Indira Gandhi Delhi Technical University for Women

(Established by Govt. of Delhi vide Act 09 of 2012)

Kashmere Gate, Delhi–110006

Scheme of Examination

&

Detailed Syllabus

(w.e.f. Academic Year 2019-2020 onwards)

For

Master of Computer Applications

Department of Information Technology

PROGRAMME OUTCOMES

Post Graduates of Master of Computer Application will be able to:

PO1. Apply knowledge of Computing fundamentals, Computing specialization, Mathematics, and domain knowledge appropriate for the computing specialization to the abstraction and conceptualization of computing models from defined problems and requirements.

PO2. Design and develop applications to analyze and solve all computer science related problems.

PO3. Design applications for any desired needs with appropriate considerations for any specific need on societal and environmental aspects.

PO4. Analyze and review literatures to invoke the research skills to design, interpret and make inferences from the resulting data.

PO5. Integrate and apply efficiently the contemporary IT tools to all computer applications.

PO6. Solve and work with a professional context pertaining to ethics, social, cultural and cyber regulations.

PO7. Involve in perennial learning for a continued career development and progress as a computer professional.

PO8. Function effectively both as a team leader and team member on multi-disciplinary projects to demonstrate computing and management skills.

PO9. Communicate effectively and present technical information in oral and written reports.

PO10. Understand and assess societal, environmental, health, safety, legal, and cultural issues within local and global contexts, and the consequential responsibilities relevant to professional computing practice.

PO11. Function effectively as an individual and as a member or leader in diverse teams and in multidisciplinary environments.

PO12. Identify a timely opportunity and using innovation to pursue that opportunity to create value and wealth for the betterment of the individual and society at large.

PROGRAMME SPECIFIC OUTCOMES

PSO1. Design, develop and implement interdisciplinary application software projects to meet the demands of industry requirements using modern tools and technologies.

PSO2. To prepare graduates who will perform both as an individual and in a team through good analytical, design and implementation skills.

PSO3. To prepare graduates who will be lifelong learners through continuous professional development.

FIRST SEMESTER

S. No.	Code	Subject	L-T-P	Credits	Category
1	MCA-101	Fundamentals of Information Technology	3-0-2	4	DCC
2	MCA-103	Problem Solving using C Programming	3-0-4	5	DCC
3	MCA-105	Discrete Mathematics	3-1-0	4	DCC
4	MCA-107	Computer Organization	3-0-2	4	DCC
5	HMC-101	Professional Skills	3-0-0	3	HMC
TOTAL				20	

SECOND SEMESTER

S. No.	Code	Subject		L-T-P	Credits	Category
1	MCA-102	Data Structures		3-0-4	5	DCC
2	MCA-104	Object Oriented Programming with C++		3-0-4	5	DCC
3	MCA-106	Software Engineering		3-0-2	4	DCC
4	MCA-108	Operating Systems		3-0-2	4	DCC
5	HMC-102	Human Values and Professional Ethics		3-0-0	3	HMC
	TOTAL				21	

THIRD SEMESTER

S. No.	Code	Subject	L-T-P	Credits	Category
1	MCA-201	Design and Analysis of Algorithms	3-0-2	4	DCC
2	MCA-203	Cloud Computing	3-0-2	4	DCC
3	MCA-205	Database Management Systems	3-0-2	4	DCC
4	MCA-207	Web Technologies	3-0-2	4	DCC
5	GEC-201	Generic Open Elective-1	0-0-4	2	GEC
6	HMC-201	Principles of Management	3-0-0	3	HMC
7	MCA-253	Industrial Training/Internship	-	1	DCC
TOTAL				22	

FOURTH SEMESTER

S. No.	Code	Subject	L-T-P	Credits	Category
1	MCA-202	Java Programming	3-0-2	4	DCC
2	MCA-204	Artificial Intelligence	3-0-2	4	DCC
3	MCA-206	Data Communications and Computer Networks	3-0-2	4	DCC
4	DEC-2xx	Departmental Elective-1	3-1-0	4	DCC
5	HMC-202	Disaster Management	2-0-0	2	HMC
6	HMC-204	Organizational Behavior	3-0-0	3	HMC
TOTAL				21	

FIFTH SEMESTER

S. No.	Code	Subject	L-T-P	Credits	Category
1	MCA-301	Software Testing	3-0-2	4	DCC
2	DEC-303	Machine Learning and Data Analytics	3-0-2	4	DCC
3	DEC-3xx	Departmental Elective -2	3-0-2	4	DEC
4	DEC-3xx	Departmental Elective-3	3-0-2	4	DEC
5	GEC-301	Generic Open Elective-2	0-0-4	2	GEC
6	MCA-351	Minor Project	3-0-0	3	DCC
7	MCA-353	Industrial Training/Internship	1-0-0	1	DCC
TOTAL				22	

SIXTH SEMESTER

S. No.	Code	Subject	L-T-P	Credits	Category
1	MCA-352	Major Project	-	20	DCC

LIST OF DEPARTMENTAL ELECTIVE COURSES

Category	Course Code	Subject	Credits
Departmental Elective Course-1	MCA-208	Computer Graphics and Multimedia Technologies	3-0-2
	MCA-210	Soft Computing	3-0-2
	MCA-212	Cyber Security and Forensics	3-1-0
	MCA-214	Software Project Management	3-0-2
Departmental Elective Course-2	MCA-305	Network Security	3-0-2
	MCA-307	Advanced DBMS	3-0-2
	MCA-309	E-Commerce	3-0-2
	MCA-311	Software Quality Assurance	3-1-0
Departmental Elective Course-3	MCA-313	Internet of Things (IoT)	3-0-2
	MCA-315	Advanced Data Structures	3-0-2
	MCA-317	Theory of Computation	3-1-0
	MCA-319	Mobile Computing	3-1-0

Fundamentals of Information Technology

Course Code: MCA-101 **Contact Hours:** L-3 T-0 P-2 **Course Category:** DCC	**Credits :** 4 **Semester** : 1

Introduction

The course Fundamentals of Information Technology has become essential in the present age of computer technology and information, as the applications of information technology can be found in all aspects of our lives. This course is designed to meet the requirements of students having very little knowledge of computers and help them to learn from the basic fundaments of computers through applications of information technology.

Course Objectives

- To introduce skills relating to IT basics, computer applications, programming, Operating systems and computer network basics etc.
- To help students to understand specialized advanced courses in the Information Technology.

Pre-requisite: Preliminary knowledge of computer, their operations and applications.

Course Outcomes:

CO1: Understanding the concept of input and output devices of computers.

CO2: Learn the functional unit and classify the type of computers, how they process information and how the individual computer interact with the other computing system and devices.

CO3: Understand an operating system and its working, and solve the common problems related to operating system.

CO4: Study to use the computer safely, legally, and responsibly.

Pedagogy

Lectures will be delivered via discussions, whiteboard, slideshows and assignments.

UNIT-I	**10 hrs**
Information Concepts and Processing : Definition of Information Technology, Quality, need of information system, levels of information, data processing, definition of knowledge, Range of application : Scientific, business, educational, e-commerce, web publishing, Management Information System, Decision Support System, inventory control, and industrial control. **Number System**: Bit, byte, binary, decimal, hexadecimal, and octal systems, conversion from one system to the other, Binary Arithmetic: Addition, subtraction and multiplication. **Representation of Information:** Integer and floating-point representation, Complement schemes, Character codes (ASCII, EBCDIC, BCD, Excess-3, Grey).	
UNIT-II	**10 hrs**
Introduction to Computer software: Introduction to system software, categories of system and application, Distinction between systems software and Application software, Introduction to Software Development activities (Requirement, Design (algorithm and flowchart), Coding, Testing, Installation & Maintenance). **Introduction to Computer Hardware**: CPU, Memory, different types of memories (Cache memory, virtual memory and Auxillary memory) , Various I/O devices. **Programming languages and Translators:** Low- and high-level languages, assembly language, 4GL and 5GL Introduction to assemblers, compilers, interpreters, linkers and loaders.	
UNIT-III	**10 hrs**
Operating systems (Only introductory level): Evolution, introduction to OS , functions and facilities, Different types of operating systems (Batch, multi-programming, time sharing, multiprocessing, PC operating system, real time operating system, single tasking and multitasking OS , single user and multi-user OS), Introduction to process management: process, threads, scheduling, characteristics of MS-DOS and Unix operating systems , DOS and UNIX commands, Introduction to Database Management System and its types.	
UNIT-IV	**10 hrs**
Communication and Computer Network: **-** Basic elements of a Communication System, , Data transmission media, Digital and Analog, Network Types (LAN, WAN and MAN), inter networking devices and Communication Protocols, Intranet and Extranet, Hypertext Markup Language, WWW, HTTP, HTTPs, FTP, Telnet, Web Browsers, Search Engines, Email, Digital Signatures, Firewall.	
Text Books	
1. Anoop Mathew ,Fundamentals of Information Technology, Alpha Science International Ltd, 2013	
2. P. K. Sinha and Priti Sinha , "Computer Fundamentals", BPB Publications, 2011.	
3. Forouzan, Data Communication and Networking, McGraw Hill Education, 2017	
Reference Books	
1. V. Rajaraman, "Fundamentals of Computers", PHI; 6th Revised edition, 2014	
2. Morris Mano, "Computer System Architecture", Pearson, 3rd Ed, 2017.	

Problem Solving using C Programming

Course Code: MCA-103 **Contact Hours**: L-3 T-0 P-4 **Course Category:** DCC	**Credits:** 5 **Semester:** 1

Introduction:

This course provides an introduction to computer concepts, logic, and computer programming. It includes designing, coding, debugging, testing, and documenting programs using a high-level programming language.

Course Objectives:

- To learn the fundamental programming concepts and methodologies, essential to build efficient C programs.
- To practice the fundamental programming methodologies in the C programming language via lab sessions.
- To code, document, test, and implement a well-structured, robust computer program using the C programming language.
- To write reusable modules (collections of functions) in C.

Pre-requisite: None

Course Outcomes:

CO1: Recall the basic principles of C Programming.

CO2: Illustrate the use of Conditional Statements & Looping Concepts.

CO3: Develop the Concepts of programming Language.

CO4: Create a program using File operations.

Pedagogy: The class will be taught using theory and tutorial based methods which include board teaching and presentations/slides, discussions etc. Along with classroom teaching, students will also be given assignments regarding the topics covered.

UNIT I	11 hrs
Introduction to Programming and its Environment: Need for programming, Levels (High and Low) of programming, Development process (Preprocessor, Compiler, Linker and Loader), Linux –commonly used commands like mkdir, cd, ls, etc., compiler –gcc, editor –vim **C Language Introduction:** Program Structure through simple C programs, Constants and Variables, Data Types –Basic and Advanced, Operators and Expressions, Managing input and output operations using printf and scanf, Command line input, Conditional constructs, Looping constructs. Problem solving exercises based on –conditional and looping constructs	
UNIT II	**11 hrs**
Pointers, Arrays and Strings: Concept of memory, Definition, Usage –address of and value at operation, Pointer arithmetic. Pointer to pointer, Arrays (Single and Multi-dimensional) and Strings–with emphasis on role of pointers in them, Pointer to Array, Array of pointers. Problem solving exercises based on –pointers, arrays and strings. **Procedural programming:** Functions (Function Prototyping, passing parameters through call by value and call by reference, returning values, recursion), Program organization using functions, Emphasis on reusability through C examples. Problem solving exercises based on –functions.	
UNIT III	**10 hrs**
File handling: Concept of streams, File pointer, Reading and Writing to file, Closing a file, Random access in a file, Error handling during file I/O operations. Problem solving exercises based on –files. **Problem Solving:** Algorithm, Flowchart and Pseudo code. Program design.	
UNIT IV	**10 hrs**
Advanced concepts: Pointers to functions and Callback functions. Storage classes (auto, extern, static, register), The C Preprocessor (#define, #undef, #include, #if conditional inclusion and other pre-processor directives), Defining New Data Types–Structures, Unions, Enumerated Types **Dynamic Memory Management:** malloc, calloc, realloc, size of, free. **Introduction to Data Structure**: Linked Lists and dynamic data structures. Problem solving exercises based on –advanced concepts and data structure	
Text Books	
1. Yashwant Kanetkar, “Let us C”, BPB Publications, 16th edition, 2018.	
2. B. Kernighan and D. Ritchie, “The ANSI C Programming Language”, 2nd edition.	
Reference Books	
1. Paul Deitel and Harvey Dietel, “How to Program”, PHI, 8th Ed., 2015.	
2. Behrouz A. Forouzan and Richard F. Gilberg, “Computer Science A Structured Programming Approach Using C”, PHI, 3rd Ed., 2007	
3. Jeri R. Hanley and Elliot B. Koffman, “Problem Solving and Programming in C”, Pearson, 8th Ed. 2015.	

Discrete Mathematics

Course Code: MCA-105 **Contact Hours:** L-3 T-1 P-0 **Course Category**: DCC	**Credits** 4 **Semester** 1

Introduction

Discrete mathematics forms the mathematical foundation of computer and information science. This course familiarizes with a broad range of mathematical objects like sets, functions, relations, graphs, that are omnipresent in computer science.

Course Objectives

- To explain formal statements and their proofs; coming up with rigorous proofs themselves; and coming up with interesting results.
- To show at least one interesting and non-trivial result and give a full proof of introduced concepts.

Pre-requisites:

Basic mathematical operations

Course Outcomes:

CO1: Apply logical reasoning to solve a variety of problems

CO2: Develop understanding of logic, sets and functions.

CO3: Apply algebraic structure in combinatorial mathematics.

CO4: Develop an understanding of how graph and tree concepts are used to solve problems arising in the computer science.

Pedagogy:

The materials are delivered mostly through lectures videos to make complex subject easy to comprehend. More details on certain lessons are delivered through examples to provide more explanation.

