MANGALORE UNIVERSITY

Credits, Scheme of Examination and Syllabus

for

Master of Computer Applications (MCA)

Degree Programme

Choice Based Credit System (CBCS)

Department of Post-Graduate Studies and Research in Computer Science, Mangalore University

Mangalagangothri-574199

MASTER OF COMPUTER APPLICATIONS (MCA) DEGREE PROGRAMME

PROGRAMME EDUCATIONAL OBJECTIVES (PEOs):

The Programme Objectives of MCA programmes are:

PO1:To prepare graduates who will be successful professionals in industry, government, academia, research, entrepreneurial pursuit and consulting firms

PO2:To prepare graduates who will contribute to society as broadly educated, expressive, ethical and responsible citizens with proven expertise

PO3:To prepare graduates who will achieve peer-recognition; as an individual or in a team; through demonstration of good analytical, design and implementation skills

PO4:To prepare graduates who will thrive to pursue life-long learning to fulfill their goals

PROGRAMME OUTCOMES (POs):

MCA programme has been designed to prepare graduates for inculcating the following program outcomes:

PO1:Applying knowledge of mathematics, computer science and management in practice

PO2: To identify, critically analyze, formulate and develop computer applications

PO3: An ability to select modern computing tools and techniques and use them with dexterity

PO4: To design a computing system to meet desired needs within realistic constraints such as safety, security and applicability

PO5: An ability to devise and conduct experiments, interpret data and provide well informed conclusions

PO6: To understand the impact of system solutions in a contemporary, global, economical, environmental, and societal context for sustainable development

PO7: An ability to function professionally with ethical responsibility as an individual as well as in multidisciplinary teams with positive attitude

PO8: An ability to communicate effectively

PO9: An ability to appreciate the importance of goal setting and to recognize the need for life-long learning.

PROGRAMME SPECIFIC OUTCOMES (PSOs):

PSO1: Ability to test and analyze the quality of various subsystems and to integrate them in order to evolve a larger computing system.

PSO2: Ability to design and develop computing systems using concepts of Mathematics, Computer Engineering and other related disciplines to meet customers' business objectives.

I SEMESTER M.C.A.								
Subject Code	**Subjects**	**Theory Hours/ Week**	**Practical Hours/ Week**	**Duration of exams (Hours)**	**Marks & Credits**			
					IA	**Exam**	**Total**	**Credits**
HARD CORE								
MCAH101	Discrete Mathematical Structures	4L	-	3	30	70	100	4
MCAH102	Digital Computer Fundamentals	4L	-	3	30	70	100	4
MCAH103	Microprocessors and Peripherals	4L	-	3	30	70	100	4
MCAH104	Object Oriented Programming Using C++	4L	-	3	30	70	100	4
SOFT CORE								
MCAS105	Web Programming	4L	-	3	30	70	100	4
MCAP106	OOP Lab		4	3	30	70	100	2
MCAP107	Web Programming Lab		4	3	30	70	100	2
	Total	**20**	**8**	**21**	**210**	**490**	**700**	**24**

II SEMESTER M.C.A.								
Subject Code	**Subjects**	**Theory Hours/ Week**	**Practical Hours/ Week**	**Duration of exams (Hours)**	**Marks & Credits**			
					IA	**Exam**	**Total**	**Credits**
HARD CORE								
MCAH201	Advanced Operating System	4L	-	3	30	70	100	4
MCAH202	System Software	4L	-	3	30	70	100	4
MCAH203	Data Communications and Computer Networks	4L	-	3	30	70	100	4
MCAH204	Advanced Data Structures	4L	-	3	30	70	100	4
SOFTCORE								
MCAS205	Java Programming	4L	-	3	30	70	100	4
MCAP206	Advanced Data Structures Lab	-	4	3	30	70	100	2
MCAP207	Java Programming Lab	-	4	3	30	70	100	2
	Total	**20**	**8**	**21**	**210**	**490**	**700**	**24**

III SEMESTER M.C.A.								
Subject Code	**Subjects**	**Theory Hours/ Week**	**Practical Hours/ Week**	**Duration of exams (Hours)**	**Marks & Credits**			
					IA	**Exam**	**Total**	**Credits**
HARD CORE								
MCAH301	Advanced Database Management Systems	4L	-	3	30	70	100	4
MCAH302	Object Oriented Data Modeling Using UML	4L	-	3	30	70	100	4
MCAH303	Computer Graphics and Multimedia	4L	-	3	30	70	100	4
MCAH304	.Net Technology	4L	-	3	30	70	100	4
SOFT CORE								
MCAS305	Advanced Java Programming	4L	-	3	30	70	100	4
MCAP306 MCAP307	.NET Technology Lab DBMS LaB	-	4	3	30	70	100	2
MCAP308 MCAP309	Computer Graphics Lab UML Modeling Lab	-	4	3	30	70	100	2
Total		**20**	**8**	**21**	**210**	**490**	**700**	**24**

IV SEMESTER M.C.A.								
Subject Code	**Subjects**	**Theory Hours/ Week**	**Practical Hours/ Week**	**Duration of exams (Hours)**	**Marks & Credits**			
					IA	**Exam**	**Total**	**Credits**
HARD CORE								
MCAH401	Software Engineering	4L	-	3	30	70	100	4
MCAH402	Distributed Computing	4L	-	3	30	70	100	4
MCAH403	Advanced Web Programming	4L	-	3	30	70	100	4
SSOFTCORE								
MCAS404 MCAS405 MCAS406 MCAS407	Advanced Computer Network Image Processing Software Architecture Mobile Computing	4L	-	3	30	70	100	4
MCAS408 MCAS409 MCAS410 MCAS411	Wireless Communications Software Testing and Automation E-Commerce Operational Research	4L	-	3	30	70	100	4
MCAP412 MCAP413 MCAP414	Distributed Computing Lab Computer Network Lab Image Processing Lab	-	4	3	30	70	100	2
MCAP415 MCAP416 MCAP417	Advanced Web Programming Lab E Commerce Lab Operational Research Lab	-	4	3	30	70	100	2
Total		**20**	**8**	**21**	**260**	**490**	**700**	**24**

V SEMESTER M.C.A.								
Subject Code	**Subjects**	**Theory Hours/ Week**	**Practical Hours/ Week**	**Duration of exams (Hours)**	**Marks & Credits**			
					IA	**Exam**	**Total**	**Credits**
HARD CORE								
MCAH501	Data Mining Techniques	4L	-	3	30	70	100	4
MCAH502	Python Programming	4L	-	3	30	70	100	4
MCAH503	Android Applications Development	4L	-	3	30	70	100	4
SOFT CORE								
MCAS504 MCAS505 MCAS506 MCAS507	Big Data Analytics Cloud and Grid Computing Machine Learning Internet of Things	4L	-	3	30	70	100	4
MCAS508 MCAS509 MCAS510 MCAS511	Cryptography and Network Security Natural Language Processing Embedded Systems Artificial Intelligence	4L	-	3	30	70	100	4
MCAP512 MCAP513 MCAP514	Data Mining Lab Python Programming Lab Data Analytics Lab	-	4	3	30	70	100	2
MCAP515 MCAP516 MCAP517	Android Applications Lab Machine Learning Lab Artificial Intelligence Lab	-	4	3	30	70	100	2
Total		**20**	**8**	**21**	**260**	**490**	**700**	**24**

VI SEMESTER M.C.A.								
Subject Code	**Subjects**	**Theory Hours/ Week**	**Practical Hours/ Week**	**Duration of exams (Hours)**	**Marks & Credits**			
					IA	**Exam**	**Total**	**Credits**
MCAH601	Major Project Internal Assessment Project Report Valuation	-	40	-	120 --	-- 200	120 200	16
	Viva-Voce	-	-	-	--	80	80	-
Total			**40**		**120**	**280**	**400**	**16**

Grand Total Marks of all the SIX Semesters: 3900

Total Number of Credits: 136

Hard Core (88 Credits): 64.71%

Soft Core (48 Credits): 35.29%

I Semester MCA

MCAH101: DISCRETE MATHEMATICAL STRUCTURES

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exams. Marks: 70**

Course Outcomes:

CO1: To appreciate the basic principles of Boolean algebra, Logic, Set theory,

CO2: Permutations and combinations and Graph Theory.

CO3: Be able to construct simple mathematical proofs

CO4: Be able to understand logical arguments and logical constructs.

CO5: Have a better understanding of sets, functions, and relations.

CO6: Acquire ability to describe computer programs in a formal mathematical manner.

UNIT-I

12 Hours

Logic: Introduction, propositional logic, propositional equivalences, predicates and quantifiers, rules of inference.

Proofs: Introduction to proofs, proof methods.

12 Hours

UNIT-II

Sets, Functions and Relations: Sets, set operations, functions, relations, equivalence relations and partial ordering.

Counting: Basics of counting, the pigeonhole principle, permutations and combinations, Binomial Co-efficient, recurrence relations.

12 Hours

UNIT-III

Probability: Introduction to probability, axioms of probability, independence and conditional Probability, inclusion-exclusion principle.

UNIT-IV

12 Hours

Graph Theory: Graphs, terminology and special types of graphs, representation of graphs, Isomorphism, connectivity, Euler and Hamiltonian paths, shortest path problems, Planar graphs, graph coloring.

REFERENCE BOOKS

1. Kenneth H Rosen, **Discrete Mathematics and its Applications**, McGraw Hill, 2011, 7th edition.
2. Ralph P. Grimaldi and B V Ramana, **Discrete and Combinatorial Mathematics: An Applied Introduction**, Pearson, 2011, 5th edition.
3. Narsingh Deo, **Graph Theory with Applications to Engineering and Computer Science**, Prentice Hall India, 2004.
4. J. P. Tremblay and R. Manohar, **Discrete Mathematical Structures with Applications to Computer Science**, McGraw Hill

MCAH102 : DIGITAL COMPUTER FUNDAMENTALS

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exams. Marks: 70**

Course Outcomes:

CO1: Upon the completion of this course students will have full understanding of the following concepts

CO2: Bridge the fundamental concepts of computers with the present level of knowledge of the students.

CO3: Familiarize operating systems, programming languages, peripheral devices, networking, multimedia and internet

CO4: Understand binary, hexadecimal and octal number systems and their arithmetic.

CO5: Understand how logic circuits and Boolean algebra forms as the basics of digital computer.

CO6: Demonstrate the building up of Sequential and combinational logic from basic gates.

UNIT-I **12 Hours**

Computer Operation – Application of Computers to Problems, Scientific Applications, Business Applications, Components of Digital computer, Number Systems – Decimal, binary, octal and hexadecimal Number systems, Number Base Conversion, Binary addition and subtraction, Complements, BCD, negative number representation and operations, floating point representation.

UNIT-II **12 Hours**

Boolean algebra and Gate Networks – Boolean algebra, Evaluation of logical expressions, and standard forms, simplification of Boolean functions – map, tabulation method Logic Design, Digital Logic Gates, Wired OR and Wired AND Gates**.** Combinational logic – NAND, NOR circuits, Adders, Decoders, Multiplexers, ROM, PLA's and PALS

UNIT-III **12 Hours**

Sequential logic - Flip Flops, shift registers, counters, state diagram & state tables, Design of sequential circuits, Programmable Array of Logic cells. Arithmetic-logic unit – Half adder, Full adder, BCD adder, Magnitude Comparator, Multipliers, operation – study of typical ALU unit.

UNIT-IV **12 Hours**

Memory Element – RAMS – static, dynamic, ROMS, flexible disk storage system, magnetic disk memories and optical memories, Magnetic Tape, Tape Cassettes & Cartridges, and Digital Recording Techniques. Input-Output devices – Keyboards, Terminals, Printers, Alphanumeric codes, Cathode Ray tube Output Devices, Error detecting and correcting codes.

REFERENCE BOOKS

1. Thomas C. Bartee, "**Digital Computer Fundamentals**", Mc-Graw Hill, 1985, 6th edition.
2. Morris Mono M., "**Digital Logic and Computer Design**", PHI.
3. Morris Mano M, Kime R. Charles, "**Logic And Computer Design Fundamentals**", 2015, 5th edition

MCAH103 : MICROPROCESSORS AND PERIPHERALS

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exams. Marks: 70**

Course Outcomes:

CO1: The student will be able to analyze, specify, design, write and test assembly language programs of moderate complexity.

CO2: The student will be able to select an appropriate 'architecture' or program design to apply to a particular situation; e.g. an interrupt-driven I/O handler for a responsive real-time machine. Following on from this, the student will be able to design and build the necessary programs.

CO3: The student will be able to calculate the worst-case execution time of programs or parts of programs, and to design and build, or to modify, software to maximize its run time memory or execution-time behavior.

CO4: The student will be able to characterize and predict the effects of the properties of the bus on the overall performance of a system.

CO5: The student will be able to describe some of the characteristics of RISC and CISC architectures.

UNIT-I **12 Hours**

Microcomputer Structure
Overview of microcomputer structure and operation, microprocessor evolution and types. Microprocessor and 8086 Architecture: 8086 internal architecture, introduction to 8086 programming, 8086Instruction Set: 8086 instruction description and assembler directives

UNIT-II **12 Hours**

Programming the Microprocessor
8086 family assembly language programming – instruction templates, MOV instruction coding format and examples, writing programs for use with an assembler, assembly language program development tools. Implementing Standard Program Structures in 8086 Assembly Language – simple sequence programs, jumps, flags, and conditional jumps, if-then, if-then-else, and multiple if-then-else programs, while-do programs, repeat-until programs, instruction timing and delay loops

UNIT-III **12 Hours**

Strings, Procedures and Macros: String instructions in 8086, writing and using procedures, writing and using assembler macros.
Arithmetic Co-processor: Data formats for arithmetic co-processor, 80x87 architecture and instruction set.

