

School of Computer & Information Sciences

Master of Computer Applications

SCHOOL OF COMPUTER & INFORMATION SCIENCES

Vision Statement:

- To invent, create and bring computing technology solutions to the common man, to the privileged and underprivileged sections of India, to bridge the digital divide and eradication of computer ignorance and digital illiteracy and to build a prosperous and technologically advanced nation.

Mission Statements:

MS-1: To pursue academic and research excellence, nationally and internationally

MS-2: To provide training, advisory, and consultancy to all the stakeholders.

MS-3: To lead the efforts in creative and newer modes of instruction delivery & supervision

School of Computer and Information Sciences

Name of the Academic Program: Master of Computer Applications

Program Educational Objectives (PEOs)

PEO-1: To train the graduates to acquire in depth knowledge of fundamental concepts and programming skills for holistic development

PEO-2: To prepare the graduates for productive careers in software industry, corporate sector and Government Organizations.

PEO-3: To prepare graduates to acquire excellent computing ability so that they can analyze, design and create Solutions for real time problems.

PEO-4: To apply the current tools and techniques to create systems for solving Industry oriented problems.

PEO-5: To prepare graduates to gain multidisciplinary knowledge through real time case studies, projects and industry internship to meet the industry needs.

Mapping Program Educational Objectives (PEOs) with Mission Statements (MS)

	MS-1	MS-2	MS-3
PEO-1			
PEO-2			
PEO-3			
PEO-4			
PEO-5			

Write '3' in the box for 'high-level' mapping, 2 for 'Medium-level' mapping, 1 for 'Low-level' mapping.

School of Computer & Information Sciences

Name of the Academic Program: Master of Computer Applications

Program Outcomes (POs)

PO-1: Computational Knowledge	Apply knowledge of computing fundamentals, computing specialisation, mathematics, and domain knowledge to conceptualise computing models
PO-2: Problem Analysis	Identify, formulate, research literature, and solve complex computing problems reaching substantiated conclusions using fundamental principles of mathematics, computing sciences, and relevant domain disciplines.
PO-3: Design/Development of Solutions	Design and evaluate solutions for complex computing problems, and design and evaluate systems, components, or processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.
PO-4: Conduct Investigation of Complex Computing Problems	Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
PO-5: Model Tool Usage	Create, select, adapt and apply appropriate techniques, resources, and modern computing tools to complex computing activities, with an understanding of the limitations.
PO-6: Professional Ethics	Understand and commit to professional ethics and cyber regulations, responsibilities, and norms of professional computing practice.
PO-7: Life Long Learning	Recognise the need, and have the ability, to engage in independent learning for continual development as a computing professional.
PO-8: Project Management and Finance	Demonstrate knowledge and understanding of the computing and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
PO-9: Communication Efficacy	Communicate effectively with the computing community, and with society at large, about complex computing activities by being able to comprehend and write effective reports, design documentation, make effective presentations, and give and understand clear instructions.
PO-10: Societal and Environmental Concern	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.
PO-11: Individual Team Work	Function effectively as an individual and as a member or leader in diverse teams and in multidisciplinary environments.
PO-12: Innovation and Entrepreneurship	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.

Mapping of Program Outcomes (POs) and Program Specific Outcomes (PSOs) with Program Educational Objectives (PEOs)

	PEO-1	PEO-2	PEO-3	PEO-4	PEO-5
PO-1					
PO-2					
PO-3					
PO-4					
PO-5					
PO-6					
PO-7					
PO-8					
PO-9					
PO-10					
PO-11					
PO-12					

Mapping of Program Specific Outcomes (PSOs) where applicable.

Write '3' in the box for 'high-level' mapping, 2 for 'Medium-level' mapping, 1 for 'Low-level' mapping.

