

**INSTITUTE OF DISTANCE EDUCATION
UNIVERSITY OF MADRAS**

**MCA-MASTER OF COMPUTER APPLICATIONS (Two years programme)
(With effect from the academic year 2020-2021 onwards)**

SCHEME OF EXAMINAIONS

BRIDGE COURSE

1. **Computer Science Department** is encouraged to offer need based following **Bridge Courses** and foundation courses to meet prerequisite requirements and academic needs.
 - i. Programming in C
 - ii. Problem Solving Techniques
 - iii. Mathematical Foundations of Computer Science.
 - iv. Information Technology
 - v. Coding Practices
2. Based on the qualifications of the students admitted the mentoring team of the department shall recommend to carry out the bridge and foundation courses as mandatory courses for that candidate.
3. Two weeks bridge courses need to be organized before the commencement of the first semester.
4. Nurturing and evaluation process of bridge and foundation courses is left the respective academic units (IDE Department).

Course Components	Name of Course	Credits	Max.Marks		Total
			Int.	Ext.	
FIRST YEAR-I SEMESTER					
Core Paper - I	C++ & Data Structures	4	20	80	100
Core Paper - II	Digital Logic Fundamentals	4	20	80	100
Core Paper - III	Database Management Systems	4	20	80	100
Core Paper - IV	Practical-I: DS using C++ Lab	2	40	60	100
Core Paper - V	Practical-II: RDBMS Lab.	2	40	60	100
Inter Disciplinary- I	Accounting & Financial Management	3	20	80	100
Elective Paper-I	Operating Systems	3	20	80	100
II SEMESTER					
Core Paper - VI	Design and Analysis od Algorithms	4	20	80	100
Core Paper - VII	Object Oriented Analysis and Design	4	20	80	100
Core Paper - VIII	Artificial Intelligence	4	20	80	100
Core Paper - IX	Practical-III: OOAD Lab	2	40	60	100
Core Paper -X	Practical-IV: Web Based Application Development Lab.	2	40	60	100
Inter Disciplinary- II	Web Based Application Development	3	20	80	100
Elective Paper -II	Computer Network	3	20	80	100
SECOND YEAR- III SEMESTER					
Core Paper -XI	Machine Learning	4	20	80	100
Core Paper -XII	Practical-V: Machine Learning Lab	2	40	60	100
Core Paper -XIII	Practical-VI: Mini Project (Group Project)	2	40	60	100
Elective Paper -III	Software Project Management	3	20	80	100
Elective Paper -IV	Cloud Computing	3	20	80	100
Elective Paper -V	Mobile Application Development	3	20	80	100
IV SEMESTER					

Core Paper -XIV	Project & Viva-Voce	20	20	60+20	100
-----------------	---------------------	----	----	-------	-----

List of Elective Papers

Sl.No	Elective-I	Sl.No	Elective - II
1	Operating Systems (OR)	1	Computer Networks (OR)
2	Theory of Computation(OR)	2	Digital Image Processing (OR)
3	Explorative Data Analysis with R	3	Software Engineering
	Elective - III		Elective - IV
1	Software Project Management (OR)	1	Mobile Computing (OR)
2	Supply Chain Management(ANS) (OR)	2	Cloud Computing (OR)
3	Management Information Systems	3	Soft Computing
	Elective – V		
1	Mobile Application Development (OR)		
2	Information Security (OR)		
3	IoT		

Students are encouraged to do courses from the resources like SWAYM, NPTEL etc

*Operation systems	*Software testing
*Principles of Programming Languages,	*Bigdata Analytics
*Computer Networks	*Robotics
*Compiler design	*Agile technologies
*Natural Language Processing	*Robotics Process Automation
*Software Engineering	*Organizational Behaviors.

SYLLABUS

SEMESTER-I

Core Paper - I : C++ and Data Structures	Year-I	Semester -I	Credits : 4
Objective of the course : This course introduces the basic concepts of programming in C++ and discuss the implementation of			
Course Outcomes : After successful completion of this course, the students should be able to Understand the behavior of basic data structures. Analyze a problem and determine the appropriate data structure for the problem.			
Unit-I : Introduction to C++; Tokens, Keywords, Identifiers, Variables, Operators, Manipulators, Expressions and Control Structures in C++; Pointers - Functions in C++ - Main Function -Function Prototyping - Parameters Passing in Functions - Values Return by Functions - Inline Functions - Friend and Virtual Functions			
Unit-II: Classes and Objects; Constructors and Destructors; and Operator Overloading and Type Conversions - Type of Constructors - Function overloading. Inheritance : Single Inheritance - Multilevel Inheritance - Multiple Inheritance - Hierarchical Inheritance - Hybrid Inheritance. Pointers, Virtual Functions and Polymorphism; Managing Console I/O operations.			
Unit-III: Working with Files: Classes for File Stream Operations - Opening and Closing a File - End-of-File Deduction - File Pointers - Updating a File - Error Handling during File Operations - Command-line Arguments. Data Structures: Definition of a Data structure - primitive and composite Data Types, Asymptotic notations, Arrays, Operations on Arrays, Order lists.			
Unit-IV: Stacks - Applications of Stack - Infix to Postfix Conversion, Recursion, Maze Problems - Queues- Operations on Queues, Queue Applications, Circular Queue. Singly Linked List- Operations, Application - Representation of a Polynomial, Polynomial Addition; Doubly Linked List - Operations, Applications.			
Unit-V: Trees and Graphs: Binary Trees - Conversion of Forest to Binary Tree, Operations -Tree Traversals; Graph - Definition, Types of Graphs, Hashing Tables and Hashing Functions, Traversal - Shortest Path; Dijkstra's Algorithm.			

1. Recommended Texts :

- a. E.Horowitz, S.Sahni and Mehta, 1999, Fundamentals of Data Structures in C++, Galgotia.
- b. Herbert Schildt, 1999, C++ - The complete Reference, Third Edition, Tata McGraw -Hill.