UNIT I	**10 hrs**
Set Theory: Notations, Types of sets, Multisets, Ordered pairs, Cartesian product, Combination of sets, Set Algebra, Proofs of some general identities on sets. **Relations:** Representation, Relation types and properties, Operations on relations, Equivalence relations, Equivalence Partitions, Equality of relations, Order of relations, Partial ordering, Recursive definition of relation, **Closure:** Reflexive, Symmetric and Transitive closures, Warshall's algorithm to compute transitive closure of a relation, Composite Relations, Functions, Classification of functions, Operation on functions. **Boolean Algebra:** Introduction, Boolean functions, Representations and simplification of Boolean functions, Theorems of Boolean algebra, Algebraic manipulation of Boolean expressions, Logic implications, Karnaugh maps, Application of Boolean functions to synthesis of circuits. Partially Ordered Sets and Lattices: Posets, lattices, Combination of partial order sets, Properties of lattices, Lattices as Algebraic systems, Sub lattices, Homomorphism, Hasse's diagram, Bounded, Complemented, Modular and Complete lattice.	
UNIT II	**10 hrs**
Combinatorics: Principle of mathematical induction, Selected problems on mathematical induction, Fundamental principles of counting, Pigeonhole principle, Principle of inclusion and exclusion. Discrete Numeric Functions and Recurrence Relations: Introduction, Asymptotic behaviour, Linear recurrence relations with constant coefficients (homogeneous and non-homogeneous case, Solution of linear recurrence relations using generating functions. **Logic:** Propositional logic, Tautology, Predicate Algebra, Quantifiers, Operators, Methods of **Proofs:** direct, formal, informal, contradiction, induction, contraposition, exhaustive.	
UNIT III	**10 hrs**
Discrete Probability: Sample space, Discrete Sample space, Types of Events: mutually exhaustive, mutually exclusive, Axioms of probability, Conditional probability, Total probability, Bayes' theorem, Univariate and bivaiate probability distributions , Discrete random variables, Probability mass function and cumulative distribution function, Mode and median and variance of a univariate and bivariate discrete probability distribution, Mathematical Expectation (Univariate and bivariate Random Variable), Expectation of a function of a random variable, Effect of change of origin and scale on mean and variance. Expectation and variance of sums of random variables. Conditional expectation and prediction. The Central Limit Theorem, Algebraic Structures: Definition, Groups, Subgroups and order, Cyclic Groups, Cosets, Lagrange's theorem, Semi groups and monoids, Cyclic semigraphs and submonoids, Congruence relations on semigroups, Normal Subgroups, Dihedral groups, Permutation and Symmetric groups, Group Homomorphisms, Properties of Rings and Fields, Integers Modulo n, polynomial arithmetic, quadratic residues, reciprocity, discrete logarithms, elliptic curve arithmetic.	
UNIT IV	**11 hrs**
Graph theory: Path, cycles, handshaking theorem, bipartite graphs, sub-graphs, graph isomorphism, operations on graphs, Eulerian graphs and Hamiltonian graphs, planar graphs, Euler formula, traveling salesman problem, shortest path algorithms. Graphs, Euler tours, planar graphs, Hamiltonian graphs, Euler's formula, applications of Kuratowski's theorem, graph coloring, chromatic polynomials, trees, weighted trees, shortest path algorithms, spanning trees, the max-flow min-cut theorem. **Applications of Discrete Mathematics in Computer Science:** Information Theory, Semantic Web, Formal Software Verification, Theorem Proving, Game Theory, Cryptography	
Text Books	
1. Rosen, Kenneth H., and Kamala Krithivasan. Discrete mathematics and its applications: with combinatorics and graph theory. Tata McGraw-Hill Education, 2012.	
2. Seymour Lipschutz, Marc Laras Lipson, Varsha H. Patil, Discrete Mathematics (Schaum's Outlines) McGraw Hill Education; Revised Third edition (1 July 2017)	
Reference Books	
1. Deo, Narsingh. Graph theory with applications to engineering and computer science. Courier Dover Publications, 2017.	

Computer Organization

Course Code: MCA- 107 **Contact Hours:** L-3 T-0 P-2 **Course Category**: DCC	**Credits** : 4 **Semester** :1

Introduction:

The course aims to provide students with an understanding of the design of fundamental blocks of a computer system and interfacing techniques of these blocks to achieve different configurations of a computer system. It covers the basic topics in the design of computational units, instruction organization, memory systems, control and data flow, and interconnections.

Course Objective:

- To have a thorough understanding of the basic structure and operation of a digital computer.
- To study the different ways of communicating with I/O devices and standard I/O interfaces.

Pre-requisite: Digital Systems and Computer Design

Course Outcome:

CO1: Understand different number systems, binary addition and subtraction, 2's complement representation and operations with this representation.

CO2: Comprehend the theory and architecture of central processing unit, pipelining, interrupt handling and memory organization

CO3: Analyze some of the design issues in terms of speed, technology, cost, performance.

CO4: Design combinational circuits for basic components of computer system and applications using multiplexers, decoders, flip flops etc.

Pedagogy:

The class will be taught using theory and tutorial-based methods which includes board teaching and presentations/slides, case studies, discussions etc. Along with classroom teaching, students will also be given assignments regarding the topics covered. The course instructor will demonstrate and explain about applications of Computer organisation techniques with research orientation.

UNIT-I	**10 Hours**
Introduction and overview: Multiplexes, Demultiplexers, Decoders, Adders **Flip-flops:** S-R, JK, D, T, Master Slave and Edge triggered, Registers, shift registers, Bi-direction shift registers. **Register Transfer and Microoperation:** Register transfer language, register transfer, bus and memory transfer, arithmetic microoperations, logic microoperations, shift microoperations.	
UNIT-II	**11 Hours**
Basic Computer Organization and Design: Instruction codes, computer registers, computer instructions, timing & control, instruction cycle, memory reference instructions, input-output and interrupts, design of basic computer, design of accumulator logic. **Micro programmed Control Unit:** Control memory, address sequencing. **Central Processing Unit:** Introduction, general register organization, stack organization, instruction formats, addressing modes.	
UNIT-III	**11 Hours**
Pipeline and Vector processing: Parallel Processing, pipelining, arithmetic pipeline, RISC Pipeline, Vector Processing, Array Processors. **Input-Output Organization:** Peripheral devices, input-output interface, asynchronous data transfer, modes of data transfer, priority interrupt, direct memory access, input-output processor.	
UNIT-IV	**10 Hours**
Memory organization: Memory hierarchy, main memory, auxiliary memory, associative memory, cache memory, virtual memory, memory management hardware. **Multiprocessors:** Characteristics of multiprocessor, Interconnection Structure, Interprocessor Communication & Synchronization.	
Text Books	
1	Mano M, “Computer System and Architecture”, Pearson, 3rd Ed., 2009
2	Stallings W, “Computer Organization & Architecture”, PHI, 9th Ed., 2013.
Reference Books	
1	Hayes, J. P. “Computer Architecture and Organization”, McGraw Hill, 3rd edition, 2017.
2	Andrew S. Tanenbaum, “Structured Computer Organization”, PHI, 6th Ed., 2016.
3	P. V. S Rao, “Computer System Architecture”, PHI, 5th Ed., 2009.

Professional Skills	
Course Code: HMC-101 **Contact Hours:** L-3 T-0 P-0 **Course Category**: HMC	**Credits**: 3 **Semester**: 1

Introduction - This course aims to enhance the students' professional communication skills by providing adequate exposure in verbal and nonverbal skills and related sub skills. The course is designed to provide awareness of appropriate communication strategies with social, organizational and cultural awareness. The course empowers students in day to day professional soft skills like listening skills, presentation skills, and group discussion etc.

Course Objectives:

- To know the process of professional communication and its various components.
- To improve language skills i.e. Listening Skills, Speaking Skills, Reading Skills and Writing Skills (LSRW).
- To create literary sensibility and enhance comprehension skills.
- To develop confidence for communicating in English language.

Pre-requisites: None

Course Outcome – After completion of the course, the students should be able to:

CO1: Understand the importance of flawless communication in professional environment.

CO2: Enrich knowledge and improve skills required for corporate world.

CO3: Evaluate theoretical frameworks and concepts for the study of communication.

CO4: Develop ethical professional habits.

Pedagogy:

To provide knowledge of various communication processes through innovative and interactive classroom teaching sessions. To evaluate students' progress through practical sessions including Group discussion, Presentations, role plays and JAMs.

UNIT-I	**10 Hours**
Self analysis through SWOT, Johari window, Personality Development, Intra personal communication vs. Inter personal Communication and Relationships, Leadership Skills, Team Building, Public speaking, Individual Communication, Self advertising, Over stating and under stating, Time Management.	
UNIT-II	**12 Hours**
Communication Boosters: Body language, Voice, Posture and gesture, Eye contact, Dress codes, Verbal crutches, Pronunciation, Contextualization: creating and understanding contexts, Aura words. **Interview:** Types of Interview, Preparing for the Interviews, Attending the Interview, Interview Process, Employers Expectations, General Etiquette.	
UNIT-III	**10 Hours**
Group Discussions: Guidelines, Expressions, Evaluation. Video conferencing, Telephone skills, Teleconferencing, Participation in meetings: chairing sessions. Presentation Skills, Types of presentation, Capturing Data, Guidelines to make an effective presentation, Body Language, Voice Modulation, Integrating voice & picture, Audience Awareness, Presentation Plan, Visual Aids, Forms of Layout, Styles of Presentation, Management presentations.	
UNIT-IV	**10 Hours**
Letter writing: Types of Letters, Business letters, E-mail, Fax, Pro-forma culture, Drafting the Applications, Format, Style, Effectiveness, study of sample letters, Elements of structure, Preparing a CV / Resume, Statement of Purpose, Paragraph Writing, Greeting, Memos, Reports, Minutes, Business correspondence.	
Text Books	
1.	Rajendra Pal, J S Korlahhi. Essentials of Business Communication, Sultan Chand & Sons, 2017.
2.	Andre J. Rutherford . Basic Communication Skills for Technology, Pearson Education Asia, 2014.
3.	KR Lakshiminarayana: English for Technical Communication, Scitech Publications, 2015.
Reference Books	
1.	RK Madhukar. Business Communication, Vikas Publishing House Pvt. Ltd. 2018.
2.	English in Mind, Herbert Puchta and Jeff Stranks, 2nd Edition, Cambridge University Press 2010.
3.	Suresh K, P. Srihari, J Savithri, Communication Skills and Soft Skills: An Integrated Approach,1st Edition, Pearson Education, 2010.

Data Structures	
Course Code: MCA-102 **Contact Hours:** L-3 T-0 P-4 **Course Category**: DCC	**Credits**: 5 **Semester:** 2

Introduction:

This course covers the design, analysis, and implementation of data structures and algorithms to solve engineering problems using an object-oriented programming language. Topics include elementary data structures, (including arrays, stacks, queues, and lists), advanced data structures (including trees and graphs), the algorithms used to manipulate these structures, and their application to solving practical engineering problems.

Course Objectives:

- To learn efficient storage mechanisms of data for an easy access.
- To design and implement various basic and advanced data structures.
- To introduce various techniques for representation of the data in the real world.
- To develop applications using data structures.

Pre-requisite: Standard programming language C/C++, mathematical knowledge, knowledge of basic probability.

Course Outcomes:

CO1: Understand the basics of data structures to represent data items in the real world.

CO2: Evaluate the time and space complexities of Algorithms.

CO3: Apply and implement the application of sorting and pattern-matching algorithms.

CO4: Create projects using a variety of data structures such as stacks, queues, hash tables, binary trees, search trees, heaps, graphs, and B-trees.

Pedagogy: The class will be taught using theory and tutorial-based methods which include board teaching, presentations/slides, discussions and case-based studies. Along with classroom teaching, students will also be given assignments regarding the topics covered.

UNIT-I	**11 Hours**
Introduction: Abstract Data Type, Elementary Data Organization, Measuring efficiency of an Algorithm, Time and Space Complexity, Asymptotic notations. Arrays: Single and Multidimensional Arrays, **Representation of Arrays:** Row Major Order, and Column Major Order, Application of arrays, Sparse Matrices. **Linked lists:** Array and Dynamic Implementation of Single Linked Lists, Doubly Linked List, Circularly Linked List, Operations on a Linked List. Insertion, Deletion, Traversal, Polynomial Representation and Addition. **Stacks:** Stack operations: Push & Pop, Array and Linked list implementation of Stack, Applications: Prefix and Postfix Expressions, Evaluation of postfix expression, Recursion.	
UNIT-II	**11 Hours**
Queues: Operations: Create, Add, Delete, full and empty queues, Array and linked implementation of queues, Dequeue, Circular queues and Priority Queue. Hashing: Hash Function, Hash Table, Collision Resolution Strategies. **Trees:** Basic terminology, Binary Trees, Array and linked list implementation, Types of Binary Tree, Extended Binary Trees, Algebraic Expressions , Tree Traversal algorithms: Inorder, Preorder and Postorder, Threaded Binary trees, Search, Addition and deletion of an element in a binary tree, AVL Trees, Heaps, B Trees, Trees and their applications, Evaluating an expression tree.	
UNIT-III	**10 Hours**
Searching: Sequential search, Binary Search. Sorting: Insertion Sort, Selection, Bubble Sort, Quick Sort, Merge Sort, Heap Sort, Radix Sort, Bucket Sort, Shell Sort, Graphs: Representation (Matrix and Linked), Traversals, Shortest path, Topological sort. Dijkstra's Algorithm, Floyd Warshall's Algorithm, Minimum Spanning Tree Algorithms (Kruskal's Algorithm, Prim's Algorithm).	
UNIT-IV	**10 Hours**
Files: Creation and Processing of files, File handling using command line arguments, File opening, closing, modes, formatted inputs, output to file, reading/writing of files, accessing records randomly, updating files. Operations on files, Library functions, File Indexing (primary, secondary, clustered, unclustered, dense, sparse), File streams, Hierarchy of file stream classes, Error handling during file operations.	
Text Books	
1	Aaron Tanenbaum, "Data Structures Using C", 2nd edition, 2016
2	Ellis Horowitz and Sartaj Sahni, "Fundamentals of data structures" 2nd edition, 2017
Reference Books	
1	Seymour Lipschutz, "Data Structures", 2nd Edition, 2015
2	Donald Knuth, "The Art of Computer Programming", 2015.

Object Oriented Programming with C++

Course Code: MCA-104 **Contact Hours:** L-3 T-0 P-4 **Course Category**: DCC	**Credits**: 5 **Semester:** 2

Introduction:

This course provides in-depth coverage of object-oriented programming principles and techniques. Topics include classes, objects, overloading, data abstraction, information hiding, encapsulation, inheritance, polymorphism, file processing, templates, exceptions, container classes etc. The course material embraces the C++ language standard with numerous examples demonstrating the benefits of C++. In the end some basics of Java will be covered.

Course Objective:

To learn object-oriented programming (OOP) principles and get a flavour of modular programming

Pre-requisite: Basics of C Programming

Course Outcomes:

CO1: Distinguish between the various programming paradigms available and understand the basic syntax of object-oriented programming.

CO2: Build the classes and apply the various features of the language.

CO3: Able to develop programs with reusability.

CO4: Implement programme using namespace, templates, exception handling and file I/O to improve effective programming skills.

Pedagogy: Emphasis on lab sessions where students will be given programming assignments to code in C++/Python/Java based on topics learnt in previous lectures.