School of Computer & Information Sciences
Master of Computer Applications
2 Year Program (w.e.f 2020-2021)

Scheme

I-Semester			II-Semester		
Code	Course Title	Credits	Code	Course Title	Credits
CA401	Theoretical Foundations of Computer Science	4-0-0	CA451	Computer Networks	3-1-0
CA402	Programming Methodology	3-0-0	CA452	Operating Systems	4-0-0
CA403	Computer Organization and Architecture	3-0-0	CA453	Data Structures	3-0-0
CA404	Computer Based Optimization Techniques	3-0-0	CA454	Object Oriented Design	3-0-0
CA405	Database Management Systems	3-0-0		Elective-I	3/4-0-0
CA406	PM Lab	0-0-2	CA455	DS Lab	0-0-2
CA407	DBMS Lab	0-0-2	CA456	OOP Lab	0-0-2
		20			21/22
III-Semester			IV-Semester		
CA501	Algorithms	3-1-0	CA551	Project/Internship	12
CA502	Software Engineering	3-0-0			
CA503	Software Project Management	3-0-0			
	Elective-II	3/4-0-0			
	Elective-III	3/4-0-0			
CA504	Software Engineering Lab	0-0-2			
CA505	Internet Technologies Lab	1-0-2			12
		21/23			

Detailed Syllabus:

UNIT-I: Sets, Relations and Functions

Sets, relations and functions; Methods of proof; Equivalence relations; Cardinality; Countable and uncountable sets

UNIT-II: Introductory Logic

Fundamentals of Logic; Logic operators such as AND, OR etc., Truth tables; Logical inferences; Methods of proofs of an implication; First order logic; Predicate calculus Predicates and Quantifiers; Rules of inference for quantified propositions

UNIT-III: Recurrence Relations:

Recursion, Forming and solving recurrence relations by substitution method and generating functions; Method of characteristic roots; solving inhomogeneous recurrence relations

UNIT-IV: Boolean Algebra:

Partial order relations; Lattices; Boolean algebra; Combinatorial circuits; Minimization of Boolean functions using Karnaugh maps

UNIT-V: Theory of Graphs

Graphs, subgraphs, isomorphism, proofs; Types of graphs; paths and cycles; Adjacency matrices; Transitive closure; Connectivity; Directed acyclic graphs; Planar graphs and Euler's formula; Dual of a graph; Hamiltonian and Eulerian graphs; Applications like matching and colouring graphs; Graph traversals (BFS and DFS); Trees; Spanning trees.

Reference Books:

1. Kenneth H Rosen (2012), *"Discrete Mathematics and Its Applications"*, 7th Edition, McGraw Hill, NY
2. Ralph P Girimaldi(2003), *"Discrete and Combinatorial Mathematics –An Applied Introduction"*, 5th Edition, Pearson Addison Wesley, Indian Edition
3. J.R Mott, A Kandel, T.P Baker (2015), *"Discrete Mathematics for Computer Scientists and Mathematicians"*, Pearson
4. Ronald L Graham, Donald E Knuth, Oren Patashnik(1994), *"Concrete Mathematics- A Foundation of Computer Science"*, 2nd Edition, Addison Wesley .
5. Susanna S. Epp(2010), *"Discrete Mathematics with Applications"*, 4th Edition, Brooks/Cole Cengage Learning.

Detailed Syllabus:

UNIT-I: Introduction to problem solving:

Problems and problem instances; Informal approach to program design: generalisation, special cases, and algorithms, breaking down a problem into functions, input and output.

UNIT-II: Introduction to the 'C' programming language:

Program structure; main() function; unnamed and named blocks; basic data types, variables, declaration and definition; initialisation and assignment; arithmetic operators and precedence; implicit and explicit type conversions; arrays; boolean variables and logical operators.

UNIT-III: Control structures:

Branching and iteration; functions and parameters; break(), return() and exit() functions; local and global variables; function prototypes.

UNIT-IV: Pointer variables and dynamic structures:

Static and dynamic (run-time) memory structures; static variables; breaking a program across multiple files; creating and linking libraries.

UNIT-V: Detecting and correcting common errors:

Debugging and debuggers; documenting programs; good programming practices; programming exercise (writing a program of at least 200 lines split across multiple files).