2. Reference Books:

- a. GregoryL.Heileman, 1996, Data Structures , Algorithms and Object Oriented Programming Mc-Graw Hill International Editions.
- b. A.V.Aho, J.D. Ullman, J.E. Hopcraft: 2001, Data Structures and Algorithms-Adisson Wesley Pub.

Core –II – Digital Logic Fundamentals	Year –I	Semester - I	Credits : 4
<p>Objectives of the course: To acquire fundamental principles of two-valued logic and various devices used to implement logical operations on variables. To lay the foundation for further studies in areas such as communication, VLSI, computer, microprocessor.</p>			
<p>Course Outcomes: After successful completion of this course, the students should be able to Use the basic logic gates and various reduction techniques of digital logic circuit in detail. Design combinational and sequential circuits. Design and implement hardware circuit to test performance and application. Understand the architecture and use of microcontrollers for the basic operations and simulate using simulation software.</p>			
<p>Unit-I : Number System – Converting numbers from one base to another – Complements – Binary Codes – Integrated Circuits – Boolean algebra – Properties of Boolean algebra – Boolean functions – Canonical and Standard forms – Logical Operations – Logic gates – Karnaugh Map up to 6 variables – Don't Care Condition – Sum of Products and Products of Sum simplification – Tabulation Method.</p>			
<p>Unit-II : Adder – Subtractor – Code Converter – Analyzing a combinational Circuit – Multilevel NAND and NOR circuits – Properties of XOR and equivalence functions – Binary Parallel Adder – Decimal Adder – Magnitude Comparator – Decoders – Multiplexers – ROM – PLA.</p>			
<p>Unit-III : Flip Flops – Triggering of flip-flops – Analyzing a sequential circuit – State reduction – excitation tables – Design of sequential circuits – Counters – Design with state equation – Registers – Shift Registers – Ripple and synchronous Counters.</p>			
<p>Unit-IV : Memory Unit – Processor Organization - Bus Organization – Scratch Pad memory – ALU – Design of ALU – Status Register – Effects of Output carry – Design of Shifter – Processor Unit – Microprogramming – Design of specific Arithmetic Circuits</p>			
<p>Unit-V : Accumulator – Design of Accumulator – Computer Design – System of Configuration – Instruction and Data formats – Instruction sets – Timing and Control – Execution of Instruction – Design of Computer – Hardwired control – PLA Control and Microprogram control</p>			

Recommended Texts:

- a. M. Morris Mano, 2011, Digital Logic and Computer Design, Thirteenth Impression, Pearson Education, Delhi

Reference Books:

- b. M. M. Mano and C.R.Kime, 2001, Logic and Computer Design Fundamentals, 2nd Edition, Pearson Education, Delhi.
- c. Givone, 2002, Digital Principles Design, Tata McGraw Hill, New Delhi.
- d. C. H. Roth , Jr, 2005, Fundamentals of Logic Design ,5th Edition, Thomson Learning, Singapore.

Core Paper - III: Database Management Systems	Year – I	Semester-I	Credits: 4
Objective of the course: This course introduces the concepts of database systems design and to understand the context, phases and techniques for designing and building database information systems in business. To understand the components of a computerized database information system.			
Course Outcomes: After successful completion of this course, the students should be able to Design a correct, new database information system for a business functional area and implement the design in either SQL or NoSQL To understand the concepts of open source databases.			
Unit-I: Introduction to Database Systems – Relational Model – Structure – Relational Algebra – Null Values – SQL – Set Operation – Views – Advanced SQL – Embedded SQL – Recursive Queries – The Tuple Relational Calculus – Domain Relational Calculus.			
Unit-II: E-R Model – Constraints – E-R- Diagrams Weak Entity Sets – Reduction to Relational Schemes – Relational Database Design – Features of Relational Design – Automatic Domains and First Normal Form – Decomposition using Functional Dependencies – Multivalued Dependencies – More Normal Forms – Web Interface – Object – Based Databases – Structured Types and inheritance in SQL – Table inheritance – Persistent.			
Unit-III: Storage and File Structure – RAID – File Organisation – Indexing and Hashing – B Tree – B Tree Index files - Static and Dynamic Hashing – Query Processing – Sorting & Join Operators – Query Optimization – Choice of Evaluation Plans.			
Unit-IV: Transaction Management – Implementation of Atomicity and Durability –Serializability – Recoverability – Concurrency Control – Dead Lock Handling – Recovery System – Buffer Management.			
Unit-V: Database – System Architecture – Client Server – Architectures – Parallel System –Network Types – Distributed Database – Homogeneous and Heterogeneous Database – Directory System – Case Study – Oracle – MSSQL Server.			

Recommended Texts

- 1) A. Silberschatz, H.F. Korth and S. Sudharshan, 2006, Database System Concepts, 5th Edition, Tata McGraw Hill, New Delhi.

Reference Books

- 1) J. D. Ullman, 1988, Principles of Database Systems, Galgotia Publishers, New Delhi
- 2) C.J. Date, 1985, An Introduction to Database Systems, Third Edition, Narosa, New Delhi.
- 3) Elmasri and Navathe, 1999, Fundamentals of Database Systems, Third Edition, Pearson Education, Delhi.
- 4) C. Ritchie, 2004, Relational Database Principals, 2ndEdition,Thomson, Singapore.

Website, E-learning resources

- 1) <http://www.cse.iitb.ac.in/dbms/Data/Papers-Local/DBConceptsBook/slide-dir/>

Core Paper - IV: Practical – I : Data Structure Using C++ Lab.	Year-I	Semester - I	Credits: 2
Objective of the course: This course train the students to implement the data structure concepts. For the implementation of the following problems, the students are advised to use all possible object-oriented features. The implementation based on structured concepts will not accepted.			
Course Outcomes: After successful completion of this course, the students should be able to understand the array, Linked list, stack and queue, tree and binary tree traversal.			
<ol style="list-style-type: none"> 1. Implementation of Arrays (Single and Multi-Dimensional) 2. Polynomial Object and necessary overloaded operators. 3. Singly Linked Lists. 4. Circular Linked Lists. 5. Doubly Linked Lists. 6. Implementation of Stack (using Arrays and Pointers) 7. Implementation of Queue (Using Arrays and Pointers) 8. Implementation of Circular Queue (using Arrays and Pointers) 9. Evaluation of Expressions. 10. Binary Tree implementations and Traversals. 11. Binary Search Trees. 			

