.
PO6	**Ethics**: To understand contemporary legal, social & ethical issues in computing.
PO7	**Presentation and Communication**: To be able to present and communicate precisely and effectively. Be able to comprehend and write effective reports and design documents and presentations professionally and be able to perceive and give clear instructions.
PO8	**Life-long learning**: To have passion for acquiring technical advancements in the field of computer science and engineering and apply new technology for solving local/regional/national or global problems
PO9	**Social responsibilities**: To apply skills for social causes and work towards sustainable solutions

🟩 Local Green 🟨 Regional Yellow 🟦 National Blue ⬛ Global Grey

Program Specific Outcomes (PSOs)

PSO-1	To be able to understand problem, think of best suitable approach to solve the problem, develop and evaluate effective solutions as per the local/ regional/ national/ global requirements and availability of resources/ technologies.
PSO-2	To be able excel in contemporary technologies being adopted by the industry and academia for providing sustainable solutions
PSO-3	To be able to excel in various programming/ project competitions and technological challenges laid by professional bodies

Program Educational Outcomes (PEOs)

PEO-1	To make the students ready for successful career leading to higher education/ industry and to apply expertise in solving global problems.
PEO-2	To empower students achieve personal and professional success with awareness and commitment to their ethical and social responsibilities, both as individuals and in team environments.
PEO-3	To encourage students maintain and improve their technical competence through lifelong learning.

🟩 Local Green 🟨 Regional Yellow 🟦 National Blue ◼ Global Grey

Curricular Components

Semester-I

Sl.NO.	Paper Code	Paper Title	Internal Assessment	External Assessment	Total Marks
1	MCS -101	Unix and C Programming	30	70	100
2	MCS -102	Computer Organization	30	70	100
3	MCS -103	Professional Communication	30	70	100
4	MCS -104	Data Base Management System	30	70	100
5	MCS -105	Lab Unix and C Programming	15	35	50
6	MCS -106	Lab Data Base Management System	15	35	50

Total: 500

Semester-II

Sl.NO.	Paper Code	Paper Title	Internal Assessment	External Assessment	Total Marks
1	MCS -201	Operating System	30	70	100
2	MCS -202	Data Structures	30	70	100
3	MCS -203	Computer Network	30	70	100
4	MCS -204	Web Technology	30	70	100
5	MCS -205	Lab Web Technology	15	35	50
6	MCS -206	Lab Data Structure	15	35	50

Total: 500

Semester-III

Sl.NO.	Paper Code	Paper Title	Internal Assessment	External Assessment	Total Marks
1	MCS -301	Object Oriented Programming using JAVA	30	70	100
2	MCS -302	Theory of Computing and Compiler Design	30	70	100
3	MCS -303	Design and Analysis of Algorithms	30	70	100
4	MCS -304	Elective-I	30	70	100
5	MCS -305	Lab Object Oriented Programming using JAVA	15	35	50
6	MCS -306	Mini Project	15	35	50

Total: 500

Semester-IV

Sl.NO.	Paper Code	Paper Title	Internal Assessment	External Assessment	Total Marks
1	MCS -401	Computer Graphics	30	70	100
2	MCS -402	Artificial Intelligence	30	70	100
3	MCS -403	Software Engineering	30	70	100
4	MCS -404	Elective-II	30	70	100
5	MCS -405	Major Project	15	35	100

Total: 500

🟩 Local Green 🟨 Regional Yellow 🟦 National Blue ⬛ Global Grey

Detailed Syllabus

Course Code: MCS – S101 **Breakup: 3 – 0 – 3**
Course Name: Programming & Computing(C & UNIX)

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

CO1	Recollect various programming constructs and to develop C programs
CO2	Understand the fundamentals of C programming
CO3	Choose the right data representation formats based on the requirements of the problem
CO4	Implement different Operations on arrays, functions, pointers, structures, unions and files

Unit – I

The Free Software Movement, Open source definition, Open source business strategy, Problem Solving and its tools, Flow chart, Pseudo code, Modular programming. Fundamentals of Unix Operating System, Login & Password, Different Commands, Unix directory, Structure and working with directories, Vi-editor, Basic Structure and execution of C programs, Constants, Variables, and Data Types and various type of declarations, Different type operators and Expressions, Evaluation of Expressions, Operator Precedence and Associability, Mathematical Functions.

Unit–II

Managing Input and Output operations, Decision Making and Branching, Decision Making and Looping. One – dimensional Arrays and their declaration and Initializations, Two-dimensional Arrays and their initializations, Multidimensional Arrays, String Variables, Reading and Writing Strings, Arithmetic Operations on characters, Putting Strings together, Comparison of Two Strings, String – handling functions.

Unit –III

Need and Elements for user –defined Functions, Definition of Functions, Return values and their types, Function calls and Declaration, Arguments and corresponding return values, Functions that return multiple values, Nesting of functions, Recursion, Passing arrays and strings to functions, The Scope, Visibility and Life time of variables.

Unit –IV

Defining Structure, Declaring Structure Variable and Accessing Structure Members, Initialization of Structure, Comparing Structure Variables, Operation on Individual Members, Arrays of Structures, Structures within structures, Structures and Functions, Unions, Size of Structures, Bit Fields.

Unit – V

Pointers to Functions, Pointers and Structures, File Management in C. use of fopen(), fclose(), Formatted file I/O, Searching through files using fseek(), ftell(), rewind().

Text Books and References:

1. Programming in C, Schaum Series, 3rd edition, BPB Publication, Byron S. Gottfried
2. The ‘C’ Programming, Denis Ritchi, Second edition, PHI, 1988
3. Mastering C, Venugopal, Second edition, TMH, 2006
4. Let Us C, Yashavant Kanetkar, 18th Edition, BPB, 2021
5. Programming in ANSI C, Balaguruswami, Eighth Edition, TMH, 2019

Course Code: **MCS-102** **Breakup: 3 – 0 – 0**
Course Name: **Computer Organization**
Course outcomes (CO): At the end of the course, the student will be able to:

CO1	Explain the basics of organizational and architectural issues of a digital computer and Classify and compute the performance of machines, Machine Instructions.
CO2	Describe various data transfer techniques in digital computer and the I/O interfaces.
CO3	Analyze the performance of various classes of Memories, build large memories using small memories for better performance and analyse arithmetic for ALU implementation
CO4	Describe the basics of hardwired and micro-programmed control of the CPU, pipelined architectures , Hazards and Superscalar Operations

Course Details:
Brief review of digital logic, Boolean algebra, flip flops, etc.
Data Representation: Integer representation-- number systems (binary, octal, Decimal, Hexadecimal), 1's and 2's Complements, Floating point numbers - - IE standard, normalization.
Computer Arithmetic: Half adder, Full adder, ripple carry and carry look-ahead adders, Multipliers - - Booth's algorithm. Processor Organization, Registers, Instruction cycle, ALU design, Instruction set of a processor, types of operands, types of operations, addressing modes, instruction formats.
Memory: RAM, ROM, DRAM Vs SRAM, Organization of memory cells inside a memory chip, Interfacing of memory with processor; Cache memory - mapping function emplacement algorithm, Write policy.
Input Output Organization: Program controlled, Interrupt driven (priority interrupts Daisy chaining), Direct memory access.
Control Unit: Micro–operations - - hardwired implementation, Micro -programming.
Computer Peripheral Organization: Keyboard, Monitor, Hard disk, CD-ROMs, Printers, etc.

Text Books and References :

1. V.C. Hamacher, Z.G. Vranesic and S.G.Zaky, Computer Organization, Fourth Edition, McGraw Hill, 1996.
2. Computer Organization & Architecture, Stallings, Eleventh Edition, Pearson, 2022
3. Computer Organization & Design, David A Paterson and John L. hennery, fifth edition, Morgan Kaufmann,
4. Computer System & Architecture, Morris Mano, TMH,,Third edition, 2007

Course Code: **MCS-103** **Breakup: 3 –1 – 0**
Course Name: **PROFESSIONAL COMMUNICATION**
Course outcomes (CO): At the end of the course, the student will be able to:

CO1	Plan, draft, revise, and edit documents for use in professional settings
CO2	Adapt writing to different audiences, purposes, and contexts
CO3	Synthesize and report on the professional and technical literature in the field
CO4	Write in a clear, coherent, and direct style appropriate for engineering communication
CO5	Understand and employ common documents in engineering writing, including proposals, failure
CO6	Analyses technical descriptions, research reports, and professional correspondence
CO7	Avoid plagiarism search, evaluate, and cite primary and secondary sources
CO8	Format documents in IEEE, the formatting style used in engineering communication

Course Details:

UNIT I

Introduction to Soft Skills– Hard skills & soft skills – employability and career Skills—Grooming as a professional with values—Time Management—General awareness of Current Affairs

UNIT II

Self-Introduction-organizing the material – Introducing oneself to the audience – introducing the topic – answering questions – individual presentation practice— presenting the visuals effectively – 5 minute presentations

UNIT III

Introduction to Group Discussion— Participating in group discussions – understanding group dynamics – brainstorming the topic -– questioning and clarifying –GD strategies- activities to improve GD skills

UNIT IV

Interview etiquette – dress code – body language – attending job interviews– telephone/skype interview -one to one interview &panel interview – FAQs related to job interviews

UNIT V

Recognizing differences between groups and teams- managing time-managing stress-networking professionally- respecting social protocols-understanding career management-developing a long-term career plan-making career.