UNIT-IV **12 Hours**

Interrupt Service Routine
8086 interrupts and interrupt responses, hardware interrupt applications, 8259A priority interrupt controller, software interrupt applications
Introduction To Advanced Microprocessors
Salient features of 80186,80286,80386,80486 and Pentium family processors.

Digital Interfacing
Programmable Parallel Ports and handshake I/O, methods of data transfer, implementing handshake data transfer.

REFERENCE BOOKS

1. Douglas V. Hall, **Microprocessors and Interfacing**, Revised 2nd Edition
2. Barry B. Brey, **Advanced Microprocessors**, 4th Edition
3. Liu and Gibson, **Microprocessors**, 2nd Edition
4. Barry B. Brey, **The Intel Microprocessors**, Prentice Hall, 2008, 8th Edition

MCAH104: OBJECT ORIENTED PROGRAMMING USING C++

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exams. Marks: 70**

Course Outcomes:

CO1: Understand the features of C++ supporting object oriented programming

CO2: Understand the relative merits of C++ as an object oriented programming language

CO3: Understand how to produce object-oriented software using C++

CO4: Understand how to apply the major object-oriented concepts to implement object oriented programs in C++, encapsulation, inheritance and polymorphism

CO5: Understand advanced features of C++ specifically stream I/O, templates and operator overloading

UNIT-I **12 Hours**

Language Basics: Object – oriented programming. Encapsulation. Polymorphism, Inheritance. The C++ Program – Pre-processor directives, A word about comments, a first look at input/output, C++ data types – pointer types, string types, const qualifier, reference types, bool type, enumeration types, array types, complex number types, typedef names, volatile qualifier, The new and delete expressions, Type conversions.

UNIT-II **12 Hours**

Procedural Programming: Functions – overview, function prototype, argument passing, Returning a value, recursion, inline functions, linkage directives, main(): handling command line options, pointers to functions, Scope and lifetime – scope, global objects and functions, local objects, dynamically allocated objects. Overload function – overloaded function declarations, the three steps of overload resolution, argument type conversions.

UNIT-III **12 Hours**

Class and Objects: Introduction, constructors and destructors, structures and classes, unions and classes – anonymous unions, friend functions, friend classes, inline functions. Parameterized constructors, static class members – Static data members, Static member function, Execution of constructors and destructors, scope resolution operator, nested classes, local classes, passing objects to functions, returning objects, object assignment. Arrays of objects – initialization v/s un initialization, Pointers to objects, Type checking pointers, this pointer, Pointers to derived types, pointers to class members. Creating a member operator function – overloading shorthand operators, operator overloading, and restriction. Operator overloading using a friend function – Using a friend to overload ++ or --, overloading some special operator: [], ->, comma operator.

UNIT-IV **12 Hours**

Core Concepts: Inheritance – Base class access control, Inheritance and protected members – protected base class inheritance. Inheriting multiple base classes, Constructors, Destructors and Inheritance – Execution of constructors and destructors, passing parameters to base class constructors, granting access, virtual base classes.

Virtual Function – Calling a virtual function through a base class reference, The virtual attribute is inherited, virtual functions are hierarchical, pure virtual function – abstract classes, using virtual function, early and late binding, Templates, Exception handling, File Handling.

REFERENCE BOOKS

1. Stanley B. Lippman and Josee Lajore, **C++ Primer,** Addision Wesley, 3rd Edition
2. Robort Lafore, **Object- Oriented Programming in Turbo C++**, Galgotia Publisher.
3. Herbert Schildt, **C++, The Complete Reference,** TMH, 3rd Edition
4. Bjarne Stroustrup, The C++ Programming Language, Pearson Education, 3rd Edition.
5. E. Balagurusamy, Object oriented Programming using C++, Tata MacGraw Hill Publishers

MCAS 105 : WEB PROGRAMMING

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: Implement interactive web page(s) using HTML, CSS and JavaScript.

CO2: Design a responsive web site using HTML5 and CSS3.

CO3: Build Dynamic web site using server side PHP Programming and Database connectivity.

CO4: Describe and differentiate different Web Extensions and Web Services.

CO5: Describe different Web Services Standards.

UNIT-I **12 Hours**

Introduction: Web Publishing, Web Browsers, Web Servers, URL; Essential Web Developer Tools; Web hosting. **HTML5 and CSS3:** Introduction, Basics – Structure, Essential Tags, Lists, Links; Formatting Text with HTML and CSS, Including Style Sheets in a Page, Varieties of Selectors, Units of Measure, Box Model, Using Images on Web Pages, Image Formats, Using Images – Basics, Text Alignments, Links, Scale, Backgrounds, Bullets; Image-map, Image Etiquettes. Tables, Creating Table, Parts of Table; Formatting Tables – Size, Borders, Cells; Alignment and Spacing; Spanning; Advanced Enhancements.

UNIT-II **12 Hours**

Using CSS to Position Elements: Positioning Schemes, Absolute Positioning, Fixed Positioning, Controlling Stacking, Creating Drop-Down Menus. **Designing HTML5 Forms:** Basics; Creating Controls, Buttons and Fields; Grouping Controls; Displaying Updates; Applying Styles. **Structuring a Page with HTML5:** History, Laying Out a Page, Structural Tags, Page Outline, Structural Elements. **Advanced CSS Page Layouts:** Laying Out Page, The Role of CSS in Web Design.

UNIT-III **12 Hours**

JavaScript and jQuery: JavaScript – Significance, Basics, Environment, Events, Validating Forms, Hiding and Showing Content, Adding New Content to a Page. **Using jQuery:** Introduction, JavaScript Libraries, Selecting Elements from the Document, Binding Events, Modifying Styles on the Page, Modifying Content on the Page, Special Effects, AJAX and jQuery.

UNIT-IV **12 Hours**

PHP: Introduction, Basics, Loops, Built-In Functions, User-Defined Functions, Processing Forms, Using PHP Includes, Database Connectivity, Regular Expressions, Sending Mail, Object-Oriented PHP, Cookies and Sessions, File Uploads.

REFERENCE BOOKS:

1. Laura Lemay et.al., **Sams Teach Yourself HTML, CSS & JavaScript Web Publishing in One Hour a Day**, Pearson Education, 2016, 7th Edition.
2. Jon Duckett, **Web Design with HTML, CSS, JavaScript and jQuery** (set), Wiley, 2014
3. Robert W. Sebesta, **Programming the World Wide Web**, Pearson Education

MCAP106: OOP LAB

Course Outcomes:

CO1: Understand the features of C++ supporting object oriented programming
CO2: Understand the relative merits of C++ as an object oriented programming language
CO3: Understand how to produce object-oriented software using C++
CO4: Understand how to apply the major object-oriented concepts to implement object oriented programs in C++, encapsulation, inheritance and polymorphism
CO5: Understand advanced features of C++ specifically stream I/O, templates and operator overloading

MCAP107: WEB PROGRAMMING LAB

Course Outcomes:

CO1: Analyze a web page and identify its elements and attributes.
CO2: Create web pages using XHTML and Cascading Style Sheets.
CO3: Build dynamic web pages using JavaScript (Client side programming).
CO4: Create XML documents and Schemas.

II Semester MCA

MCAH201 : ADVANCED OPERATING SYSTEM

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: Understanding the difference between a distributed and "traditional" system.

CO2: Identifying characteristics of distributed systems.

CO3: Ability to estimate if a system takes distributed system characteristic into account in a reasonable way.

CO4: Knowing the basic structures (e.g. client-server) and knowing the existing middleware frameworks.

CO5: Ability to estimate framework suitability for different applications.

CO6: Ability to implement a simple distributed software laboratory work with socket and RMI interfaces.

CO7: Understanding the mathematical principles behind validity of algorithms solving the problems of distribution.

CO8: Understanding the problems that will arise if atomicity and timing issues are not handled in a distributed application.

UNIT-I **12 Hours**

Operating System Overview : Operating System Objectives and Functions, The Evolution of Operating Systems, Major Achievements, Developments Leading to Modern Operating Systems, Microsoft Windows Overview, Traditional UNIX Systems, Modern UNIX Systems, Linux. **Process description & control:** What is a Process?, Process States, Process Description, Process Control, Execution of the Operating System, Security Issues, UNIX SVR4 Process Management.

UNIT-II **12 Hours**

Threads, SMP, and Microkernel: Processes and Threads, Symmetric Multiprocessing (SMP), Microkernels, Windows Vista Thread and SMP Management, Solaris Thread and SMP Management, Linux Process and Thread Management.. **Virtual Memory :** Hardware and Control Structures, Operating System Software, UNIX and Solaris Memory Management, Linux Memory Management, Windows Vista Memory Management, Summary.

UNIT-III **12 Hours**

Multiprocessor and Real-Time Scheduling: Multiprocessor Scheduling, Real-Time Scheduling, Linux Scheduling, UNIX PreclsSl) Scheduling, Windows Vista Scheduling.

Distributed Process Management: Process Migration, Distributed Global States, Distributed Mutual Exclusion, Distributed Deadlock. **Security:** Security Threats, Attacks, and Assets, Intruders, Malicious Software Overview, Viruses, Worms, and Bots, Rootkits.

UNIT-IV **12 Hours**

Kernel Organization: Using Kernel Services, Daemons, Starting the Kernel, Control in the Machine, Modules and Device Management, Module Organization, Module Installation and Removal, Process and Resource Management, Running Process Manager, Creating a new Task, IPC and Synchronization, The Scheduler, Memory Manager, The Virtual Address Space, The Page Fault Handler, File Management. **The windows NT/2000/XP kernel:** Introduction, The

NT kernel, Objects, Threads, Multiplication Synchronization, Traps, Interrupts and Exceptions, The NT executive , Object Manager, Process and Thread Manager, Virtual Memory Manager, I/o Manager, The cache Manager , Kernel local procedure calls and IPC, The native API, subsystems.

REFERENCE BOOKS

1. William Stallings: **Operating Systems: Internals and Design Principles**, Prentice Hall, 2013, 6th Edition.
2. Gary Nutt: **Operating Systems**, Pearson, 2014, 3rd Edition.
3. Silberschatz, Galvin, Gagne: **Operating System Concepts**, Wiley, 2008, 8th Edition.
4. Andrew S. Tanenbaum, Albert S. Woodhull: **Operating Systems, Design and Implementation**, Prentice Hall, 2006, 3rd Edition.
5. Pradeep K Sinha: **Distributed Operating Systems, Concept and Design**, PHI, 2007.

MCAH202 : SYSTEM SOFTWARE

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: To compare the machine dependent and machine independent assembler features

CO2: To implement assembly language programs using MASM

CO3: To implement the assembly code using MSDOS linker

CO4: To analyze the functions and capabilities of debugging system

CO5: To estimate the user interface criteria

UNIT-I **12 Hours**

Introduction

System software and machine architecture The Simplified Instructional Computer (SIC) – Machine architecture - Data and instruction formats - addressing modes - instruction sets - I/O and programming.

Assemblers

Basic assembler functions - A simple SIC assembler – Assembler algorithm and data structures Machine dependent assembler features Instruction formats and addressing modes - Program relocation- Machine independent assembler features - Literals Symbol defining statements - Expressions – Onepass assemblers and Multi pass assemblers - Implementation example - MASM assembler.

UNIT-II **12 Hours**

Loaders and linkers

Basic loader functions - Design of an Absolute Loader – A Simple Bootstrap Loader -Machine dependent loader features - Relocation – Program Linking –Algorithm and Data Structures for Linking Loader -Machine-independent loader features -Automatic Library Search – Loader Options - Loader design options - Linkage Editors –Dynamic Linking – Bootstrap Loaders - Implementation example - MSDOS linker.

UNIT-III **12 Hours**

Macro processors Basic macro processor functions - Macro Definition and Expansion Macro Processor Algorithm and data structures, Machine independent macro processor features - Concatenation of Macro Parameters Generation of Unique Labels Conditional Macro Expansion – Keyword Macro Parameters-Macro within Macro-Implementation example - MASM Macro Processor – ANSI C Macro language.

UNIT-IV **12 Hours**

System software tools

Text editors - Overview of the Editing Process - User Interface – Editor Structure. -Interactive debugging systems - Debugging functions and capabilities –Relationship with other parts of the system – User-Interface Criteria.

REFERENCE BOOKS

1. Leland L. Beck, "**System Software –An Introduction to Systems Programming**", 3rd
2. Edition, Pearson Education Asia, 2000
3. D. M. Dhamdhere, "**Systems Programming and Operating Systems**", Second
4. Revised Edition, Tata McGraw-Hill, 1999.
5. John J. Donovan "**Systems Programming**", Tata McGraw-Hill Edition, 1972.
6. John R. Levine, **Linkers & Loaders**, Harcourt India Pvt. Ltd., Morgan Kaufmann Publishers, 2000.