Core Paper - V: Practical – II: RDBMS Lab.	Year-I	Semester - I	Credits: 2
<p>Objective of the course: This course gives training in design and implementation of data bases for the selected problems. Students are advised to use the concepts like Data Normalization, Link between table by means of foreign keys and other relevant data base concepts for developing databases for the following problems. The implementation of each problem should have necessary input screen Menu-driven query processing and pleasing reports. The choice or RDBMS is left to the students. Necessary validations must be done after developing database.</p>			
<p>Course Outcomes: After successful completion of this course, the students should be able to understand and design an application for library management system, student marksheet, bank transaction, quiz using controls and fields.</p>			
<ol style="list-style-type: none"> 1. Building Simple Applications. 2. Working with Intrinsic Controls and ActiveX Controls. 3. Application with multiple forms. 4. Application with Dialogs. 5. Application with Menus. 6. Application using Data Controls. 7. Application using Common Dialogs. 8. Drag and Drop Events. 9. Database Management. 10. Creating ActiveX Controls. 11. Library Management System. 12. Students Marksheet Processing. 13. Bank Transactions. 14. Personal information system. 15. Question Database and conducting Quiz. 			

Inter Disciplinary-I: Accounting and Financial Management	Year- I	Semester-I	Credits:3
Objective of the course: This course introduces the basic concepts of Accounting & Financial Management.			
Course Outcomes: After successful completion of this course, the students should be able to understand the basic idea and they can able to work in accounts and finance for managing the business.			
Unit-I: Principles of Accounting: Principles of double entry -Assets and Liabilities - Accounting records and systems - Trial balance and preparation of financial statements - Trading, Manufacturing, Profit and Loss accounts, Balance Sheet including adjustments(Simple problems only).			
Unit-II: Analysis and Interpreting Accounts and Financial Statements: Ratio analysis - Use of ratios in interpreting the final accounts (trading accounts and loss a/c and balance sheet) - final accounts to ratios as well as ratios to final accounts.			
Unit-III: Break-even analysis and Marginal Costing: Meaning of variable cost and fixed cost - Cost-Volume-Profit analysis – calculation of breakeven point, Profit planning, sales planning and other decision – making analysis involving break - even analysis - Computer Accounting and algorithm.(differential cost analysis to be omitted).			
Unit-IV: Budget/Forecasting: preparation of and Characteristics of functional budgets, Production, sales, Purchases, cash and flexible budgets.			
Unit-V: Project Appraisal: Method of capital investment decision making: Payback method, ARR method - Discounted cash flows - Net Present values - Internal rate of return - Sensitivity analysis - Cost of capital.			

Reference Books:

- 1) Shukla M.C. & T.S. Grewal, 1991, Advanced Accounts, S.Chand & Co. New Delhi.
- 2) Gupta R.L. & M. Radhaswamy, 1991, Advanced Accounts Vol. II, Sultan Chand & Sons, New Delhi.
- 3) Man Mohan & S.N. Goyal, 1987, Principles of Management Accounting, Arya Sahithya Bhawan.
- 4) Kuchhal, S.C., 1980, Financial Management, Chaitanya, Allahabad.
- 5) Hingorani, N.L. & Ramanthan, A.R, 1992, Management Accounting, 5th edition, Sultan Chand, New Delhi.

Elective Paper- I: Operating Systems	Year -I	Semester - I	Credits: 3
Objective of the course: This course introduces the fundamental concepts of operating systems.			
Course Outcomes: After successful completion of this course, the students should be able to understand the behavior of the operating system.			
Unit-I: Introduction – Multiprogramming - Time sharing - Distributed system - Real-Time systems - I/O structure - Dual-mode operation - Hardware protection _ General system architecture - Operating system services - System calls - System programs - System design and implementation. Process Management: Process concept - Concurrent process - Scheduling concepts - CPU scheduling - Scheduling algorithms, Multiple processor Scheduling			
Unit-II: Process Management: Process Synchronization - Critical section - Synchronization hardware - Semaphores, classical problem of synchronization, Interprocess communication. Deadlocks: Characterization, Prevention, Avoidance, and Detection.			
Unit-III: Storage management - Swapping, single and multiple partition allocation - paging - segmentation - paged segmentation, virtual memory - demand paging - page replacement and algorithms, thrashing. Secondary storage management - disk structure - free space management - allocation methods – disk scheduling - performance and reliability improvements - storage hierarchy.			
Unit-IV: Files and protection - file system organization - file operations - access methods - consistency semantics - directory structure organization - file protection - implementation issues - security – encryption			
Unit-V: Case Studies: UNIX and Windows operating systems.			

Recommended Texts

- 1) A. Silberschatz P.B. Galvin, Gange, 2002, Operating System Concepts, 6th Edn., Addison-Wesley Publishing Co., Boston.

Reference Books

- 1) H.M. Deitel, 1990, An Introduction to Operating Systems, Addison Wesley Publishing Co., Boston.
- 2) D.M. Dhamdhare , 2002, Operating System, Tata McGraw-Hill, New Delhi.
- 3) A.S. Tanenbaum , Operating Systems: Design and Implementation, Prentice-Hall of India, New Delhi.
- 4) Nutt, 2005, Operating Systems, 3 rd Edition, Pearson Education, Delhi.