Text Books and References:

1. English for Engineers and Technologists (Combined edition, Vol. 1 and 2), Orient Blackswan 2010.
2. Meenakshi Raman and Sangeetha Sharma,"Technical Communication: Principles and Practice", 2nd Edition, Oxford University Press, 2011
3. Stephen E. Lucas, "The Art of Public Speaking", 10th Edition; McGraw Hill Education, 2012.
4. Ashraf Rizvi, "Effective Technical Communication", 2nd Edition, McGraw Hill Education, 2017.
5. William Strunk Jr. & E.B. White, "The Elements of Style", 4th Edition, Pearson, 1999.
6. David F. Beer and David McMurrey, Guide to writing as an Engineer, John Willey. New York, 2004.

Course Code: **MCS-104** **Breakup: 3 – 0 – 0**
Course Name: **Database Management Systems**
Course outcomes (CO): At the end of the course, the student will be able to:

CO1	Describe the fundamental elements of relational database management systems
CO2	Explain the basic concepts of relational data model, entity-relationship model, relational database design, relational algebra and SQL.
CO3	Design ER-models to represent simple database application scenarios
CO4	Convert the ER-model to relational tables, populate relational database and formulate SQL queries on data.
CO5	Improve the database design by normalization.
CO6	Familiar with basic database storage structures and access techniques: file and page organizations, indexing methods including B tree, and hashing.

Course Details:

Introduction:
Database-System Applications
Purpose of Database Systems
File processing disadvantages
View of Data
Data Abstraction
Data Models
Database Languages
Relational Databases
DBMS Architecture
Introduction to the Relational Model
Structure of Relational Databases
Database Schema
Attributes and Keys
Schema Diagrams
Introduction to SQL
SQL Data Definition
Basic Structure of SQL Queries
Basic Operations
Set Operations
Null Values
Aggregate Functions
Nested Subqueries
Modification of the Database
Database Design and the E-R Model
Overview of the Design Process
The Entity-Relationship Model
Constraints
Removing Redundant Attributes in Entity Sets
Entity-Relationship Diagrams
Reduction to Relational Schemas
Entity-Relationship Design Issues
The Relational Algebra
The Tuple Relational Calculus
The Domain Relational Calculus
Functional Dependencies
Extraneous Attribute
Left irreducible FD
Prime/non-prime attributes
Logically Implied FD
Closure of a FD
Rules for logical inference of FD
Algorithm to determine closure of a FD set
Canonical Cover of a FD
Algorithm to determine Canonical Cover of a FD set
Algorithm to determine closure of an attribute set under FD set
Relational Database Design
Features of Good Relational Designs
Atomic Domains and First Normal Form
Decomposition Using Functional Dependencies
Lossless Join Decomposition
Dependency preserving Decomposition
Normalization
Introduction to Concurrency Control
Introduction to Transaction Management

Text Books and References:

1. Database System Concepts, Abraham Silberschatz), Henry F. Korth, S. Sudarshanl, McGraw Hill; 7th edition, 2021

2. Database Management Systems, Raghu Ramakrishnan, Johannes Gehrke, McGraw Hill Education; Third edition 2014
3. Fundamentals of Database Systems, Elmasri Ramez , Navathe Shamkant, Pearson Education; Seventh edition 2017

Course Code: MCS-105 **Breakup: 0 – 0 – 3**
Course Name: Lab Unix and C Programming

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

CO1	Recollect various programming constructs and to develop C programs
CO2	Understand the fundamentals of C programming
CO3	Choose the right data representation formats based on the requirements of the problem
CO4	Implement different Operations on arrays, functions, pointers, structures, unions and files

Course Details:

Learning OS Commands
Practice of all Internal and External DOS Commands, Writing simple batch programs, Exposure to Windows environment, Practice of UNIX commands and Vi editor, Writing simple shell script
C Programming:
Practicing programs to get exposure to basic data types, algebraic expressions, Conditional statements, Input Output Formatting, Control structures, arrays, functions, structures, pointers and basic file handling.

Course Code: MCS-106 **Breakup: 0 – 0 – 3**
Course Name: Lab Data Base Management System
Course outcomes (CO): At the end of the course, the student will be able to:

CO1	Describe the fundamental elements of relational database management systems
CO2	Explain the basic concepts of relational data model, entity-relationship model, relational database design, relational algebra and SQL.
CO3	Design ER-models to represent simple database application scenarios
CO4	Convert the ER-model to relational tables, populate relational database and formulate SQL queries on data.
CO5	Improve the database design by normalization.
CO6	Familiar with basic database storage structures and access techniques: file and page organizations, indexing methods including B tree, and hashing.

Course Details:

1. Creating tables for various relations (in SQL)
2. Implementing the queries in SQL for
a) Insertion
b) Retrival (Implement all the operation like Union, Intersect, Minus, in, exist, aggregate functions (Min.,Max…) etc…
c) Updation d) Deletion
3. Creating Views
4. Writing Assertions
5. Writing Triggers
6. Implementing Operations on relations (tables) using PI/SQL
7. Creating FORMS
8. Generating REPORTS.

Course Code: MCS-201 **Breakup: 3 – 0 – 0**
Course Name: Operating System

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

CO1	Understands the different services provided by Operating System at different level.
CO2	They learn real life applications of Operating System in every field
CO3	Understands the use of different process scheduling algorithm and synchronization techniques to avoid deadlock.
CO4	They will learn different memory management techniques like paging, segmentation and demand paging etc

Course Details:
Introduction and history of operating system
Process Management: Process Synchronization and mutual exclusion, Two process solution and Dekker's algorithm, semaphores monitors, Examples (Producer – consumer, reader-writer, dining philosophers, etc.)
CPU Scheduling: Multiprogramming and time sharing, Scheduling approaches (shortest–job–first, first–in–first–out, Round Robin, etc.)
Deadlock: Modeling, detection and recovery, prevention and avoidance.
Interprocess communication: Shared memory, message passing pipes.
Input/ output: Devices controllers and device drivers, disk scheduling, other devices
Memory Management: with and without swapping, virtual memory- paging and segmentation, page replacement algorithm, Implementation.
File System: FS services, Disk source management, Directory and data structure .Security, Protection, Access right.

Text Books and References:

1. Operating system concepts, A.Silberschatz and P.B. Galvin, , Wiley, 8th edition, 2017
2. Schaum's Outline of Operating Systems, J. Archer Harris, McGraw-Hill Education, 2001
3. Modern Operating Systems, Andrew Tanenbaum, Pearson; 4th edition 2014
4. Operating Systems Concepts And Design, Milan Milenkovic, McGraw Hill Education; 2nd edition 2001
5. Operating Systems: Internals and Design Principles, William Stallings, Pearson, 9th edition, 2018
6. Operating Systems : A Design-Oriented Approach, Charles Crowley, McGraw Hill Education, 2017

Course Code: **MCS-202** **Breakup: 3 – 0 – 0**
Course Name: **Data Structure**
Course outcomes (CO): At the end of the course, the student will be able to:

CO1	Learn the basic types for data structure, implementation and application.
CO2	Know the strength and weakness of different data structures.
CO3	Use the appropriate data structure in context of solution of given problem.
CO4	Develop programming skills, which require to solve given problem.

Course Details:

Basic concepts and notations, Mathematical background, Revision of arrays and pointers, Recursion and implementation of Recursion
Stacks and Queues : Sequential representation of stacks and queues
Lists: List representation techniques, Dynamics Storage allocation, Representation of stacks and queues using linked list, operations on linked list, Introduction to Doubly linked list.
Sorting Algorithms: Insertion sort, Bubble sort, Quick sort, Merge sort, Heap sort, Shell sort, Time and Space complexity of sorting algorithms
Tables: Searching sequential tables, Index sequential searching, Hash tables, Heaps.
Trees: Definition and basic concepts, Linked tree representations, Binary tree traversal algorithms,(Preorder, Inorder, Postorder), Binary search tree, Insertion and Deletion in Binary search tree, Multiway search trees, B trees, B+ tree and their applications, Digital search trees and Trie structure.
Graphs: Introduction to Graphs, Implementation of Graphs, Depth first search, Breadth first search.
Introduction to External Sorting

Text Books and References:

1. Data Structure Using C and C++, Y. Langsam, M.J. Augenstein and A.M. Tenenbaum, Second Edition, Pearson education, 2002.
2. Data Structures with C (Schaum's Outline Series), Seymour Lipschutz, McGraw Hill, first edition, 2017
3. Data Structures Using C, Aaron M. Tenenbaum, McGraw Hill, first edition, 1989

Course Code: **MCS-203** **Breakup: 3 – 0 –0**
Course Name: **Computer Network**
Course outcomes (CO): At the end of the course, the student will be able to:

CO1	Recognize the technological trends of Computer Networking.
CO2	Discuss the key technological components of the Network.
CO3	Evaluate the challenges in building networks and solutions to those

Course Details:

Introduction: history and development of computer networks, Local area networks, Metropolitan area networks, wide area networks, networks topology ISO/OSI seven layer architecture, connectionless versus connection oriented.
Data Communication: Data encoding and transmission ,data link control, Multiplexing, packet switching, LAN Architecture, LAN Systems(Ethernet, Token Ring), Network devices switches, Gateways , Routers
Physical Layer: transmission media, analog transmission, digital transmission.
Data link layer: framing error detection and correction, stop-and wait protocol, sliding window protocols, HSLC protocol.
MAC Layer: Aloha protocols, CSMA/CD: Ethernet, token ring, token bus Logical link control, Bridges and switches, FDDI, fast Ethernet, FDM, TDM.
Network layer: Virtual circuit, datagrams, Routing Algorithms shortest path, distance vector, link state routing, flooding, hierarchical routing, congestion control algorithms.
Internetworking tunneling, Encapsulation , Fragmentation. Multicasting, Inter network protocols (IP) – header structure, addresses, option, etc. Routing protocols, (Example : RIP,HELLO,OSPF,BGP)classless Inter- domain routi9ng other protocols, ICMP,ARP, RARP,BOOTP,DHCP.
Asynchronous Transfer mode (ATM); cell format, connection setup, switching, quality –of – services, ATM adaptation layers.