Reference Books:

1. Brian W. Kernighan, Dennis M. Ritchie. "The C Programming Language, 2nd Edition", Prentice-Hall India.
2. G. Michael Schneider. "Introduction to Programming and Problem Solving with PASCAL", John Wiley and Sons.
3. Paul Deitel and Harvey Deitel. "C How to Program", Pearson Education India.
4. Stephen Kochan. "Programming in C", Pearson Education India.
5. Brian W. Kernighan and R. Pike. "The Unix Programming Environment", Prentice-Hall India.
6. Chakravarthy Bhagvati. "How to Program (An Informal Guide)", <https://scis.uohyd.ac.in/~chakcs/howtoprogram.pdf>

Detailed Syllabus:

UNIT - I: Computer Evolution & Arithmetic:

A Brief History of computers, Designing for Performance, Von Neumann Architecture, Hardware architecture, Computer Components, Interconnection Structures, Bus Interconnection, Scalar Data Types, Fixed and Floating point numbers, Signed numbers, Integer Arithmetic, 2's Complement method for multiplication, Booths Algorithm, Hardware Implementation, Division, Restoring and Non Restoring algorithms, Floating point representations, IEEE standards, Floating point arithmetic

UNIT - II: The Central Processing Unit:

Machine Instruction characteristics, types of operands, types of operations, Addressing modes, Instruction formats, Instruction types, Processor organization, Processor as running example, Programmers model of , max/min mode, Register Organization, Instruction cycles, Read Write cycles, assembly instruction examples to explain addressing modes

UNIT - III: The Control Unit:

Single Bus Organization, Control Unit Operations: Instruction sequencing, Micro operations and Register Transfer. Hardwired Control: Design methods – State table and classical method, Design Examples - Multiplier CU. Micro-programmed Control: Basic concepts, Microinstructions and micro- program sequencing

UNIT - IV:

Memory Organization: Characteristics of memory systems, Internal and External Memory, Types of memories: ROM: PROM, EPROM, EEPROM, RAM: SRAM, DRAM, SDRAM, RDRAM, High-Speed Memories: Cache Memory, Organization and Mapping Techniques, Replacement Algorithms, Cache Coherence, Virtual Memory: Main Memory allocation, Segmentation, Paging, Address Translation Virtual to Physical. Secondary Storage: Magnetic Disk, Tape, DAT, RAID, Optical memory, CDROM, DVD

UNIT - V: I/O Organization:

Input/ Output Systems (features and principles), Programmed I/O, Interrupt Driven I/O, Interrupt structure, Direct Memory Access (DMA), features Buses and standard Interfaces: Synchronous, Asynchronous, Parallel I/O features, Serial I/O features, PCI, SCSI, USB Ports Working mechanisms of Peripherals: Keyboard, Mouse, Scanners, Video Displays, Touch Screen panel, Dot Matrix, Desk-jet and Laser Printers.

UNIT - VI:

Case Studies: Concepts RISC: Instruction execution characteristics,, RISC architecture and pipelining. RISC Vs CISC. ARM and Embedded Systems PowerPC, Intel X86 Evolution from 32bit to 64bit architectures. AMD Opteron

Reference Books

1. Patterson D.A. & Hennesy J.L., *Computer Organisation & Design: The Hardware/Software Interface*.
2. Computer Organization and Architecture, 10/E William Stallings ISBN-10: 0134101618 • ISBN-13: 9780134101613- See more at:
http://www.pearsonhighered.com/pearsonhigheredus/educator/product/products_detail.page?isbn=9780134101613&forced_logout=forced_logged_out#sthash.WVVJbZUb.dpuf

Detailed Syllabus:

UNIT-I:

Reference Books:

1. W.

Detailed Syllabus:

UNIT 1: Introduction: Aims and Objectives, Technology involved and current uses of the technology.

UNIT 2: Data Models: Entity-Relationship model, Network model, Hierarchical model.