SEMESTER - II

Core Paper – VI: Design and Analysis of Algorithms	Year-I	Semester - II	Credits: 4
<p>Objective of the course: This course introduces the basic concepts of Algorithm and discuss the implementation of different methods.</p>			
<p>Course Outcomes: After successful completion of this course, the students should be able to Understand the basics of algorithm. Determine the appropriate data structure designs for the different concepts.</p>			
<p>Unit-I: Introduction - Definition of Algorithm – pseudocode conventions – recursive algorithms – time and space complexity –big- “oh” notation – practical complexities – randomized algorithms – repeated element – primality testing - Divide and Conquer: General Method - Finding maximum and minimum – merge sort.</p> <p>Unit-II: Divide and conquer contd. – Quicksort, Selection, Strassen's matrix multiplication – Greedy Method: General Method –knapsack problem - Tree vertex splitting - Job sequencing with deadlines – optimal storage on tapes.</p> <p>Unit-III: Dynamic Programming: General Method - multistage graphs – all pairs shortest paths – single source shortest paths - String Editing – 0/1 knapsack. Search techniques for graphs – DFS-BFS-connected components – biconnected components.</p> <p>Unit-IV: Back Tracking: General Method – 8-queens - Sum of subsets - Graph Coloring – Hamiltonian cycles. Branch and Bound: General Method - Traveling Salesperson problem.</p> <p>Unit-V: Lower Bound Theory: Comparison trees - Oracles and advisory arguments - Lower bounds through reduction - Basic Concepts of NP-Hard and NP-Complete problems.</p>			

Recommended Texts

- 1) E. Horowitz, S. Sahni and S. Rajasekaran, 2008, Computer Algorithms, 2nd Edition, Universities Press, India.

Reference Books

- 1) G. Brassard and P. Bratley, 1997, Fundamentals of Algorithms, PHI, New Delhi.
- 2) A.V. Aho, J.E. Hopcroft, J.D. Ullmann, 1974, The Design and Analysis of Computer Algorithms, Addison Wesley, Boston.
- 3) S.E.Goodman and S.T.Hedetniemi, 1977, Introduction to the Design and Analysis of algorithms, Tata McGraw Hill Int. Edn, New Delhi.

E-learning resources

- 1) <http://www.cise.ufl.edu/~raj/BOOK.html>

Core Paper -VII – Object Oriented Analysis and Design	Year -I	Semester - II	Credits: 4
<p>Objectives of the course: To understand the Object-based view of Systems. To develop robust object-based models for Systems and to inculcate necessary skills to handle complexity in software design.</p>			
<p>Course Outcomes: After successful completion of this course, the students should be able to Ability to analyze and model software specifications, Ability to abstract object-based views for generic software systems, Ability to deliver robust software components.</p>			
<p>Unit-I: System Development - Object Basics - Development Life Cycle - Methodologies - Patterns - Frameworks - Unified Approach - UML.</p> <p>Unit-II: Use-Case Models - Object Analysis - Object relations - Attributes - Methods – Class and Object responsibilities - Case Studies.</p> <p>Unit-III: Design Processes - Design Axioms - Class Design - Object Storage - Object Interoperability - Case Studies.</p> <p>Unit-IV: User Interface Design - View layer Classes - Micro-Level Processes - View Layer Interface - Case Studies.</p> <p>Unit-V: Quality Assurance Tests - Testing Strategies - Object orientation on testing - Test Cases - test Plans - Continuous testing - Debugging Principles - System Usability - Measuring User Satisfaction - Case Studies.</p>			

Recommended Texts

- 1) Ali Bahrami, Reprint 2009, Object Oriented Systems Development, Tata McGraw Hill International Edition.

Reference Books

- 1) G. Booch, 1999, Object Oriented Analysis and design, 2nd Edition, Addison Wesley, Boston
- 2) R. S.Pressman, 2010, Software Engineering A Practitioner’s approach, Seventh Edition, Tata McGraw Hill, New Delhi.
- 3) Rumbaugh, Blaha, Premerlani, Eddy, Lorensen, 2003, Object Oriented Modeling And design , Pearson education, Delhi.

Core Paper-VIII: Artificial Intelligence	Year – I	Semester - II	Credits: 4
Objective of the course: The primary objective of this course is to introduce the basic principles, techniques, and applications of Artificial Intelligence. Emphasis will be placed on the teaching of these fundamentals, not on providing a mastery of specific software tools or programming environments.			
Course Outcomes: After successful completion of this course, the students should be able to Demonstrate fundamental understanding of the history of artificial intelligence (AI) and its foundations. Apply basic principles of AI in solutions that require problem solving, inference, perception, knowledge representation, and learning.			
UNIT-I: Introduction: What Is AI? - Foundations of Artificial Intelligence-The History of Artificial Intelligence- The State of the Art- Risks and Benefits of AI. Intelligent Agents: Agents and Environments - The Concept of Rationality - The Nature of Environments- The Structure of Agents.			
UNIT-II: Solving problem by Searching: Problem-Solving Agents - Example Problems - Search Algorithms: Best-first search - Search data structures - Redundant paths - Measuring problem-solving performance - Uninformed Search Strategies: BFS-DFS- Depth limited and iterative deepening search. Heuristic Search Strategies: Greedy best-first search - A* search - Search contours - Inadmissible heuristics and weighted A* - Heuristic Functions.			
UNIT-III: Local Search and Optimization Problems: Hill-climbing search - Simulated annealing - Local beam search - Local Search in Continuous Spaces - Search with Nondeterministic Actions: The erratic vacuum world - AND—OR search trees. Optimal Decisions in Games: The minimax search algorithm - Optimal decisions in multiplayer games - Alpha--Beta Pruning. Heuristic Alpha--Beta Tree Search: Evaluation functions - Cutting off search - Forward pruning - Monte Carlo Tree Search - Stochastic Games- Limitations of Game Search Algorithms.			
UNIT-IV: Constraint Satisfaction Problems: Defining Constraint Satisfaction Problems - Constraint Propagation: Inference in CSPs - Backtracking Search for CSPs - Local Search for CSPs - The Structure of Problems. Logical agent and Logics: Propositional Logic - Propositional Theorem Proving - Effective Propositional Model Checking - Agents Based on Propositional Logic - First-Order Logic: Syntax and Semantics of First-Order Logic - Using First-Order Logic - Knowledge Engineering in First-Order Logic. Inference in First-Order Logic: Unification and First-Order Inference - Forward Chaining - Backward Chaining – Resolution.			
UNIT-V: Knowledge Representation and Reasoning : Ontological Engineering - Categories and Objects - Events - Mental Objects and Modal Logic - Reasoning Systems for Categories - Reasoning with Default Information. Automated Planning: Definition of Classical Planning - Algorithms for Classical Planning - Heuristics for Planning. Quantifying Uncertainty: Acting under Uncertainty - Basic Probability Notation - Inference Using Full Joint Distributions - Independence - Bayes' Rule and Its Use - Naive Bayes Models			