UNIT-I	**10 Hours**
Need for Object Oriented Programming, Comparison of Programming paradigms, Characteristics of Object-Oriented Programming Languages, Introduction to Object Oriented concepts (classes, objects, encapsulation, inheritance, data hiding, abstraction, polymorphism), Fundamentals Data Types & Literals Variables, Arrays, Operators, Control of Flow in OOP, Compilation and Execution of Process , Reference vs. Pointer variable, Classes and Objects: class declaration, Role of private, public and protected access specifiers, Memory organization of class, inline function, friend function, static members , constructor and destructors, instantiation of objects, default parameter value, object types	
UNIT-II	**11 Hours**
Garbage collection, dynamic memory allocation, new and delete operator Polymorphism: Function overloading, Constructor overloading, Compile time polymorphism, Overloading Rules, Operator Overloading (Unary and Binary) as member function/friend function. Inheritance, Types of Inheritance, Use of protected access specifier, Virtual base class, Ambiguity resolution using scope resolution operator and Virtual base class, Overriding inheritance methods, Constructors and Destructor in derived classes, Runtime polymorphism,	
UNIT-III	**11 Hours**
Pointer to objects, Virtual Functions (concept of virtual table), pure virtual functions, Abstract Class, Managing Input / Output, Concept of streams, console I/O – formatted and unformatted, Manipulators, File I/O – Predefined classes, file opening & closing, file manipulation, read & write operations, sequential and random file access, Exception Handling: Basic mechanism, Throwing, Catching and Re-throwing. Namespace: Basic concept, role of scope resolution operator and using keyword	
UNIT-IV	**10 Hours**
Introduction to Java- Overview and characteristics of Java, Data types, Organization of the Java Virtual Machine, Compilation and Execution Process in java , Java Classes, Packages and interfaces, Case Studies using C++ to build highly extensible software: System Sort, Apache Traffic Server, Apache Open Office Document Suite	

Text Books	
1	Josée Lajoie and Stanley B. Lippman, “C++ Primer”, 5th Edition, 2013
2	Herbert Schildt , “Java: The Complete Reference”, 7th Edition, TMH.
3	Martin C. Brown, “Python: The Complete Reference”, 4th Edition, TMH, 2018
Reference Books	
1	Herbert Schildt, “C++: The Complete Reference”, 4th Edition, TMH, 2017
2	Mark Lutz, “Learning Python”3rd Edition, O’’reilly Media, 2007
3	Bjarne Stroustrup , “The C++ Programming Language”, Pearson, 3rd Ed, 2000

Software Engineering

Course Code: MCA-106 **Contact Hours:** L-3 T-0 P-2 **Course Category**: DCC	**Credits**: 4 **Semester:** 2

Introduction

This course introduces students to the different software development lifecycle (SDLC) phases used in developing, delivering, and maintaining software products. Students will also acquire basic software development skills and understand common terminology used in the software engineering profession. The aim of the course is to provide an understanding of the working knowledge of the techniques for estimation, design, testing and quality management of software development projects.

Course Objectives:

- To introduce the concepts of software engineering, software processes and its models.
- To understand the software requirements analysis, transform the requirements using DFD, create software requirement specification document and validation of the software requirements.
- To understand fundamental of software design, software quality and software maintenance.
- To understand the project planning process, size and cost estimation techniques for development of software.

Pre-requisite: Basic knowledge of Programming Languages

Course Outcomes:

CO1: Understand the concepts of Software engineering, Software process and its models.

CO2: Evaluate the Software Requirements Specification, Interpret and Create Software Requirements Specification Document.

CO3: Apply appropriate software architectures and patterns to carry out high level design of a system and be able to critically compare alternative choices, evaluate the quality and maintenance of the software through software testing.

CO4: Create the software project plan for size and cost estimation including risk analysis.

Pedagogy: This course is structured around continuous progress. It will include a combination of lectures, and group activities focused on experiential learning, in-class discussions, regular assessments and case studies. The topics will be presented to students using real-world scenarios and problem-solving activities.

UNIT-I	**10 Hours**
Introduction: Software Crisis, Software Processes & Characteristics, Software life cycle models, Waterfall, Prototype, Evolutionary and Spiral Models. Software Requirements analysis & specifications: Requirement engineering, requirement elicitation techniques like FAST, QFD & Use case approach, requirements analysis using DFD, Data dictionaries & ER Diagrams, Requirements documentation, Nature of SRS, Characteristics & organization of SRS, Requirement Management, IEEE Std. for SRS.	
UNIT-II	**11 Hours**
Software Project Planning: Size Estimation like lines of Code & Function Count, Cost Estimation Models, COCOMO, Putnam resource allocation model, Validating Software Estimates, Risk Management. Software Design: Cohesion & Coupling, Classification of Cohesiveness & Coupling, Function Oriented Design, Object Oriented Design.	
UNIT-III	**11 Hours**
Software Metrics: Software measurements: What & Why, Token Count, Halstead Software Science Measures, Data Structure Metrics, Information Flow Metrics. Software Reliability: Importance, Hardware Reliability & Software Reliability, Failure and Faults, Reliability Models- Basic Model, Logarithmic Poisson Model, Software Quality Models, CMM & ISO 9001.	
UNIT-IV	**10 Hours**
Software Testing: Testing process, Design of test cases, Introduction to functional testing & Structural testing, Unit Testing, Integration and System Testing, Debugging, Alpha & Beta Testing. Software Maintenance: Management of Maintenance, Maintenance Process, Maintenance Models, Regression Testing, Reverse Engineering, Software Re-engineering, Configuration Management, Documentation.	
Text Books	
1	K.K.Aggarwal, Yogesh Singh: Software Engineering, New Age International Ltd, 3rd Ed. 2008.
2	Pankaj Jalote, An Integrated Approach to Software Engineering, Narosa Publishing, 2010.
Reference Books	
1	R.S. Pressman, Software Engineering – A Practitioner’s Approach, 8th Edition, McGraw Hill, 2019.
2	Ian Sommerville, Software Engineering, 10th Edition, Pearson, 2017.

Operating Systems

Course Code: MCA-108 **Contact Hours:** L-3 T-0 P-2 **Course Category:** DCC	**Credits**: 4 **Semester**: 2

Introduction

This course aims at introducing classical internal algorithms and structures of modern operating systems including CPU scheduling, memory management, and device management. Topics including file systems, virtual memory, disk scheduling, concurrent processes, deadlocks, security, and integrity will be covered.

Course Objectives

- To learn the fundamentals of Operating Systems.
- To learn the mechanisms of OS to handle processes and threads and their communication.
- To learn the mechanisms involved in memory management in contemporary OS.
- To gain knowledge on OS architecture, mutual exclusion algorithms, deadlock detection algorithms etc.
- To know the components and management aspects of concurrency management.

Pre-requisite: Analysis of algorithms, algorithm design techniques, programming knowledge in C, C++ or JAVA.

Course Outcome:

CO1: To understand various types of OS, basic concepts, various functions of different OS, process management & CPU scheduling.

CO2: To compare and contrast various memory management schemes like paging, segmentation and to apply different deadlock handling algorithms.

CO3: To implement different disk scheduling algorithms, to apply and use various process synchronization techniques and device management strategies.

CO4: To understand management of I/O and different file handling & directory implementation schemes in OS.

Pedagogy: The class will be taught using theory and tutorial-based methods which includes board teaching and presentations/slides, discussions, case studies etc. Along with classroom teaching, students will also be given assignments regarding the topics covered. The course instructor will demonstrate and explain about applications of Operating Systems techniques with real-time examples.

UNIT-I	**10 Hours**
Introduction: Definition, Role, Types of Operating System, Batch Systems, multi programming, time–sharing, parallel, distributed and real-time systems, Operating system structure, Operating system components and services, System calls, System programs, Virtual machines. **Processes:** Process Concept, Process Scheduling, Operation on Processes, Cooperating Processes, Threads. **CPU Scheduling:** Basic Concepts, Scheduling Criteria, Scheduling Algorithms, Multiple Processor Scheduling, Real-Time Scheduling.	
UNIT-II	**11 Hours**
Interprocess Communication and Synchronization: Background, The Critical-Section Problem, Synchronization Hardware, Semaphores, Classical Problems of Synchronization, Critical Regions, Monitors, Message Passing. **Deadlocks:** System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock. **Memory Management:** Background, Logical vs. Physical Address space, swapping, Contiguous allocation, Paging, Segmentation, Segmentation with Paging.	
UNIT-III	**11 Hours**
Virtual Memory: Demand Paging and its performance, Page-replacement Algorithms, Allocation of Frames, Thrashing, page size and other Considerations, Demand Segmentation. **Device Management:** Techniques for Device Management, Dedicated Devices, Shared Devices, Virtual Devices, Independent Device Operation, Buffering, Device Allocation Consideration **Secondary-Storage Structure:** Disk Structure, Disk Scheduling, Disk Management, Swap Space Management, Disk Reliability.	
UNIT-IV	**10 Hours**
File-System Interface: File Concept, Access Methods, Directory Structure. **File-System Implementation**: Introduction, File-System Structure, Basic File System, Allocation Methods, Free-Space Management, Directory Implementation. **Security:** The Security problem, Goals of protection, Access matrix, Authentication, Program threats, System threats, Intrusion detection.	
Text Books	
1	Silberschatz and Galvin, “Operating System Concepts”, John Wiley, 9th Ed., 2016
2	Tannenbaum, “Operating Systems”, PHI, 5th Ed., 2000.
3	Deitel, Deitel and Choffnes, “Operating Systems”, Pearson, 3rd Edition, 2003
Reference Books	
1	Madnick E. and Donovan J., “Operating Systems”, McGraw Hill, 2017.
2	Flynn McHoes, “Operating System”, Cengage Learning, 6th edition, 2013.
3	Sibsankar Halder and Alex A. Arvind, “Operating System”, Pearson, 2009
4	William Stallings, “Operating Systems Internals & Design Principles”, Pearson Education, 9th Ed., 2018

Human Values and Professional Ethics

Course Code: HMC-102 **Contact Hours**: L-3 T-0 P-0 **Course Category**: HMC	**Credits**: 3 **Semester**: 2

Introduction: Values and Ethics are very relevant in today's environment of conflicts and stress in every profession, with obligations to be met by one person in many directions. A formal study will certainly improve one's ability and judgment and refine one's behaviour, decisions, and actions in performing the duty to the family, organization, and to the society.

Course Objectives:

- To facilitate the development of a Holistic perspective among students towards life, profession and happiness, based on a correct understanding of the Human reality and the rest of Existence. Such a holistic perspective forms the basis of Value based living in a natural way.
- To inculcate Ethics and Human Values into the young minds and develop moral responsibility and mould them as best professional which will create ethical vision and achieve harmony in life.

Prerequisite: None

Course Outcomes:

Having successfully completed this course, the student will be able to

CO1: Develop the capability of shaping themselves into outstanding personalities through a value-based life.

CO2: turn themselves into champions of their lives.

CO3: take things positively, convert everything into happiness, and contribute to the happiness of others.

CO4: become potential sources for contributing to the development of the society around them institutions/organizations they work in.

Pedagogy: The learning and teaching methods include the use of weekly three-hour lectures to illustrate the subject and provide examples of the practical application of such topics. Lecture materials will be supported by directed reading and assignments.

UNIT-I	**10 Hours**
Human Values Morals, Values and Ethics, Integrity, Work Ethic, Respect for Others, Living Peacefully, Caring, Sharing, Honesty, Valuing Time, Co-operation, Commitment, Empathy, Self-Confidence, Character, Spirituality. Indian values (on the conceptual framework of Vedas): Purusharth, Niskama karma, Religion and Human Values, Towards a World Religion, Ethical Living and Harmony in Life.	
UNIT-II	**10 Hours**
Ethics and Engineering Profession Profession and Professionalism, Ethical Theories: Kohlberg's Theory, Gilligan's Theory, Moral Dilemmas, Types of Enquiry, Uses of Ethical Theories, Engineering Profession, Engineering Professionals- Training, Skill Set, Life Skills, Engineering Ethics: Making Senses and Issues, Ethical Obligations of Engineers, Ethical Codes for Engineers.	
UNIT-III	**12 Hours**
Engineering as a Social Experimentation, Safety Responsibility and Rights: Engineering as experimentation, Engineers as responsible Experimenters, Concept of Safety and Risk, Engineer's Responsibility for Safety, Risk – Benefit Analysis, Case Studies: The challenger case study, The Three Mile Island, Fukushima Nuclear Disaster, Bhopal Gas Tragedy. Disaster Management, Professional Rights, Employee Rights, Intellectual Property Rights (IPRs), Human Rights and Human Responsibilities. Major Ethical Issues.	
UNIT-IV	**10 Hours**
Ethics and Global Issues Ethics in Global Scenario, Multinational corporations, Environmental ethics, computer ethics, Business Ethics. Corporate Social responsibility, Weapons Development, Research Ethics.	

Text Books	
1.	Govindarajan M., Natarajan S., Senthil Kumar V. S., "Engineering Ethics", Prentice Hall India Learning Private Limited, New Delhi, 2004.
2.	Subramaniam R., "Professional Ethics", Oxford University Press, New Delhi, 2013.
3.	Mike Martin and Roland Schinzinger, "Ethics in engineering", 4th Edition,McGraw-Hill Education 2004.
4.	RR Gaur, R Sangal, GP Bagaria, "A Foundation Course in Human values and Professional Ethics", Excel Books Pvt. Ltd, New Delhi 2009.
5.	A.N.Tripathi, "Human Values", New Age International Publishers, New Delhi, 2nd Edition, 2004.
Reference Books	
1.	B.P. Banerjee, "Foundation of Ethics and Management", Excel Books, 2005.
2.	Fleddermann, Charles D., "Engineering Ethics", Pearson Education. 2004.
3.	Harris, Charles E., Protchard, Michael S. And Rabins, Michael, J., Wadsworth, "Engineering Ethics- Concepts and Cases", Thompson Learning, 2000
4.	Boatright, John R., "Ethics and the Conduct of Business", Pearson Education, New Delhi, 2003.
5.	Swami Ranganathananda, "Universal Message of the Bhagavad Gita: An exposition of the Gita in the light of modern thought and modern needs", Vol. I – III, Advaita Ashrama (Publication Department), Kolkata. 2000.
6.	Peter Singer, "Practical Ethics", Oxford University Press, 1993

Design And Analysis of Algorithms

Course Code: MCA-201 **Contact Hours:** L-3 T-0 P-2 **Course Category**: DCC	**Credits:** 4 **Semester** 3

Introduction: Algorithms play a crucial and fundamental role in computer science. Given that algorithms are present in all domains of computer science, it is important for students to learn techniques to analysis a given algorithm. In addition, different approaches to design algorithms are important to write one's own algorithm.

Course Objectives:

- Introduction, learning and analysis of performances of algorithmic efficiency of approaches such as searching, sorting etc.
- Introduction, learning and analysis of greedy paradigms.
- Introduction, learning and analysis of dynamic programming and back tracking.
- Introduction, learning and analysis of computational complexity and branch & bound.

Pre-requisites: Knowledge of data structures and programming

Course Outcomes:

CO1: Understand asymptotic complexities of the algorithms and design algorithms using Divide and Conquer strategy.

CO2: Apply greedy and dynamic programming approaches for designing algorithms.

CO3: Implement various graph algorithms and design algorithms using backtracking approach and branch and bound techniques

CO4: Implement different string-matching algorithms and understand the concept of NP-complete problems.

Pedagogy: The teaching-learning of the course would be organized through lectures, tutorials, assignments, projects/ presentations and quizzes. Faculty members strive to make the classes interactive so that students can correlate the theories with practical examples for better understanding. Use of ICT, web-based resources as well as flipped class room teaching will be adopted.

UNIT – I **10hrs**

Introduction to Algorithms: Need for algorithm, Growth of Functions, Exercises based on Asymptotic Notations, Solving Recurrence Relations – Iterative method, Substitution method & Master method. Space vs Time Complexity Tradeoff.

Divide and Conquer Technique: Merge Sort, Quick Sort, Median and Order Statistics, Maximum-subarray Problem, Strassen's Matrix Multiplication.

UNIT – II **10hrs**

Dynamic Programming: Elements of Dynamic Programming, Matrix Chain Multiplication, Longest Common Subsequence, 0/1 Knapsack and Optimal Binary Search Tree problems.