Text Book and References:

1. Computer Networks, S. Tanenbaum, Pearson Education India; Sixth edition, 2022
2. Data and Computer Communication, Stallings William, Pearson Education; Tenth edition, 2017
3. Data Communications and Networking with TCPIP Protocol Suite, Behrouz A. Forouzan, 6/e, McGraw Hill Education (India) Private Limited, 2022
4. Unix Network Programming Volume 1, Steavens/ Bill Fenner / Rudoff, Vol. 1, Pearson Education India; 3rd edition, 2015
5. Computer Networks: A Systems Approach, Peterson, Elsevier, Fifth edition 2011

Course Code: **MCS-204** **Breakup: 3 – 0 –0**
Course Name: **Web Technology**

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

CO1	Students are able to develop a dynamic webpage by the use of java script and DHTML
CO2	Students will be able to write a well formed / valid XML document
CO3	Students will be able to connect a java program to a DBMS and perform insert, update and delete operations on DBMS table
CO4	Students will be able to write a server side java application called Servlet to catch form data sent from client, process it and store it on database
CO5	Students will be able to write a server side java application called JSP to catch form data sent from client and store it on database, applying professional ethics

Course Details:

Introduction and Web Development Strategies, History of Web, Protocols governing Web, Creating Websites for individual and Corporate World, Cyber Laws, Web Applications, Writing Web Projects, Identification of Objects, Target ,Users, Web Team, Planning and Process Development.
HTML, XML and Scripting: List, Tables, Images, Forms, Frames, CSS Document type definition, XML schemes, Object Models, Presenting XML, Using XML Processors: DOM and SAX, Introduction to Java Script, Object in Java Script, Dynamic HTML with Java Script.
Java Beans and Web Servers: Introduction to Java Beans, Advantage, Properties, BDK, Introduction to EJB, Java Beans API. Introduction to Servelets, Lifecycle, JSDK, Servlet API, Servlet Packages: HTTP package, Working with Http request and response, Security Issues.
JSP: Introduction to JSP, JSP processing, JSP Application Design, Tomcat Server, Implicit JSP objects, Conditional Processing, Declaring variables and methods, Error Handling and Debugging, Sharing data between JSP pages- Sharing Session and Application Data.
Database Connectivity Database Programming using JDBC, Studying Javax.sql.*package, accessing a database from a JSP page, Application-specific Database Action, Developing Java Beans in a JSP page, Introduction to Struts framework.

Text Books and References:

1. Collaborative Web Development: Strategies and Best Practices for Web Teams, Jessica Burdman, Addison Wesley 1999
2. Chris Bates, “Web Programing Building Internet Applications”, John Wiley & Sons Inc; 3rd edition 2006
3. Joel Sklar , “Principal of web Design” Cengage; 5th edition 2012
4. Core Java: An Integrated Approach, R. Nageswara Rao, Dreamtech Press 2016
5. Herbert Schieldt, The Complete Reference:Java, McGraw Hill; Standard Edition 2022
6. Hans Bergsten, “Java Server Pages”, O′Reilly; 1st edition 2001

Course code: MCS-205 **Breakup: 0 – 0 – 3**
Course Name: Lab Web Technology

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

CO1	Students are able to develop a dynamic webpage by the use of java script and DHTML
CO2	Students will be able to write a well formed / valid XML document
CO3	Students will be able to connect a java program to a DBMS and perform insert, update and delete operations on DBMS table
CO4	Students will be able to write a server side java application called Servlet to catch form data sent from client, process it and store it on database
CO5	Students will be able to write a server side java application called JSP to catch form data sent from client and store it on database

Course Objective:

- To design interactive web pages using Scripting languages.
- To learn server side programming using servlets and JSP.
- To develop web pages using XML/XSLT.
- To embed an image map in a web page.
- Show all the related information when the hot spots are clicked
- Create a web page with all types of Cascading style sheets.
- Client Side Scripts for Validating Web Form Controls using DHTML.
- Installation of Apache Tomcat web server.
- Write script for web pages.
- To invoke dat from HTML forms.
- Session Tracking.

Course code: MCS-206 **Breakup: 0 – 0 – 3**
Course Name: Lab Data Structures
Course outcomes (CO): At the end of the course, the student will be able to:

CO1	Learn the basic types for data structure, implementation and application.
CO2	Know the strength and weakness of different data structures.
CO3	Use the appropriate data structure in context of solution of given problem.
CO4	Develop programming skills, which require to solve given problem.

Course Detail: Write Program in C for following:

1. Array implementation of Stack, Queue, Circular Queue
2. Linked list implementation using Dynamic memory Allocation, deletions and insertions, Linked Implementation of Stack, Queue, Circular Queue
3. Implementation of Tree Structures, Binary Tree, Tree Traversals, Binary Search Tree, Insertion and Deletion in BST, Simple implementation of Multiway search trees
4. Implementation of Searching and Sorting Algorithms
5. Graph Implementation, BFS, DFS.

Course Code: **MCS-301** **Breakup: 3 – 0 – 0**
Course Name: **Object Oriented Programming using Java**

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

CO1	Understand the basic concepts of Procedure–Oriented Programming and object-oriented programming.
CO2	Achieve the Knowledge of developing simple java programs.
CO3	Develop computer programs to solve real world problems.
CO4	Design simple GUI interfaces to interact with users, using Applets and swings.
CO5	Achieve Knowledge of multi-threading and to comprehend the event-handling techniques.

Course Details:

Basic Concepts: Object, Class, Inheritance, Instant, Instant variable, Attribute, Encapsulation, Information hiding, Multiple Inheritance, Typing, Dynamic typing, Object analysis, Object–oriented issues

Programming in Java: Variables, Simple I/O, file I/O, Class data types, derived classes, Functions, function overloading, Overloading operators, Abstract classes, Class inheritance, Interface, Multiple Inheritance, Templates, Java Library.

Text Books and References:

1. Java: The Complete Reference, Herbert Schildt, McGraw Hill, Twelfth Edition, 2021
2. Java 2 Unleashed, Stephen Potts, Alex Pestrikov, Sams Publishing; 6th edition 2002

Course Code: **MCS - 302** **Breakup: 3 – 0 – 0**
Course Name: **Theory of Computation and Compiler Design**

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

CO1	Design and analyse Finite Automata machines for given problems.
CO2	Design Pushdown Automata, Turing machine for given CF language(s).
CO3	Generate the strings/sentences of a given context-free languages using its grammar.
CO4	Understand fundamentals of compiler and identify the relationships among different phases of the compiler.
CO5	. Use modern tools and technologies for designing new compiler.

Course Details:

Model of Computation
Classification, Properties and equivalence's
Regular languages models:
finite state machine (deterministic and non – deterministic). Regular grammars, regular expression, Equivalence of deterministic and non – deterministic machines, Properties: closure, decidability, minimization of automata, iteration theorems.
Context – free languages models: Context – free grammars, simplification if CFGs , Chomsky normal form , Greibach normal form. Pushdown Automata, and their equivalence with context free languages, Properties closure , iteration theorems, parsing.
Recursive and recursively innumerable sets models: Turing machines, computable languages and function, Modification of Turning machines, Restricted Turning machines equivalents to the basic model, grammars recursive function , and their equivalence Church's thesis, Properties: closure, decidability, undecidability/ non – computability, notion of reductions.
Lexical Analysis: Interface with input , parser and symbol table, Token, lexeme and patterns. Difficulties in lexical analysis. Error reporting . Implementation, Regular definition, Transition Diagrams, Lex.
Syntax Analysis: CFGs, Ambiguity, associativity, precedence, top down parsing, recursive descent parsing, transformation on the grammars predictive parsing , bottom up parsing, operator precedence grammars, LR parsers (SLR.LALR, LR), YACC.
Syntax Directed definition: Inherited and synthesized attributes, dependency graph, Evaluation order, bottom up and top down evaluation of attributes, L- and s-attributes definition.