MCAH 203 : DATA COMMUNICATIONS AND COMPUTER NETWORKS

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: The outcome of this course is to master the fundamentals of data communications networks by gaining a working knowledge of data transmission concepts

CO2: Learning the methods used for line control and line sharing

CO3: Understanding the operation of compression algorithms used to optimizing data transfer

CO4: Learning the methods used to provide security for data sent over a network

CO5: Understanding the operation of physical and data link protocols

CO6: Learning network performance techniques

CO7: Understanding the concepts and operation of local and IP-based networks.

UNIT-I **12 Hours**

Introduction: Data communications fundamentals, Communication model, computer communications architecture, Data Communication tasks, Data Communication Systems Applications, Data Communication System Characteristics features, Data Communication Network criteria, Protocols and standards, Standards Organizations, Line Configuration, Topology, Transmission mode, Categories of Networks. Signals: Analog /Digital data and Signals, Periodic and Aperiodic Signals, Time and Frequency Domains, Composite Signals. Transmission rate, Bit rate, Baud rate and signal levels, Channel capacity using Nyquist and Shannon's relation.

UNIT-II **12 Hours**

Encoding and Modulating: Digital to Digital Conversion, Analog to Digital Conversion, Analog to Analog Conversion, Digital to Analog Conversion, Modulation and Demodulation: Data modulation methods: ASK, FSK, PSK, QAM, PCM, PAM, POLAR, BIPOLAR, NRZ, RZ. Transmission of Digital data: Interfaces and Modems: Digital Data transmission, DTE- DCE interface, Other Interface Standards, Modem features, Types of Modem and functions of MODEM. Transmission media, Guided media, Unguided media, Transmission impairments and Performance. Multiplexing Techniques.

UNIT-III **12 Hours**

Definition of Computer Networks, Goals and Applications. ISO-OSI Architecture, Functions and Services of Physical, Data link, Network, Transport, Session, Presentation and Application Layers. Classifications of Computer Networks: Local Area Network(LAN), Wide Area Network WAN, Metropolitan Area Network, Storage Area Network(SAN), Public and Private Networks, Value Added Network(VAN), Internet works, TCP/I Preference Model, Novell Netware Reference Model. Standards of Networks. Distributed Applications

UNIT-IV **12 Hours**

Physical Layer Services, Data Link Layer Services and Network Layer Services: Point–to – Point Protocol (PPP), Networking and Internet working Technology Devices, Repeaters, Bridges, Routers, Gateways, TCP/IP Protocol Suit: Overview of TCP/IP, Classes of IP, Addressing, Address Resolution Protocol (ARP), Reverse Address Resolution Protocol (RARP), Internet Control MESSAGE Protocol (ICMP), Internet Group Message Protocol

(IGMP). Upper OSI Layers: Transport Layer, Session Layer, Presentation and Application Layer services. BOOTP, Dynamic Host Configuration Protocol (DHCP), Domain Name System (DNS), Telnet, Fle transfer Protocol (FTP), Trivial File Transfer Protocol (TFTP), Simple Mail Transfer Protocol (SMTP), Post Office Protocol (POP), Simple Network Management Protocol (SNMP), Hyper Text Transfer Protocol(HTTP), World Wide Web (WWW).

REFERENCE BOOKS

1. William Stallings– Data & Computer Communications, PHI, 6th ed.
2. Behrouz A Forouzan-Data Communication & Networking, McGrawHill, 2000, 2nd edition
3. W. Tomasi– Advanced Electronic Communication Systems.
4. Forouzan, B.A., "TCP/IPProtocol", TMH
5. Laura Chappell (ed),"Introduction to Cisco Router Configuration", Techmedia, 1999.
6. Tananbaum A.S., "Computer Networks", 3rd Ed, PHI, 1999.
7. Black U.,"Computer Networks-Protocols, Standards and Interfaces", PHI,1996.
8. Stallings W., "SNMP, SNMPv2, SNMPv3, RMON 1&2", 3rd Ed., Addison Wesley.

MCAH204 : ADVANCED DATA STRUCTURES

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exams. Marks: 70**

Course Outcomes:

CO1: Ability to analyze algorithms and algorithm correctness.

CO2: Ability to summarize searching and sorting techniques

CO3: Ability to describe stack, queue and linked list operation.

CO4: Ability to have knowledge of tree and graphs concepts.

CO5: Learn the notions of data structure, Abstract Data Type.

CO6: Understand Big(O) notation and role of algorithm complexity in computing

CO7: To evaluate various methods of linked list formulation. Also explore different kinds of linked lists and their applications in day to day problem solving.

CO8: To evaluate various formulation of queues. Also explore different kinds queues and their applications and implementations in simulations.

UNIT-I **12 Hours**

Introduction: Algorithms, performance analysis-time complexity and space complexity, Pseudo-Code, Quick Mathematical Review, O-notation, Omega notation and Theta notation. Stacks, Queues, Linked Lists, Double-Ended Queues. Trees: The Tree Abstract Data Type, Basic Algorithms on Trees, Binary Trees, Data Structures for Representing Trees, Priority Queues Abstract Data Type , Heaps

UNIT-II **12 Hours**

Search Trees: Binary Search Trees, definition, ADT, implementation, operations-searching, insertion and deletion, Balanced search trees- AVL trees, definition, height of an AVL tree, representation, operations-insertion, deletion and searching. Introduction to Red – Black trees and Splay Trees, B-Trees, insertion, deletion and searching, Comparison of Search Trees.

UNIT-III **12 Hours**

Introduction to Set, Implementation, Basic Operations on Set, Graphs, Directed Graphs ,Shortest Path Problem ,Undirected Graph ,Spanning Trees ,Graph Traversals, hash table representation, hash functions, collision resolution, separate chaining, open addressing, linear probing, quadratic probing double hashing, rehashing.

UNIT-IV **12 Hours**

Searching Techniques, Sorting, Internal Sorting, Bubble Sort, Insertion Sort, Quick, Sort, Heap Sort, Bin Sort, Radix Sort, External Sorting, Merge Sort, Multiway Merge Sort, Polyphase Sorting, Design Techniques: Divide and Conquer, Dynamic Programming, Greedy Algorithm, Backtracking, Local Search Algorithms

REFERENCE BOOKS

1. Mark A. Weiss, “Data structures and Algorithm analysis in C++(Java)”, Fourth Edition, PHI ,2013
2. Michael T. Goodrich, R. Tamassia and D. Mount “Data structures and Algorithms in C++”, Wiley student edition, John Wiley and Sons.
3. Data Structures and Algorithms in C++, Second Edition, Adam Drozdek, Vikas Publishing House, Thomson International Student Edition.
4. Ellis Horowitz, Sartaj Sahni and Sanguthevar Rajasekaran, “Computer Algorithms/ C++”, Second Edition, Universities Press, 2007.
5. Michael T. Goodrich and Roberto Tamassia, “Data Structure and Algorithms in Java”, 3rd edition, ISBN: 0-471-46983-1.

MCAS205: JAVA PROGRAMMING

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: Knowledge of the structure and model of the Java programming language, (knowledge)

CO2: Use the Java programming language for various programming technologies, (understanding)

CO3: Develop software in the Java programming language, (application)

CO4: Evaluate user requirements for software functionality required to decide whether the Java programming language can meet user requirements (analysis)

CO5: Propose the use of certain technologies by implementing them in the Java programming language to solve the given problem (synthesis)

CO6: Choose an engineering approach to solving problems, starting from the acquired knowledge of programming and knowledge of operating systems. (evaluation)

UNIT-I 12 Hours

Introduction: Java and Java Applications, Features, Bytecode and Interpretation, JDK, JVM; Object-Oriented Programming, Simple Programs; Data Types, Variables, Arrays and Type Conversions; Operators and Expressions; Control Statements: Selection Statements, Iteration Statements and Jump Statements. **Classes and Objects:** Classes in Java, Declaring a Class, Creating Instances of Class, Members of a Class, Method Overloading; Different Types of Constructors, Inner Class; Uses of this Keyword; Garbage Collection; Recursion; Access Control; Static Members.

UNIT-II 12 Hours

Inheritance: Introduction; Method Overriding and Dynamic Method Dispatch; Uses of super and final Keywords; Command Line Arguments; Varargs; Enumerations; **Exception Handling:** Exception Handling in Java. **Packages and Interfaces:** Packages, Importing Packages; Interfaces. **I/O:** Basics, Console I/O, Reading and Writing Files; **Generics**: Overview, Examples, Multiple Generic Parameters, Bounds, Wildcards, Generic Methods, Interfaces and Classes. **Collections:** Overview, Interfaces, Classes – Array List, LinkedList, HashSet and Map.

UNIT-III 12 Hours

Multi-threaded Programming: Introduction; Creating Threads: Extending Threads; Implementing Runnable; Synchronization, Priorities, Inter-Thread Communication, Thread States and Methods on Thread Objects. **Event Handling:** Two Event Handling Mechanisms; The Delegation Event Model; Event Classes; Sources of Events; Event Listener Interfaces; Using the Delegation Event Model; Adapter Classes; Inner Classes.

UNIT-IV **12 Hours**

Lambda Expressions: Introduction, Block Lambda Expressions, Generic Functional Interfaces, Passing Lambda Expressions as Arguments, Exceptions, Variable Capture, Method References, Constructor References, Predefined Functional Interfaces. **Swing:** The Origins of Swing; Two Key Swing Features; Components and Containers; The Swing Packages; A Simple Swing Application; JLabel; ImageIcon; JTextField; The Swing Buttons; Understanding Layout Managers; JTabbedPane; JScrollPane; JList; JComboBox; JTable; Overview of Menu.

REFERENCE BOOKS:

1. Herbert Schildt, **Java - The Complete Reference** – McGraw Hill Education, 2014, 9th Edition.
2. Kathy Sierra and Bert Bates, **Head First Java**, O'Reilly, 2005, 2nd Edition.
3. Joshua Bloch, **Effective Java**, Addison Wesley, 2008, 2nd Edition.

MCAP 206: ADVANCED DATA STRUCTURES LAB

Course Outcomes:

CO1: Develop solutions for a range of problems using object oriented programming.
CO2: Apply divide and conquer strategy to searching and sorting problems using iterative and / or recursive solutions.
CO3: Use critical thinking skills and creativity to solve the problems.
CO4: Design scenarios to explain behaviors and demonstrate correctness of programs.
CO5: Determine which algorithm or data structure to use in different scenarios.
CO6: Choose the appropriate data structure and algorithm design method for a specified application.
CO7: Analyze performance of algorithms.

MCAP 207: JAVA PROGRAMMING LAB

Course Outcomes:

CO1: Implement Object Oriented programming concept using basic syntaxes of control Structures, strings and function for developing skills of logic building activity.
CO2: Identify classes, objects, members of a class and the relationships among them needed for a finding the solution to specific problem
CO3: Demonstrates how to achieve reusability using inheritance, interfaces and packages and describes faster application development can be achieved.
CO4: Demonstrate understanding and use of different exception handling mechanisms and concept of multithreading for robust faster and efficient application development.
CO5: Identify and describe common abstract user interface components to design GUI in Java using Applet & AWT along with response to events Identify, Design & develop complex
CO6: Graphical user interfaces using principal Java Swing classes based on MVC architecture

III Semester MCA

MCAH 301 : ADVANCED DATABASE MANAGEMENT SYSTEMS

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: Explain the features of database management systems and Relational database.

CO2: Design conceptual models of a database using ER modeling for real life applications and also construct queries in Relational Algebra.

CO3: Create and populate a RDBMS for a real life application, with constraints and keys, using SQL.

CO4: Retrieve any type of information from a data base by formulating complex queries in SQL.

CO5: Analyze the existing design of a database schema and apply concepts of normalization to design an optimal database.

CO6: Build indexing mechanisms for efficient retrieval of information from a database.

UNIT-I **12 Hours**

Review on Fundamentals of Databases: Concepts, Architecture, ER Modelling, Relational Databases, Fundamental and extended operations in Relational Database Model, SQL, basic and Complex queries in SQL. **Database Normalization and Security**: Normalization of Database Tables, Transaction and concurrency control, Database security, Authorization and Encryption.

UNIT-II **12 Hours**

Object Relational Database Systems: Objects, Object Identity, and Reference Types, Inheritance, Features of Object-relational Systems, Database Design for an ORDBMS, New Challenges in Implementing an ORDBMS, OODBMS, Comparing RDBMS with OODBMS and ORDBMS. **Emerging Database Technologies**: Active Database Concepts and Triggers, Temporal Database, Multimedia Databases, Spatial Databases, Geographic Information Systems(GIS) and Deductive Databases.

UNIT-III **12 Hours**

Distributed Databases: Introduction to Distributed DBMS, Client-Server Model, Data Fragmentation, Replication, and Allocation Techniques for Distributed Database Design. Types of distributed database systems: - Federated database systems, Multi-database systems; Query processing in distributed databases. **XML and Internet Databases:** Structured, unstructured and semi structured data, XML Hierarchical data model, XML document, DTD and XML Schema XML documents and databases, and XML query.

UNIT-IV **12 Hours**

Data Warehousing: Characteristics of data warehouses, Data warehousing Components – Building a Data warehouse, Typical functionality of a data warehouse: Mapping the Data Warehouse to a Multiprocessor Architecture – DBMS Schemas for Decision Support – Data Extraction, Cleanup, and Transformation Tools –Metadata.