Recommended Texts

- 1) Stuart Russel and Peter Norvig: Artificial Intelligence – A Modern Approach- 4th Edition Pearson Education, 2020.

Reference Books:

- 1) Elaine Rich and Kevin Knight: Artificial Intelligence- Tata McGraw Hill 2nd Ed, 1991.
- 2) N.P. padhy: Artificial Intelligence and Intelligent Systems- Oxford Higher Education- Oxford University Press, 2005.
- 3) George F Luger: Artificial Intelligence- Structures and Strategies for complex Problem Solving- 4 th Ed. Pearson Education, 2002.

Core Paper – IX : Practical – III: OOAD Lab	Semester - II	Credits: 2
Objective of the course:		
This course gives training to understand the Object-based view of Systems. To develop robust object-based models for Systems. To inculcate necessary skills to handle complexity in software design		
Course Outcomes:		
After successful completion of this course, the students should be able to analyze and model software specifications, Ability to abstract object-based views for generic software systems. Ability to deliver robust software components.		
<ol style="list-style-type: none"> 1. Student information system. 2. Stock Maintenance System. 3. Banking system. 4. Online course reservation system. 5. Exam Registration. 6. Employee Management System. 7. Project Tracking System. 8. Library Information System. 9. E-ticketing 10. E-book management system. 11. Recruitment system. 12. Conference Management System. 13. BPO Management System. 14. Credit card processing. 15. Gas Booking System. 		

Core Paper – X : Practical – IV: Web Based Application Development Lab.	Year – I	Semester - II	Credits: 2
Objective of the course:			
To understand the importance of the web as an effective medium of communication. To develop basic skills in analyzing the usability of a web site. To develop hands on experience using open source technologies such as HTML, CSS, JavaScript, PHP and MySQL.			
Course Outcomes:			
After successful completion of this course, the students should be able to Design, develop and host a user-friendly website, Know the usage of APIs and Layout management in line with current trend.			
<ol style="list-style-type: none"> 1. Create a Multilevel inheritance for Employee using appropriate data members and methods using C#. 2. Create an application form to apply for a new course in a college, fill the information and submit it (Use Basic Web Server controls). 3. Create a web application using Global.asax file which will count the number of visitors on web page. 4. Design a web page to implement upload and download files functionality using File Upload Control. 5. Develop a web page to implement the concept of state management using Cookies 6. Develop a web page to implement the concept of state management using Session and Application 7. Develop a web page to implement the concept of state management using ViewState and QueryString. 8. Design Sign Up form and validate User Name (Minimum 8 character Maximum 15 and only characters and underscore), Password (Minimum 8 Characters) and Confirm_Password (Both should be same), Phone No (Only digits), Email-id etc. (Use Validation controls). 9. Create a web site using Master Page Concept having two content pages. 10. Write sample application to connect to database, Fetching and inserting data from database and using Data Reader 11. Create Employee database and develop a web application to Add, Update, View and Delete records from database using in Gridview control. 12. Create Student database and develop a web application to Add, Update, View and Delete records from database using in Gridview control. 13. Create a web form for Online Library data entry and manipulate records using C#.NET. 14. Design a web page to display the XML content. 15. Design a web application to demonstrate form authentication and authorization. 			

Inter Disciplinary- II Web Based Application Development	Year –I	Semester - II	Credits: 3
<p>Objective of the course: Student will be familiar with client server architecture and able to develop a web application using java technologies. They will gain the skills and project-based experience needed for entry into web application and development careers.</p>			
<p>Course Outcomes: After successful completion of this course, Students are able to develop a dynamic webpage by the use of java script and DHTML and write a well formed / valid XML document. They are able to connect a java program to a DBMS and perform insert, update and delete operations on DBMS table. Students will be able to write a server-side java application called Servlet to catch form data sent from client, process it and store it on database.</p>			
<p>Unit – I: OVERVIEW OF ASP.NET - The .NET framework – The C# Language: Data types – Declaring variables- Scope and Accessibility- Variable operations- Object Based manipulation- Conditional Structures- Loop Structures- Methods. Types, Objects and Namespaces: The Basics about Classes- Value types and Reference types- Understanding name spaces and assemblies - Advanced class programming.</p>			
<p>Unit – II: Developing ASP.NET Applications - The Anatomy of a Web Form – Writing Code - Visual Studio Debugging. Web Form Fundamentals: The Anatomy of an ASP.NET Application - Introducing Server Controls - HTML Control Classes - The Page Class - Application Events - ASP.NET Configuration. Web Controls: Web Control Classes - List Controls - Web Control Events and AutoPostBack - A Simple Web Page.</p>			
<p>Unit – III: Error Handling, Logging, and Tracing: Common Errors - Exception Handling - Handling Exceptions - Throwing Your Own Exceptions - Logging Exceptions - Page Tracing. State Management: View State - Transferring Information Between Pages – Cookies - Session State - Session State Configuration - Application State. Validation: Understanding Validation - The Validation Controls.</p>			
<p>Unit – IV: Rich Controls: The Calendar - The AdRotator - Pages with Multiple Views - User Controls and Graphics - User Controls - Dynamic Graphics. Website Navigation: Site Maps - URL Mapping and Routing - The SiteMapPath Control - The TreeView Control - The Menu Control. ADO.NET Fundamentals: The Data Provider Model - Direct Data Access - Disconnected Data Access.</p>			
<p>Unit – V: Data Binding: Single-Value Data Binding - Repeated-Value Data Binding - Data Source Controls - The Data Controls: The GridView - Formatting the GridView - Selecting a GridView Row - Editing with the GridView - Sorting and Paging the GridView - Using GridView Templates - The DetailsView and FormView – XML: The XML Classes - XML Validation - XML Display and Transforms. Website Security: Security Fundamentals - Understanding Security - Authentication and Authorization - Forms Authentication - Windows Authentication.</p>			