Text Books and References:

1. J.E. Hopcroft and J.D.Ullman & Motwani Introduction to Automata Theory, Language and Computation,3rd edition Addision wesley, 2007.
2. Peterlinz – An Introduction to formal Language & automata (Narosa Publication House), 6th edition, Jones & Bartlett, 2016
3. Theory of Computer Science: Automata, Languages and Computation, Mishra K.L.P, Prentice Hall India Learning Private Limited, 3rd edition, 2006
4. Introduction to Computer Theory, Daniel I.A. Cohen, Wiley; Second edition 2007
5. Theory of Computation (TMH), John Martin, McGraw Hill Education; 3rd edition 2007
6. Introduction to Theory of Computation, Michael Sipser, 2nd Edition, Thomson course technology, 2014

7. A.V. Aho, R. Sethi and J.D. Ullman, Compilers: Principle Techniques and Tools, Addition- Wesley 2007, 2nd edition.
8. Steven Muchnick – Advance Compiler Design Implementation (Elsevier India), 2008
9. Theory and Practice of Compiler Writing, Jean Paul Trembla, Paul Gordon Sorenson, McGraw-Hill Inc.,US, 1985
10. Compiler Design in C, Holub (Author), Allen, Prentice Hall India Learning Private Limited, 1992

Course Code: **MCS - 303** **Breakup: 3 – 0 – 0**
Course Name: **Design and Analysis of Algorithms**

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

CO1	Ability to decide the appropriate data type and data structure for a given problem
CO2	Ability to select the best algorithm to solve a problem by considering various problem characteristics, such as the data size, the type of operations, etc
CO3	Ability to compare algorithms with respect to time and space complexity

Course Details:

Notion of algorithm, Big Oh, Small–oh, Theta and Omega notations, Space and Time complexities of an algorithm
Sorting and Order Statistics: Revision of complexity analysis of different sorting algorithms and introduction to recurrence relations
Introduction: A first problem: Stable matching
Graph Algorithms: Breadth First search, Depth First search, single source shortest paths, minimum spanning trees, all pair shortest paths, Traveling sales person problem
Fundamental design paradigms:
Divide and Conquer: Mergesort, Binary search, Quick sort, Matrix multiplication,etc
Greedy methods: Shortest path algorithms, fractional knapsack problem, task scheduling problem,etc
Dynamic Programming: 0/1 knapsack problem, Longest common subsequence, Matrix chain multiplication, etc
Network Flow: The maximum flow problem and Ford Fulkerson algorithm, maximum flows and minimum cuts in a network
Theory of NP completeness: Polynomial time, NP complete problems, concept of reducibility.
Measure of approximation: ratio bound and relative error, Polynomial time approximation scheme.

Text Books and References:

1. E. Horowitz and S. Sahni, Fundamentals of Computer Algorithms, Galgotia, 2011
2. Algorithm Desig, Jon Kleinberg, Pearson Education India; 1st edition, 2013
3. Introduction to Algorithms, Charles E. Leiserson, Thomas H. Cormen, MIT Press; 4th edition 2022
4. Computer Algorithms: Introduction To Design And Analysis, Sara Baase and Van Gelder, Pearson Education,2000
5. Design & Analysis of Computer Algorithms, AHO, Pearson Education India; 1st edition, 2002

Course Code: **MCS-305** **Breakup: 0 – 0 – 3**
Course Name: **Lab Object Oriented Programming using Java**

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

CO1	Understand the basic concepts of Procedure–Oriented Programming and object-oriented programming.
CO2	Achieve the Knowledge of developing simple java programs.
CO3	Develop computer programs to solve real world problems.
CO4	Design simple GUI interfaces to interact with users, using Applets and swings.
CO5	Achieve Knowledge of multi-threading and to comprehend the event-handling techniques.

Course Details:

1. Programs illustrating the use of destructor and the various types of constructors (constructors without arguments, constructors with arguments, copy constructor etc).
2. Program illustrating use of functions and parameter passing
3. Programs illustrating overloading of various operators
4. Ex: Binary operators, Unary operators, New and delete operators etc.
5. Programs illustrating the use of following functions:
   a) Friend functions. b) Inline functions c) Static Member functions d) Overloaded Functions
6. Programs to create singly and doubly linked lists and perform insertion and deletion Operations. Using self referential classes, new and delete operators.
7. Programs illustrating various forms of inheritance: Ex. Single, Multiple, multilevel inheritance etc.
8. Programs on abstract class and derived classes
9. Programs illustrating the use of virtual functions.
10. Programs illustrating file handling operations:
    Ex. a) Copying a text file b) Displaying the contents of the file etc.
11. Write programs illustrating the console I/O operations.
12. Write programs illustrating how exceptions are handled (ex: division-by-zero, overflow and underflow in stacks etc.).

Course Code: **MCS - 401** **Breakup: 3 – 0 – 0**
Course Name: **Computer Graphics**

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

CO1	Understand the basics of computer graphics, different graphics systems and applications of computer graphics
CO2	Discuss various algorithms for scan conversion and filling of basic objects and their comparative analysis
CO3	Use of geometric transformations on graphics objects and their application in composite form
CO4	Extract scene with different clipping methods and its transformation to graphics display device
CO5	Explore projections and visible surface detection techniques for display of 3D scene on 2D screen.
CO6	Render projected objects to naturalize the scene in 2D view and use of illumination models for this

Course Details:

Introduction: Areas of Graphics, What is Computer Graphics, Video Display Devices, Fundamental problem in geometry.
Line drawing algorithm, Circle and Ellipse generating algorithms, Polynomial and Spline curves, Filling (Boundary fill, Flood fill etc.), Attributes of lines, Curves, Filling, Characters, etc.
Geometric Manipulation:
Transformation (Translation, Rotation, Scaling, Reflection etc), Matrix representation, Homogeneous coordinate systems
Two dimensional viewing:
Viewing coordinate reference frame, line clipping, polygon clipping
Elementary 3D Graphics:
Plane projection, Perspective, Orthographic projection, Surface vendering, Hidden lines Removal, Vanishing points, Specification of 3D view.
3D Transformations :
Rotation, Scaling, Shearing, Translation, Reflection.
Visibility: Image and Object precision, z-buffer algorithm, Area-based algorithm.

Text Book and References:

1. Hill – Computer Graphics using openGL, Pearson; 3rd edition 2007
2. Foley, Feiner & Hughes – Computer Graphics Principles & Practices in C (Addision wesley)
3. Computer Graphics C Version, Hearn, Pearson Education India; 2nd edition 2002
4. Procedural Elements of Computer Graphics, David Rogers, McGraw Hill Education; 2nd edition 2017
5. Yashwant Kanetkar – Computer Graphics Programming in C, BPB, 1998

Course Code: **MCS - 402** **Breakup: 3 – 0 – 0**
Course Name: **Artificial Intelligence**

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

CO1	Demonstrate fundamental understanding of the history of artificial intelligence (AI) and its foundations.
CO2	Apply basic principles of AI in solutions that require problem solving, inference, perception, knowledge representation, and learning.
CO3	Demonstrate awareness and a fundamental understanding of various applications of AI techniques in intelligent agents, expert systems, artificial neural networks and other machine learning models.
CO4	Demonstrate proficiency-developing applications in an 'AI language', expert system shell, or data-mining tool.
CO5	Demonstrate proficiency in applying scientific method to models of machine learning, apply AI to solve global problems

🟩 Local Green 🟨 Regional Yellow 🟦 National Blue ⬛ Global Grey

Course Details:

Introduction:
Introduction to AI, Foundations of AI, History of AI, Concept of AI techniques, the underlying assumptions, the state of art
Intelligent agents:
Agents and Behavior, The concept of rationality, Agent Architecture
Problem solving:
Problems, problem space and search – Formulating problems, Designing the problems as state space search, Issues in the design of search programs
Uninformed Search Techniques: Breadth first, Depth first, Depth limited, Iterative deepening, bidirectional, etc
Heuristic/Informed Search Techniques:
Generate and test, Best first search, A* search, Memory bounded heuristic search, Hill climbing search, Simulated annealing search, local beam search, genetic algorithms
Constraint Satisfaction Problem, Means End Analysis Adversial Search: Optimal decitions in games, Minmax algorithm, Alpha Beta Pruning
Knowledge Representation – knowledge representation issues, the predicate calculus representing knowledge using rules, symbolic reasoning, uncertainty, Probabilistic reasoning.
Languages and programming technique for AI:
An Introduction to PROLOG or LISP

Text Books and References:

1. S.J. Russell and P. Norvig , Artificial intelligence : A Modern Approach , Pearson; 3rd edition 2010
2. Elaine Rich and Kaven Knight – Artificial Intellegence McGraw Hill Education; 3rd edition, 2017
3. Introduction to Artificial Intelligence, Mariusz Flasiński, Springer, 1st ed. 2016
4. Introduction to Artificial Intelligence, Patterson, Pearson, 2015

Course Code: **MCS - 403** **Breakup: 3 – 1 – 0**
Course name: **Software Engineering.**

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

CO1	Basic knowledge and understanding of the analysis and design of complex systems
CO2	Ability to apply software engineering principles and techniques.
CO3	Ability to develop, maintain and evaluate large-scale software systems
CO4	To produce efficient, reliable, robust and cost-effective software solutions, applying professional ethics.
CO5	Ability to perform independent research and analysis.
CO6	To communicate and coordinate competently by listening, speaking, reading and writing English for technical and general purposes.
CO7	Ability to work as an effective member or leader of software engineering teams.

Course Details:

Software and Software Engineering
Software Process a Generic View
Software Process Models
Requirements Engineering
Project management Concepts
Software Process, Project and Product Metrics
Metrics for Design Model
Estimation for Software Projects
Analysis Concepts and Modeling
Software Testing

Text Books and References:

1. Software Engineering: A Practitioner's Approach, Bruce R. Maxim (Author), Roger S. Pressman, McGraw Hill Education; Eighth edition, 2019
2. Integrated approach to software engineering, Pankaj Jalote, Narosa, 2005
3. Software Engineering: A Precise Approach, Pankaj Jalote, Wiley, 2010
4. Fundamentals of Software Engineering, Rajib Mall, PHI Learning; 5th edition, 2018
5. Sommerville – S/W Engineering, Pearson Education; First edition, 2020

Course Code: **MCS-501** **Breakup:3– 1 – 0**
Course Name: **Digital Image Processing**

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

CO1	Review the fundamental concepts of a digital image processing system
CO2	Analyse images in the frequency domain using various transforms
CO3	Evaluate the techniques for image enhancement and image restoration.
CO4	Categorize various compression techniques
CO5	Interpret Image compression standards
CO6	Interpret image segmentation and representation techniques

Course Details:

UNIT-I
The image model and image acquisition image shape, sampling, intensify images, color images, range images, image capture, scanners.
UNIT-II
Statistical and spatial operations Grey Level transformations, histogram equilization, multi image operations. Spatially dependent transformations, templates and convolution window operations, Directional smoothing, other smoothing techniques.
UNIT-III
Segmentation and Edge detection region operations, Basic edge detection, second order detection, crack edge detection edge following, gradient operators, compass & laplace operators.
UNIT-IV
Morphological and other area operations, basic morphological operations, opening and closing operations, area operations morphological transforms.
UNIT-V
Image compression: Types and requirements, statistical compression, spatial compression, contour coding, quantizing compression.