REFERENCE BOOKS

1. Raghu Ramakrishnan and Jhonnes Gehrke: Database Management Systems, McGraw Hill 2000, Second Edition.
2. Elmasri and Navathe: Fundamentals of Database Systems, Addison-Wesley, 1999, Sixth Edition.

MCAH 302 : OBJECT-ORIENTED DATA MODELLING USING UML

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: To learn the Unified Modeling (UML) : Use Case diagrams, state diagrams, sequence diagrams, Communication diagrams and Activity diagrams

CO2: To learn the concepts of objects, Classes, Methods

CO3: To learn relation of a functional to object and dynamic models

CO4: Basic principles of Software engineering: system analysis, design, testing and debugging

CO5: To learn the concepts of abstract classes and interfaces

CO6: To translate a design into an implementation

UNIT-I **12 Hours**

The object Model, the evolution of object model, elements of object model, applying the object model, Classes and Objects, Relationships among objects, the nature of a class, relationship among classes, the interplay of classes and objects, on building quality classes and objects (selected topics from Grady Booch)

UNIT-II **12 Hours**

Advanced object Modelling, Aggregation, Abstract Classes, Generalization as extension and Restriction, Multiple inheritance, Metadata, Candidate Keys, Constraints. Dynamic Modelling – events and states, operations nested state diagrams, Concurrency, Functional Modelling, Data Flow Diagrams, specifying operations, Constraints, Relation of Functional to object and Dynamic Models.

UNIT-III **12 Hours**

Design Methodology, OMT as a software engineering methodology, Analysis, overview of analysis, Problem statement, overview of system Design, Breaking a system into subsystems, identifying Concurrency, Allocating subsystems to processes and tasks, Management of data stores, Handling global resources, choosing software control implementation, Handling Boundary condition, setting trade off priorities, Common architectural frameworks.

UNIT-IV **12 Hours**

Object Design, Overview of object design, Combining the three models, Design algorithms, Design optimization, implementation of Control adjustment of inheritance, Design of Association, object representation, Physical packaging. Implementation, from Design to implementation object-oriented style, Reusability, extensibility, Robustness, Object Oriented languages, translating a Design into an implementation.

REFERENCE BOOKS

1. James Rumbaugh et.al, Object-Oriented Modelling and Design, PHI, 1991.
2. Grady Booch et.al, Object-Oriented Analysis and Design with Applications, 2007, Wesley, 3rd Edition

MCAH 303 : COMPUTER GRAPHICS AND MULTIMEDIA

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exams. Marks: 70**

Course Outcomes:

CO1: To list the basic concepts used in computer graphics.
CO2: To implement various algorithms to scan, convert the basic geometrical primitives, transformations, Area filling, clipping.
CO3: To describe the importance of viewing and projections.
CO4: To define the fundamentals of animation, virtual reality and its related technologies.
CO5: To understand atypical graphics pipeline
CO6: To design an application with the principles of virtual reality.

UNIT-I **12 Hours**

An Introduction Graphics System: Computer Graphics and Its Types, Application of computer graphics, Graphics Systems: Video Display Devices, Raster Scan Systems, Random Scan Systems, Graphics Monitors and Work Stations, Input Devices, Hard Copy Devices, Graphics Software.

UNIT-II **12 Hours**

Output Primitives and Attributes of Output Primitives: Output Primitives Points and Lines, Line Drawing Algorithms, Circle Generating Algorithms, Scan-Line Polygon Fill Algorithm, Inside-Outside tests, Boundary-Fill Algorithm, Flood Fill Algorithm, Cell Array, Character Generation, Attributes of Output Primitives: Line attributes, Color and Grayscale Levels, Area fill Attributes, Character Attributes, Bundled Attributes. Anti-aliasing.

UNIT-III **12 Hours**

Two-dimensional Geometric Transformations: Basic Transformations, Matrix Representations and Homogeneous Coordinates, Composite Transformations, Reflection and Shearing **Two-Dimension Viewing:** The viewing Pipeline, Window to view port coordinate transformation, Clipping Operations, Point Clipping, Line Clipping, Polygon Clipping, Text Clipping, Exterior Clipping Three–Dimensional Concepts: Three Dimensional Display Methods, 3D Transformations, Parallel Projection and Perspective Projection.

UNIT-IV **12 Hours**

Multimedia: Introduction to Multimedia: Classification of Multimedia, Multimedia Software, Components of Multimedia – Audio: Analog to Digital conversion, Sound card fundamentals, Audio play backing and recording Video, Text: Hyper text, Hyper media and Hyper Graphics, Graphics and Animation: Classification of Animation. Authoring Process and Tools. **Case Study:** graphics software MatLab, Use of MatLab in graphics application, Features of MatLab, Generalize application by using MatLab.

REFERENCE BOOKS

1. Donald Hearn and M.Pauline Baker, Computer Graphics, PHI
2.Roy A. Plastock, Theory & Problem of Computer Graphics, Tata McGraw Hill.
3. J D Foley and Van Dam, Fundamentals of Interactive Computer Graphics, Addison-Wesley.
4. Newman, Principles of Interactive Computer Graphics, McGraw Hill.
5. Tosijasu, L.K., Computer Graphics, Springer.
6. S Gokul, Multimedia Magic, BPB Publication.
7. Bufford, Multimedia Systems, Addison Wesley.
8. Jeffcoate, Multimedia in Practice, Pretice-Hall.

MCAH 304 : NET TECHNOLOGY

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: Students will learn to develop applications using C# and VB.NET.

CO2: They will also learn to apply these languages to develop server-side applications which make use of ADO.NET, ASP.NET, Web Services etc.

CO3: Create user interactive web pages using ASP.Net. 2.

CO4: Create simple data binding applications using ADO.Net connectivity.

CO5: Performing Database operations for Windows Form and web applications.

UNIT-I **12 Hours**

The philosophy of .NET: Introducing the Building Blocks of the .NET Platform (the CLR, CTS, and CLS), The Role of the Base Class Libraries and Managed vs. Unmanaged Code. An overview of .NET Assemblies. Understanding the CTS, CLS, CLR.

Building C# applications: Building C# Applications Using csc.exe, Building .NET Applications Using Visual Studio.

C# Language Fundamentals: C# programming constructs, System Data Types and Corresponding C# Keywords, The System. Console Class, Data Type Conversions, Operators and Expressions, Working with String Data, C# Iteration Constructs, Decision Constructs and the Relational/Equality Operators, Methods and Parameter Modifiers, Understanding C# Arrays, Understanding the enum Type, Understanding the Structure Type, Understanding Value Types and Reference Types, Understanding C# Nullable Types.

Object-Oriented Programming with C#: Encapsulation, Defining the Pillars of OOP,C# Access modifiers, Properties, Inheritance and Polymorphism.

UNIT-II **12 Hours**

Understanding Structured Exception Handling: The Role of .NET Exception Handling, System level Exceptions, Application level Exceptions.

Working with Interfaces: Defnition, Implementation, Advanced keywords in C#, Interface as parameters, IEnumerable, IEnumerator, IConvertible Interfaces.

Delegates, Events, and Lambda Expressions: Definition, Multicast Delegate, Generic Delegates, Event Keyword, C# Anonymous methods, Lambda Expressions.

Advanced C# Language Features: Indexer Methods, Operator Overloading, CustomType Conversions, Pointer Types, Extension Methods.

LINQ to Objects: LINQ-Specific Programming Constructs, Role of LINQ, Applying LINQ Queries to Primitive Arrays, Returning the Result of a LINQ Query.

UNIT-III 12 Hours

Understanding .Net Assembles: An Overview of .NET Assembly, Building and Configuring Class Libraries, The Role of .NET Assemblies, Building the Multi-file Assembly, Using Assembly, Understanding Private Assemblies, Probing for Private Assemblies (The Basics), Understanding Shared Assembly, Understanding Shared Names, Building a Shared Assembly, Understanding Delay Signing, Installing/Removing Shared Assembly.

Dynamic Types and the Dynamic Language Runtime: The Role of the C# dynamic Keyword, The Role of the Dynamic Language Runtime (DLR).

ADO.NET Part I: The Connected Layer: Definition,ADO.NET Data Providers,ADO.NET Namespaces, Understanding the Connected Layer of ADO.NET, Building a Reusable Data Access Library, Understanding Database Transactions.

UNIT-IV 12 Hours

ADO.NET Part II: The Disconnected Layer: Understanding the Disconnected Layer of ADO.NET, Understanding the Role of the Dataset, The Windows Forms Database Designer Tools.

ASP.NET Web Forms: Introduction, Role of HTTP, Understanding Web Applications and Web Servers, HTML, Client-Side Scripting, Overview of the ASP.NET API, Building an ASP.NET Web Page Using Code Files, ASP.NET Web Sites vs. ASP.NET Web Applications.

ASP.NET Web Controls, Master Pages, and Themes: Nature of Web Controls, Control and Web Control Base Classes, Categories of ASP.NET Web Controls, Role of the Validation Controls, Role of the Validation Controls.

REFERENCE BOOKS:

1. Andrew Troelsen, Sixth Edition, **Pro C# 5.0 and the .NET 4.5 Framework** Apress, India, 2012, 6th Edition.
2. E. Balagurusamy, **Programming in C#,** Tata McGraw Hill. (For Programming Examples)
3. Tom Archer, **Inside C#**, WP Publishers, 2001.
4. Herbert Schildt, **C#: The Complete Reference** Tata McGraw Hill, 2004.

MCAS 305 : ADVANCED JAVA PROGRAMMING

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: Implement Object Oriented programming concept using basic syntaxes of control Structures, strings and function for developing skills of logic building activity.

CO2: Demonstrate understanding and use of different exception handling mechanisms and concept of multithreading for robust faster and efficient application development.

CO3: Learn to access database through Java programs, using Java Data Base Connectivity (JDBC).

CO4: Students will also be exposed to advanced topics including multithreading, internet networking, and JDBC database connectivity.

CO5: Develop server side programs in the form of servlets.

UNIT-I **12 Hours**

Java 2 Enterprise Edition Overview: Overview of J2EE.

Java Database Connectivity (JDBC): The Concept of JDBC; JDBC Driver Types; JDBC Packages; A Brief Overview of the JDBC Process; Database Connection; Statement Objects; Result Set; Transaction Processing; Metadata, Data Types; Exceptions.

UNIT-II **12 Hours**

Networking: Basics, Useful Classes and Interfaces, Inet Address, Sockets, URI, URL, URL Connection, Http URL Connection, Datagrams.

Java Beans: Introduction, Advantages, Introspection, Bound and Constrained Properties, Persistence, Customizers, Java Beans API.

UNIT-III **12 Hours**

Servlets: Background, Life Cycle, Development Options, Tomcat, Example, Servlet API, Reading Parameters, javax.servlet.http Package, Handling HTTP Requests and Responses, Using Cookies, Session Tracking.

Java Server Pages (JSP): JSP; JSP Tags; Tomcat; Request String; User Sessions; Cookies; Session Objects.

UNIT-IV **12 Hours**

Enterprise Java Beans: Enterprise Java Beans; Deployment Descriptors; Session Java Bean; Entity Java Bean; Message-Driven Bean; The JAR File.

Hibernate: Persistence, Problems in Object/Relational Mapping, Object/Relational Mapping and Java Persistence API; Introduction to Hibernate, Creating a JPA Project.

REFERENCE BOOKS:

1. Jim Keogh, **J2EE - The Complete Reference**, Tata McGraw Hill, 2008.
2. Herbert Schildt, **Java - The Complete Reference**, McGraw Hill Education, 2014, 9th Edition.
3. Gavin King et.al., **Java Persistence with Hibernate**, Manning Publications, 2016,2nd Edition.
4. Cameron McKenzie, **Hibernate Made Easy**, Pulpjava, 2008
5. Phil Hanna, **JSP 2.0: The Complete Reference**, Osborne

MCAP 306: NET PROGRAMMING

Course Outcomes:

CO1: Create user interactive web pages using ASP.Net
CO2: Create simple data binding applications using ADO.Net connectivity.
CO3: Performing Database operations for Windows Form and web applications.
CO4: Develop server-side applications

MCAP307: DBMS LAB

Course Outcomes:

CO1: Design and implement a database schema for a given problem domain.
CO2: Populate and query a database using SQL DDL/DML commands.
CO3: Program in PL/SQL including stored procedures, stored functions, cursors, packages.
CO4: Design and build a GUI application using a 4GL

MCAP308: Computer Graphics Lab

Course Outcomes:

CO1: Using OpenGL for Graphics
CO2: Programming User-interface issues
CO3: Concepts of 2D & 3D object representation
CO4: Implementation of various scan & clipping algorithms
CO5: 2D modeling
CO6: Implementation of illumination model for rendering 3D objects
CO7: Visibility detection & 3D viewing
CO8: Implementation of a project based on learned concepts.

MCAP309: UML Modelling Lab

Course Outcomes:

CO1: Demonstrate the Conceptual model of UML and SDLC.
CO2: Define classes modeling techniques and instances modeling techniques.
CO3: Describe interaction diagrams and their modeling techniques.
CO4: Demonstrate activity diagram and their modeling techniques.
CO5: Demonstrate component and deployment diagram

IV Semester MCA

MCAH401 : SOFTWARE ENGINEERING

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: Define various software application domains and remember different process model used in software development.

CO2: Explain needs for software specifications also they can classify different types of software requirements and their gathering techniques.