Recommended Texts

- 1) Matthew MacDonald, "Beginning ASP.NET 4 in C# 2010", Apress 2010.

Reference Books

- 1) Crouch Matt J, “ASP.NET and VB.NET Web Programming”, Addison Wesley 2002.
- 2) Mathew Mac Donald, “ASP.NET Complete Reference”, TMH 2005
- 3) J.Liberty, D.Hurwitz, “Programming ASP.NET”, Third Edition, O’REILLY, 2006.

E-learning resources

- 1) [https://msdn.microsoft.com/en-in/library/aa288436\(v=vs.71\).aspx](https://msdn.microsoft.com/en-in/library/aa288436(v=vs.71).aspx)
- 2) <http://www.asp.net/>

Elective-II: Computer Network	Semester - II	Credits: 3
<p>Objective of the course: Understand the theoretical concepts used in the present-day computer networks. Understand the design of wireless, cellular and adhoc networks. To have hands on experience with the protocol analyzer and simulation tools.</p>		
<p>Course Outcomes: After successful completion of this course, the students should be able to identify the various computer network protocol design models and the usage of various types of transmission media and working of LAN technology. Familiarize the various wireless LAN concepts and the design of adhoc wireless network.</p>		
<p>Unit 1: Introduction – Network Hardware – Software – Reference Models – OSI and TCP/IP models – Example networks: Internet, 3G Mobile phone networks, Wireless LANs –RFID and sensor networks - Physical layer – Theoretical basis for data communication - guided transmission media Unit-2: Wireless transmission - Communication Satellites – Digital modulation and multiplexing - Telephones network structure – local loop, trunks and multiplexing, switching. Data link layer: Design issues – error detection and correction. Unit 3: Elementary data link protocols - sliding window protocols – Example Data Link protocols – Packet over SONET, ADSL - Medium Access Layer – Channel Allocation Problem – Multiple Access Protocols. Unit 4: Network layer - design issues - Routing algorithms - Congestion control algorithms – Quality of Service – Network layer of Internet- IP protocol – IP Address – Internet Control Protocol. Unit 5: Transport layer – transport service- Elements of transport protocol - Addressing, Establishing & Releasing a connection – Error control, flow control, multiplexing and crash recovery - Internet Transport Protocol – TCP - Network Security: Cryptography.</p>		

Recommended Texts:

- 1) A. S. Tanenbaum, 2011, Computer Networks, Fifth Edition, Pearson Education, Inc.

Reference Books

- 1) B. Forouzan, 1998, Introduction to Data Communications in Networking, Tata McGraw Hill, New Delhi.
- 2) F. Halsall, 1995, Data Communications, Computer Networks and Open Systems, Addison Wesley.
- 3) D. Bertsekas and R. Gallager, 1992, Data Networks, Prentice hall of India, New Delhi.
- 4) Lamarca, 2002, Communication Networks, Tata McGraw Hill, New Delhi.

Website, E-learning resources

- 1) <http://peasonhighered.com/tanenbaum>

Core Paper - XI : Machine Learning	Year -2	Semester - 3	Credits : 4
---	---------	---------------------	-------------

Objective of the course :

To introduce the basic concepts and techniques of Machine Learning and to develop skills of using recent machine learning software for solving practical problems.

Course Outcomes :

After successful completion of this course, the students should be able to recognize the characteristics of machine learning that make it useful to real-world problems. Understand the foundation of generative models.

Unit 1:

The Fundamentals of Machine Learning: The Machine Learning Landscape - Types of Machine Learning Systems - Main Challenges of Machine Learning - Testing and Validating. End-to-End Machine Learning Project - Look at the Big Picture - Get the Data - Discover and Visualize the Data to Gain Insights - Prepare the Data for Machine Learning Algorithms - Select and Train a Model - Fine-Tune Your Model - Launch, Monitor, and Maintain Your System.

Unit 2:

Ingredients of machine learning: Tasks – Models – Features. Supervised Learning: Classification – Binary classification and related tasks – Scoring and ranking – class probability estimation – Multi-class classification. Unsupervised Learning: Regression – Unsupervised and descriptive learning. Concept Learning: The hypothesis space – paths through the hypothesis space – beyond conjunctive concepts – learnability.

Unit 3:

Tree Models: Decision trees – Ranking and probability estimation trees – tree learning as variance reduction. Rule Models: Learning ordered rule lists – learning unordered rule sets – descriptive rule learning – first-order rule learning. Linear Models: The least-squares method – The perceptron – Support vector machines.

Unit 4:

Distance-based Models: Neighbours and exemplars – Nearest-neighbour classification – Distance-based clustering – K-Means algorithm – Hierarchical clustering. Probabilistic Models: The normal distribution and its geometric interpretations – probabilistic models for categorical data – Naïve Bayes model for classification – probabilistic models with hidden values – Expectation-Maximization.

Unit 5:

Features: Kinds of features – Feature transformations – Feature construction and selection. Model ensembles: Bagging and random forests – Boosting – Mapping the ensemble landscape. Machine Learning experiments: What to measure – How to measure it – How to interpret it.