UNIT 3: Database design: Normalization principles and their uses. Secondary data storage and retrieval techniques.

UNIT 4: Query Processing: Studies on query processing strategies and cost estimation.

UNIT 5: Transaction Processing: Defining Properties and studies on recovery and concurrency. Security and Integrity.

UNIT 6: Distributed Databases: Introduction, Issues on design, concurrency, recovery, deadlock handling and coordinator selection.

Reference Books:

1. A. Silberschatz, H. F. Korth and S. Sudarshan, *Database Systems & Concepts*, 6th Edition McGrawHill Publications, 1376 pages.
2. R. Elmasri, S. B. Navathe: *Fundamentals of Database Systems*, 7th Edition, Pearson Publication, US, 1168 pages.
3. Stefano Ceri, G. Pellagatti: *Distributed Databases Principles & Systems*, McGrawHill, India, 408 pages.

Detailed Syllabus:

Unit 1: Introduction to SQL. Features of SQL, DDL Statements and DML commands.

Unit 2: Writing simple SQL queries using DDL statements and DML commands

Unit 3: Introduction to inner, outer and natural joins. Writing nested queries and correlated nested queries to retrieve and update the data.

Unit 4: Writing SQL queries using EXISTS, NOT EXISTS, explicit join operation, aggregate functions, group by and having clauses.

Unit 5: Creating virtual tables (views). Using views in SQL queries.

Unit 6: PL/SQL programming

Unit 7: Mini project: Implement a real-time web application which makes use of database concepts

Reference Books:

1. R. Elmasri, S. B. Navathe: *Fundamentals of Database Systems*, 7th Edition, Pearson Publication, US, 1168 pages.
2. Raghu Ramakrishnan, Johannes Gehrke: *Database management systems*, McGrawHill, Singapore, 1098 pages.

Detailed Syllabus:

UNIT - I: Physical Layer: Modulation Techniques: Amplitude, Frequency and Phase, ADSL, Multiplexing Techniques: Frequency division multiplexing, Time division multiplexing, wave length division multiplexing, Differential PCM, Switching Techniques: Circuit, message and packet switching.

UNIT - II: Data Link Layer: PPP, PPPoE, MAC Layer: Ethernet (incl. manchester encoding), Switched Ethernet, VLANs, Spanning Tree Protocol.

UNIT - III: Network Layer: Data Plane: Internet Protocol Addressing: CIDR, Internet Protocol Datagram (including fragmentation and reassembly, routing options), IP Forwarding Algorithm, ARP, ICMP (including ICMP Redirect, ICMP Path MTU discovery, ICMP Destination Unreachable options).

UNIT - IV: Transport Layer: UDP, TCP sliding window protocol, TCP connection establishment, TCP reliability including cumulative and delayed acknowledgements, Nagle algorithm, Karn's algorithm for RTT and RTO estimation, TCP AIMD Congestion Control Algorithm, TCP half-close connections including TCP keepalive timer and probe timer, TCP Fast Retransmit and Fast Recovery.

UNIT - V: Network Layer: Control Plane: Distance Vector Algorithm and Routing Information Protocols V1 and V2, Link State Algorithm and Open Shortest Path First Protocol (OSPF).

UNIT - VI: Application Layer: Domain Naming System (DNS) and Dynamic Host Configuration Protocol (DHCP), Network Management using SNMP.

Reference Books:

1. James F. Kurose and Keith W. Ross. Computer Networking: A top-down approach, 6th edition, Pearson Education.
2. Douglas Comer. Computer Networks and Internets Sixth Edition, 2014. ISBN 0133587932/9780133587937, Pearson Education.
3. Douglas Comer. Internetworking With TCP/IP Volume 1: Principles Protocols, and Architecture, 6th edition, 2013. ISBN-10: 0-13-608530-X ISBN-13: 9780136085300, Pearson Education.
4. Kevin R. Fall and W.Richard Stevens. TCP/IP Illustrated, Volume 1: The Protocols, 2/E, 2012, ISBN-10: 0321336313 ISBN-13: 9780321336316, Pearson Education.
5. Radia Perlman. Interconnections: Bridges, Routers, Switches, and Internetworking Protocols, 2/E, 2000, ISBN-10: 0201634481 ISBN-13: 9780201634488. Pearson Education.