CO3: Convert the requirements model into the design model and demonstrate use of software and user interface design principles.

CO4: Distinguish among SCM and SQA and can classify different testing strategies and tactics and compare them.

CO5: Justify role of SDLC in Software Project Development and they can evaluate importance of Software Engineering in PLC.

CO6: Generate project schedule and can construct, design and develop network diagram for different type of Projects. They can also organize different activities of project as per Risk impact factor.

UNIT-I **12 Hours**

Introduction: Professional Software Development, Software Engineering Ethics. Case Studies. **Software Processes**: Models. Process activities. Coping with Change. The Rational Unified Process.

UNIT-II **12 Hours**

Agile Software Development: Agile methods. Plan-driven and agile development. Extreme programming. Agile project management. Scaling agile methods. **Requirements Engineering**: Functional and non-functional requirements. The software Requirements Document. Requirements Specification. Requirements Engineering Processes. Requirements Elicitation and Analysis. Requirements validation. Requirements Management.

UNIT-III **12 Hours**

System Models: Context models. Interaction models. Structural models. Behavioural models. Model-driven engineering. **Design and Implementation**: Object-oriented design using the UML. Design patterns. Implementation issues. Open source Development.

UNIT-IV **12 Hours**

Software Testing: Development testing, Test-driven development, Release testing, User testing. **Software Evolution**: Evolution processes. Program evolution dynamics. Software maintenance. Legacy system management. **Project Planning**: Software pricing. Plan-driven development. Project scheduling. Agile planning. Estimation techniques. **Quality management:** Software quality. Software standards. Reviews and inspections. Software measurement and metrics.

REFERENCE BOOKS

1. **Ian Somerville**, Software Engineering, Pearson Education, 2012, 9th Edition.
2. **Roger.S.Pressman**, Software Engineering - A Practitioners approach, Tata McGraw Hill, 7th Edition.
3. **PankajJalote**: An Integrated Approach to Software Engineering, Wiley India.

MCAH402 : DISTRIBUTED COMPUTING

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: Study software components of distributed computing systems.

CO2: Know about the communication and interconnection architecture of multiple computer systems.

CO3: Recognize the inherent difficulties that arise due to distributed-ness of computing sources.

CO4: Understanding of networks & protocols, mobile & wireless computing and their applications to real world problems.

CO5: At the end students will be familiar with the design, implementation and security issues of distributed system.

UNIT-I **12 Hours**

Principles of distributed computing: Fundamentals:- What is Distributed Computing Systems?, Distributed Computing System Models, What is DOS?, Issues in designing a DOS, Why gaining popularity in DOS?, Introduction to Distributed Computing Environment (DCE). Distributed databases: Definition, Introduction, Features Of Distributed Versus Centralized Databases, Why Distributed Databases?, Distributed Database Management Systems (DDBMSs), Reference Architecture For Distributed Databases, Types Of Data Fragmentation, Distributed database design, Objectives of the design of data distribution, Top-down and Bottom-up approaches to the design of data distribution.

UNIT-II **12 Hours**

Distributed objects and remote invocation: Introduction, Communication between distributed objects, Remote procedure calls, Events and notifications.

Distributed Operating System Support: Introduction, The operating system layer, Processes and threads, Communication and invocation, operating system architecture. Security: Overview of security techniques, Cryptographic algorithms, Digital signatures.

UNIT-III **12 Hours**

Distributed File Systems: Introduction, File service architecture, Sun Network File System (NFS).Distributed Time and clocks: Introduction, Clocks, events and process states, Synchronizing physical clocks, Logical time and logical clocks. Coordination and Agreement: Distributed mutual exclusion algorithms, Election algorithms, and multicast communication.

UNIT-IV **12 Hours**

Transactions and Concurrency Control: Introduction, Transactions, Nested transactions, Locks, Optimistic concurrency control, Time stamp ordering. Distributed transactions:- Introduction, Flat and nested distributed transactions, Atomic commit protocols, Concurrency control in distributed transactions, Distributed deadlocks, Transaction recovery.

REFERENCE BOOKS

1. Pradeep K Sinha, Distributed Operating Systems - Concepts and design, PHI, 2009, 1st Edition.
2. Stefano Ceri, Giuseppe PelagattiDistributed Databases - Principles and systems, McGraw-Hill International editions.
3. Distributed Systems: Concepts and Design, George Coulouris, Jean Dollimore, Tim Kindberg, Gordon Blair PHI, 2012, 5th edition.

MCAH403 : ADVANCED WEB PROGRAMMING

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: Develop Webpages using HTML/XHTML markup languages.

CO2: Develop Webpages applying CSS for Real World Scenarios.

CO3: Illustrate the use of Java Scripts in dynamic WebPages.

CO4: Design and develop a web based project.

UNIT-I 12 Hours

RESTful Web Services: Introduction, Architectural Principles; Designing RESTful Services - The Object Model, Model and URIs, Defining the Data Format, Assigning HTTP Methods.

JAX-RS: Developing a JAX-RS RESTful Service, Deploying, Writing a Client. **HTTP Method and URI Matching:** Binding HTTP Methods, @Path, Sub-resource Locators, Gotchas in Request Matching; **JAX-RS Injection:** Basics, Injecting Different Types of Parameters, Common Functionality.

UNIT-II 12 Hours

JAX-RS Content Handlers: Built-in Content Marshalling, JAXB, Custom Marshalling. **Server Responses and Exception Handling:** Default Response Codes, Complex Responses, Exception Handling. **JAX-RS Client API:** Introduction, Client Builder, Client and Web Target, Building and Invoking Requests. Configuration Scopes.

UNIT-III 12 Hours

Node.js: JavaScript and Node.js – Introduction, Server-Side JavaScript, Hello Word application; Node.js Use Cases, Application Stack. Building the Application Stack: Basic HTTP Server and Analysis, Passing Functions Around, Server Working, Event-Driven Asynchronous Callbacks, Handling Requests, Placing Server Module, Route Requests – Basics and Real Handlers, Execution, Responding, Serving, Handling POST Request, Handling File Uploads.

UNIT-IV 12 Hours

Search Engines and SEO: What is SEO? Needs for SEO; SEO in Social Media; Doing SEO on Your Own; Finding Sites; Working of Search Engines – Google, Bing, Yahoo!, International Searches; SEO Techniques – Friendliness, Keywords and Keyword Research, Creating Content for Customers, Myths and Facts; Tools for Tracking and Managing SEO: Sitemaps, robots.txt, Canonical Links, Duplicate Content, Checking, Tracking, Paying for Links.

REFERENCE BOOKS:

1. Bill Burke**, RESTful Java with JAX-RS 2.0** – Designing and Developing Distributed Web Services, O'Reilly, 2013, 2nd Edition.
2. Manuel Kiessling**, The Node Beginner Book**, Leanpub, 2016
3. Laura Lemay et.al**, Sams Teach Yourself HTML, CSS & JavaScript Web Publishing in One Hour a Day**, Pearson Education, 2016, 7th Edition
4. Martin Kalin**. Java Web Services: Up and Running**, O'Reilly Media, 2013, 2nd Edition.
5. SandroPasquali**, Mastering Node.js**, Packt, 2013

MCAS404 : ADVANCED COMPUTER NETWORKS

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: Illustrate reference models with layers, protocols and interfaces.

CO2: Summarize functionalities of different Layers.

CO3: Combine and distinguish functionalities of different Layers.

CO4: Describe and Analysis of basic protocols of computer networks, and how they can be used to assist in network design and implementation.

CO5: Identify and describe development history of routing protocols.

CO6: Describe Sub-netting and Addressing of IP V4.

CO7: Demonstrate Data Communications System and its components.

CO8: Identify the different types of network devices and their functions within a network.

CO9: Diagnose and resolve problems of a LAN and WAN.

UNIT-I **12 Hours**

Introduction: Introduction to Computer Networks. Understanding Network architecture. Introduction to TCP/IP Architecture, TCP/IP addressing, services, FTP, SMTP, TFTP, SNMP, and Network file system, domain name system, transport layer protocols, user datagram protocol, transmission control protocol, Class addresses, ARP, RARP.

UNIT-II **12 Hours**

Inter process communications: File and record locking, pipes, FIFO's, stream and messages, message queues, semaphores

UNIT-III **12 Hours**

Sockets: Sockets system calls, reserved ports, stream pipes, socket option, asynchronous I/O, Sockets and signals. Understanding the Internet Protocols SLIP versus PPP. Understanding the Socket interface. Concepts of the Windows sockets API. Importance of Raw Sockets. Internet Application Services, E-Mail, File Transfer Protocols, Characteristic Features of the Firewall.

UNIT-IV **12 Hours**

Transport Layer Interface: Elementary TLI functions, stream and stream pipes, asynchronous I/O multiplexing. **Remote Procedure calls:** Remote login, remote command execution, external data representation. UUCP.

REFERENCE BOOKS

1. A. Stevens, "TCP/IP Illustrated", Vol. 1-3, Addison Wesley, 1998.
2. R. Stevens, "Unix Network Programming", PHI, 1998
3. J. Martin, "TCP/IP Networking – Architecture, Administration and programming", Prentice Hall, 1994.
4. D.E. Comer, "Internetworking with TCP/IP, Vol. 1, Principles, Protocols, and architecture, PHI, 2000.
5. Internet Programming by Kris Jamsa, Galgotia publishers, 2001.

MCAS405 : IMAGE PROCESSING

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: Remember the fundamental concepts of image processing.

CO2: Explain different Image enhancement techniques

CO3: Understand and review image transforms

CO4: Analyze the basic algorithms used for image processing &image compression with morphological image processing.

CO5: Contrast Image Segmentation and Representation

CO6: Design & Synthesize Color image processing and its real world applications

UNIT-I **12 Hours**

Introduction:-Fundamental Steps in Digital Image Processing, Components of an Image Processing System, Image Sensing and Acquisition, Image Sampling and Quantization. **Image Enhancement in the Spatial Domain**: - Basic Gray Level Transformations, Histogram Processing, Enhancement Using Arithmetic/Logic Operations. Basics of Spatial Filtering: Smoothing Spatial Filters, Sharpening Spatial Filters.

UNIT-II **12 Hours**

Image Transforms such as FT, DCT, and HAAR Transform etc. **Image Enhancement in the Frequency:-** Introduction to the Fourier Transform and the Frequency Domain, Smoothing Frequency- Domain Filters, Sharpening Frequency Domain Filters.

UNIT-III **12 Hours**

Image Restoration: Model of the Image Degradation/Restoration Process, Noise Models, Restoration in the Presence of Noise Only Spatial Filtering, Periodic Noise Reduction by Frequency Domain Filtering, Linear, Position-Invariant Degradations, Estimating the Degradation Function. **Image Compression**: Fundamentals, Image Compression Models, Error Free Compression, Lossy Compression. **Morphological Image Processing:** Preliminaries, Dilation and Erosion, Opening and Closing The Hit – or Miss Transformation, Some Basic Morphological Algorithms

UNIT-IV **12 Hours**

Image Segmentation: Detection of Discontinuities, Edge Linking and Boundary Detection, Thresholding, Region-Based Segmentation, The Use of Motion in Segmentation. **Representation and Description:** Representation, Boundary Descriptors, Regional Descriptors, Use of Principal Components for Description, Relational Descriptors.

REFERENCE BOOKS

1. R.C. Gonzalez and R.E. Woods**, Digital Image Processing**, Prentice Hall.
2. B. Chanda, and D. Dutta Majumder**, Digital Image Processing and Analysis**, Prentice-Hall Pvt. Ltd, 2000

MCAS406 : SOFTWARE ARCHITECTURE

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: Students will cite knowledge of various approaches to document a software system (Remembering)

CO2: Students will be able to describe functional and non-functional requirements (Understanding)

CO3: Students will be able to use proper architecture for software (Applying)

CO4: Students will be able to categorize different components used in the software system (Analyzing)

CO5: Students will be able to choose from different architectural styles (Evaluating)

CO6: Students will be able to improve quality of software by selecting proper architecture (Creating)

UNIT-I **12 Hours**

Introduction: The Architecture Business Cycle: Where do architectures come from? Software processes and the architecture business cycle; What makes a "good" architecture? What software architecture is and what it is not; Other points of view; Architectural patterns, Reference models and reference architectures; Importance of software architecture; Architectural structures and views. **Architectural Styles And Case Studies:** Architectural styles; Pipes and filters; Data abstraction and object-oriented organization; Event-based, implicit invocation; Layered systems; Repositories; Interpreters; Process control; Other familiar architectures; Heterogeneous Architectures. Case Studies: Keyword in Context; Instrumentation software; Mobile robotics.

UNIT-II **12 Hours**

Quality: Functionality and architecture; Architecture and quality attributes; System quality attributes; Quality attribute scenarios in practice; Other system quality attributes; Business qualities; Architecture qualities. Achieving Quality: Introducing tactics; Availability tactics; Modifiability tactics; Performance tactics; Security tactics; Testability tactics; Usability tactics.

UNIT-III **12 Hours**

Architectural Patterns: Introduction, Distributed Systems: Broker; Interactive Systems: MVC, Presentation-Abstraction-Control. Adaptable Systems: Microkernel; Reflection. **Some Design Patterns:** Structural decomposition: Whole – Part; Organization of work: Master – Slave; Access Control: Proxy.