Text Books:

1. Flach, P, “Machine Learning: The Art and Science of Algorithms that Make Sense of Data”, Cambridge University Press, 2012
2. Aurélien Géron, “Hands-On Machine Learning with Scikit-Learn and Tensor Flow: Concepts, Tools, and Techniques to Build Intelligent Systems”, First Edition, 2017 (Chapters 1 and 2)

References

1. John D. Kelleher, Brian Mac Namee, Aoife D'Arcy, “Fundamentals of Machine Learning for Predictive Data Analytics: Algorithms, Worked Examples, and Case Studies”, The MIT Press, First Edition, 2012
2. Kevin P. Murphy, “Machine Learning: A Probabilistic Perspective”, MIT Press, 2012
3. Ethem Alpaydin, “Introduction to Machine Learning”, MIT Press, Third Edition, 2014
4. Tom Mitchell, "Machine Learning", McGraw-Hill, 1997
5. Stephen Marsland, “Machine Learning - An Algorithmic Perspective”, Chapman and Hall/CRC Press, Second Edition, 2014.

Core-XII: Practical- V: Machine Learning Lab	II Year & III Semester	Credit: 2
<p>Course Objective: To introduce the basic concepts and techniques of Machine Learning. To develop skills of using recent machine learning software for solving practical problems and gain experience of doing independent study and research.</p>		
<p>Course Outcomes: After successful completion of this course, student will be able to ability to identify the characteristics of datasets and compare the trivial data and big data for various applications.</p>		

Machine Learning Tools and Applications:

Machine learning platform: WEKA machine learning workbench, R platform, Python Scipy.

Machine Learning Library: scikit-learn in Python, JSAT in Java, Accord Framework in .NET

GUIs: KNIME, RapidMiner, Orange.

Applications: Prediction using data, Speech recognition, Healthcare, Object recognition in images, Natural Language Processing, Online search

1. Data Preprocessing:
 - a. Data Cleaning
 - b. Data Transformation
 - c. Data Reduction
 - d. Feature extraction
2. Supervised learning:
 - a. Decision tree classification
 - b. Classification using Support Vector Machines
 - c. Classification using Multilayer perceptron
3. Unsupervised learning:
 - a. Regression
 - b. K-Means clustering
 - c. Hierarchical clustering

Mini Project: Application of Data Preprocessing techniques and Machine Learning techniques on a data set selected from UCI repository / Kaggle / Government and submission of a report.

Core – XIII	Practical – VI: Mini Project	II Year & III Semester	Credit: 2
Course Objectives : To provide basic experience of small real time application development			
Course outcomes : After doing a small software the student can get confident to implement industries need projects.			

Individual students will develop and implement application software based on any emerging latest technologies.

Elective - III : Software Project Management	Year - II	Semester-3	Credit :3
Course objectives: Prescribe the conventional and evolution of software. Resolve the process of managing a software from conventional to modern. Analyze the architecture of a model-based software and the process flow.			
Course Outcomes: After successful completion of this course, the students should be able to design various estimation levels of cost and effort. Acquire the knowledge of managing, economics for conventional, modern and future software projects. Sketch various artifacts sets for better understanding of software development.			
<p>UNIT-I</p> <p>Introduction to Software Project Management- Software project versus other types of project- problems-management control- Stakeholders- Requirement Specification – Information and control in organizations Introduction to step wise project planning- Select-identify scope and objectives-identify project infrastructure-Analyse project characteristics- products and activities- Estimate effort for each activity- Identify activity risks- Allocate resources- Review/ publicize plan- Execute plan and lower levels of planning. Project evaluation- Introduction – Strategic assessment- technical assessment-cost benefit analysis- cash flow forecasting- cost- benefit evaluation techniques-risk evaluation</p> <p>UNIT-II</p> <p>Selection of an appropriate project approach- choosing technologiestechnical plan contents list- choice of process models- structured methods-rapid application development- waterfall model -process model-spiral model software prototyping- ways of categorizing prototypes- tools- incremental delivery- selecting process model -Software effort estimation- introduction where- problems with over and under estimates- basis for software estimating software effort estimation technique- expert judgement- Albercht function point analysis- Function points Mark II- Object points- procedural code oriented approach- COCOMO -Activity Planning- Objectives- Project schedules projects and activities- sequencing and scheduling activities- network planning models- formulating a network model- using dummy activities- representing lagged activities- adding time dimension- forward pass- backward pass identifying the critical path- Activity float- shortening project duration – identifying critical activities-precedence networks</p> <p>UNIT-III</p> <p>Risk Management- nature of risk- managing- identification-analysis reducing- evaluating- z values. Resource allocation- nature of resources requirements- scheduling- critical paths- counting the cost-resource schedule cost schedule- scheduling sequence. Monitoring and control- creating the frame work- collecting the data- visualizing the progress- cost monitoring-earned value- prioritizing, monitoring-Change control.</p> <p>UNIT=IV</p> <p>Managing contracts- types of contract- stages in contract placement terms of a contract-contract management-acceptance. Managing people and organizing teams- organizational behaviour background- selecting the right person for the job- instruction in the best methods-motivation- decision making leadership- organizational structures. Software quality- importance- defining – ISO 9126- practical measures- product versus process quality management external standards-techniques to help enhance software quality.</p> <p>UNIT-V</p> <p>Ethics in Information Technology – an Overview of ethics – Ethics for IT professionals and IT users, Computer and internet crime – privacy workplace monitoring – advanced surveillance technology.</p>			

Text Books

(i) B. Hughes and M. Cotterell, 2005, Software Project Management, 4th Edition, Tata McGraw Hill, New Delhi.

(ii) W. Royce, 1998, Software Project Management: A Unified Frame Work, Addison Wesley, Boston

(iii) G.Reynolds, 2003, Ethics in Information Technology, Thomson Learning, Singapore.