Greedy Algorithms: Elements of Greedy strategy, Activity Selection problem, Huffman Codes, 0/1 Fractional Knapsack, Task Scheduling problem.

UNIT – III **10hrs**

Graph Algorithms: Representation of Graphs, Breadth First Search, Depth First Search, Topological Sort, Strongly Connected Components, Algorithm for Kruskal's and Prim's for finding Minimum cost Spanning Trees, Dijkstra's and Bellman Fort Algorithm for finding Single source shortest paths. All pair shortest paths and matrix multiplication, Floyd – Warshall algorithm for all pair shortest paths.

UNIT - IV **10hrs**

String Matching: The naïve String-Matching algorithm, The Rabin-Karp Algorithm, String Matching with finite automata, The Knuth-Morris Pratt algorithm.**NP-Completeness:** Polynomial-time verification, NP-Completeness and Reducibility, NP- Completeness Proof, NP-Complete problems.

TEXT BOOKS:

1. T.H. Cormen, C.E. Leiserson, R.L. Rivest and C. Stein, "Introduction to Algorithms" PHI, 3rd Ed.
2. Jon Kleinberg and Eva Tardos, "Algorithm Design", Pearson Edition.

REFERENCE BOOKS:

1. Johnsonbaugh, "Algorithms", Pearson.
2. Anany Levitin, "Introduction to the Design and Analysis of Algorithm", Pearson Education.
3. Sara Baase and Allen Van Gelder, "Computer Algorithms - Introduction to Design and Analysis", Pearson Education.
4. A.V. Aho, J. E. Hopcroft and J.D.Ullman, "The Design and Analysis of Computer Algorithms", Pearson Education.
5. R. S. Salaria, Khanna, "Data Structure & Algorithms", Book Publishing Co. (P) Ltd.
6. R. Panneerselvam, "Design and Analysis of Algorithm", PHI.
7. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, "Fundamental of Computer Algorithms", Orient Longman.

Cloud Computing

Course Code: MCA-203 **Contact Hours**: L-3 T-0 P-2 **Course Category**: DCC	**Credits:** 4 **Semester:** 3

Introduction: Cloud computing is a scalable service provider platform that provides on-demand and pay per use computing service for various types of shared pool of resources such as memory, servers, storage, networking, software, database, applications designing etc., with the help of the internet. This course will introduce various aspects of cloud computing including fundamentals of cloud computing, load balancing techniques, security challenges, case studies and industrial applications of cloud computing. This will help students to use and explore the cloud computing platforms.

Course Objectives:

- To learn the use of various cloud computing services and cloud deployment models.
- Understand the concept of virtualization in cloud computing.
- To apply the concepts of cloud computing for designing, evaluating, simulating and comparing various applications in a cloud computing environment.
- To gain the confidence in resource management and load balancing algorithms in a cloud computing environment.
- To gain the confidence of security attacks and their provisions at various levels of cloud computing.

Prerequisite: Basic understanding of Operating System.

Course Outcomes:

CO1: To articulate key concepts of cloud computing and computing techniques, strength and limitations of cloud computing with possible application domains.

CO2: To identify the architecture and infrastructure of cloud computing including SaaS, PaaS, IaaS, public cloud, private cloud and hybrid cloud.

CO3: To interpret various data, scalability and cloud services to acquire efficient database for cloud storage.

CO4: To explain the core issues of cloud computing such as security, privacy and interoperability and deal with controlling mechanism for accessing sage cloud service.

Pedagogy: The teaching-learning of the course would be organized through lectures, tutorials, assignments, projects/ presentations and quizzes. Faculty members strive to make the classes interactive so that students can correlate the theories with practical examples for better understanding. Use of ICT, web-based resources as well as flipped class room teaching will be adopted.

UNIT – I	**10hrs**
Introduction: Introduction of cloud computing, History of cloud computing, NIST definition, properties and characteristics, Cloud as green and smart, Cloud as IaaS, PaaS, Saas, BPaaS, HaaS, Public, Private, Hybrid and community cloud, Benefits and Challenges, Application availability, performance, security and disaster recovery; next generation Cloud Applications, Technology providers vs. Cloud providers vs. Cloud vendors .	
UNIT - II	**10hrs**
Cloud Architecture: Virtualization concept, cloud building blocks, ROI Model, Service models, deployment models, storage models, security model. **Introduction to IaaS:** Resource Virtualization, Server, Storage, Network **Introduction to PaaS:** Cloud platform & Management, Computation, Storage. **Introduction to SaaS:** Web services, Web 2.0, Web OS. **Cloud Storage Infrastructure:** Storage strategy and governance; security and regulations Storage Network Design: Architecture of storage, analysis and planning. Storage network design considerations, Cloud Optimized Storage, Designing backup/recovery solutions	
UNIT – III	**10hrs**
Cloud issues and challenges: Cloud provider Lock-in, Security challenges and approaches (Infrastructure security, Network level security, Host level security, Application-level security, Data security and Storage, Data privacy and security Issues, Jurisdictional issues raised by Data location, Identity & Access Management, Access Control, Trust, Reputation, Risk Authentication in cloud computing, Client access in cloud, Cloud contracting Model, Commercial and business considerations.	
UNIT – IV	**10hrs**
Application Development: Service creation environments to develop cloud-based applications, Development environments for service development; Amazon, Azure, Google App, Salesforce.com, IBM Cloud, Google MapReduce, Yahoo Hadoop, Eucalyptus, Nimbus, OpenStack.	
TEXT BOOKS:	
1. Barrie Sosinsky, “Cloud Computing Bible”, Wiley-India 1st edition, 2011	
2. Gautam Shroff, “Enterprise Cloud Computing Technology Architecture Applications” Cambridge University Press 1st edition, 2010	
3. Greg Schulz, “Cloud and Virtual Data Storage Networking”, Auerbach Publications, 1st edition, 2009	
REFERENCE BOOKS:	
1. Miller Michael, “Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online”, Pearson Education India ,1st edition, 2008,	
2. Ronald L. Krutz, Russell Dean Vines, “Cloud Security: A Comprehensive Guide to Secure Cloud Computing”, Wiley-India 1st edition, 2010	
3. Rajkumar Buyya, James Broberg, Andrzej M. Goscinski, “Cloud Computing: Principles and Paradigms”, Wiley-India , 2011.	

Database Management System

Course Code: MCA-205 **Contact Hours**: L-3 T-0 P-2 **Course Category:** DCC	**Credits** 4 **Semester** 3

Introduction: Database Management System (DBMS) is used for creating and managing the databases. The main aim of a DBMS is to supply a way to store-up and retrieve the desired database information as per the application requirement, which is both convenient and efficient.

Course Objectives:

- Describe the fundamental elements of relational database management systems, relational data model, entity-relationship model, relational database design, relational algebra and SQL.
- To design relational databases by applying normalization techniques to normalize the database.
- Strong practice in SQL programming through a variety of database problems.
- Understand the needs of database processing and learn techniques for controlling the consequences of concurrent data access.

Pre-requisites: Basic concepts of set theory

Course Outcomes:

CO1: To have a high-level understanding of major DBMS components and their functions.

CO2: To model an application's data requirements using conceptual modeling tools like ER diagrams and design database schemas based on the conceptual model.

CO3: To develop structured query language (SQL) queries to create, read, update, and delete relational database data.

CO4: To understand the concept of Transaction, concurrency and Query processing.

Pedagogy: The teaching-learning of the course would be organized through lectures, tutorials, assignments, projects/ presentations and quizzes. Faculty members strive to make the classes interactive so that students can correlate the theories with practical examples for better understanding. Use of ICT, web-based resources as well as flipped class room teaching will be adopted.

UNIT I	**10 hrs**
Introduction: Database system concepts and its architecture, Data models schema and instances, Data independence and database language and interface, Data definition languages, DML. Overall database structure. **Data modeling using Entity Relationship Model:** ER model concept, notation for ER diagrams mapping constraints, Keys, Concept of super key, candidate key, primary key generalizations, Aggregation, reducing ER diagrams to tables, extended ER model. **Relational Data Model and Language:** Relational data model concepts, integrity constraints, Keys domain constraints, referential integrity, assertions, triggers, foreign key.	
UNIT II	**12 hrs**
Relational algebra, relational calculus, SQL Queries, SQL Functions, Nested Queries, Joins, Advanced Queries, Views, Indexing, Sequence, Grant and Revoke, Materialized View, Introduction to PL/SQL	
UNIT III	**10 hrs**
Data Base Design: Functional dependencies, normal forms, 1NF, 2NF, 3NF and BCNF, multi-valued dependencies fourth normal form, join dependencies and fifth normal form. Inclusion dependencies, lossless join decompositions, normalization using FD, MVD and JDs, Denormalization.	
UNIT IV	**10 hrs**
Transaction processing concepts: Transaction processing system, schedule and recoverability, Testing of serializability, Serializability of schedules, conflict & view serializable schedule, recovery from transaction failures, deadlock handling. **Concurrency Control Techniques:** Locking Techniques for concurrency control, time stamping protocols for concurrency control, concurrency control in distributed systems. Multiple granularities and multi-version schemes.	
Text Books	
1. Elmasri Ramez and Navathe Shamkant, Fundamentals of Database System, Pearson, 7th Ed. (June 2017)	
2. Abraham Silberschatz, Henry F. Korth, S. Sudarshan, Database System Concepts, McGraw Hill, 7th Ed(2019)	
Reference Books	
1. Ceri and Pelagatti, Distributed Databases : Principles & Systems, McGraw-Hill, 2017.	
2. Conolly & Begg, Database Management Systems, Pearson Education Asia., 5th Edition, 2010	

Web Technologies

Course Code: MCA-207 **Contact Hours:** L-3 T-0 P-2 **Course Category:** DCC	**Credits** 4 **Semester** 3

Introduction: This course aims at introducing the fundamental of internet and concepts of web technology**.**

Course Objectives:

- To understand the basics of Internet and the Web phenomena.
- To create the web pages and essential areas of developing the website.
- To introduce PHP language for server-side scripting
- To introduce XML and processing of XML Data
- To introduce Client-side scripting with Javascript and AJAX

Pre-requisites: Basic knowledge of programming.

Course Outcome:

CO1: Understand the basic terminology of web and to implement CSS and HTML in web development.

CO2: Design and Explain the basic concept of XML and Create XML documents and Schema.

CO3: Develop web-based application using suitable client side and server-side web technologies

CO4: Develop a web-based portal to provide requisite services to the users.

Pedagogy: Students will design web pages using static and dynamic pages, with the introduction on clientside and server-side programming. Emphasis on developing web applications. The teaching-learning of the course would be organized through lectures, tutorials, assignments, projects/ presentations and quizzes. Faculty members strive to make the classes interactive so that students can correlate the theories with practical examples for better understanding. Use of ICT, web-based resources as well as flipped class room teaching will be adopted.

UNIT-I	**10 hrs**
Web Basics- Introduction, Concept of Internet- History of Internet, World Wide Web, URL, Understanding websites and Web Server, Web Browser. **Introduction to HTML:** HTML overview, Basics of HTML Document, HTML tags, HTML Elements, HTML Attributes, Tables, Frames, Creating Forms, Images, Multimedia, Links, Application of HTML, HTML examples **Separating style from structure with style sheets:** Inline style specification and internal style specifications within html, external linked style specification using CSS.	
UNIT-II	**10hrs**
Introduction to XML: XML vs. HTML, uses of xml, simple xml, xml key components, DTD and schemas, well formed, XML trees, XML Namespace, XML examples, using xml with application, XSL and XSLT. Introduction to XSL, XML transformed simple example, XSL elements, transforming with XSLT **Client-side programming:** Introduction to JavaScript, JavaScript programming, variables, functions, conditions, loops, JavaScript object model, event handling, forms handling, cookies, hidden fields, images, applications.	
UNIT-III	**10 hrs**
DHTML: Combining HTML, CSS and Javascript, DHTML document object model (DOM) **Server side programming:** PHP introduction, Declaring variables, data types, arrays, strings,operators, expressions, control structures, functions, Reading data from web form controls like textboxes, radio buttons, lists etc., Handling File Uploads, Connecting to database (MySQL asreference), executing simple queries, handling results, Handling sessions and cookies File Handling in PHP: File operations like opening, closing, reading, writing, appending, deleting etc. on text andbinary files, listing directories	
UNIT-IV	**10 hrs**
Introduction to AJAX: Introduction, AJAX Database, Working of AJAX with PHP, Ajax PHP Database Form, AJAX PHP MySQL Select Query **Web services:** components and working of web services, web services architecture, introduction to service-oriented architecture, overview of web analytics and web mining	
Text Books	
1. Deitel and Deitel, Internet and World Wide Web, How to Program, Pearson Edu., 5th Ed., 2011	
2. Luke Welling and Laura Thomson, PHP and MySQL Web Development, Pearson Education; Fifth Edition (2016)	
3. Raj Kamal, Internet and Web Technologies, McGraw Hill, 2017	
Reference Books	
1. Wendy Willard, HTML: A Beginner’s Guide, McGraw-Hill Education; 5th Edition (2013)	
2. Anders Moller, Michael Schwartzeach, An Introduction to XML and Web Technologies, Pearson, 2009	

General Elective Course	
Course Code: GEC-201 **Contact Hours**: L-0 T-0 P-4 **Course Category**: GEC	**Credits** 2 **Semester** 3

Introduction: A Generic Elective (GE) course is an inter-disciplinary course provided to the students chosen generally from an unrelated discipline/subject and allowing them a chance at comprehensive education. GEs are introduced as part of the CBCS. The students can choose their preference from a pool of courses from various disciplines/subjects. Elective courses do much more than filling in the gaps to fulfill the high school graduation requirements. It gives a chance to explore new options, allowing students to study more about the subject they are passionate about, and enables them to 'test drive' new activities. They provide students with the necessary skills to improve creativity that they might not find in the classroom. The main purpose of the elective course is to seek exposure to a new discipline/subject and to provide the students with an alternative option for desired fields.

Course objectives:

- Students will have exposure to a new discipline/subject.
- Prepare students to look for inter-disciplinary research.
- Fulfill the limitation to pursue master's study in desired field.
- Help discover new things that never existed and might change the course of student's life.

Prerequisite: Basic knowledge of the selected domain of elective course

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

CO1: Identify new discipline and learn new subject for future careers.

CO2: Apply their knowledge to understand and solve the real-life problems.

CO3: Analyse creative design process through the integration and application of diverse technical knowledge and expertise to address social issues.

CO4: Develop the habit of working independently to attain self-motivation, discipline, and confidence to achieve their goals.

Industrial Training/Internship	
Course Code: MCA 253 **Contact Hours**:- **Course Category:** DCC	**Credits** 1 **Semester** 3

Introduction: Students will carry on the industrial training/internship for at least six weeks in the summer break of previous academic session. The idea of the training is to make them capable of handling the implementation of their theoretical knowledge in the practical field. To facilitate the development of a holistic perspective among students towards life, industry experts teach advanced technologies. Through Industrial training, students get familiarize with the environment of an organization and a company. Students get a certificate which validates their skills and helps them in getting a job quickly. The assessment for the same will be done within the first two weeks of opening of academic session by the respective department.

Course Outcomes

CO1: Understand the Organizational Structure of a company.