UNIT-IV **12 Hours**

Designing And Documenting Software Architecture: Architecture in the life cycle; designing the architecture; Forming the team structure; Creating a skeletal system. Uses of architectural documentation; Views; choosing the relevant views; Documenting a view; Documentation across views.

REFERENCE BOOKS

1. Len Bass, Paul Clements, Rick Kazman**, Software Architecture in Practice,** Pearson Education, 2003, 2nd Edition.
2. Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, Michael Stal**, Pattern-Oriented Software Architecture, A System of Patterns -Volume 1,** John Wiley and Sons, 2006.
3. **Mary Shaw and David Garlan**, Software Architecture-Perspectives on an Emerging Discipline, Prentice-Hall of India, 2007.
4. E. Gamma, R. Helm, R. Johnson, J. Vlissides, **Design Patterns- Elements of Reusable Object-Oriented Software**, Addison- Wesley, 2003.

MCAS407 : MOBILE COMPUTING

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: Grasp the concepts and features of mobile computing technologies and applications;

CO2: Have a good understanding of how the underlying wireless and mobile communication networks work, their technical features, and what kinds of applications they can support;

CO3: Identify the important issues of developing mobile computing systems and applications;

CO4: Organize the functionalities and components of mobile computing systems into different layers and apply various techniques for realizing the functionalities;

CO5: Develop mobile computing applications by analyzing their characteristics and requirements, selecting the appropriate computing models and software architectures, and applying standard programming languages and tools;

CO6: Organize and manage software built for deployment and demonstration.

CO7: Analyze requirements and solve problems using systematic planning and development approaches.

CO8: Search for and read critically the information required in solving problems

UNIT-I **12 Hours**

Introduction: Mobile Computing – Mobile Computing Vs wireless Networking – Mobile Computing Applications – Characteristics of Mobile computing – Structure of Mobile Computing Application. MAC Protocols – Wireless MAC Issues – Fixed Assignment Schemes – Random Assignment Schemes – Reservation Based Schemes

UNIT-II **12 Hours**

Mobile Internet Protocol And Transport Layer: Overview of Mobile IP – Features of Mobile IP – Key Mechanism in Mobile IP – route Optimization. Overview of TCP/IP – Architecture of TCP/IP- Adaptation of TCP Window – Improvement in TCP Performance. Mobile Telecommunication System: Global System for Mobile Communication (GSM) – General Packet Radio Service (GPRS) – Universal Mobile Telecommunication System (UMTS).

UNIT-III **12 Hours**

Mobile Ad-Hoc Networks: Ad-Hoc Basic Concepts – Characteristics – Applications – Design Issues – Routing – Essential of Traditional Routing Protocols –Popular Routing Protocols – Vehicular Ad Hoc networks (VANET) – MANET vs VANET – Security.

UNIT-IV **12 Hours**

Mobile Platforms And Applications: Mobile Device Operating Systems – Special Constrains & Requirements – Commercial Mobile Operating Systems – Software Development Kit: iOS,

Android, BlackBerry, Windows Phone – M Commerce – Structure – Pros & Cons – Mobile Payment System – Security Issues.

REFERENCE BOOKS

1. Prasant Kumar Pattnaik, Rajib Mall, “Fundamentals of Mobile Computing”, PHI Learning Pvt. Ltd, New Delhi – 2012.
2. Jochen H. Schller, “Mobile Communications”, Second Edition, Pearson Education, New Delhi, 2007.
3. Dharma Prakash Agarval, Qing and An Zeng, "Introduction to Wireless and Mobile systems", Thomson Asia Pvt Ltd, 2005.
4. Uwe Hansmann, Lothar Merk, Martin S. Nicklons and Thomas Stober, “Principles of Mobile Computing”, Springer, 2003.
5. William. C.Y. Lee,“Mobile Cellular Telecommunications-Analog and Digital Systems”, Second Edition, TataMcGraw Hill Edition,2006.
6. C.K. Toh, “AdHoc Mobile Wireless Networks”, First Edition, Pearson Education, 2002.

MCAS408 : WIRELESS COMMUNICATION

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: Explain the basic concepts of wireless network and wireless generations.

CO2: Demonstrate the different wireless technologies such as CDMA, GSM, GPRS etc.

CO3: Appraise the importance of Ad-hoc networks such as MANET and VANET and Wireless Sensor networks

CO4: Describe and judge the emerging wireless technologies standards such as WLL, WLAN, WPAN, WMAN. 5. Explain the design considerations for deploying the wireless network infrastructure.

CO5: Differentiate and support the security measures, standards. Services and layer wise security considerations

UNIT-I **12 Hours**

Wireless Channels : Large scale path loss – Path loss models: Free Space and Two-Ray models - Link Budget design – Small scale fading- Parameters of mobile multipath channels – Time dispersion parameters Coherence bandwidth – Doppler spread & Coherence time, Fading due to Multipath time delay spread – flat fading – frequency selective fading – Fading due to Doppler spread – fast fading – slow fading.

UNIT-II **12 Hours**

Cellular Architecture: Multiple Access techniques - FDMA, TDMA, CDMA – Capacity calculations–Cellular concept Frequency reuse - channel assignment- hand off- interference & system capacity- trunking & grade of service – Coverage and capacity improvement.

UNIT-III **12 Hours**

Digital Signalling For Fading Channels: Structure of a wireless communication link, Principles of Offset-QPSK, p/4-DQPSK, Minimum Shift Keying, Gaussian Minimum Shift Keying, Error performance in fading channels, OFDM principle – Cyclic prefix, Windowing, PAPR.

UNIT-IV **12 Hours**

Multipath Mitigation Techniques: Equalization – Adaptive equalization, Linear and Non-Linear equalization, Zero forcing and LMS Algorithms. Diversity – Micro and Macro diversity, Diversity combining techniques, Error probability in fading channels with diversity reception, Rake receiver. Multiple Antenna Techniques: MIMO systems – spatial multiplexing -System model -Pre-coding - Beam forming - transmitter diversity, receiver diversity- Channel state information-capacity in fading and non-fading channels

REFERENCE BOOKS

1. Rappaport, T.S., “Wireless communications”, Second Edition, Pearson Education, 2010.
2. Andreas F. Molisch, “Wireless Communications”, John Wiley – India, 2006.
3. David Tse and Pramod Viswanath, “Fundamentals of Wireless Communication”, Cambridge University Press, 2005.
4. Upena Dalal, “Wireless Communication”, Oxford University Press, 2009.
5. Van Nee, R. and Ramji Prasad, “OFDM for wireless multimedia communications”, Artech House, 2000.

MCAS409: SOFTWARE TESTING AND AUTOMATION

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: Investigate the reason for bugs and analyze the principles in software testing to prevent and remove bugs.

CO2: Implement various test processes for quality improvement

CO3: Design test planning.

CO4: Manage the test process

CO5: Apply the software testing techniques in commercial environment

CO6: Use practical knowledge of a variety of ways to test software and an understanding of some of the tradeoffs between testing techniques.

UNIT-I **12 Hours**

Introduction to Testing – why and what, Why is testing necessary? What is testing? Role of Tester, Testing and Quality, Overview of STLC Software Testing Life Cycle - V model**,** SDLC vs. STLC, different stages in STLC, document templates generated in different phases of STLC, different levels of testing, different types of testing**,** Static Testing**,** Static techniques, reviews, walkthroughs.

UNIT-II **12 Hours**

Basics of test design techniques**,** Various test categories, test design techniques for different categories of tests. Designing testcases using MS-Excel. Test management**,** Documenting test plan and test case, effort estimation, configuration management, project progress management. Use of Testopia for test case documentation and test management.

UNIT-III **12 Hours**

Defect management**,** Test Execution, logging defects, defect lifecycle, fixing / closing defects. Use of Bugzilla for logging and tracing defects. Test Data Management**,** Test Data Management –Overview, Why Test Data Management, Test Data Types, and Need for Test Data Setup, Test Data Setup Stages, and Test data management Challenges. Creating sample test data using MS-Excel Basics of Automation testing**,** Introduction to automation testing, why automation, what to automate, tools available for automation testing.

UNIT-IV **12 Hours**

Basics of Automation testing using Selenium**,** Introduction to Selenium, using Selenium IDE for automation testing, using Selenium Webdriver for automation testing, understanding TestNG framework with Selenium Web driver for automation testing

REFERENCE BOOKS

1. Rex Black, Managing the Testing Process, John Wiley, 2001, 2 nd Edition
2. Dorothy Graham, Erik van Veenendaal, Isabel Evans, Foundations of Software Testing, Rex Black
3. Elfriede Dustin , Implementing Automated Software Testing - How to Save Time and Lower Costs While Raising Quality

MCAS410 : E-COMMERCE

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: Define and differentiate various types of Ecommerce.

CO2: Describe Hardware and Software Technologies for Ecommerce.

CO3: Explain payment systems for E - commerce.

CO4: Describe the process of Selling and Marketing on web.

CO5: Define and Describe E-business and its Models.

CO6: Discuss various E-business Strategies.

UNIT-I **12 Hours**

IT and business, various applications of IT in business field**.** History of e-commerce, definition, classification- B2B, B2C, C2C, G2C, B2G sites, e-commerce in education, financial, auction, news, entertainment sectors, Doing ecommerce., EDI and its components

UNIT-II **12 Hours**

Electronic payment systems – credit cards, debit cards, smart cards, e-credit accounts, e-money, EFT, security concerns in e commerce, authenticity, privacy, S-HTP, Secure e-mail protocols, integrity, non-repudiation, encryption, secret key cryptography, public key cryptography, SET, SSL, digital signatures, firewalls.

UNIT-III **12 Hours**

Internet Marketing Phase, Marketing on the web, marketing strategies, creating web presence, advertising, customer service and support, web branding strategies, web selling models.

UNIT-IV **12 Hours**

M-commerce; case study of two internationally successful e-commerce web sites and two Kerala-based e-commerce web sites; IT act (India) and e-commerce.

REFERENCE BOOKS

1. C. S. V.Murthy, E-Commerce, Himalaya Publishing House.
2. NIIT, Basics of E-Commerce, PHI.
3. Erfan Turban et. al., Electronic Commerce–A Managerial Perspective, Pearson Education.
4. R Kalokota, Andrew V. Winston, Electronic Commerce – A Manager's Guide, Pearson Education.

MCAS411 : OPERATION RESEARCH

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: Identify and develop operational research models from the verbal description of the real system.

CO2: Understand the mathematical tools that are needed to solve optimization problems.

CO3: Use mathematical software to solve the proposed models.

CO4: Develop a report that describes the model and the solving technique, analyze the results and propose recommendations in language understandable to the decision-making processes in Management Engineering.

UNIT-I **12 Hours**

Introduction: Nature and developments of operations research, characteristics of operations research, necessity of operations research in industry, scope of OR in management, objectives of OR, models in OR, role of computers in OR, limitations of OR. Linear Programming: Requirements of linear programming problems, formulation of linear programming problem, graphical solution, simplex algorithm, computational procedure in simplex, duality and its concept, application of L.P. model to product mix and production scheduling problems, limitations of linear programming.

UNIT-II **12 Hours**

Transportation model: Definition of transportation model, formulation and solution methods, and degeneracy in transportation problems. Assignment Model: Definition of assignment model, comparison with transportation model, formulation and solution methods, the travelling salesman problem.

UNIT-III **12 Hours**

Queuing Models: Application of queuing models, characteristics of queuing models, single channel queuing theory, solution to single channel with poison arrivals and exponential service infinite population model, Industrial applications of queuing theory. Simulation: When to use simulation, Advantages and limitations of the simulation technique, generation of random numbers, Monte-Carlo simulation, And computer-aided simulation: applications in maintenance and inventory management.

UNIT-IV **12 Hours**

Game Theory and Network Analysis: PERT and CPM: Work breakdown structure, network logic, critical path, CPM Vs PERT, slack and floats. Game theory: Pure strategies and Mixed strategies. Application of software skill in Operations Research.

REFERENCE BOOKS

1. P.K. Gupta and D.S. Hira, S Chand, **Operations Research**, S Chand and company.
2. A.H. Taha, **Operation Research - An Introduction**, Macmillan Publishing Co.
3. W.D. Miller and M.K Starr , **Executive Decisions and Operations Research**, Prentice Hall
4. Hillier and Lieberman, **Introduction of Operations Research**.
5. Ackoff and Sasiene, **Fundamentals of Operations Research**.
6. Jerry Banks, David M. Nicole, Barry L. Nelson, **Discrete-event system simulation**

MCAP412: Distributed Computing Lab

Course Outcomes:

CO1: Study software components of distributed computing systems.
CO2: Know about the communication and interconnection architecture of multiple computer systems.
CO3: Recognize the inherent difficulties that arise due to distributed-ness of computing resources.
CO4: Understanding of networks & protocols, mobile & wireless computing and their applications to real world problems.
CO5: At the end students will be familiar with the design, implementation and security issues of distributed system.

MCAP413: Computer Network Lab

Course Outcomes:

CO1: Understand fundamental underlying principles of computer networking
CO2: Understand details and functionality of layered network architecture.
CO3: Apply mathematical foundations to solve computational problems in computer networking
CO4: Analyze performance of various communication protocols.
CO5: Compare routing algorithms
CO6: Practice packet /file transmission between nodes

MCAP414: Image Processing Lab

Course Outcomes:

CO1: Apply different Point processing operations on images and show the output.
CO2: Develop a set of filtering methods for removing noise present in images.
CO3: Demonstrate different Arithmetic operations on images and visualize the effects.
CO4: Show how histogram equalization improves images.
CO5: Illustrate the segmentation process over an image.