Text Books and References:

1. Andrion Low-Introductory computer Vision and Image Processing MCGraw Hill International Edition, 1991
2. Digital Image Processing, Rafael Gonzalez , Richard Woods, Pearson; 4th edition 2017)

Course Code: **MCS – 502** **Breakup: 3 – 1 – 0**
Course Name: **Wireless & Mobile Computing**

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

CO1	Demonstrate knowledge on : cellular concepts like frequency reuse, fading, equalization, CDMA.
CO2	Demonstrate knowledge hand-off and interface and apply the concept to calculate link budget using path loss model.
CO3	Demonstrate knowledge equalization and different diversity techniques.
CO4	Apply the concept of GSM in real time applications.
CO5	Compare different multiple access techniques in mobile communication.
CO6	Study &amp; applications of different types of MANET’s Algorithm.

Course Details:

Introduction:
History of wireless communication, Cellular Telephone system, Mobile & Wireless devices, GSM, CDMA standards, Mobile services.

Wireless Transmission:
Frequencies for radio Transmission, Signals, Antennas, Signal propagation, Multiplexing, Modulation.

Modern Wireless Communication System:
2G Cellular networks, 3G wireless networks, WLL, WLANs, Bluetooth & Personal Area Network.

The Cellular Concept:
Frequency Reuse, channel assignment strategies, Handoff strategies, Interference & system capacity, improving coverage & capacity.

Mobile Radio Propagation: (Large Scale Path Loss):
Introduction to radio wave propagation, free space propagation model, Relating power to electric field, Three basic propagation mechanisms, Reflection, Ground reflection.

Small Scale Fading & Multipath: Small scale multipath propagation, Impulse response model of a multipath channel, small scale multipath measurements, parameters of mobile multipath channels.

Wireless Networking:
Introduction, Difference b/w fixed & wireless telephone networks, Development of Wireless Networking, Traffic Routing in wireless networks, CCS, ISDN.

Speech coding:
Introduction, characteristics of speech signals, Quantization Techniques, ADPCM, Frequency Domain Coding of Speech, Vocoders.

Text Books and References:

1. Wireless Communication –Theodore . S. Rappaport, (PHI 2002),2nd edition
2. Mobile Communication - Jochen Schiller, Adison Wisley, 2nd Edition 2003

Course Code: **MCS-503** **Breakup: 3 – 1 – 0**
Course Name: **Parallel Processing**

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

CO1	Understand the basic construction and use of parallel computers,
CO2	Use of the terminology for how one measures the performance of parallel algorithms and parallel computers
CO3	Develop computer programs for different types of parallel computers
CO4	Optimize sequential code for fastest possible execution
CO5	Analyze sequential programs and determine if they are worthwhile to parallelize
CO6	Develop, analyze, and implement algorithms for parallel computers. This applies both to computers with shared memory and with distributed memory

Course Details:

Introduction to Parallel Processing:
Supercomputers and grand challenge problems, Modern Parallel Computers, Data Dependence Graph, Data Parallelism, Functional Parallelism, Pipelining and Data Clustering.

Interconnection Networks:
Switch Network Topologies, Direct and Indirect Network Topology, Bus, Star, Ring, Mesh, Tree, Binary Tree Network, Hyper Tree Network, Hybrid, Hypercube, Perfect Shuffle Network, Torus and Butterfly Network.

Performance Analysis:
Introduction, Execution Time, Speedup, Linear and Superlinear Speedup, Efficacy and Efficiency, Amdahl’s Law and Amdahl Effect, Gustafson-Barsis’s Law, Minsky's Conjecture, The Karp-Flatt Metric, The Isoefficiency Metric, Isoefficiency Relation, Cost and Scalability.

Parallel Computational Models:
Flynn’s Taxonomy, PRAM, EREW, CREW, ERCW, CRCW, Simulating CRCW, CREW and EREW, PRAM algorithms.

Introduction to Parallel Algorithms:
Parallel Programming Models, PVM, MPI Paradigms, Parallel Programming Language, Brent’s Theorem, Simple parallel programs in MPI environments, Parallel algorithms on network, Addition of Matrices, Multiplication of Matrices.

Text Books and References:

1. Hwang and Briggs, advance Computer Architecture and Parallel Processing, McGraw Hill Education 2017
2. Crichlow, Introduction to Distributed and Parallel Computing, Prentice-Hall 1987
3. M.J.Quinn, Designing Efficient Algorithms for Parallel Computers, McGraw-Hill College; First Edition 1987
4. V.Rajaraman, Elements of Parallel Computing, Prentice-Hall of India.
5. Joseph JA JA, Introduction to Parallel Algorithms, Addison Wesley.
6. S.G.Akl, The Design and Analysis of Parallel Algorithms, Prentice Hall; First Edition 1989

Course Code: **MCS-504** **Breakup:** **3 – 1 – 0**
Course Name: **Cryptography and Network Security**

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

CO1	Provide security of the data over the network.
CO2	Do research in the emerging areas of cryptography and network security
CO3	Implement various networking protocols
CO4	Protect any network from the threats in the world, applying professional ethics

Course Details:

Unit I:

Introduction to security attacks and mechanisms, Introduction to cryptology.
Conventional Encryption: Convention encryption model, Classical encryption techniques – substitution ciphers & transposition ciphers, cryptanalysis,
stereography, stream & block ciphers.
Modern Block Ciphers: Block ciphers principles, Shannon's theory of confusion and diffusion, fiestal structure, Data Encryption Standards (DES), Strength of DES, Differential & Linear Cryptanalysis of DES, Block Cipher modes of Operation, Triple DES, IDEA encryption and decryption. Strength of IDEA, Confidentiality using conventional encryption, traffic confidentiality, key distribution, random number generation.

Unit II:

Introduction to graph, ring and field, prime and relative prime numbers, modular arithmetic, Fermat's & Euler's Theorem, primarily testing, Euclid's Algorithm, Chinese remainder theorem, Discrete algorithms.
Principles of Public-Key cryptosystems, RSA algorithm, Security of RSA, Key management, Diffle- Hellman key exchange algorithm, Introductory idea of Elliptic curve cryptography, Elganal encryption.

Unit III:

Message authentication and hash functions: Authentication requirements, Authentication functions, message authentication codes, hash function, birthday attacks, security of hash function. & MACS, MD5 message digest algorithm, Secure Hash Algorithm (SHA).
Digital signatures: Digital signatures, Authentication protocol, digital signature standard (DSS), proof of digital signature algorithm.

Unit IV:

Authentication Application: Kerberos & X.509, directory authentication service, electronic mail security- Pretty Good Privacy (PGP), S7 MIME.

Unit V:

IP Security: Architecture, Authentication Header, Encapsulating security payloads, combining security associations, Key management.
Web security: Secure Socket Layer & Transport security, Secure electronic Transaction (SET). System security: Intruders, Viruses and related threads, Firewall design principles, trusted systems.

Text Books and References:

1. William Stallings, "Cryptography and Network Security: Priciples and Practice", Prentice Hall, New Jersey,5th edition 2010
2. Johannes A. Buchmann, Introduction to Cryptography", Springer- Verlag,2nd edition 2004.
3. Bruce Schneier, Practical Cryptography, Pearson Education India

Course Code:	**MCS-505**	**Breakup:**	**3 – 1 – 0**
Course Name:	**VLSI Design**		

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

CO1	Model digital circuit with HDL, simulate, synthesis and prototype in PLDs.
CO2	Understand chip level issues and need of testability.
CO3	Design analog & digital CMOS circuits for specified applications

Course Details:

Basic MOS Transistor
Enhancement mode & Depletion mode – Fabrication (NMOS, PMOS, CMOS, BiCMOS) Technology – NMOS transistor current equation – Second order effects – MOS Transistor Model.

NMOS & CMOS Inverter and Gates
NMOS & CMOS inverter – Determination of pull up / pull down ratios – Stick diagram – lamda based rules– Super buffers, BiCMOS & steering logic.

Sub System Design and Layout
Structured design of combinational circuits – Dynamic CMOS & clocking – Tally circuits – (NAND- NAND, NOR-NOR and AOI logic) – EXOR structure – Multiplexer structures – Barrel shifter.

Design Of Combinational Elements and Regular Array Logic
NMOS PLA – Programmable Logic Devices - Finite State Machine PLA – Introduction to FPGA.

VHDL Programming
RTL Design – Combinational logic – Types – Operators – Packages – Sequential circuit – Sub-programs – Test benches. (Examples: address, counters, flipflops, FSM, Multiplexers / Demltiplexers).