CO2: Develop work habits and attitudes necessary for job success (technical competence, professional attitude, organization skills etc.)

CO3: Develop written communication and technical report writing skills.

CO4: Develop an awareness for the need and applications of standards in the industry.

Java Programming

Course Code: MCA-202 **Contact Hours**: L-3 T-0 P-2 **Course Category:** MCA	**Credits** 4 **Semester** 4

Introduction: Java Programming is one of the most widely used programming language among developers and are preferred over other languages. This course introduces students to object-oriented design methods and GUI like Applet, swing, AWT etc. The objective is to provide students with the use of the Java programming language for writing complex and stand-alone applications at the Intermediate level.

Course Objectives

- To provide knowledge of Object-Oriented programming features and fundamentals of program development using java.
- Students will learn how to write, test, and debug Object-Oriented programs using Java and learn advanced concepts.

Pre-requisites: The student may have experience in a high-level programming language such as C/C++.

Course Outcome: After completion of course, students will able to:

CO1: Understand object-oriented concepts and use the concepts of inheritance, polymorphism, interfaces, packages with exception handling reusable Java programs.

CO2: Identify operations commonly used to implement thread-based applications, network-based application, file I/O operations, and exception handling

CO3: Implement simple GUI interfaces for a computer program to interact with users and understand the event handling

CO4: Understand Servlet and implement programs using JDBC

Pedagogy: The teaching-learning of the course would be organized through lectures, tutorials, assignments, projects/ presentations and quizzes. Faculty members strive to make the classes interactive so that students can correlate the theories with practical examples for better understanding. Use of ICT, web-based resources as well as flipped class room teaching will be adopted.

UNIT-I	10 hrs
Overview of java: Class Fundamentals: introduction of classes, objects and methods using program example, creating objects and object reference, object lifetime and garbage collection, Arrays and String: Creating an array, one- and two-dimensional arrays, String, String Buffer and String Builder classes, Constructors, Class inheritance, use of super, Multilevel hierarchy, Abstract Class and final classes, Object class **Packages and interfaces:** Extending Interfaces, Organizing Classes and Interfaces in Packages, Package as Access Protection, Defining Package, CLASS PATH Setting for Packages. **Exception Handling:** Exception types, uncaught exceptions, try-catch, throw, throw and finally, built in exception, Creating your own exceptions **Multithreaded Programming:** Life Cycle of Thread, Creating and running thread, Multiple thread synchronization, Thread communication, Thread group, Thread priorities, Daemon Thread, suspending, resuming and stopping threads.	
UNIT-II	**11 hrs**
The Collection Framework(java.util): The Collection Interface, Collection Classes, Working with Maps & Sets, Wrappers classes **Networking(java.net):** Networking concepts, using java.net package, networking classes and interfaces, socket programming, TCP/IP client and server sockets **RMI (Remote Method Invocation):** Introduction, Steps in creating a Remote Object, Generating Stub & Skeleton, RMI Architecture, RMI packages **Input/Output Programming and file operations(java.io):** Java.io, Byte and Character Stream, predefined streams, Reading and writing from console and files	
UNIT-III	**10 hrs**
Applet, Event handling and AWT: Applet design, parameters to applets, Event Handling: Different Mechanism, the Delegation Event Model, Event Classes, Event Listener Interfaces, Applet event handling, Adapter and Inner Classes, AWT packages, Components and Containers, using AWT controls, Layout managers, AWT components, Adding menu to window **Swing:** Introduction to JFC (Java Foundation Classes), Features of Swing and Comparison with AWT, Advanced Control in swing (JTree and JTable)	
UNIT-IV	**11 hrs**
JDBC packages: Introduction to JDBC, Types of JDBC drivers, obtaining a Connection, Connection, statement, ResultSet, Prepared Statement, Callable Statement, Program example using JDBC. **Servlets:** Using Servlets - Servlet Package - Servlet lifecycle - init() , method - service() method , doGet() method, doPost() method **Java Bean:** Introduction, Bean Architecture, Using the Bean Development Kit, creating simple bean-properties, methods and events, Packing beans- the manifest & the jar, Java bean package, Introduction to NetBean.	
Text Books	
1. The Complete Reference Java,, Herbert Schildt, ISBN: 978-0-07163177-8, Publisher: McGraw Hill(7th Edition)	
2. Thinking in Java, Bruce Eckel, ISBN: 0-13-187248-6, Publisher: Prentice Hall, 4th Edition	
Reference Books	
1. The Java Programming Languages, Ken Arnold, ISBN-13:978-032134980, Publisher: Sun,4th edition	

2. Paul Dietel and Harvey Deitel, “Java How to Program”, PHI, 8th Ed., 2010.

3. Java in Nutshell, Benjamin,ISBN: 9781449371296, Publisher: O’Reilly Media, Inc.,6th edition

Artificial Intelligence

Course Code: MCA-204 **Contact Hours:** L-3 T-0 P-2 **Course Category**: DCC	**Credits** 4 **Semester** 4

Introduction: "AI is the new electricity" -Andrew Ng. This course aims to give the fundamental knowledge and practical skills needed to design, build, and apply AI systems in one's chosen area of specialization. This course is an introduction to the basic Knowledge representation, problem solving and learning methods of artificial intelligence. After learning this course, the student should be able to understand the basic concepts of problem solving and learning in intelligent system engineering.

Course Objectives:

- To learn the meaning behind common AI terminology
- To understand what AI realistically can--and cannot—do
- To spot opportunities to apply AI to problems in your own organization

Course Outcomes: On successful completion of this course, the students should be able to:

CO1: Understand basic terminology of modern AI frameworks.
CO2: Understand, and implement problem solving agents in AI.
CO3: Learn to understand decision making systems.
CO4: Understand and apply learning-based agents.

Pedagogy: Students will analyse and design AI applications in Python using hands-on, engaging activities. At the end of each Unit, example application/case study will be discussed and relevant research paper reading will be carried out.

<table>
<tr><td><b>UNIT-I</b></td><td><b>8 hrs</b></td></tr>
<tr><td colspan="2">AI terminology, data, workflow of a data science project, what makes a company good at AI, Bias in AI, adversarial attacks on AI, AI application areas, tools and techniques, what AI can and cannot do, AI and developing economies, AI team and job functions, case studies: smart speaker and self-driving car</td></tr>
<tr><td><b>UNIT-II</b></td><td><b>12 hrs</b></td></tr>
<tr><td colspan="2">Search: Formalism, BFS, DFS, Uninformed Search, A* and Heuristics, Adversarial Search, CSP: Constraint Satisfaction, Local Search, and Optimization, Logic: Ontology, Propositional Logic, First order predicate logic, resolution, fuzzy logic, case study: restaurant tip planner</td></tr>
<tr><td><b>UNIT-III</b></td><td><b>12 hrs</b></td></tr>
<tr><td colspan="2">Uncertainty, Probabilistic Reasoning Systems, Making Simple Decisions, Making Complex Decisions, Markov Decision Processes: Bayesian Networks: Representation, Independence, Inference, Markov Models, Hidden Markov Models, case study: search string completion</td></tr>
<tr><td><b>UNIT-IV</b></td><td><b>10 hrs</b></td></tr>
<tr><td colspan="2">Learning: Learning from Observations, inductive learning, active learning, decision trees, statistical learning: learning with complete data (naïve Bayes), instance-based learning (nearest neighbour), learning with hidden variables (clustering), learning in Neural and Belief Networks, Reinforcement Learning, case study: malware detection</td></tr>
<tr><td colspan="2"><b>Text Books</b></td></tr>
<tr><td colspan="2">1. Elaine Rich, Kevin Knight, Shivashankar B Nair: Artificial Intelligence, McGraw Hill 3rd Edition. 2017
2. Parag Kulkarni, Prachi Joshi, Artificial Intelligence: Building Intelligent Systems, Prentice Hall India Learning Private Limited; 1st Edition (2015)</td></tr>
<tr><td colspan="2"><b>Reference Books</b></td></tr>
<tr><td colspan="2">1. S. Russell and P. Norvig, Artificial Intelligence: A modern approach, Pearson Education, 3rd Edition, 2015</td></tr>
<tr><td colspan="2">2. Online resources: AI for everyone, Andrew Ng, Coursera, https://www.coursera.org/learn/ai-for-everyone</td></tr>
</table>

Data Communications and Computer Networks

Course Code: MCA-206 **Contact Hours:** L-3 T-0 P-2 **Course Category**: DCC	**Credits** 4 **Semester** 4

Introduction: Data communications refers to the transmission of this digital data between two or more computers and a computer network or data network is a telecommunications network that allows computers to exchange data. The physical connection between networked computing devices is established using either cable media or wireless media. The best-known computer network is the Internet.

Course Objectives:

- The students should understand the layers of networking devices.
- They should be familiar with a few networking protocols.
- They should study the different types of networks and topologies of networks.

Pre-requisite: Data Structures and Algorithms

Course Outcomes:

CO1: Describe the fundamental concepts and layered architecture of computer networking.

CO2: Explain the basic concepts of link layer properties to detect error and develop the solution for error control and flow control. Design, calculate, and apply subnet masks and addresses to fulfill networking requirements. Also, compare various routing protocols.

CO3: Comprehend the duties of transport layer and congestion control techniques.

CO4: Illustrate the features and operations of various application layer protocols such as DNS, HTTP, FTP, e-mail protocols and other applications; and focus on network security issues to secure communication towards society.

Pedagogy: The teaching-learning of the course would be organized through lectures, tutorials, assignments, projects/ presentations and quizzes. Students would be encouraged to develop an understanding of the existing real life cyber security issues and how they are solved. Emphasis would be given on assignments where students will be given numerical/ programming assignments based on topics studied in previous lectures. Course will have a blend of theory and practice for the benefit of students. Use of ICT, web-based sources as well as blackboard teaching will be adopted.

UNIT-I	**10 hrs**
Introduction: Goals and Applications of Networks, Layering Concept, OSI Reference Model, TCP/IP Protocol Suite, Networks Topology, **Physical Layer:** Signals, Digital Transmission – Analog to Digital & Digital to Digital, Analog Transmission – Digital to Analog & Analog to Analog, Multiplexing – FDM & TDM, Media – Guided and Unguided, Switching – Packet based & Circuit based, Shannon Capacity; Network Topologies, Connecting Devices	
UNIT -II	**11 hrs**
Data Link Layer: Addressing, Error Detection & Correction, Checksum & CRC; Medium Access – ALOHA, CSMA, CSMA/CD & CA; Protocols – Ethernet, ARP & RARP; Switching Techniques. **Network Layer:** Need for internetworking, IP Addressing, Subnetting, Supernetting, Basic Routing (or Forwarding) Mechanism; IPv4 frame format and functions; Key features of IPv6, ICMP, IGMP, Routing protocols – RIP, OSPF & BGP and algorithms – Distance Vector and Link State. Linux Network Commands: arp, route, ifconfig, netstat, traceroute, ping.	
UNIT-III	**11 hrs**
Transport Layer: Port Addresses; ARQ - Simple, Stop and Wait, Go Back-N, Selective Repeat; UDP – Services & Applications; TCP – header format, connection setup & termination, state transition diagram, flow control, error control, Congestion Control: causes for congestion, effects of congestion, various open-loop and close-loop congestion control techniques: The leaky bucket algorithm, The token bucket algorithm	
UNIT -IV	**10 hrs**
Application Layer: Web & HTTP, FTP, Email, Telnet, SSH, DNS, RPC. Advanced Protocols: SNMP, RTP, SIP, BitTorrent.	
Text Books	
1. L. L. Peterson and B. S. Davie, Computer Networks: A Systems Approach, Fifth Edition, Elsevier, 2011.	
2. A. S. Tanenbaum and D.J. Wetherall, Computer Networks, Fifth Edition, Pearson, 2013.	
3. B. Forouzan, Data Communications and Networking, Fifth Edition, Mcgraw Hill, 5th Edition, 2017	
References Books	
1. Respective Internet Drafts and RFCs of IETF.	
2. William Stallings, “Data and Computer Communications”, PHI 6th Edition	

Software Testing

Course Code: MCA-301 **Contact Hours:** L-3 T-0 P-2 **Course Category**: DCC	**Credits:** 4 **Semester**: 5

Introduction: Software testing is a course based on knowledge dissemination of investigating software's to ensure that its quality under test is in line with the requirements of the client. This course will introduce the students to a number of techniques to design and analyze test cases, teach them how software testing is carried out in a systematic manner with the intent of finding defects and evaluating the systems. It helps students in solving computational problems across a variety of areas in testing software.

Course Objective:

- To understand that software testing is a fundamental part of the software life cycle.
- To learn the essential theories, types, tools, and methods of software testing
- To learn about various software testing problems.

Course Outcomes:

CO1: Understand the fundamental concepts of a testing.

CO2: Derive test cases using black box and white box testing strategies.

CO3: Generate and prioritize test cases to prove the correctness of program and understand levels of testing and object-oriented testing and web testing

CO4: Understand verification methods, verification of documents, testing metrics and quality models to improve the quality of software.

Pedagogy: The teaching-learning of the course would be organized through lectures, tutorials, assignments, projects/ presentations and quizzes. Faculty members strive to make the classes interactive so that students can correlate the theories with practical examples for better understanding. Use of ICT, web- based resources as well as flipped classroom teaching will be adopted.

UNIT I	10 hours
Introduction: Testing Objectives, Software Testing Process, Software Testing Principles, Tester Role in Software Development Organization, Test Case Implementation and Execution. **Testing Concepts:** Levels of Testing, Test Cases Design and Strategy, Test Suit, Test Plan, testing as a Process, Testing and Debugging, Limitations of Testing**. Software Testing Tools:** Characteristics of Modern Tools, Static Testing Tools, Dynamic Testing Tools, Process Management Tools.	
UNIT II	**10 hours**
Functional Testing: Boundary Value Analysis, Robustness Testing, Worst Case Testing, Special Value Testing, Equivalence Class, Testing-Weak Normal, Strong Normal, Weak Robust and Strong Robust, Decision Table Based Testing, Cause Effect Graphing Technique. **Structural Testing:** Control Flow Testing-Statement, Branch, Condition and Path Coverage, Data Flow Testing, Testing Strategies, Generation of Test Cases, Slice-Based Testing, Mutation Testing, Integration Testing, Decomposition Based Integration, Call Graph Based Integration, System Testing: Thread Testing.	
UNIT III	**10 hours**
Introduction to Object Oriented Testing, State Based Testing, Class Testing, Web Testing, Issues in Object Oriented Testing, Regression testing, Selection of test cases, Reducing the number of test cases, Prioritization guidelines.	
UNIT IV	**10 hours**
Software Verification Methods, SRS Verification, SDD Verification, Source Code Reviews, Software Project Audit, Debugging Process and Approaches, Software Testing Metrics, Metrics used in Testing, Software Quality and Quality Models.	
Text Books	
1. Y. Singh, "Software Testing", Cambridge University Press, 1st Edition, 2011/Latest edition.	
2. P. C. Jorgensen, "Software Testing: A Craftsman's Approach", Auerbach Publications, 4th edition, 2013/Latest edition.	
Reference Books	
1. Burnstein, "Practical Software Testing: A Process-Oriented Approach", Springer, 1st edition, 2003/Latest edition.	
2. A. P. Mathur, "Foundations of Software Testing", Pearson, 2nd edition, 2013/Latest edition	
3. J. A. Whittaker, "How to Break Software: A Practical Guide to Testing", Pearson, 1st edition, 2002/Latest edition	
4. B. Beizer, "Software Testing Techniques", Itp – Media**,** 2nd edition,1990/Latest edition.	