MCAP415: Advanced Web Programming Lab

Course Outcomes:

CO1: Explain Search Engine Optimization Techniques and Develop Keyword Generation.
CO2: Describe different Web Services Standards.
CO3: Develop Rich Internet Application using proper choice of Framework.
CO4: Develop webpages using CSS.

MCAP416: E-Commerce Lab

Course Outcomes:

CO1: At the end of the course students should know how to apply their computer science skills to the conduct of e-commerce.
CO2: Understanding of the legal, security, commercial, economic, marketing and infrastructure issues involved.
CO3: Assess electronic payment systems
CO4: Recognize and discuss global E-commerce issues

MCAP417: Operational Research Lab

Course Outcomes:

CO1: Identify and express a decision problem in mathematical form and solve it graphically and by Simplex method
CO2: Recognize and formulate transportation, assignment problems and drive their optimal solution.
CO3: Identify parameters that will influence the optimal solution of an LP problem and derive feasible solution using a technique of O R.

V Semester MCA

MCAH501: DATA MINING TECHNIQUES

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: Understand data mining principles and techniques: Introduce DM as a cutting edge business intelligence method and acquaint the students with the DM techniques for building competitive advantage through proactive analysis, predictive modelling, and identifying new trends and behaviors. Course Objectives include:

CO2: Building basic terminology.

CO3: Learning how to gather and analyze large sets of data to gain useful business understanding.

CO4: Learning how to produce a quantitative analysis report/memo with the necessary information to make decisions.

CO5: Describing and demonstrating basic data mining algorithms, methods, and tools 6. Identifying business applications of data mining

CO6: Overview of the developing areas - web mining, text mining, and ethical aspects of data mining

UNIT-I **12 Hours**

Introduction: Motivations, Data Mining Databases-Relational Databases, Data warehouse, Transactional Databases, Advanced Database systems and advanced Database applications. Data Mining Functionalities-Concept/Class Discrimination; characterizations and Discrimination, Association Analysis, Classification and Prediction, Cluster Analysis, Outlier Analysis and Evolution Analysis. Classifications of Data Mining Systems, Major issues in Data Mining. **Data Pre-processing**: Data Cleaning, Data Integration and Transformation, Data Reduction, Discrimination and Concepts Hierarchy Generation.

Data Warehouse and OLAP technology for data Mining: Definition of data warehouse, A Multidimensional Data Model, Data warehouse architecture, Data warehouse implementation, Further development of data cubes technology, From data warehousing to data Mining.

UNIT-II **12 Hours**

Mining Primitives, Languages and Systems Architectures: Data Mining Primitives, Data Mining Query Languages, Designing Graphical User Interfaces Based onData Mining Query Languages and Architecture of Data Mining systems.

Concept Description: Characterization and Comparison: Concept Description, Data Generalization and Summarization-based Characterization, Analytical Characterization: Analysis of Attributes Relevance, Missing Class comparisons: Discriminating Between Different classes, Mining Descriptive Statistical Measures in Large Databases.

UNIT-III **12 Hours**

Mining Association Rules in Large Database: Association Rule Mining, Mining Single-Dimensional Association Rules From Transactional Databases, Mining Multi-Association Rules From Transaction Databases, Mining Multi-dimensional Association

Rules from Relational Databases and Data Warehouses, From Association Mining Correlation Analysis, Constraint-Based Association Mining.

UNIT-IV **12 Hours**

Classification and Prediction: Definition of Classification, issues regarding classification and Prediction, Classification by decision tree induction, Bayesian Classification, Classification by Back propagation, Classification based on concepts from association rules mining, other classification methods, prediction, classification accuracy. **Cluster Analysis:** Definition of Cluster, Types of data in cluster analysis, A categorization of major cluster Methods, Partitioning methods, Hierarchical methods, Density-Base Methods, Grid-based methods, Model based Methods, Outlier analysis.

REFERENCE BOOKS

1. Jaiawei Han and Micheline Kamber, Data Mining Concepts and Techniques, 3rd Edition, Morgan Kaufmann/Elsevier Science publisher, Reprint published by Harcourt (INDIA) Private Limited.
2. David L. Olson, Dursun Delen, Advanced Data Mining Techniques, Springer publishers.

MCAH502 : PYTHON PROGRAMMING

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: Explain basic principles of Python programming language

CO2: Problem solving and programming capability.

CO3: Implement object oriented concepts,

CO4: Implement database and GUI applications.

UNIT-I **12 Hours**

Introduction to python, the concept of data types; variables, assignments; immutable variables; numerical types; arithmetic operators and expressions; comments in the program; understanding error messages; Conditions, Boolean logic, logical operators; ranges; Control statements: if-else, loops (for, while); short circuit evaluation; Strings and text files; manipulating files and directories, os and sys modules; text files: reading/writing text and Numbers from/to a file; creating and reading a formatted file (csv or tab-separated).

UNIT-II **12 Hours**

String manipulations: subscript operator, indexing, slicing a string; strings and number System: converting strings to numbers and vice versa. Binary, octal, hexadecimal numbers Lists, tuples, and dictionaries; basic list operators, replacing, inserting, removing an element; searching and sorting lists; dictionary literals, adding and removing keys, accessing and Replacing values; traversing dictionaries.

UNIT-III **12 Hours**

Design with functions: hiding redundancy, complexity; arguments and return values; formal vs. actual arguments, named arguments. Program structure and design. Recursive functions. Classes and OOP: classes, objects, attributes and methods; defining classes; design with classes, data modelling; persistent storage of objects inheritance, polymorphism, operator overloading (_eq_,_str_, etc); abstract classes; exception handling, try block

UNIT-IV **12 Hours**

Python database application programmer's interface (DB- API), connection and cursor objects, Type objects and constructors, python database adapters. Creating simple web clients, introduction to CGI, CGI module, building CGI applications, python web application frameworks.

REFERENCE BOOKS

1. Kenneth A. Lambert, The Fundamentals of Python: First Programs, 2011, Cengage Learning,
2. Magnus Lie Hetland, Beginning Python from Novice to Professional, Second Edition.
3. Mark Summerfield, Programming in Python 3 - A Complete Introduction to the Python Language, Second Edition.
4. Y. Daniel Liang, “Introduction to Programming Using Python”, Pearson
5. Chun, J Wesley, Core Python Programming, 2nd Edition, Pearson, 2007 Reprint 2010.
6. David Beazley and Brian K. Jones, Python Cookbook, Third Edition, Shroff Publishers & Distributors Pvt. Ltd.
7. Mark Lutz, Learning Python FIFTH EDITION Mark Lutz.
8. Mark Lutz, Programming Python (English) 4th Edition.
9. Testing Python, David Sale, Wiley India (P) Ltd.,

MCAH 503: ANDROID APPLICATION DEVELOPMENT

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: By the end of the course, student will be able to write simple GUI applications

CO2: Use built-in widgets and components

CO3: Work with the database to store data locally, and much more.

CO4: Students will gain fundamental knowledge essential to not only Android development, but mobile development in general.

UNIT-I **12 Hours**

Introduction to Android Operating System: Android OS design and Features – Android development framework, SDK features, Installing and running applications on Eclipse platform, Creating AVDs, Types of Android applications, Best practices in Android programming, Android tools Android application components – Android Manifest file, Externalizing resources like values, themes, layouts, Menus etc, Resources for different devices and languages, Runtime Configuration Changes Android Application Lifecycle – Activities, Activity lifecycle, activity states, monitoring state changes.

UNIT-II **12 Hours**

Android User Interface: Measurements – Device and pixel density independent measuring units Layouts – Linear, Relative, Grid and Table Layouts User Interface (UI) Components – Editable and non editable Text Views, Buttons, Radio and Toggle Buttons, Checkboxes, Spinners, Dialog and pickers Event Handling – Handling clicks or changes of various UI components Fragments – Creating fragments, Lifecycle of fragments, Fragment states, Adding fragments to Activity, adding, removing and replacing fragments with fragment transactions, interfacing between fragments and Activities, Multi-screen Activities

UNIT-III **12 Hours**

Intents and Broadcasts: Intent – Using intents to launch Activities, Explicitly starting new Activity, Implicit Intents, Passing data to Intents, Getting results from Activities, Native Actions, using Intent to dial a number or to send SMS Broadcast Receivers – Using Intent filters to service implicit Intents, Resolving Intent filters, finding and using Intents received within an Activity Notifications – Creating and Displaying notifications, Displaying Toasts.

UNIT-IV **12 Hours**

Persistent Storage: Files – Using application specific folders and files, creating files, reading data from files, listing contents of a directory Shared Preferences – Creating shared preferences, saving and retrieving data using Shared Preference Database – Introduction to SQLLite database, creating and opening a database, creating tables, inserting retrieving and deleting data, Registering Content Providers, Using content Providers (insert, delete, retrieve and update). Advanced Topics: Alarms – Creating and using alarms Using Internet Resources – Connecting to internet resource, using download manager Location Based Services – Finding Current Location and showing location on the Map, updating location.

REFERENCE BOOKS

1. RetoMeier,,Professional Android 4 Application Development, Wiley India, (Wrox) , 2012.
2. James C Sheusi,,Android Application Development for Java Programmers, Cengage Learning, 2013
3. Wei-MengLee,,Beginning Android 4 Application Development, Wiley India (Wrox), 2013

MCAS504 : BIG DATA ANALYTICS

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: Work with big data platform and explore the big data analytics techniques business applications.

CO2: Design efficient algorithms for mining the data from large volumes.

CO3: Analyze the HADOOP and Map Reduce technologies associated with big data analytics.

CO4: Explore on Big Data applications Using Pig and Hive.

CO5: Understand the fundamentals of various big data analytics techniques.

CO6: Build a complete business data analytics solution

UNIT-I 12 Hours

Introduction To Big Data: Introduction – distributed file system – Big Data and its importance, Four Vs, Drivers for Big data, Big data analytics, Big data applications. Algorithms using map reduce, Matrix-Vector Multiplication by Map Reduce.

UNIT-II 12 Hours

Introduction Hadoop: Big Data – Apache Hadoop & Hadoop Eco System – Moving Data in and out of Hadoop – Understanding inputs and outputs of Map Reduce - Data Serialization.
Hadoop Architecture: Hadoop Architecture, Hadoop Storage: HDFS, Common Hadoop Shell commands , Anatomy of File Write and Read., Name Node, Secondary Name Node, and Data Node, Hadoop Map Reduce paradigm, Map and Reduce tasks, Job, Task trackers - Cluster Setup – SSH &Hadoop Configuration – HDFS Administering –Monitoring & Maintenance.

UNIT-III 12 Hours

Hadoop Ecosystem And Yarn: Hadoop ecosystem components - Schedulers - Fair and Capacity, Hadoop 2.0 New Features Name Node High Availability, HDFS Federation, MRv2, YARN, Running MRv1 in YARN.

UNIT-IV 12 Hours

Hive And Hiveql, Hbase : Introduction to No Query Language, Hive Architecture and Installation, Comparison with Traditional Database, HiveQL - Querying Data - Sorting And Aggregating, Map Reduce Scripts, Joins & Sub queries, HBase concepts Advanced Usage, Schema Design, Advance Indexing - PIG, Zookeeper - how it helps in monitoring a cluster, Hbase uses Zookeeper and how to Build Applications with Zookeeper.

REFERENCE BOOKS

1. Boris Lublinsky, Kevin T. Smith, Alexey Yakubovich, "Professional Hadoop Solutions", Wiley, 2015.
2. Chris Eaton, Dirk Deroos et al. , "Understanding Big data ", McGraw Hill, 2012.
3. Tom White, "HADOOP: The definitive Guide" , O Reilly, 2012.
4. Vignesh Prajapati, "Big Data Analytics with R and Haoop", Packet Publishing 2013.
5. Tom Plunkett, Brian Macdonald et al, "Oracle Big Data Handbook", Oracle Press, 2014.
6. Jy Liebowitz, "Big Data and Business analytics", CRC Press, 2013.

MCAS 505 : CLOUD AND GRID COMPUTING

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: Demonstrate an in-depth understanding characteristics of grid and cloud computing;

CO2: Demonstrate an in-depth understand of the design principles of grid and cloud computing;

CO3: Illustrate security mechanisms in grid and cloud computing applications;

CO4: Design and demonstrate distributed computing applications.

UNIT-I 12 Hours

Introduction : Evolution of Distributed computing: Scalable computing over the Internet – Technologies for network based systems – clusters of cooperative computers - Grid computing Infrastructures – cloud computing - service oriented architecture – Introduction to Grid Architecture and standards – Elements of Grid – Overview of Grid Architecture. Grid Services: Introduction to Open Grid Services Architecture (OGSA) – Motivation – Functionality Requirements – Practical & Detailed view of OGSA/OGSI – Data intensive grid service models – OGSA services

UNIT-II 12 Hours

Virtualization: Cloud deployment models: public, private, hybrid, community – Categories of cloud computing: Everything as a service: Infrastructure, platform, software - Pros and Cons of cloud computing – Implementation levels of virtualization – virtualization structure – virtualization of CPU, Memory and I/O devices – virtual clusters and Resource Management – Virtualization for data center automation.