Text Books & References:

1. D.A.Pucknell, K.Eshraghian, ‘Basic VLSI Design’, 3rd Edition, Prentice Hall of India, New Delhi, 2011.
2. Eugene D.Fabricius, ‘Introduction to VLSI Design’, Tata McGraw Hill, 1990.
3. N.H.Weste, ‘Principles of CMOS VLSI Design’, Pearson Eduction, India, 2002.
4. Charles H.Roth, ‘Fundamentals of Logic Design’, Jaico Publishing House, 1992.
5. Zainalatsedin Navabi, ‘VHDL Analysis and Modelling of Digital Systems’, 2nd Edition, Tata McGraw Hill, 1998.
6. Douglas Perry, ‘VHDL Programming by example’, Tata McGraw Hill, 4th Edition, 2002.

Course Code: **MCS-506** **Breakup:** **3 – 1 – 0**
Course Name: **Advanced Computer Networks**

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

CO1	To identify and discuss the concepts underlying IPv6 protocol, and their main characteristics and functionality
CO2	To understand the principles and functionality of mobile IP, explaining its concretization in IPv6; to understand the needs of optimization of the mobility mechanisms and description of some extensions that aim to reduce handover latency and requirements from terminals;
CO3	To recognize the need for service integration and discuss how it can be accomplished;
CO4	To explain and exemplify current QoS architectures and mechanisms, and the QoS support challenges in future networks;
CO5	To understand and explain the design issues in transport services in face of applications and services requirements

Course Details:

Revision of Computer Networks, Seven Layer Architecture, TCP/IP Suite of protocols etc.
Transport Layer: Flow and error control, multiplexing, establishing and releasing a connection, Transmission control protocol – header, services, connection management, convention control, sliding window and timers. User datagram protocol, Domain name services.
Unix network programming, socket abstraction client – server architecture.
Session presentation, application layers, Example protocols: Email (SMTP) Telnet, FTP, etc.
Internet security: firewalls. Network managements: SNMP.
IPV6: IPV6 Versus IPV4, Structure of IPV6 Protocol : general header structure , extension headers , IPV6 addressing : Types , notation, prefix notation , unicast, anycast , multicast addresses etc.
Security in IPV6: Basic Security Requirement and techniques, open security issues in current internet, IPSec frame work Quality of service in IPV6
ICMPV6: error messages, neighbor discovery, Auto configuration, path MTU discovery.
Wireless networks: Overview of 802.11 networks, 802.11 MAC, wired Equivalent privacy, Wireless communication technology: FHSS, DSSS, CDMA etc.
Mobility networks: Mobile IP, security related issues

Text Books and References:

All books used in the computer network

1. 802.11 wireless networks : The definitive guide, Mathew S. Gast, O’relly,2nd edition 2005
2. Wireless communication & networks: William Stallings, Pearson; 1st edition 2015
3. IPV6 Essentials , Silvia Hagen ,O'Reilly Media; 3rd edition 2014
4. TCP/IP Clearly Explained, Peter Loshin, Morgan Kaufmann; 4th edition 2003
5. Mobile IP design , Principle & Practices , Charles Perkin, Bobby Woolf, Sherman R. Alpert, Prentice Hall; First Edition 1998

Course Code: **MCS-507** **Breakup: 3 – 1 – 0**
Course Name: **Soft Computing (Neural Networks, fuzzy logic and Genetic algorithms)**

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

CO1	Learn about Neural Network, Fuzzy Logic and Genetic Algorithms, which are the major building blocks of an Intelligent Systems
CO2	Develop intelligent systems leveraging the paradigm of soft computing techniques.
CO3	Implement, evaluate and compare solutions by various soft computing approaches for finding the optimal solutions
CO4	Design hybrid system to revise the principles of soft computing in various applications

Course Details:

Neural network
Basic Concepts of Neural Network, Models of artificial Neural network, Characteristics of Neural Networks Network Architectures, Artificial Intelligence and Neural Networks
Leaning Processes
Introduction, Error-Correction Learning, Memory-Based Learning, Hebbian Learning, Competitive Learning,Boltzmann Learning, Credit Assignment Problem, Learning with a Teacher, Learning Tasks, Statistical Nature of the Learning Process, Statistical Learning Theory, Probably Approximately Correct Model of Learning
Single Layer Perceptrons
Adaptive Filitering Problem, Unconstrained Optimization Techniques, Linear Least-Squares Filters, Learning Curves, Learning Rate Annealing Techniques, Perceptron, Perceptron Convergence Theorem
Multi Layer Perceptrons
Some Preliminiaries, Back-Propagation Algorithm, Summary of the Back-Propagation Algorithm, XOR Problem, Heuristics for Making the Back-Propagation Algorithm Perform Better, Output Representation and Decision Rule, Computer Experiment, Feature Detection, Back-Propagation and Differentiation
Fuzzy Logic
Fuzzy Set Theory: Fuzzy verses crisp , crisp sets , Fuzzy sets, Crisp relations, Fuzzy relations
Fuzzy systems: Crisp logic, predicate logic , fuzzy logic , fuzzy rule based system , De fuzzification systems , applications
Genetic Algorithms
Fundamental of genetic algorithm ,Genetic algorithms , basic concept of genetic algorithm , creation of rings, working principal , encoding, fitness function, reproduction
Genetic Modeling:
Inheritance operators, cross over , inversion and deletion , mutation operation, bitwise operators, bitwise operators used in genetic algorithm , generational cycle, convergence of genetic algorithm.

Text Books and references :

1. Neural Networks, Fuzzy Logic, And Genetic Algorithms : Synthesis And Applications, S. Rajasekaran (Author), G. A. Vijayalakshmi Pai, PHI, 2013
2. Introduction to neural network James A. Anderson, MIT Press 1995
3. Introduction to genetic algorithm by Melanie Mitchell, MIT Press 1996
4. Genetic algorithm by Goldberg , Addison Wesley; 13th ed. edition 1989

Course Code: **MCS-508** **Breakup: 3 – 1 – 0**
Course Name: **Computational Geometry**

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

CO1	Analyze randomized algorithms for small domain problems
CO2	Use line-point duality to develop efficient algorithms.
CO3	Apply geometric techniques to real-world problems in graphics
CO4	Solve linear programs geometrically.

Course Details:

Convex hulls:
construction in 2d and 3d, lower bounds; Triangulations: polygon triangulations, representations, point-set triangulations, planar graphs; Voronoi diagrams: construction and applications, variants; Delayney triangulations: divideand- conquer, flip and incremental algorithms, duality of Voronoi diagrams, min-max angle properties;

Geometric searching:
pointlocation, fractional cascading, linear programming with prune and search, finger trees, concatenable queues, segment trees, interval trees;

Visibility:
algorithms for weak and strong visibility, visibility with reflections, art-gallery problems; Arrangements of lines: arrangements of hyperplanes, zone theorems, many-faces complexity and algorithms;

Combinatorial geometry:
Ham-sandwich cuts, Helly's theorems, k-sets, polytopes and hierarchies, polytopes and linear progarmming in d-dimensions, complexity of the union of convex sets, simply connected sets and visible regions; Sweep techniques: plane sweep for segment intersections, Fortune's sweep for Voronoi diagrams, topological sweep for line arrangements;

Randomization in computational geometry: algorithms, techniques for counting; Robust geometric computing; Applications of computational geometry.

Textbooks and Refrences

1. M. de Berg, M. van Kreveld, Mark Overmars & Otfried Schwarzkopf, "Computational Geometry: Algorithms and Applications," Second Edition, Springer-Verlag, 2000. ISBN: 3-540-65620-0.
2. Computational Geometry (An Introduction), by Franco P Preparata and Michael Shamos, Springer- Verlag, 1985.
3. Computational Geometry In C (Second Edition), by Joseph O'Rourke, Cambridge University Press, 1998.

Course Code: **MCS-509** **Breakup:** **3 – 1 – 0**
Course Name: **MANAGEMENT INFORMATION SYSTEMS**

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

CO1	Analyze randomized algorithms for small domain problems
CO2	Use line-point duality to develop efficient algorithms.
CO3	Apply geometric techniques to real-world problems in graphics
CO4	Solve linear programs geometrically.

Course Contents:

Module I: Basic Concepts of Information System
Role of data and information, Organization structures, Business Process, Systems Approach and introduction to
Information Systems.
Module II: Types of IS
Resources and components of Information System, integration and automation of business functions and
developing business models. Role and advantages of Transaction Processing System, Management Information
System, Expert Systems and Artificial Intelligence, Executive Support Systems and Strategic Information
Systems.
Module III: Architecture & Design of IS
Architecture, development and maintenance of Information Systems, Centralized and Decentralized Information
Systems, Factors of success and failure, value and risk of IS.
Module IV: Decision Making Process
Programmed and Non- Programmed decisions, Decision Support Systems, Models and approaches to DSS
Module V: Introduction to Enterprise Management technologies
Business Process Reengineering, Total Quality Management and Enterprise Management System viz. ERP,
SCM, CRM and Ecommerce.
Module VI: Introduction to SAD
System Analysis and Design. Models and Approaches of Systems Development

Textbooks and Refrences

1. Management Information Systems, Effy OZ, Cengage India Private Limited; 6th edition, 2013
2. Management Information Systems, James A. O'Brein, McGraw Hill Education; 10th edition, 2017
3. Management Information System, W.S Jawadekar, McGraw Hill; Sixth edition, 2020.
4. Management Information System, David Kroenke, McGraw-Hill Inc.,US; 2nd edition, 1992.