General Elective Course

Course Code: GEC-301 **Contact Hours**: L-0 T-0 P-4 **Course Category:** GEC	**Credits** 2 **Semester** 5

Introduction: A Generic Elective (GE) course is an inter-disciplinary course provided to the students chosen generally from an unrelated discipline/subject and allowing them a chance at comprehensive education. GEs are introduced as part of the CBCS. The students can choose their preference from a pool of courses from various disciplines/subjects. Elective courses do much more than filling in the gaps to fulfill the high school graduation requirements. It gives a chance to explore new options, allowing students to study more about the subject they are passionate about, and enables them to 'test drive' new activities. They provide students with the necessary skills to improve creativity that they might not find in the classroom. The main purpose of the elective course is to seek exposure to a new discipline/subject and to provide the students with an alternative option for desired fields.

Course objectives:

- Students will have exposure to a new discipline/subject.
- Prepare students to look for inter-disciplinary research.
- Fulfill the limitation to pursue master's study in desired field.
- Help discover new things that never existed and might change the course of student's life.

Prerequisite: Basic knowledge of the selected domain of elective course

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

CO1: Identify new discipline and learn new subject for future careers.

CO2: Apply their knowledge to understand and solve the real-life problems.

CO3: Analyse creative design process through the integration and application of diverse technical knowledge and expertise to address social issues.

CO4: Develop the habit of working independently to attain self-motivation, discipline, and confidence to achieve their goals.

Industrial Training/Internship

Course Code: MCA 353 **Course Category:** DCC	**Credits** 1 **Semester** 5

Introduction: Students will carry on the industrial training/internship for at least six weeks in the summer break of the previous academic session. The idea of the training is to make them capable of handling the implementation of their theoretical knowledge in the practical field. To facilitate the development of a holistic perspective among students towards life, industry experts teach advanced technologies. Through Industrial training, students get familiarized with the environment of an organization and a company. Students get a certificate which validates their skills and helps them in getting a job quickly. The assessment for the same will be done within the first two weeks of opening of academic session by the respective department.

Course Outcomes

CO1: Understand the Organizational Structure of a company.

CO2: Develop work habits and attitudes necessary for job success (technical competence, professional attitude, organization skills etc.)

CO3: Develop written communication and technical report writing skills.

CO4: Develop an awareness for the need and applications of standards in the industry.

Computer Graphics and Multimedia Technologies

Course Code: MCA-208 **Contact Hours:** L-3 T-0 P-2 **Course Category:** DCC	**Credits** 4 **Semester** 4

Introduction: Computer graphics is an art of drawing pictures, lines, charts, etc. using computers with the help of programming. Computer graphics is made up of number of pixels. Pixel is the smallest graphical picture or unit represented on the computer screen. In this course, students will learn fundamental concept and algorithms of computer graphics and multimedia.

Course Objectives:

- To learn the fundamental concepts of graphics and multimedia.
- To gain and apply the acquired knowledge pertaining to 2D and 3D concepts in graphics programming.
- To understand the basic 3D modelling and rendering techniques.
- To realize the importance of multimedia towards building the virtual environment and communication.

Pre-requisites: Nil

Course Outcomes: Upon successful completion of the course, students will be able to:

CO1: Enumerate the functionalities of pixels and coordinate systems pertaining to graphics manipulation.

CO2: Design and demonstrate the 2D and 3D objects using graphics algorithms.

CO3: Have the ability to model and render 3D objects by comprehending the complexities of illumination in virtual scenes.

CO4: Appraise and interpret the various multimedia communication standards, applications and basic principles.

Pedagogy: The teaching-learning of the course would be organized through lectures, tutorials, assignments, projects/ presentations and quizzes. Faculty members strive to make the classes interactive so that students can correlate the theories with practical examples for better understanding. Use of ICT, web-based resources as well as flipped class room teaching will be adopted.

UNIT I	10 hrs

Scan Conversion Algorithms: Scan Converting Lines (DDA, Bresenham), Scan Converting Circles (Mid-point, Bresenham), Scan Converting Ellipses (Midpoint). Clipping: Two-Dimensional Clipping, Sutherland-Cohen Subdivision Line-Clipping Algorithm 2D- Transformation: Representation of Points, Transformations and Matrix, Transformation of Straight Line, 2-D - Rotation, Reflection, Scaling, Combined Transformations, Translation and Homogeneous Coordinates, Translation, Rotation about an Arbitrary Point, Reflection through an Arbitrary Line, window-to-viewport transformation

UNIT II	12 hrs

3D-Transformation: Representation of Points, 3D- Scaling, 3D- Shearing, 3D- Rotation, Three-Dimensional Translation, 3D- Reflection, Multiple Transformations, Rotation about an Axis Parallel to a Coordinate Axis, Rotation about an Arbitrary Axis in Space. The Dimensional Perspective Geometry: Geometric Projection, Orthographic Projections, Oblique Projections, Perspective Transformations, Single-Point Perspective Transformation, Two-Point Perspective Transformation, Three-Point Perspective Transformation. Solid Modeling: Representing Solids, Regularized Boolean Set Operation primitive Instancing Sweep Representations, Boundary Representations, Spatial Partitioning Representations, Constructive Solid Geometry, Comparison of Representations.

UNIT III	10 hrs

Representing Curves & Surfaces: Polygon meshes, parametric, Cubic Curves, geometric and parametric continuities, Hermite, Bezier (4-point, 5-point, general), B-Spline, Quadric Surface Illumination and Shading: Modeling light intensities, ambient light, diffused light, specular reflection, attenuation factor, Reflection vector, Shading Models: constant shading, flat shading, gouraud shading, phong shading. Hidden-Surface Removal: Hidden Surfaces and Lines, Back-Face Detection, A-buffer, ZBuffers Algorithm, Scan-line Algorithm, The Painter's Algorithm, Area subdivision Introduction to Multimedia: Multimedia, Multimedia Terms, Introduction to making multimedia – The Stages of project, the requirements to make good multimedia, Multimedia Applications.

UNIT IV	10 hrs

IV Multimedia – making it work – Multimedia Hardware, Software and Authoring Tools, Graphics File Formats: TIFF, MIDI, JPEG, MPEG, RTF. Multimedia building blocks – Text, Sound, Images, Animation and Video, Digitization of Audio and Video objects, Data Compression: Different Compression algorithms concern to text, audio, video and images etc.

Text Books:

1. Steve Marschner, Peter Shirley, Fundamentals of Computer Graphics, CRC Press, 4th Ed. (2015)
2. D.Hearn & Baker: Computer Graphics, Prentice Hall of India
3. Foley, Van Dam, Feiner, Hughes, “Computer Graphics Principles & Practice” Tay Vaughan, “Multimedia: Making it Work”, TMH

Reference Books

1. K. Andleigh and K. Thakkar, “Multimedia System Design”, PHI, PTR
2. Rogers & Adams, “Mathematical Elements for Computer Graphics”, McGraw Hill

Soft Computing

Course Code: MCA 210 **Contact Hours**: L-3 T-0 P-2 **Course Category**: DEC	**Credits** 4 **Semester** 4

Introduction:_This course aims at introducing the fundamental theory and concepts of computational intelligence methods, in particular neural networks, fuzzy systems, genetic algorithms and their applications in the area of machine intelligence.

Course Objectives:

- To provide an introduction to the basic principles, techniques, and applications of soft computing.
- To provide an understanding of the basic areas of Soft Computing including Artificial Neural Networks, Fuzzy Logic and Genetic Algorithms.
- To provide the mathematical background for carrying out the optimization associated with neural network learning.
- To develop some familiarity with current research problems and research methods in Soft Computing by working on a research or design project.

Prerequisite: Artificial Intelligence, Data Structures and Algorithms, Programming languages.

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

CO1: Understand basic concepts of neural networks.

CO2: Understand the fuzzy logic concepts.

CO3: Learn and understand genetic algorithms.

CO4: Understand the concepts of differential evolution.

Pedagogy: The teaching-learning of the course would be organized through lectures, tutorials, assignments, projects/ presentations and quizzes. Faculty members strive to make the classes interactive so that students can correlate the theories with practical examples for better understanding. Use of ICT, web-based resources as well as flipped class room teaching will be adopted.

UNIT I	**10 hrs**
Introduction of soft computing, soft computing vs. hard computing, various types of soft computing techniques, Bayesian Networks, Probabilistic reasoning, Neural Networks: NN vs ANN, Learning networks of ANN – Perceptron's - Adaline – Back Propagation, Multilayer Perceptron – Radial Basis Function Networks – Unsupervised Learning Neural Networks – Competitive Learning Networks, Hebbian Learning.	
UNIT II	**10 hrs**
Fuzzy Set Theory: Fuzzy set theory, Fuzzy set versus crisp set, Crisp relation & fuzzy relations, introduction & features of membership functions, Extension Principle, Fuzzy If-Then Rules, Fuzzy Inference Systems, Sugeno Fuzzy Models, Fuzzification, Defuzzification, Applications, Fuzzy clustering, cluster validity measures.	
UNIT III	**10 hrs**
Genetic Algorithm: Difference between Traditional Algorithms and GA, The basic operators, Schema theorem, convergence analysis, stochastic models, applications in search and optimization. Encoding, Fitness Function, Reproduction, Cross Over, Mutation.	
UNIT IV	**10 hrs**
Differential Evolution, Hill Climbing, Tabu Search, Cuckoo Search, Harmony Search, PSO, ACO, Bat algorithm, Artificial Bee Colony optimization, meta heuristic algorithms: applications to solve complex problems.	
Text Books	
1. S. N. Sivanandam and S. N. Deepa, "Principles of Soft Computing", Wiley – India, 2nd Edition, 2011/ Latest Edition.	
2. S. Rajasekaran, "Neural Networks, Fuzzy Systems and Evolutionary Algorithms: Synthesis and Applications", PHI Learning, 2nd Edition, 2017/ Latest Edition.	
Reference Books/Materials	
1. N. P. Padhy and S.P. Simon, "Soft Computing techniques with MATLAB programming", Oxford University Press, UK Edition, 2015/ Latest Edition.	
2. X. Wang, X. Z. Gao and K. Zenger, "An introduction to harmony search optimization method", Springer International Publishing, 2015/ Latest Edition.	
3. R. Lowen and A. Verschoren, "Foundations of generic optimization: Volume 2: Applications of fuzzy control, genetic algorithms and neural networks", Springer Science & Business Media,2008/ Latest Edition.	
4. S. Kaushik and S. Tewari, "Soft Computing", McGraw Hill Education, 1st Edition, 2018/ Latest Edition.	
5. https://nptel.ac.in/courses/106/105/106105173/	

Cyber Security and Forensics

Course Code: MCA-212 **Contact Hours**: L-3 T-1 P-0 **Course Category:** DEC	**Credits** 4 **Semester** 4

Introduction: Cyber Security and Forensics is the application of investigation and analysis techniques to gather and preserve evidence from a particular computing device in a way that is suitable for presentation in a court of law. This course provides for a broad introduction of cyber security and forensics concepts, industry best practices for information security and key security concepts that will protect an organization against fraud, data breaches and other vulnerabilities. It enables the students to gain in-depth knowledge in the field of Computer forensics & Cyber Crime.

Course Objectives:

- To maintain an appropriate level of awareness, knowledge and skill to allow students to minimize the occurrence and severity of information security incidents.
- To learn techniques used to detect, respond and prevent network intrusions.
- To identify and apply appropriate forensics tools to acquire, preserve and analyze system image.
- To protect information and information systems from unauthorized access, use, disclosure, disruption, modification or destruction in order to provide confidentiality, integrity and availability.
- Identify sources of evidentiary value in various evidence sources including network logs, network traffic, volatile data.

Pre-requisites: Knowledge of Computer Networking, Linux, UNIX, Understanding of Web Application Architecture and HTTP/HTTPS communication.

Course Outcomes:

CO1: Understand the fundamentals of Cyber Security and comprehend the incident response process

CO2: Demonstrate the difference between data acquisition techniques

CO3: Apply forensic analysis tools to recover important evidence for identifying cyber-crime.

CO4: Apply investigation tools and techniques for analysis of data to identify evidence related to cyber-crime and use available digital forensics tools.

Pedagogy: The teaching-learning of the course would be organized through lectures, tutorials, assignments, projects/ presentations and quizzes. Students would be encouraged to develop an understanding of the existing real life cyber security issues and how they are solved. Emphasis would be given on assignments where students will be given numerical/ programming assignments based on topics studied in previous lectures. Course will have a blend of theory and practice for the benefit of students. Use of ICT, web-based sources as well as blackboard teaching will be adopted.

UNIT-I	**12 hrs**
Introduction to Incident Response Process, Computer Security Incident, Goals of Incident response, Who is involved in Incident response, Incidence Response Methodology, Pre-Incident preparation, Detection of Incidents, Initial response, Formulate a response strategy, Investigate the incident, Reporting and Resolution. Computer Forensics Fundamentals, Benefits of Computer Forensics, Computer Crimes, legal concerns and private issues. Live data collection from Windows systems. Live data Collection from Unix systems.	
UNIT-II	**11 hrs**
Data Acquisition of digital evidence from electronic media, Acquisition tools, Evidence collection and preservation, Sources of Digital/Electronic Evidence, Computer Forensic Analysis and Validating Forensics Data, **System Forensics:** File signatures, volatile/non-volatile data, File formats, Metadata, existing system forensics tools. **Network Forensics**: Firewalls, Intrusion Detection System. **Database Forensics**.	
UNIT-III	**10 hrs**
Windows Forensics: malware forensics**. Mobile Device Forensics**: Evidence in Cell Phone, PDA, Blackberry, iPhone, iPod, and MP3. Evidence in CD, DVD, Tape Drive, USB, Flash Memory, Digital Camera. **Google Forensics**: Analysis of search data/information gathered from various google services. **Internet Forensics**.	
UNIT-IV	**10 hrs**
Email Analysis: investigating email crime and violations. **Messenger Analysis**: AOL, Yahoo, MSN, and Chats. **Web investigation**: IP tracking, Server logs, Domain records. **Current Computer Forensics Tools**: Software/Hardware Tools. **An Indian perspective on digital forensics**: Indian IT act, Cyber laws.	
Text Books	
1. K Mandla, C. Prosise , Matt Pepe, “ Incident Response and Computer Forensics”, McGraw Hill, 2nd Edition, 2003	
2. Chad Steel, “Windows Forensics”, Wiley India, 1st Edition, 2006	
3. Nelson, B, Phillips, A, Enfinger, F, Stuart, C., “Guide to Computer Forensics and Investigations, Thomson Course Technology, 4th Edition, 2009	
Reference Books	
1. Keith J. Jones, Richard Bejtiich, Curtis W. Rose, Real Digital Forensics, Pearson Education, 1st Edition, 2005	
2. Computer Forensics, Computer Crime Investigation by John R. Vacca, Firewall Media, New Delhi	

Software Project Management

Course Code: MCA-214 **Contact Hours**: L-3 T-0 P-2 **Course Categor:** DEC	**Credits** 4 **Semester** 4

Introduction: This course is designed to enable students to learn successful development of the software project's procedures of initiation, planning, execution, regulation and closure as well as the guidance of the project team's operations towards achieving all the agreed upon goals within the set scope, time, quality and budget standard.