Detailed Syllabus:

UNIT - I: Physical Layer: Modulation Techniques: Amplitude, Frequency and Phase, ADSL, Multiplexing Techniques: Frequency division multiplexing, Time division multiplexing, wave length division multiplexing, Differential PCM, Switching Techniques: Circuit, message and packet switching.

UNIT - II: Data Link Layer: PPP, PPPoE, MAC Layer: Ethernet (incl. manchester encoding), Switched Ethernet, VLANs, Spanning Tree Protocol.

UNIT - III: Network Layer: Data Plane: Internet Protocol Addressing: CIDR, Internet Protocol Datagram (including fragmentation and reassembly, routing options), IP Forwarding Algorithm, ARP, ICMP (including ICMP Redirect, ICMP Path MTU discovery, ICMP Destination Unreachable options).

UNIT - IV: Transport Layer: UDP, TCP sliding window protocol, TCP connection establishment, TCP reliability including cumulative and delayed acknowledgements, Nagle algorithm, Karn's algorithm for RTT and RTO estimation, TCP AIMD Congestion Control Algorithm, TCP half-close connections including TCP keepalive timer and probe timer, TCP Fast Retransmit and Fast Recovery.

UNIT - V: Network Layer: Control Plane: Distance Vector Algorithm and Routing Information Protocols V1 and V2, Link State Algorithm and Open Shortest Path First Protocol (OSPF).

UNIT - VI: Application Layer: Domain Naming System (DNS) and Dynamic Host Configuration Protocol (DHCP), Network Management using SNMP.

Reference Books:

1. James F. Kurose and Keith W. Ross. Computer Networking: A top-down approach, 6th edition, Pearson Education.
2. Douglas Comer. Computer Networks and Internets Sixth Edition, 2014. ISBN 0133587932/9780133587937, Pearson Education.
3. Douglas Comer. Internetworking With TCP/IP Volume 1: Principles Protocols, and Architecture, 6th edition, 2013. ISBN-10: 0-13-608530-X ISBN-13: 9780136085300, Pearson Education.
4. Kevin R. Fall and W.Richard Stevens. TCP/IP Illustrated, Volume 1: The Protocols, 2/E, 2012, ISBN-10: 0321336313 ISBN-13: 9780321336316, Pearson Education.
5. Radia Perlman. Interconnections: Bridges, Routers, Switches, and Internetworking Protocols, 2/E, 2000, ISBN-10: 0201634481 ISBN-13: 9780201634488. Pearson Education.

Detailed Syllabus:

UNIT - I: Introduction and Operating System Structures

Operating Systems Functionality, Computer Organization and Architecture, OS Operations, Kernel Data Structures, OS Services, User interfaces to OS, Programmer interfaces to OS, OS Structure, System Boot.

UNIT - II: Process and Thread Management

Process Concept, Process operations, Process Scheduling, Extended Process State Diagram, Process Context Switch in detail; Inter process Communication: Pipes, Named Pipes, Shared Memory; Process Synchronization: Signals, Mutexes, Semaphores, Monitors; Thread Management: thread creation, thread scheduling, thread synchronization; Deadlocks: Resource Allocation Graphs, deadlock detection, prevention and avoidance, recovery from deadlock.

UNIT - III: Memory Management

Memory allocation techniques: paging and segmentation, Swapping, structure of the page table; Virtual memory: demand paging, copy-on-write, Page replacement, allocation of frames, kernel memory allocation, thrashing, memory-mapped files, Translation-Lookaside Buffer (TLB).