Reference Books

(i) K. Heldman, 2005, Project Management Professionals, 3 rd Edition, Wiley Dreamtech

(ii) Bhforooz & Hudson, 2004, Software Engineering, Oxford Press.

Website, E-learning resources

<http://highered.mcgraw-hill.com/sites/0077109899/information-center-view/>

back

Elective – IV	Cloud Computing	II Year & III Semester	Credit: 3
Course Objective: To provide foundation of the Cloud Computing so that they are able to start using and adopting Cloud Computing services and tools in their real life scenarios.			
Course Outcomes: After successful completion of this course, student will be able to Apply the fundamental concepts in data centres to understand the trade-offs in power, efficiency and cost. Identify resource management fundamentals, i.e. resource abstraction, sharing and sandboxing and outline their role in managing infrastructure in cloud computing. Analyse various cloud programming models and apply them to solve problems on the cloud.			

Unit 1: UNDERSTANDING CLOUD COMPUTING: Cloud Computing –History of Cloud Computing –Cloud Architecture –Cloud Storage –Why Cloud Computing Matters –Advantages of Cloud Computing –Disadvantages of Cloud Computing –Companies in the Cloud Today –Cloud Services

Unit 2: DEVELOPING CLOUD SERVICES: Web-Based Application –Pros and Cons of Cloud Service Development –Types of Cloud Service Development –Software as a Service –Platform as a Service- Infrastructure as a service –Web Services –On-Demand Computing –Discovering Cloud Services Development Services and Tools –Amazon Ec2 –Google App Engine –IBM Clouds

Unit 3: CLOUD COMPUTING FOR EVERYONE: Centralizing Email Communications –Collaborating on Schedules –Collaborating on To-Do Lists –Collaborating Contact Lists –Cloud Computing for the Community –Collaborating on Group Projects and Events –Cloud Computing for the Corporation

Unit 4: USING CLOUD SERVICES: Collaborating on Calendars, Schedules and Task Management –Exploring Online Scheduling Applications –Exploring Online Planning and Task Management –Collaborating on Event Management –Collaborating on Contact Management –Collaborating on Project Management –Collaborating on Word Processing –Collaborating on Databases –Storing and Sharing Files

Unit 5: OTHER WAYS TO COLLABORATE ONLINE: Collaborating via Web-Based Communication Tools –Evaluating Web Mail Services –Evaluating Web Conference Tools –Collaborating via Social Networks and Groupware –Collaborating via Blogs and Wikis

Recommended Text

- 1) Michael Miller, Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online, Que Publishing, August 2008.
- 2) Kumar Saurabh, “Cloud Computing –Insights into New Era Infrastructure”, Wiley Indian Edition, 2011.
- 3) Haley Beard, Cloud Computing Best Practices for Managing and Measuring Processes for On-demand Computing, Applications and Data Centers in the Cloud with SLAs, Emereo Pty Limited, July 2008

Elective – V	Mobile Application Development	II Year & III Semester	Credit: 3
Course Objective: To introduce mobile platforms and how to program these platforms and teamwork Presentation of methods for developing applications for mobile devices using their hardware and system capabilities.			
Course Outcomes: After successful completion of this course, student will be able to Student can define the specification of the complex program and according to the given specification - create a program for a mobile device using the right programming techniques.			

UNIT – I: Mobile Application Development - Mobile Applications and Device Platforms - Alternatives for Building Mobile Apps -Comparing Native vs. Hybrid Applications -The Mobile Application Development Life cycle-The Mobile Application Front-End-The Mobile Application Back-End-Key Mobile Application Services-What is Android-Android version history-Obtaining the Required Tools- Launching Your First Android Application-Exploring the IDE-Debugging Your Application-Publishing Your Application

UNIT – II: Understanding Activities-Linking Activities Using Intents-Fragments-Displaying Notifications-Understanding the Components of a Screen-Adapting to Display Orientation-Managing Changes to Screen Orientation- Utilizing the Action Bar-Creating the User Interface Programmatically Listening for UI Notifications

UNIT – III: Using Basic Views-Using Picker Views -Using List Views to Display Long Lists-Understanding Specialized Fragments - Using Image Views to Display Pictures -Using Menus with Views-Using WebView-Saving and Loading User Preferences-Persisting Data to Files-Creating and Using Databases.

UNIT – IV: Sharing Data in Android-Creating Your Own Content Providers -Using the Content Provider- SMS Messaging -Sending Email-Displaying Maps- Getting Location Data- Monitoring a Location.

UNIT – V: Consuming Web Services Using HTTP-Consuming JSON Services- Creating Your Own Services - Binding Activities to Services -Understanding Threading - Developing simple applications that uses radio button, image button, Alert dialog box, Layout managers and to display personal details using GUI components etc.

TEXT BOOK:

1. Jerome DiMarzio, “Beginning Android Programming with Android Studio”, 4th Edition, 2016.

REFERENCE BOOKS:

1. Dawn Griffiths, David Griffiths, “Head First Android Development: A Brain-Friendly Guide”, 2017.
2. Neil Smyth, “Android Studio 3.0 Development Essentials: Android”, 8th Edition, 2017.
3. Pradeep Kothari, “Android Application Development (With Kitkat Support)”, Black Book 2014.

WEB REFERENCES:

- <https://developer.android.com/guide>
- https://en.wikipedia.org/wiki/Android_10
- [Develop App for Free](#)
- <https://flutter.dev/>
- <http://ai2.appinventor.mit.edu>
- <https://aws.amazon.com/mobile-application-development/> (Unit 1)
- <https://flutter.dev/> (Applications)
- <http://ai2.appinventor.mit.edu> (Applications)

Core- XIV : Project & Viva-Voce	Year - II	Semester-IV	Credit :20
Course Objectives : to realize real time application environment for collecting data, design and implement according to the user requirements			
Course outcome: get confident to face industrial interview and acquire more opportunity in software development companies			

The project work is to be carried out either in a software industry or in an academic institution for the entire semester and the report of work done is to be submitted to the University.