Course Code: **MCS-510** **Breakup: 3 – 1 – 0**
Course Name: **Embedded Systems**

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

CO1	Evaluate the requirements of programming Embedded Systems, related software architectures and tool chain for Embedded Systems.
CO2	Learn to develop the hardware for embedded system application based on the processors.
CO3	Explore the features of the microcontrollers and provide apt solutions for any embedded application.
CO4	Analyze the embedded systems' specification and develop software programs.
CO5	Incorporate suitable microcontroller along with appropriate interfacing circuits and implement the same for an application with software programs

Course Details:

Introduction
Introduction to embedded systems – hardware and software components – types – examples – characteristics– challenges in Embedded computing system design – embedded system design processes.
Architecture of Embedded System
Hardware components – SOC – Processors – CPU – Types of memory – Memory management – I/O devices and interfacing – Software components – Interpreter – Compiler – Assembler – Cross Assembler – RTOS – Languages for embedded applications – Hardware and software architecture. Examples: Cellphone, Smartcard, Digital Thermometer.
OS for Embedded Systems
Introduction to real time theory – Operating System Services – Real time Operating System Concepts – Basic design using a RTOS – Underground tank monitoring system.
Performance Issues of an Embedded System
CPU performance – CPU Power Consumption – Analysis and Optimization of CPU Power Consumption program execution time – Analysis and optimization of energy and power – Analysis of program size – Hardware accelerators.
Design Examples
Personal Digital Assistants – Set Top Boxes – Ink Jet Printers – Telephone PBX. Introduction to Micro C/OS-II operating system and its uses.

Text Books and References:

1. Wayne Wolf, (2001). "Computer as Components – Principles of Embedded Computing System Design", Harcourt India Pvt Ltd.,
2. .David E Simon, (2004) "An Embedded Software Primer", Pearson Education,
3. Raj Kamal, (2003) "Embedded Systems – Architecture, Programming and Design", Tata McGraw Hill,.
4. Sriram V Iyer, Pankaj Gupta, (2004) "Embedded Realtime Systems Programming", Tata McGraw Hill,
5. K.V.K.K. Prasad, (2004) "Embedded/Realtime Systems: Concepts, Design and Programming", Dreamtech Press

Course Code: **MCS-511** **Breakup: 3 – 1 – 0**
Course Name: **Advance Computer Architecture**

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

CO1	Interpret the performance of a processor based on metrics such as execution time, cycles per instruction (CPI), Instruction count etc
CO2	Predict the challenges of realizing different kinds of parallelism (such as instruction, data, thread, core level) and leverage them for performance advancement
CO3	Apply the concept of memory hierarchy for efficient memory design and virtual memory to overcome the memory wall
CO4	Explore emerging computing trends, computing platforms, and design trade-offs Teaching Methodology: Depends on the Instructor(s)

Course Details:

Review of pipelining, example of some pipelining in modern processors, Pipeline hazards, data hazards,
control hazards, techniques to handle hazards. Performance improvement with pipelines and effect of hazards on the performance.

Vector processor – use and effectiveness, memory to memory vector architectures vector register architecture. Vector length and stride issues. Compiler effectiveness in vectorization. Example of modern vector processors. Single instruction multiple data stream (SIMD) architecture, array processors, comparison with vector processors, example of array processors such as MMX technology.

Advance pipeline techniques, instruction level parallelism, basic instruction scheduling to avoid conflicts, dynamics scheduling, effect of loop unrolling. Branch prediction and their effectiveness in reducing control stalls, multiple issue of instruction compiler support for exploiting instruction level parallelism, issues of cache design.

Memory hierarchy. Cache Introduction, technique to reduce cache misses, techniques to reduce cache penalties, techniques to reduce cache hit times. Effect of main memory bandwidth, effect of bus width memory access time virtual memory etc.

RISC architectures, addressing modes, instruction formats, effect of simplification, on the performance example processors such as MIPS, PA-RISC, SPARC, PowerPC etc.
MIMD Multiprocessors. Centralized shared memory architectures, distributed shared memory architecture, synchronization and memory consistency models, message passing architectures, compiler issues. Dataflow architectures.
Interconnection networks: World wide parallel processing projects, Architecture of multiprocessor and multicomputer machines like hypercube, MMS, mesh CM*, CMP Iliac IV, Monsoon machine, dataflow architecture CM machine, teraflop computers.

Text Book and References:

1. Kai. Hwang, Advance computer architecture, MacGraw Hill, 1993.
2. Schaum's Outline of Computer Architecture, Nick Carter, McGraw-Hill Education 2002
3. Structured Computer Organization, Tanenbaum, Pearson Education India; Sixth edition 2016
4. Parallel Computer Architecture: A Hardware/Software Approach (The Morgan

Kaufmann Series in Computer Architecture and Design), Anoop Gupta (Author), David Culler (Author), J.P. Singh, Morgan Kaufmann Publishers In 1998

5. Advance computer architecture, Amit Mishra, S.K. Kataria & Sons; 2012th edition 2012

Course Code: **MCS-512** **Breakup: 3 – 1 – 0**
Course Name: **Natural Language Processing**

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

CO1	Apply the computational knowledge for Natural Language Processing to understand the properties of natural languages, its algorithms for processing linguistic information in various tasks such as Machine translation, Information extraction and retrieval, and Speech Technology to solve global problems related to NLP.
CO2	Understand the concepts of linguistic foundations that underlie natural language processing, which would provide the knowledge for building components of NLP systems.
CO3	Discover the capabilities, analyze them and explore the limitations of current natural language technologies, and some of the algorithms and techniques that underline these technologies to take up various research challenges in the field
CO4	Demonstrate the concepts of morphology, syntactic analysis, semantic interpretation and pragmatics of the language, and understanding them to apply in different research areas

🟩 Local Green 🟨 Regional Yellow 🟦 National Blue ⬛ Global Grey

Course Details:

Introduction to Natural Language Understanding
Linguistic Background: Outline of English Syntax

Knowledge Representation and Reasoning: A Representation Based on FOPC

Grammars and Parsing: Grammars and Sentence Structure, What Makes a Good Grammar, A Top-Down parser, Bottom-Up Chart Parser, Transition Network Grammars, Top-Down Chart Parsing, Finite State Models and Morphological Processing, Grammars and Logic Programming

Features and Augmented Grammars: Feature Systems and Augmented Grammars, Augmented Transition Networks

Grammars for Natural Language: Auxiliary Verbs and Verb Phrases, Movement Phenomena in Language, Toward Efficient Parsing, Human Preferences in Parsing

Encoding Uncertainty: Shift-Reduce Parsers, A Deterministic Parser, Techniques for Efficient Encoding of Ambiguity

Ambiguity Resolution: Statistical Methods, Basic Probability Theory, Estimating Probabilities, Part of Speech Tagging, Obtaining Lexical Probabilities, Probabilistic Context Free Grammars

Semantics and Logical form: Semantics and Logical form, Word senses and ambiguity, Encoding ambiguity in the logical form, Verbs and states in logical Form, Thematic roles

Text Books & References:

1. James Allen, Natural Language Understanding, Pearson; 2nd edition 1994
2. Jurafsky & Martin – Speech & Language Processors, Prentice Hall; 2nd edition 2008

Course Code: **MCS-513** **Breakup: 3 – 0 – 0**
Course Name: **Data Mining and Data Warehousing**

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

CO1	Understand data mining principles and techniques: Introduce DM as a cutting edge business intelligence method and acquaint the students with the DM techniques for building competitive advantage through proactive analysis, predictive modelling, and identifying new trends and behaviours. Learning objectives include:
CO2	Building basic terminology.
CO3	Learning how to gather and analyse large sets of data to gain useful business understanding
CO4	Learning how to produce a quantitative analysis report/memo with the necessary information to make decisions
CO5	Describing and demonstrating basic data mining algorithms, methods, and tools
CO6	Identifying business applications of data mining
CO7	Overview of the developing areas - web mining, text mining, and ethical aspects of data mining

Course Details:

Unit I:
Data Warehousing: Need for data warehousing , Basic elements of data warehousing, Data Mart, Data Warehouse Architecture, extract and load Process, Clean and Transform data, Star ,Snowflake and Galaxy Schemas for Multidimensional databases, Fact and dimension data, Partitioning Strategy-Horizontal and Vertical Partitioning.

Unit II:
Data Warehouse and OLAP technology, Multidimensional data models and different OLAP Operations, OLAP Server: ROLAP, MOLAP, Data Warehouse implementation ,Efficient Computation of Data Cubes, Processing of OLAP queries, Indexing data.

Unit III:
Data Mining: Data Preprocessing ,Data Integration and Transformation, Data Reduction, Discretizaion and Concept Hierarchy Generation , Basics of data mining, Data mining techniques, KDP (Knowledge Discovery Process), Application and Challenges of Data Mining, Introduction of Web Structure Mining, Web Usage Mining, Spatial Mining, Text Mining, Security Issue, Privacy Issue, Ethical Issue.

Unit IV:
Mining Association Rules in Large Databases: Association Rule Mining, Single-Dimensional Boolean Association Rules, Multi-Level Association Rule, Apriori Algorithm, Fp-Growth Algorithm, Time series mining association rules, latest trends in association rules mining.

Unit V:
Classification and Clustering Distance Measures, Types of Clustering, K-Means Algorithm,Decision Tree Induction, Bayesian Classification, Association Rule Based, Other Classification Methods, Prediction, Classifier Accuracy, Categorization of methods, Partitioning methods, Outlier Analysis.