UNIT - IV: File System Management

Disk management: formatting, boot block, swap-space management, RAID structure; Disk scheduling algorithms: elevator, C-SCAN; File concept, Access methods, Directory structure, file sharing, protection, file system structure; file system implementation: file system metadata storage structures such as inode, allocation methods, free space management, efficiency and performance including disk cache and recovery from failures.

UNIT - V: I/O Management

I/O devices: polling, interrupt-driven, DMA; Application I/O interface: character and block devices, network devices; clocks and timers, nonblocking and asynchronous I/O, vectored I/O; Kernel I/O interface: I/O scheduling, Buffering, Caching.

Reference Books

1. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne. Operating System Concepts, 9th edition, Wiley.
2. Charles Crowley. Operating Systems: A Design-Oriented Approach, Prentice-Hall India.
3. W. Richard Stevens, . Advanced Programming in Unix Environment, Pearson Education.
4. W. Richard Stevens. Unix Network Programming, vol. 2, Pearson Education.
5. William Stallings. Operating Systems: Internals and Design Principles, Pearson Education.
6. Maurice J. Bach. The Design of the Unix Operating System, Prentice-Hall India.
7. Robert Love. Linux Kernel Development, Pearson Education.
8. Thomas Anderson and Michael Dahlin. Operating Systems: Principles and Practice, 2nd edition, Recursive Books.

Detailed Syllabus:

UNIT -I: Object Oriented Fundamentals and Modeling: Need for OOP paradigm, What is object orientation and OO Development, Modelling, Abstraction, Three models of OOD, Object and class concepts, Links and Association Concepts, Generalization and Inheritance, N-ary associations, Aggregation, Abstract classes, multiple inheritance, metadata, Reification, Constraints, Derived data, packages.

UNIT -II: Java Basics: History of Java, java data types, variables, scope and life time of variables, arrays, operators, expressions, control statements, type conversion and casting, simple java program, concepts of classes, objects, constructors, methods, access control, this keyword, garbage collection, overloading methods and constructors, parameter passing, recursion, nested and inner classes, exploring string class. Defining, Creating and Accessing a Package, Understanding CLASSPATH, importing packages, differences between classes and interfaces, defining an interface, implementing interface, applying interfaces, variables in interface and extending interfaces. Exploring java.io.

UNIT -III: State Modelling and Interaction Modelling: Events, states, Transitions and Conditions, State Diagram, Nested state diagram, Concurrency Use-Case model, Sequence model, Activity model, procedural sequence model, Relation between class, state model and interaction model.

UNIT -IV: Hierarchical abstractions, Generalization and Aggregation, Base class object, subclass, subtype, substitutability, forms of inheritance-specialization, specification, construction, extension, limitation, combination, benefits of inheritance, costs of inheritance. Member access rules, super uses, using final with inheritance, polymorphism-method overriding, abstract classes, the Object class Exception handling in Java: Concepts of exception handling, benefits of exception handling, Termination or resumptive models, exception hierarchy, usage of try, catch, throw, throws and finally, built in exceptions, creating own exception sub classes. String handling, Exploring java.util.

UNIT -V: System Design, Class Design and Implementation Modelling: Overview of system design, performance estimation, reuse plan, Subsystems, Management of data storage and global resources, software control strategy and boundary conditions. Overview of class design, Realizing use-cases, designing algorithms, refactoring and design optimization, Overview of implementation, fine tuning of classes, generalization, and realizing associations.

Java Lab Exercises:

1. Basic features of Java such as data types, control structures, loops and arrays (2 Lab Sessions)
2. Working with classes, constructors, methods, objects. (2 Lab Sessions)
3. Using Inheritance, Polymorphism, Interfaces and abstract classes (3 Lab Sessions). Exercises can include a case study depicting OO application design using polymorphism and inheritance. (For example) Developing a Solitaire Application (Chapter 8 of "Introduction to Object Oriented Programming by Timothy Budd")
4. Exception Handling (1 or 2 Lab Session)
5. Exploring Java IO Package (2 or 3 Lab Sessions)
6. Java GUI Programming such as Applets (2 Lab Sessions)
7. Miscellaneous Topics such as generic classes, collection framework and java.util packages (1 or 2 Lab Sessions)