Course Objectives:

- To learn Software Project management phases.
- Creating a project plan and implementing the plan to achieve the project goal.
- Learn how to plan, evaluate, and schedule components, resources, and durations of action project programs.

Pre-requisite: None

Course Outcome: Upon successful completion of this course, students will be able to:

CO1: Understand various project contexts and management approach.

CO2: Understand project management principles and methods in an IT project.

CO3: Understand key phases of project management.

CO4: Determine an appropriate project management approach, project control and risk management

Pedagogy: The teaching-learning of the course would be organized through lectures, tutorials, assignments, projects/ presentations and quizzes. Faculty members strive to make the classes interactive so that students can correlate the theories with practical examples for better understanding. Use of ICT, web-based resources as well as flipped class room teaching will be adopted.

UNIT-I	**11 hrs**
Introduction: Introduction to software project management activities, Attributes of a project, Project life cycle, Project Management process, Project selection, Preparing a request for proposal, Soliciting proposals, Proposal preparation, Pricing considerations, Proposal submission and follow up, Customer evaluation of proposals	
UNIT-II	**10 hrs**
Project Management Organizational Structures: Functional type organization, Project type organizations, Matrix-type organization, Project Planning - Project objective, Work breakdown structure, Developing the network plan, Network principles, Preparing the network diagram, Critical path analysis, PERT ,Project Scheduling- Activity duration estimates, Project schedule calculations	
UNIT-III	**10 hrs**
Schedule Control: Project control process, Effects of actual schedule performance, Incorporating project changes into the schedule, Updating the project schedule, Approaches to schedule control, Resource Considerations- Resource constrained planning, Planned resource utilization, Resource leveling, Resource limited scheduling	
UNIT-IV	**11 hrs**
Risk Management: Risk, Categories of risk, A framework for dealing with risk, Evaluating risks to the schedule, Monte Carlo simulation and critical chain concepts. Project Cost Planning and Performance – Project cost estimates, Project budgeting, Determining the actual cost, Determining the value of work performed, Cost performance analysis, Cost forecasting, Cost control, Software project metrics, Project control and closure, Project Management Issues with regard to New Technologies, Case Study & use of software project management tool	
Text Books	
1. Pankaj Jalote, “Software Project Management in Practice”, Pearson Education, 2015	
2. Jack Gido, Jim Clements, Rose Baker, “Successful Project Management”, Cengage Learning 7th Edition, 2018	
3. Hughes, Software Project Management, McGraw Hill Education; 5th Edition, 2017	
Reference Books	
1. Bob Hughes, Mike Cotterell, Rajib Mall “Software Project Management”, Fifth Edition, McGraw Hill, 2013	

Network Security	
Course Code: MCA 305 **Contact Hours**: L- 3 T- 0 P-2 **Course Category:** DEC	**Credits:** 4 **Semester**: 5

Introduction:

This course will introduce students to the basic building blocks of cryptography and applications of cryptographic protocols in real world and network security. The intent of this course is to familiarize students with security threats, cryptography, and application development in computer network protocols. The focus will be on how cryptography and its applications can maintain privacy and security in electronic communications and computer networks.

Course Objective:

- To understand the fundamentals of Cryptography.
- To acquire knowledge on standard algorithms used to provide confidentiality, integrity and authenticity.
- To explain and use modern cryptographic methods (symmetric encryption, public key encryption, hash functions, key management, digital signatures, certificates etc).
- To discuss various network security protocols.

Pre-requisite: None

Course Outcome: Upon successful completion of this course, students will be able to:

CO1: Understand applied cryptographic basics and apply to real world problems.
CO2: Select the right algorithm, protocol, and systems to develop secure systems to protect digital assets in the cyber world.
CO3: Apply the knowledge of the number theory in understanding the cryptosystems and designing the new cryptosystems with defined security requirements based on computationally hard problems.
CO4: Gain the knowledge of the algebraic structures that will enable them to work around in designing cryptosystems.

Pedagogy:

The teaching-learning of the course would be organized through lectures, tutorials, assignments, projects/ presentations and quizzes. Students would be encouraged to develop an understanding of the existing real life network security issues and how they are solved. Emphasis would be given on assignments where students will be given numerical/ programming assignments based on topics studied in previous lectures. Course will have a blend of theory and practice for the benefit of students. Use of ICT, web based sources as well as blackboard teaching will be adopted.

UNIT-I	**10 hours**
Introduction and terminology, Conventional Cryptography: Definitions, Classical encryption techniques, Substitution and Transposition Cipher, Vignere Cipher, Introduction to security attacks, services and mechanism, Security Overview, CIA model, Security Policies and Mechanisms, Threats, Block Ciphers and Stream Ciphers, Block ciphers principles, Shannon's theory of confusion and diffusion, Fiestal Structure, Data Encryption Standard (DES), Cryptanalysis of DES, Triple DES.	
UNIT-II	**11 hours**
Group, Abelian and Cyclic group, Ring, Finite Fields Advanced Encryption Standard (AES), Modes of Encryption: ECB, CBC, CFB, Counter mode, Message Padding, Asymmetric Cryptography: Number Theory, Modular Arithmetic, Fermat's and Euler's theorem, primarily testing, Euclid's Algorithm, Chinese Remainder theorem, discrete logarithms, public key cryptography: RSA, ElGamal, and Elliptic Curve Cryptography, Diffie Hellman Key management, Meet-in-the-Middle Attack, Digital Certificates: X.509.	
UNIT-III	**11 hours**
Digital Signatures, Stream Ciphers, LFSR based stream ciphers, Hash functions, Hash algorithms (MD5, SHA-2, Kecchak), Message Authentication Codes, CBC-MAC, HMAC, NMAC, Authentication Protocols: Kerberos, password, challenge-response, biometric authentication, electronic mail security-pretty good privacy (PGP), S/MIME, Malicious Logic, Trojan Horses, Defenses, Viruses, Worms Logic Bombs, Sandboxing.	
UNIT-IV	**10 hours**
IP Security: Architecture, Authentication header, Encapsulating security payloads, combining security associations, key management, Web Security: Secure Socket Layer(SSL) and transport layer security, TSP, Secure Electronic Transaction (SET), Electronic money, firewall design principals, Virtual Private Network (VPN) security.	
Text Books	
1	W. Stallings, "Cryptography and Network Security: Principles and Practice", 7th Edition, Prentice Hall, 2017.
2	B. Forouzan, D.Mukhopadhyay, "Cryptography and Network Security", 3rd Edition, McGraw Hill Education, 2015.
3	M.Bishop, "Introduction to Computer Security", 3rd Edition, Addison-Wesley Professional, 2005.
4	B. Menezes, "Network Security and Cryptography", 2nd Edition , Delmar Cengage Learning, 2012.
Reference Books	
1	A.Menezes, P.Oorschot, S.Vanstone, "Handbook of Applied Cryptography", Hardcover Edition, CRC press, 2018.
2	R. Stinson, M. Paterson, "Cryptography: Theory and Practice", 4th Edition, CRC Press, 2018.
3	C.Paar, J.Pelzl, "Understanding Cryptography: A textbook for students and practitioners", 1st Edition, Springer, 2010.

Advanced DBMS

Course Code: MCA-307 **Contact Hours**: L-3 T-0 P-2 **Course Category**: DEC	**Credits** 4 **Semester** 5

Introduction: This course will help the students to sharpen their DBMS skills in more depth. This course describes in major details about the advanced concepts of database management systems including advanced SQL, handling unstructured data, Query execution, database security and various database models.

Course Objectives:

- To sharpen the skills on writing complex and effective queries
- To handle unstructured data by using No-SQL and MongoDB
- To understand the query execution plan
- To design and implement Distributed Databases.

Prerequisite: Basic DBMS concepts and any Programming Language.

Course Outcome: On successful completion of the course, the students will be able to:

CO1: Understand and apply the concept of Advanced SQL in processing the data in large database.

CO2: Design a database for managing structured data items.

CO3: Analyze and apply emerging technologies such as Big Data, NoSQL, and MongoDB for handling unstructured data.

CO4: Apply Distributed Database Management System for handling multimedia data in distributed environment.

Pedagogy: The teaching-learning of the course would be organized through lectures, tutorials, assignments, projects/ presentations and quizzes. Faculty members strive to make the classes interactive so that students can correlate the theories with practical examples for better understanding. Use of ICT, web- based resources as well as flipped classroom teaching will be adopted.

UNIT I	**10 hrs**
Advanced SQL: Joins (Outer, Inner and Self Join), Nested Queries, Views, Indexes, Materialized Views, Embedded SQL, dynamic SQL, SQLJ, Cursor, Exception Handling, Triggers, Procedures, Functions.	
UNIT II	**10 hrs**
Indexing and Hashing, B+ Tree Index Files, B-Tree Index Files, Dynamic and Static Hashing, Query Processing, Measures of Query cost, Selection Operation, Sorting, Join operation, evaluation of expressions, Query Optimization, estimating statistics of expression results, transformation of Relational Expressions, Choice of evaluation plans, Database Security and Authorization: Levels of database security, Access control, Multilevel security, Statistical database security, Audit trails in the databases	
UNIT III	**10 hrs**
Structured versus Unstructured data, NoSQL database concepts: Types of NoSQL databases, NoSQL data modeling, Benefits of NoSQL, comparison between SQL and NoSQL database system. NoSQL using MongoDB: Introduction to MongoDB Shell, Running the MongoDB shell, MongoDB client, Basic operations with MongoDB shell, Basic Data Types, Arrays, Embedded Documents Querying with MongoDB: find () function, specifying which keys to return, query criteria, OR queries, Types specific querying, Aggregation Introduction: Aggregation Pipeline, Aggregation using Map reduce, Single purpose aggregation	
UNIT IV	**10 hrs**
Distributed Databases, Homogeneous and Heterogeneous Databases, Distributed Data Storage, Distributed Transactions and their commit protocols, Concurrency Control in Distributed Data Bases, Distributed Query Processing. Multimedia Databases, Mobile Data bases, Temporal database, Image and Semantic-based query processing, Active database	
Text Books	
1. 1. E. Ramez and S. B. Navathe, "Fundamentals of Database System", Pearson, 7th Edition, 2016/Latest Edition.	
2. 2. A. Silberschatz, H. F. Korth, S. Sudarshan, "Database System Concepts, McGraw Hill, 6th Edition, 2013/Latest edition.	
Reference Books	
1. Ceri and Pelagatti, "Distributed Databases: Principles & Systems", McGraw-Hill, 2nd Edition, 2017/Latest edition.	
2. Conolly and Begg, "Database Management Systems", Pearson Education Asia, 5th Edition, 2010/Latest edition.	
3. R. Ramakrishnan, J. Gerkhe, "Database Management Systems", McGraw Hill Publications, 3rd Edition, 2014/Latest edition.	
4. 4. W. Lemahieu, S. Broucke, B.Baesens,"Principles of Database Management: Practical Guide to Storing, Managing and Analyzing Big and Small Data", Cambridge University press, 1st edition, 2018/Latest edition.	

E-Commerce	
Course Code: MCA-309 **Contact Hours:** L-3 T-0 P-2 **Course Category**: DEC	 **Credits** 4 **Semester** 5

Introduction: E-commerce is abbreviated for Electronic Commerce. Its function is the transference of financial and other commerce related information using Information Technology and Telecommunications. E- commerce helps to simplify the business processes and makes them faster and efficient. These business transactions occur either as business-to-business (B2B), business-to-consumer (B2C), consumer-to- consumer (C2C) or consumer-to-business (C2B). Benefits of e-commerce include its around-the-clock availability, the speed of access, the wide availability of goods and services for the consumer, easy accessibility and international reach.

Course Objectives:

- To understand the advantages and disadvantages of using e-commerce platforms.
- To learn various e-business strategies.
- To understand the various payment methods associated with e-commerce.
- To learn the concepts of security at various levels of e-commerce.

Prerequisite: Knowledge on the basics of Information Security, Networking.

Course Outcome: Upon successful completion of this course, students will be able to:

CO1: Understand the basic concepts and principles of e-commerce.

CO2: Analyze the various e-business strategies.

CO3: Develop digital payment software for e-commerce applications.

CO4: Apply the knowledge of mobile technology for commercial aspects.

Pedagogy: The teaching-learning of the course would be organized through lectures, tutorials, assignments, projects/ presentations and quizzes. Faculty members strive to make the classes interactive so that students can correlate the theories with practical examples for better understanding. Use of ICT, web-based resources as well as flipped classroom teaching will be adopted.

UNIT I	**10 hrs**
Electronic Commerce Introduction: - Definition of E-Commerce, Electronic commerce and Physical Commerce, Architectural framework, Impact of E-commerce on business, different type of e-commerce, some e-commerce scenario, Economic potential of electronic commerce, Advantages and Disadvantages, Incentives for engaging in electronic commerce, forces behind E-Commerce.	
UNIT II	**10 hrs**
E-business strategy: Introduction, Characteristics of e-Business, Business models, E-Business vs E-commerce, e-business Requirements, impacts of e-business, Strategic positioning, Levels of e- business strategies, Strategic planning process, Success factors for implementation of e-business strategies, CRM, MRP. ERP: Introduction, need of ERP, Modules of ERP.	
UNIT III	**10 hrs**
Electronic Payment Methods: Overview, SET Protocol for credit card payment, E-cash, E-check, Micropayment system, Credit card, Magnetic strip card, Smart cards, Electronic Data Interchange, E-Commerce Law. Security Architecture, Encryption techniques, Symmetric & Asymmetric encryption, Digital Signatures, Virtual Private Network, IPsec, Threats, Firewalls.	
UNIT IV	**10 hrs**
M-Commerce: Introduction, Attributes, customer and provider views, Architecture, Infrastructure of m- commerce, Requirement of the m-commerce, characteristics, Mobile Information device, Mobile Computing Applications, Mobile wallet, Mobile payments, Mobile portals, Pros and Cons of m-commerce, Secure Transaction Processes: Wireless Application Protocol, Bluetooth, The role of emerging wireless LANs and 3G/4G wireless networks.	
Text Books	
1. R. Kalakota, A. Whinston, "Frontiers of Electronic Commerce", Addison Wesley, 3rd Edition, 1996/ Latest edition.	
2. B. Mennecke and T. Strader, "Mobile Commerce: Technology, Theory and Applications", Idea Group, 2003/Latest edition	
Reference Books	
1. D. Chaffey, "E-Business and E-Commerce Management", Pearson Education, 3rd Edition, 2009/Latest edition.	
2. 2. H. Chan, "E-Commerce Fundamentals and application", Wiley publication, 1st Edition, 2001/Latest edition.	
3. Bajaj and Nag, "E-Commerce the cutting edge of Business", TMH, 2nd Edition, 2005/Latest edition.	
4. P. Loshin, J. Vacca, "Electronic commerce", Firewall Media, 1st Edition, 2005/Latest edition	

Software Quality Assurance

Course Code: MCA-311 **Contact Hours:** L-3 T-1 P-0 **Course Category:** DEC	**Credits** 4 **Semester** 5

Introduction: This course introduces concepts, metrics, and models in software quality assurance. The course covers components of software quality assurance systems before, during, and after software development. It presents a framework for software quality assurance and discuss individual components in the framework such as planning, reviews, testing, configuration management, and so on. It also discusses metrics and models for software quality as a product, in process, and in maintenance. The course will include case studies and hands on experiences. Students will develop an understanding of software quality and approaches to assure software quality.