Text Books and References:

1. Data Mining: Concepts and Techniques, A volume in The Morgan Kaufmann Series in

Data Management Systems, Third Edition, 2012

2. P.Ponnian, “Data Warehousing Fundamentals”, John Wiley & Sons Inc; 2nd edition 2010
3. M.H.Dunham, “Data Mining Introductory & Advanced Topics”, Pearson Education.
4. Ralph Kimball, “The Data Warehouse Lifecycle Tool Kit”, John Wiley.
5. M.Berry , G.Linoff, “Master in Data Mining”, Wiley; 3rd edition 2008
6. W.H.Inmon, “Building the Data Ware houses”, Wiley; 4th edition 2005
7. E.G. Mallach , “The Decision Support & Data Warehouse Systems”, McGraw-Hill Education 2000
8. Data Warehousing in the Real World: A practical guide for building Decision Support Systems, D. Murray, Addison-Wesley; 1st edition 1997
9. David Hand, Heikki Manila, Padhraic Symth, “Principles of Data Mining”, PHI 2004..
10. Alex Bezon, Stephen J.Smith, “Data Warehousing, Data Mining & OLAP”, McGraw Hill Education 1 July 2017

Course Code: **MCS-514** **Breakup: 3 – 1– 0**
Course Name: **Multi-core architectures**

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

CO1	Identify the limitations of Instruction-level parallelism and the need for multicore architectures
CO2	Define fundamental concepts of parallel programming and its design issues
CO3	Solve the issues related to multiprocessing and suggest solutions
CO4	Make out the salient features of different multicore architectures and how they exploit parallelism
CO5	Demonstrate the role of OpenMP and programming concept

Course Details:

Introduction to multi-core architectures, issues involved into writing code for multi-core architectures, how to develop programs for these architectures, program optimizations techniques.
OpenMP and other message passing libraries, threads, mutex etc.

Introduction to parallel computers:
Instruction level parallelism (ILP) vs. thread level parallelism (TLP); Performance issues: Brief introduction to cache hierarchy and communication latency; Shared memory multiprocessors: General architectures and the problem of cache coherence.

Synchronization primitives:
Atomic primitives; locks: TTS, ticket, array; barriers: central and tree; performance implications in shared memory programs.

Chip multiprocessors:
Why CMP (Moore's law, wire delay); shared L2 vs. tiled CMP; core complexity; power/performance; Snoopy coherence: invalidate vs. update, MSI, MESI, MOESI, MOSI; performance trade-offs; pipelined snoopy bus design; Memory consistency models: SC, PC, TSO, PSO, WO/WC, RC; Chip multiprocessor case studies: Intel Montecito and dual-core Pentium4, IBM Power4, Sun Niagara.

Introduction to optimization:
Overview of parallelization; Shared memory programming, introduction to OpenMP; Dataflow analysis, pointer analysis, alias analysis; Data dependence analysis, solving data dependence equations (integer linear programming problem); Loop optimizations; Memory hierarchy issues in code optimization; Operating System issues for multiprocessing Need for pre-emptive OS.

Scheduling Techniques:
Usual OS scheduling techniques, Threads, Distributed scheduler, Multiprocessor scheduling, Gang scheduling; Communication between processes, Message boxes, Shared memory; Sharing issues and Synchronization, Sharing memory and other structures, Sharing I/O devices, Distributed Semaphores, monitors,spin-locks,

Text Books and References:

1. Computer Architecture: A Quantitative Approach (The Morgan Kaufmann Series in Computer Architecture and Design), John L. Hennessy (Author), David A. Patterson,

Morgan Kaufmann; 6th edition 2017

2. Parallel Computer Architecture: A Hardware/Software Approach (The Morgan Kaufmann Series in Computer Architecture and Design), Anoop Gupta (Author), David Culler (Author), J.P. Singh, Morgan Kaufmann Publishers, 1998
3. Kai hwang, Advance computer architecture. Mac Graw Hill 1993

Course Code: **MCS-515** **Breakup: 3 – 1 – 0**
Course Name: **Machine Learning**

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

CO1	Appreciate the importance of visualization in the data analytics solution
CO2	Apply structured thinking to unstructured problems
CO3	Understand a very broad collection of machine learning algorithms and problems
CO4	Learn algorithmic topics of machine learning and mathematically deep enough to introduce the required theory
CO5	Develop an appreciation for what is involved in learning from data.

Course Details:

Introduction: Introduction to machine learning, supervised learning, unsupervised learning Reinforcement learning.
Revision: Basics of Probability Theory, Basics of Linear Algebra and Statistical Decision Theory.
Supervised learning:
Linear regression: Linear Regression, Linear discriminant analysis, Polynomial Regression. Ridge Regression, Lasso Regression. Parameter Estimation: Least Square, Least Mean Square, Gradient Descent.
Classification: Two class classification, Multi-class classification, Concept of Loss functions for classification
Classification algorithms:
Logistic Regression: Introduction. to Logistic Regression, Types of Logistic Regression, Regression Models, Binary Logistic Regression Model, Multinomial Logistic Regression Model, Naive Bayes: Bayes Theorem, The Naive Bayes' Classifier.
Decision trees, Regression trees, Stopping criteria &pruning.
SVM:SVM—formulation,interpretation&analysis,SVMsforlinearlynon-separabledata,SVM kernels. SVM hinge loss formulation.
Artificial Neural Networks: Concept of Perceptron & Parameter Estimation, Early artificial neural network models , Feed forward networks, Recurrent Networks. Concept of Back propogation, Initialization, training & validation, Maximum Likelihood estimate.
Unsupervised learning:
Clustering: Partitional clustering, Hierarchical clustering, K- Means, II- NN, Dimensionality reduction, BIRCH algorithm.
Association Mining: Frequent Itemset Mining, Apriori algorithm, FP-growth algorithm.
Evaluation Measures & Hypothesis Testing: Evaluation measures, Bootstrapping & cross validation, 2 class evaluation measures, The ROC curve.
Introduction to Advance Topics: Recommendation systems, Deep learning etc.

Text Book and References:

1. Tom M. Mitchell : "Machine Learning", 2013.
2. Hal Daume III: "A Course in Machine Learning, 2012.
3. Christopher M. Bishop, "" Pattern Recognition and Machine Learning", 2010.
4. Ian Goodfellow, Yoshua Bengio, Aaron Courville, Francis Bach : "Deep Learning", 2017.

Course Code: **MCS-516** **Breakup: 3 – 0 – 0**
Course Name: **Industrial Management**

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

CO1	Choose, prepare, interpret and use cost estimates as a basis for the different situations in an industrial company
CO2	Interpret financial statements and other financial reports of industrial companies, including the income statement, the balance sheet, the cash flow statement, key measures, budget and sustainalbility analysis in these
CO3	Explain how the industrial company can be organised and managed
CO4	Explain the industrial company's value creating processes, how the company can price it's products and how the company works in it's environment.

Course Details:

Introduction to Industrial management, Brief history of industries in India, Brief definition of management, organization and administration. Characteristics of management, Principle of management, Function of management like, planning, organization, direction, co-ordination etc.

Level of management, skills of management, inter relation between skills and levels of management, scientific management, Introduction to Schools of Management thoughts, introduction to organization, study of basic type of organization for ex. Line and staff organization, project organization, metrics organization, Informal organization, Introduction to industrial Psychology, Motivation theory and study of Maxlow, Need, Hierarchy Theory, Planned Location, Planned Layout. Study of different forms of layout like line layout, process layout, product layout, combinational layout, sixth position layout etc.

Objective of planned layout, introduction to material management, scope of material management, study of inventory control method, introduction to different types of inventory control techniques, introduction to work study, motion study etc, introduction to conflict management.

Text Book and References:

1. Khanna O.P. : Industrial Engineering
2. T.R. Banga : Industrial Engineering and Management
3. Mahajan : Industrial and Process Management

Course Code: MCS-517 **Breakup: 3 –1 – 0**
Course Name: Industrial Economics

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

CO1	Have sufficient knowledge about demand and supply problems
CO2	Understand concepts of production and cost analysis
CO3	Use of microeconomic tools in problem solving
CO4	Utilisation of limited resources in meeting the rising demand in the market

Course Details:

UNIT-1
Meaning, definition and scope of economics, Basic concepts of demand and supply, Market equilibrium, Ceiling price and floor price.

UNIT-2
Price elasticity of demand: Factors affecting price elasticity of demand, Calculation, Relation between marginal revenue, demand and price elasticity, Income elasticity of demand and Cross elasticity of demand, Indifference curves, Budget Line

UNIT-3
Production and Cost analysis: Basic concepts, Production in the short- run and long-run, cost analysis
Finding the optimal combination of inputs, Returns to scale

UNIT-4
Market: Characteristics of perfect completion, Profit maximisation in short-run and long-run
Firms with market power: Measurement and determinants of market power, Profit maximisation under monopoly: output and pricing decisions, Price discrimination, capturing consumer surplus, Strategic decision making in oligopoly markets

UNIT-5
National income: Concepts, Sources, Measurement, Difficulties, circular flow of income
Inflation: Cost-push and Demand-pull inflation, Effects and control of inflation, Business cycle, Functions of RBI, GST

Text and References Books:

1. Economics by Paul. A. Samuelson
2. Managerial Economics by Christopher R. Thomas, S. Charles Maurice, Sumit Sarkar
3. Financial Management by J. V. Vaishampayan
4. Micro Economics by A. Koutsoyannis