Reference Books:

1. Herbert Schildt, Java: The complete reference, McGraw hill.
2. Paul J. Deitel and Harvey M. Deitel , Java: How to Program, Prentice Hall.
3. T. Budd, Understanding OOP with Java, Pearson Education.
4. Michael Blaha and James Rumbaugh, Object Oriented Modelling and Design with UML, 2nd edition, Eastern Economy Edition.
5. Herbert Schildt, Java: A Beginner's Guide, McGraw Hill Education (India) Private Limited.
6. Bruce Eckle, Thinking in Java, Prentice Hall.
7. Joshua Bloch, Effective Java, Createspace Independent Pub.
8. Kathy Siera, Head First Java, O'Reilly Media

Detailed Syllabus

UNIT-I: Analysis of Algorithms: Asymptotic Notation; Best, worst and average case analysis of algorithms; Solving recurrence relations using substitution method, generating functions, Master's theorem etc. Warm-up to complexity analysis: Heap data structure, priority queue application, Best, worst and average case analysis of a few sorting algorithms like heap sort, insertion, bubble, selection, counting and radix sort algorithms. Strategies for problem solving

UNIT-II: Divide and Conquer strategy: Time complexity analysis for Merge Sort and Quick Sort Algorithms

UNIT-III: Greedy strategy: Theoretical foundation of greedy strategy: Matroids Algorithms for solving problems like Knapsack Problem (Fractional), Minimum Spanning Tree problem; Shortest Paths, Job Scheduling, Huffman's code etc along with proofs of correctness and complexity analysis

UNIT-IV: Dynamic Programming strategy: Identify situations in which greedy and divide and conquer strategies may not work. Understanding of optimality principle. Technique of memorization. Applications to problems like Coin change, 0/1 and 0/n- Knapsack, Shortest Paths, Optimal Binary Search Tree (OBST), Chained Matrix Multiplication, Traveling Salesperson Problem (TSP) etc.

UNIT-V: Backtracking and Branch & Bound strategies: State space tree construction, traversal techniques and solving problems like 0/1 and 0/n knapsack, TSP, Applications of Depth First Search: Topological sorting, Finding strongly connected components and game problems.

UNIT-VI: Theory of NP-Completeness: Complexity classes of P, NP, NP-Hard, NP-Complete, Polynomial reductions, Cook's theorem. Discussion of problems: Satisfiability(SAT), CNF-SAT, Min-Vertex Cover, Max-Clique, Graph Coloring, NP-Completeness proofs.

Reference Books:

1. Introduction to Algorithms-T.Cormen, C.E.Leiserson, R.L.Rivest, PHI, 3rdEdition 2009.
2. Algorithms- R.Johnsonbaugh and M.Schaefer, Pearson, 2004.
3. Fundamentals of Algorithmics - G.Brassard and P.Bratley, PH, 1996
4. The Algorithm Design Manual- Steven S. Skiena, Springer, 2009

Detailed Syllabus:

UNIT-I: Introduction to Software Engineering

Need of software engineering, systems engineering, challenges in software engineering, Software process models, quality characteristics of software systems, Ethics in Software Engineering.

UNIT-II: Requirements Engineering

Requirements engineering process, requirements specification, structured and object oriented analysis

UNIT-III: Software Design

Architectural design, detailed design, Structured and object oriented design, user interface design

UNIT-IV: Software Testing

Verification, Validation, testing techniques, Testing Process

UNIT-V: Tools and Evolution

CASE Tools, Reverse engineering, Reengineering and Configuration management.