Course Objectives:

- Understand the basic tenets of software quality and quality factors.
- Be exposed to the Software Quality Assurance (SQA) architecture and the details of SQA components.
- Understand of how the SQA components can be integrated into the project life cycle.
- Be familiar with the software quality infrastructure.
- Be exposed to the management components of software quality.

Prerequisite: General knowledge of Software Engineering and Software development life cycle.

Course Outcome: Upon successful completion of this course, the students will be able to:

CO1: Understand the concepts, metrics and models in software quality.

CO2: Analyze the capability of software product to adopt quality standards.

CO3: Develop a model to assess the quality of software product.

CO4: Apply the concepts in preparing the quality plan & documents.

Pedagogy: The teaching-learning of the course would be organized through lectures, tutorials, assignments, projects/ presentations and quizzes. Faculty members strive to make the classes interactive so that students can correlate the theories with practical examples for better understanding. Use of ICT, web- based resources as well as flipped classroom teaching will be adopted.

UNIT I	**10 hrs**
Introduction: The Software Quality Challenge, what is Software Quality? Definition and objectives, need of Software Quality, Software Quality Factors (AQF): Classification of software requirements into SQF, McCall's quality model, Quality metrics, Quality trade-offs. SQA components: SQA system, SQA Architecture, Pre-project software quality components, Quality Assurance activities, SQA plan	
UNIT II	**10 hrs**
Software project life cycle components: Verification and Validation, model for SQA defect removal effectiveness and cost. Infrastructure components for error prevention and improvement, Development and quality plans, Software reliability models, Reviews: Review's objectives, Formal design reviews (DRs)Software Quality metrics: Objectives of quality management, Classification of Software quality metrics, Process metrics, Product metrics, Implementation of Software quality metrics, limitations	
UNIT III	**10 hrs**
Software quality Infrastructure: Procedures and work instructions, Templates, Checklists, 3S development, Staff training and certification Corrective and preventive actions, Configuration management, Software change control, Configuration management audit, Documentation control, Storage and retrieval. Management SQA components: Assuring the quality of software maintenance components, Software quality of external participants contribution: Objectives, SQA tools, CASE tools and their effect on Software quality, CMM and CMMI assessment methodology.	
UNIT IV	**10 hrs**
Cost of software quality: objectives, classic model of cost of software quality, Applications and problems, SQA Standards, certification and assessment: Software management standards-Scope, ISO 9001, 9000-3, ISO 9126 Standard, IEEE std 1028-reviews, Considerations guiding construction of an organization's SQA system. Future of SQA: Challenges and Capabilities. Risks of distinct quality assurance processes in modern software development companies (e.g., the impact of choosing among different testing techniques)	
Text Books	
1. D. Galin, "Software Quality Assurance", Pearson Publication, 2nd Edition, 2009/Latest Edition.	
2. S. H. Kan, "Metrics and Models in Software Quality Engineering", Pearson Education, 2nd Edition, 2003/ Latest Edition.	
Reference Books	
1. A. C. Gillies, "Software Quality: Theory and Management", International Thomson Computer Press, 1997/Latest edition.	
2. M. Ben-Menachem "Software Quality: Producing Practical Consistent Software", International Thompson Computer Press, 1997/Latest edition.	
3. G. Blokdyk, "Software QA Complete Self-Assessment Guide", 5STARCook's publishers, 1st edition, 2018/Latest edition.	
4. L. Iancu, "QA Quality Assurance & Software Testing Fundamentals", KDP, 1st Edition, 2019/Latest edition.	

Advanced Data Structures

Course Code: MCA-315 **Contact Hours**: L-3 T-0 P-2 **Course Category**: DEC	**Credit**s 4 **Semester** 5

Introduction: This course builds upon the introductory courses in data structures. It introduces students to a number of highly efficient data structures for solving data driven computational problems across a variety of areas.

Course Objectives:

- To impart knowledge of computational and advanced concepts of Data structures and algorithms.
- To understand concepts about searching algorithms, lists, graphs and trees.
- To understand about writing algorithms and sequential approach in solving problems with advanced Data structures

Prerequisite: Knowledge of fundamentals of Data Structures, Algorithms and Analysis.

Course Outcome:

CO1: Define advanced highly efficient data structures and their properties.

CO2: Understand the concept of space and time complexity and compare the efficiency of algorithms.

CO3: Apply the advanced highly efficient data structures to solve computational problems.

CO4: Design and employ network flow algorithms to solve real world problems.

Pedagogy: The teaching-learning of the course would be organized through lectures, tutorials, assignments, projects/ presentations and quizzes. Faculty members strive to make the classes interactive so that students can correlate the theories with practical examples for better understanding. Use of ICT, web- based resources as well as flipped classroom teaching will be adopted.

UNIT I	**10 hrs**
Review of data structures: Arrays, Stacks, Linked Lists, Queues. Hash tables – collision resolution, Hash functions, Open addressing. Dictionary. Data Frames and operations. Multi-dimensional Arrays (NumPy) and operations.	
UNIT II	**10 hrs**
Binary trees and their properties, threaded binary trees - differentiation, leftist trees, tournament trees, use of winner trees in merge sort as an external sorting algorithm, bin packing, Binary search trees, search efficiency, insertion and deletion operations, importance of balancing, AVL trees, searching, insertion and deletions in AVL trees, Tries, 2-3 tree, B-tree	
UNIT III	**10 hrs**
Review of Graphs – DFS and BFS, MST, Shortest Path – Single Source and All Pair. Degree Distribution, Paths, Distances, Connectedness, Clustering Coefficient, Random Networks – Evolution, Small World, Barabasi-Albert Model.	
UNIT IV	**10 hrs**
Network Flow: Max-Flow problem, Ford-Fulkerson algorithm, Augmenting paths, Bipartite Matching problem, Applications: Airline Scheduling, Image Segmentation. Evolving Networks: Bianconi-Barabasi Model.	
Text Books	
1. A. Aho, J. Ullman, J. Hopcroft.,"Data Structures and Algorithms", Pearson Education India, 1st Edition, 2002/Latest edition	
2. J. Kleinberg and E. Tardos. "Algorithm Design", Pearson Publication, 1st Edition, 2005/Latest edition	
Reference Books	
1. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, "Introduction to Algorithms", MIT Press, 3rd Edition, 2009/Latest edition.	
2. Al. Barabasi. "Network Science", Cambridge University Press, 2016/Latest edition.	
3. P. Brass, "Advanced Data Structures", Cambridge University Press, 1st Edition, 2008/Latest edition.	
4. T. Cormen, C. Leiserson, R Rivest, C. Stein, "Introduction to Algorithms", MIT Press, 3rd Edition, 2009/Latest edition.	

Theory of Computation

Course Code: MCA-317 **Contact Hours:** L-3 T-1 P-0 **Course Category**: DEC	 **Credits** 4 **Semester** 5

Introduction: The study of automata and the theory of computation deal with the concepts of working of automatic machines and processing of input formal language data. This subject provides an important background material to students involved in understanding the basic functionalities of automata theory.

Course Objectives:

- Introduce students to the mathematical foundations of computation including automata theory; the theory of formal languages and grammars; the notions of algorithm, decidability, complexity, and computability.
- Identify different formal language classes and their relationships
- Design grammars and recognizers for different formal languages

Prerequisite: Strong background in Discrete mathematics, Data structures and algorithms

Course Outcome:

CO1: Describe the Finite Automata, their capabilities and limitations.
CO2: Classify the different types of grammars, languages and machines.
CO3: Discover the equivalence of languages described by finite state machines and regular expressions.
CO4: Design the FA, CFG, Push Down Automata and Turing recognizable languages.

Pedagogy: The teaching-learning of the course would be organized through lectures, tutorials, assignments, projects/ presentations and quizzes. Faculty members strive to make the classes interactive so that students can correlate the theories with practical examples for better understanding. Use of ICT, web- based resources as well as flipped classroom teaching will be adopted.

UNIT I	10 hrs
Introduction to Theory of Computation: Definitions: Languages, Grammar, Automata, Applications of Theory of Computation, Finite Automata: DFA, NDFA, Equivalence of DFA and NDFA, DFA Minimization Regular Languages, Regular Grammars, Properties of Regular Languages, Pumping Lemma	
UNIT II	**10 hrs**
Context Free Language: Introduction, Parsing and Ambiguity, Pushdown Automata (PDA), Non-Deterministic PDA, Context Free Grammar, Chomsky Normal Form, Greibach Normal Form, Parse Tree representation of Derivation Tree, Equivalence of PDA and CFGs, Properties of Context Free Grammars	
UNIT III	**10 hrs**
Pumping Lemmas: Pumping Lemma for context free languages, Pumping lemma for linear languages. Turing Machine: Definition, TM as language acceptors, TM as transducers, Hierarchy of Formal Languages and Automata, Chomsky Hierarchy, Context Sensitive Languages and LBA, Unrestricted Grammars	
UNIT IV	**10 hrs**
Turing machine Models and complexity, Some NP Problems, Complexity classes P and NP, Unsolvable Problem, Halting problem, Finite State Transducers: Introduction, Mealy Machines, Moore Machines, Mealy and Moore Equivalence, Limitations of Finite State transducer	
Text Books	
1. P. Linz, “An Introduction to Formal Languages and Automata”, Narosa Publishers, 4th Edition, 2013/Latest edition	
2. J. Ullman, J. Hopcroft, “Introduction to Automata Theory, Languages and Computation”, Pearson Education India, 3rd Edition, 2008/Latest edition.	
Reference Books	
1. M. Sipser “Introduction to the Theory of Computation”, Cengage, 3rd Edition, 2014/Latest edition.	
2. C. K. Nagpal, “Formal Languages and Automata Theory”, Oxford University Press, 2015/Latest edition.	
3. H. Lewis, C. H. Papadimitriou, “Elements of the Theory of Computation”, Pearson, 1st Edition, 1993/Latest edition.	
4. B. M. Moret, “The Theory of Computation”, Pearson, 1st Edition, 2002/Latest edition.	

Mobile Computing	
Course Code: MCA-319 **Contact Hours**: L-3 T-1 P-0 **Course Category**: DEC	**Credits** 4 **Semester** 5

Introduction: Mobile Computing is a technology that allows transmission of data, voice and video via a computer or any other wireless enabled device without having to be connected to a fixed physical link. Mobile Computing involves mobile communication, software and hardware.

Course Objectives:

- To introduce the basic concepts and principles in mobile computing. This includes major techniques involved, and networks as well as systems issues for the design and implementation of mobile computing systems and applications.
- To understand the basic concepts of mobile computing.
- To learn Wireless technologies and planning Ad-hoc Network.
- To understand telecommunication systems and gain knowledge about different mobile platforms and application development.

Prerequisite: Basic knowledge of Computer fundamentals and networking.

Course Outcome: Upon successful completion of this course, students will be able to:

CO1: Explain the basics concepts of mobile computing and types of mobile communication systems.

CO2: Understand the need of Mobile IP and TCP Protocol and architectures of Mobile Telecommunication System.

CO3: Illustrate the wireless network concepts and its routing protocols

CO4: Evaluate a large network-based system with network components.

Pedagogy: The teaching-learning of the course would be organized through lectures, tutorials, assignments, projects/ presentations and quizzes. Faculty members strive to make the classes interactive so that students can correlate the theories with practical examples for better understanding. Use of ICT, web- based resources as well as flipped classroom teaching will be adopted.

UNIT I	**10 hrs**
Introduction to Mobile Computing: History, Types, Benefits, Application, Evolution, Characteristics of Mobile computing, Security Concern regarding Mobile Computing, Different Propagation Modes, Wireless Architecture and its types. First-Generation Analog, Second-Generation TDMA, Second- Generation CDMA, Third-Generation Systems; **Cellular Concept:** Cellular Systems and Principles of Cellular Networks, Hexagonal geometry cell and concept of frequency reuse, Channel Assignment Strategies, Distance to frequency reuse ratio; Electromagnetic Spectrum, **Antennas and Propagation**- Antennas, Propagation Modes, Line-of-Sight Transmission, Fading in the Mobile Environment, Signal Characteristics; Channel Capacity, Multiplexing, Spread Spectrum: DSSS & FHSS, CDMA.	
UNIT II	**10 hrs**
Telecommunication Systems: GSM Architecture**,** Channel allocation, call routing, PLMN interface, GSM addresses and identifiers, network aspects, frequency allocation, authentication and security, Handoffs Technique; GPRS: network architecture, network operation, data services, Applications, Billing and charging; UTRAN, UMTS; Mobile Networking: Medium Access Protocol, Internet Protocol and Transport layer, Medium Access Control: Motivation for specialized MAC, Introduction to multiple Access techniques (MACA).	
UNIT III	**10 hrs**
Mobile IP: Features of Mobile IP and its need, IP packet delivery, Key Mechanism in Mobile IP, Agent Discovery, Registration, Tunnelling and encapsulation, Reverse Tunnelling, Routing (DSDV,DSR), Route optimization, IP Handoff; Mobile TCP: Traditional TCP, Classical TCP Improvements like Indirect TCP, Snooping TCP & Mobile TCP, Fast Retransmit/ Fast Recovery, Transmission/Timeout Freezing, Selective Retransmission; Wireless Application Protocol: Introduction, Application, Architecture, Protocol Stack and Challenges.	
UNIT IV	**10 hrs**
Bluetooth: Introduction, User Scenario, Architecture, protocol stack; IP Mobility, Macro Mobility and Micro Mobility, Introduction to 4G and 5G; LTE, HIPERLAN, Mobile Device Operating Systems, Special Constraints & Requirements, Commercial Mobile Operating Systems, Software Development Kit: iOS, Android, BlackBerry, Windows Phone, M-Commerce, Structure, Mobile Payment System.	

Text Books

1. J. H. Schiller, “Mobile Communications”, Pearson Education, 2nd Edition, 2003/Latest edition.
2. A.K. Talukder, H. Ahmed, R.R. Yavagal, “Mobile Computing: Technology, Applications and Service Creation”, McGraw Hill Education, 2nd Edition, 2017/Latest edition.

Reference Books

1. R. Kamal, “Mobile Computing”, Oxford University Press, 3rd Edition, 2018/Latest edition.
2. F. Adelstein, S.K.S. Gupta, G.G. Richard III and L. Schwiebert, “Fundamentals of Mobile and Pervasive Computing”, McGraw-Hill Professional, 1st Edition, 2004/Latest edition.
3. A. F. Molisch, “Wireless Communications”, Wiley - IEEE Press, 2nd Edition, 2010/Latest edition.
4. P. K. Pattnaik, R. Mall, “Fundamentals of Mobile Computing”, PHI Learning, 1st Edition, 2015/Latest edition.