Reference Books:

1. Ian Sommerville (2016), *"Software Engineering"*, 10th Edition, Pearson Education Limited, Global Edition.
2. Roger S Pressman, Bruce R Maxim(2015), *"Software Engineering, A Practitioner's Approach"*, 8th Edition, TataMcGraw Hill, Indian Edition
3. Grady Booch, James Rumbaugh, Ivor Jacobson(2005), *"The Unified Modeling Language User Guide"*, 2nd Edition, Addison Wesley Professional.US

Detailed Syllabus:

For a given case study/problem statement, the following deliverables are to be realized

- Define stories
- Identify tasks and develop test plan for stories/task (with the help of specifications)
- Design and develop increments
- Test the increments and release the increment
- Apply object oriented and structured modelling
- Implement the case study for plan driven approach by writing use case specification, designing the system and implementing the same.

Reference Books:

1. Ian Sommerville (2016), "*Software Engineering*", 10th Edition, Pearson Education Limited, Global Edition
2. Roger S Pressman, Bruce R Maxim(2015), "*Software Engineering, A Practitioner's Approach*", 8th Edition, TataMcGraw Hill, Indian Edition

School of Computer & Information Sciences

Name of the Academic Program: Master of Computer Applications (MCA-III)

Course Code: CA506

Title of the Course: Enterprise Resource Planning

L-T-P: 3-0-0

Credits: 3

Prerequisite Course / Knowledge (If any):

Course Outcomes (COs)

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

CO-1: Describe the Industry verticals / micro-verticals / Business Domains and synergy in a multi-company set up.(Understand)

CO-2: Explain processes in Manufacturing Module

CO-3: Differentiate the processes that are essential in Financial Module and its use across (Analyze)

CO-4 : Develop the ability to talk to Industry experts on these processes and optimizing / implementing on the same when necessary (Create)

CO-5: Assess the current trends in the industry in implementing / customizing / configuring the software(Evaluate)

**Mapping of Course Outcomes (COs) with Program Outcomes (POs)
and Program Specific Outcomes (PSOs)**

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1												
CO2												
CO3												
CO4												
CO5												

Each Course Outcome (CO) may be mapped with one or more Program Outcomes (POs). Write '3' in the box for 'High-level' mapping, 2 for 'Medium-level' mapping, 1 for 'Low'-level' mapping

Mapping with PSOs, where applicable.

Detailed Syllabus:

UNIT-I: ERP Overview: Integrated management information systems, Supply chain management, Integrated data model, Benefits of ERP, Evolution of ERP and Modern enterprise, BPR (Business Process Reengineering) & ERP, Business modelling for ERP

UNIT-II: Customer Service

UNIT-III: Production Planning and execution

UNIT-IV: Purchasing and goods receipt

UNIT-V: Financial and other metrics

UNIT-VI: ERP Packages

UNIT VII: Case studies, Insurance industry, Banking industry, Pharmaceutical industry, Health care, Consumer products, Retail industry, University, Transport Industry, Telecom industry, Public Sector Industry

UNIT VIII: Current Trends in Implementations; Hardware / Network selections ; Data Management requirements; Integration requirements and techniques; Other Non-functional requirements necessary for implementations

Reference Books:

Learning Resources/Text Books:

1. 3. Infor ERP - LN Workbook
2. 4. Enterprise Resource Planning – Alexis Leon, Tata McGraw Hill

Suggested Reading

3. INTRODUCTION TO MATERIALS MANAGEMENT 6/E (English) 6th Edition Authors: Tony Arnold / Stephen Chapman Publishers: Pearson India
4. Manufacturing Planning and Control for Supply Chain Management (APICS / CPIM Certification Edition) Authors: Thomas E. Vollmann, CFPIM, William L. Berry, D. Clay Whybark, and F. Robert Jacobs Publishers: McGraw Hill Education
5. Accounting Handbook 6th Edition Authors: Shim Siegal Publishers: Barrons Educational Series
6. Operations Strategy 4th Edition Authors: Nigel Slack and Mike Lewis Publishers: Pearson
7. CONCEPTS IN ENTERPRISE RESOURCE PLANNING Authors: Ellen F. Monk, Bret J. Wagner
8. Enterprise Resource Planning – Ashim Raj Singla, Cengage Learning