UNIT-III 12 Hours

Programming Model 9 Open source grid middleware packages – Globus Toolkit (GT4) Architecture , Configuration – Usage of Globus – Main components and Programming model - Introduction to Hadoop Framework - Mapreduce, Input splitting, map and reduce functions, specifying input and output parameters, configuring and running a job – Design of Hadoop file system, HDFS concepts, command line and java interface, dataflow of File read & File write.

UNIT-IV 12 Hours

Security 9 Trust models for Grid security environment – Authentication and Authorization methods – Grid security infrastructure – Cloud Infrastructure security: network, host and application level – aspects of data security, provider data and its security, Identity and access management architecture, IAM practices in the cloud, SaaS, PaaS, IaaS availability in the cloud, Key privacy issues in the cloud.

REFERENCE BOOKS

1. Kai Hwang, Geoffery C. Fox and Jack J. Dongarra, "Distributed and Cloud Computing: Clusters, Grids, Clouds and the Future of Internet", First Edition, Morgan Kaufman Publisher, an Imprint of Elsevier, 2012.
2. Jason Venner, "Pro Hadoop- Build Scalable, Distributed Applications in the Cloud", A Press, 2009.
3. Tom White, "Hadoop The Definitive Guide", First Edition. O"Reilly, 2009.
4. Bart Jacob (Editor), "Introduction to Grid Computing", IBM Red Books, Vervante, 2005.
5. Ian Foster, Carl Kesselman, "The Grid: Blueprint for a New Computing Infrastructure", Morgan Kaufmann, 2nd Edition.
6. Frederic Magoules and Jie Pan, "Introduction to Grid Computing" CRC Press, 2009.
7. 7. Daniel Minoli, "A Networking Approach to Grid Computing", John Wiley Publication, 2005.

MCAS 506 : MACHINE LEARNING

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: Gain knowledge about basic concepts of Machine Learning

CO2: Identify machine learning techniques suitable for a given problem

CO3: Solve the problems using various machine learning techniques

CO4: Apply Dimensionality reduction techniques.

CO5: Design application using machine learning techniques.

UNIT-I **12 Hours**

Introduction - Well-posed learning problems, Designing a learning system, Perspectives and issues in machine learning Concept learning and the general to specific ordering – Introduction, A concept learning task, Concept learning as search, Find-S: finding a maximally specific hypothesis, Version spaces and the candidate elimination algorithm, Remarks on version spaces and candidate elimination, Inductive bias

UNIT-II **12 Hours**

Decision Tree learning – Introduction, Decision tree representation, Appropriate problems for decision tree learning, The basic decision tree learning algorithm, Hypothesis space search in decision tree learning, Inductive bias in decision tree learning, Issues indecision tree learning Artificial Neural Networks – Introduction, Neural network representation, Appropriate problems for neural network learning, Perceptions, Multilayer networks and the back propagation algorithm, Remarks on the back propagation algorithm, An illustrative example face recognition Advanced topics in artificial neural networks Evaluation Hypotheses – Motivation, Estimation hypothesis accuracy, Basics of sampling theory, A general approach for deriving confidence intervals, Difference in error of two hypotheses, Comparing learning algorithms

UNIT-III **12 Hours**

Bayesian learning – Introduction, Bayes theorem, Bayes theorem and concept learning, Maximum likelihood and least squared error hypotheses, Maximum likelihood hypotheses for predicting probabilities, Minimum description length principle, Bayes optimal classifier, Gibs algorithm, Naïve Bayes classifier, An example learning to classify text, Bayesian belief networks The EM algorithm Computational learning theory – Introduction, Probability learning an approximately correct hypothesis, Sample complexity for Finite Hypothesis Space, Sample Complexity for infinite Hypothesis Spaces, The mistake bound model of learning - Instance-Based Learning- Introduction, k -Nearest Neighbour Learning, Locally Weighted Regression, Radial Basis Functions, Case-Based Reasoning, Remarks on Lazy and Eager Learning Genetic Algorithms – Motivation, Genetic Algorithms, An illustrative Example, Hypothesis Space Search, Genetic Programming, Models of Evolution and Learning, Parallelizing Genetic Algorithms

UNIT-IV **12 Hours**

Combining Inductive and Analytical Learning – Motivation, Inductive-Analytical Approaches to Learning, Using Prior Knowledge to Initialize the Hypothesis, Using Prior Knowledge to Alter the Search Objective, Using Prior Knowledge to Augment Search Operators, Reinforcement Learning – Introduction, The Learning Task, Q Learning, Non-Deterministic, Rewards and Actions, Temporal Difference Learning, Generalizing from Examples, Relationship to Dynamic Programming.

REFERENCE BOOKS

1. Tom M. Mitchell, Machine Learning, MGH.
2. Stephen Marshland, Taylor & Francis, Machine Learning: An Algorithmic Perspective.
3. William WHsieh, Machine Learning Methods in the Environmental Sciences, Neural
4. Networks, Cambridge Univ Press.
5. Richard O. Duda, Peter E. Hart and David G. Stork, pattern classification, John Wiley & Sons Inc., 2001.
6. Chris Bishop, Neural Networks for Pattern Recognition, Oxford University Press, 1995
7. Peter Flach, Machine Learning, Cambridge.

MCAS 507 : INTERNET OF THINGS

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: Identify the requirements for the real world problems.
CO2: Conduct a survey of several available literatures in the preferred field of study.
CO3: Study and enhance software/ hardware skills.
CO4: Demonstrate and build the project successfully by hardware requirements, coding, emulating

UNIT-I **12 Hours**

Introduction to Internet of Things –Definition and Characteristics of IoT, Physical Design of IoT – IoT Protocols, IoT communication models, Iot Communication APIs IoT enabled Technologies – Wireless Sensor Networks, Cloud Computing, Big data analytics, Communication protocols, Embedded Systems, IoT Levels and Templates Domain Specific IoTs – Home, City, Environment, Energy, Retail, Logistics, Agriculture, Industry, health and Lifestyle

UNIT-II **12 Hours**

IoT and M2M – Software defined networks, network function virtualization, difference between SDN and NFV for IoT Basics of IoT System Management with NETCOZF, YANGNETCONF, YANG, SNMP NETOPEER

UNIT-III **12 Hours**

Introduction to Python - Language features of Python, Data types, data structures, Control of flow, functions, modules, packaging, file handling, data/time operations, classes, Exception handling Python packages - JSON, XML, HTTPLib, URLLib, SMTPLib

UNIT-IV **12 Hours**

IoT Physical Devices and Endpoints - Introduction to Raspberry PI-Interfaces (serial, SPI, I2C) Programming – Python program with Raspberry PI with focus of interfacing external gadgets, controlling output, reading input from pins. IoT Physical Servers and Cloud Offerings – Introduction to Cloud Storage models and communication APIs Webserver – Web server for IoT, Cloud for IoT, Python web application framework Designing a RESTful web API

REFERENCE BOOKS

1. Arshdeep Bahga and Vijay Madisetti, Internet of Things - A Hands-on Approach, Universities Press, 2015.
2. Matt Richardson & Shawn Wallace, O'Reilly (SPD), Getting Started with Raspberry Pi, 2014.

MCAS 508: CRYPTOGRAPHY AND NETWORK SECURITY

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: Identify information security goals, classical encryption techniques and acquire fundamental knowledge on the concepts of finite fields and number theory.

CO2: Understand, compare and apply different encryption and decryption techniques to solve problems related to confidentiality and authentication

CO3: Apply the knowledge of cryptographic checksums and evaluate the performance of different message digest algorithms for verifying the integrity of varying message sizes

CO4: Apply different digital signature algorithms to achieve authentication and create secure applications

CO5: Apply network security basics, analyze different attacks on networks and evaluate the performance of firewalls and security protocols like SSL, IPSec, and PGP.

UNIT-I **12 Hours**

Introduction to information systems, Types of information Systems, Development of Information Systems, Introduction to information security, Need for Information security, Threats to Information Systems, Information Assurance, Cyber Security, and Security Risk Analysis.

UNIT-II **12 Hours**

Application security (Database, E-mail and Internet), Data Security Considerations-Backups, Archival Storage and Disposal of Data, Security Technology-Firewall and VPNs, Intrusion Detection, Access Control. Security Threats -Viruses, Worms, Trojan Horse, Bombs, Trapdoors, Spoofs, E-mail viruses, Macro viruses, Malicious Software, Network and Denial of Services Attack, Security Threats to E-Commerce- Electronic Payment System, e- Cash, Credit/Debit Cards. Digital Signature, public Key Cryptography.

UNIT-III **12 Hours**

Developing Secure Information Systems, Application Development Security, Information Security Governance & Risk Management, Security Architecture & Design Security Issues in Hardware, Data Storage & Downloadable Devices, Physical Security of IT Assets, Access Control, CCTV and intrusion Detection Systems, Backup Security Measures

UNIT-IV **12 Hours**

Security Policies, Why Policies should be developed, WWW policies, Email Security policies, Policy Review Process-Corporate policies-Sample Security Policies, Publishing and Notification Requirement of the Policies. Information Security Standards-ISO, IT Act, Copyright Act, Patent Law, IPR. Cyber Laws in India; IT Act 2000 Provisions, Intellectual Property Law: Copy Right Law, Software License, Semiconductor Law and Patent Law.

REFERENCE BOOKS

1. Charles P. Pfleeger, Shari Lawerance Pfleeger, “Analysing Computer Security”, Pearson.
2. V.K. Pachghare, “Cryptography and information Security”, PHI Learning Private Limited, Delhi India.
3.Dr. Surya Prakash Tripathi, Ritendra Goyal, Praveen Kumar Shukla, “Introduction to Information Security and Cyber Law”, Willey.
4. Schou, Shoemaker, “Information Assurance for the Enterprise”, Tata McGraw Hill.
5. Chander, Harish, “Cyber Laws And It Protection”, PHI.

MCAS 509: NATURAL LANGUAGE PROCESSING

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: After successful completion of this course, student will be able to
CO2: Understand approaches to syntax and semantics in NLP.
CO3: Understand approaches to discourse, generation, dialogue and summarization within NLP.
CO4: Understand current methods for statistical approaches to machine translation.
CO5: Understand machine learning techniques used in NLP, including hidden Markov models and probabilistic context-free grammars, clustering and unsupervised methods, log-linear and discriminative models, and the EM algorithm as applied within NLP

UNIT-I **12 Hrs.**

OVERVIEW AND LANGUAGE MODELING: *Overview*: Origins and challenges of NLP-Language and Grammar-Processing Indian Languages- NLP Applications-Information Retrieval. Language Modeling: Various Grammar- based Language Models-Statistical Language Model.

UNIT-II **12 Hrs.**

WORD LEVEL AND SYNTACTIC ANALYSIS: Word Level Analysis: Regular Expressions-Finite-State Automata-Morphological Parsing-Spelling Error Detection and correction-Words and Word classes-Part-of Speech Tagging. Syntactic Analysis: Context-free Grammar-Constituency-Parsing-Probabilistic Parsing.

UNIT-III **12 Hrs.**

SEMANTIC ANALYSIS AND DISCOURSE PROCESSING: Semantic Analysis: Meaning Representation-Lexical Semantics- Ambiguity-Word Sense Disambiguation. Discourse Processing: cohesion-Reference Resolution- Discourse Coherence and Structure.

UNIT-IV **12 Hrs.**

NATURAL LANGUAGE GENERATION AND MACHINE TRANSLATION:
Natural Language Generation: Architecture of NLG Systems- Generation Tasks and Representations- Application of NLG. Machine Translation: Problems in Machine Translation-Characteristics of Indian Languages- Machine Translation Approaches-Translation involving Indian Languages.

REFERENCE BOOKS:

1. Edward Loper, Ewan Klein, and Steven Bird, Natural Language Processing with Python, , O'Reilly Publication 2009.;
2. Christopher D. Manning, Hinrich Schütze , Foundations of Statistical Natural Language Processing , MIT press,1999.
3. Dan Jurafsky, James H. Martin, Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition, Prentice Hall, 2009.

MCAS510 : EMBEDDED SYSTEMS

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: Understand what is a microcontroller, microcomputer, embedded system.

CO2: Understand different components of a micro-controller and their interactions.

CO3: Become familiar with programming environment used to develop embedded systems.

CO4: Understand key concepts of embedded systems like IO, timers, interrupts, interaction with peripheral devices.

CO5: Learn debugging techniques for an embedded system.

UNIT-I **12 Hours**

Introduction to embedded systems hardware needs; typical and advanced, timing diagrams, memories (RAM, ROM, and EPROM) Tri state devices, Buses, DMA, UART and PLD's Built-in on the microprocessor. Interrupts basics, ISR; Context saving, shared data problem. Atomic and critical section, Interrupt latency.

UNIT-II **12 Hours**

Survey of software architectures, Round Robin, Function queue scheduling architecture, Use of real time operating system. RTOS, Tasks, Scheduler, Shared data reentrancy, priority inversion, mutex binary semaphore and counting semaphore. Inter task communication, message queue, mailboxes and pipes, timer functions, events Interrupt routines in an RTOS environment.

UNIT-III **12 Hours**

Embedded systems of forwarded sign RTOS Hard real-time and soft real time system principles, Task division, need of interrupt routines, shared data. Embedded Software development tools.

UNIT-IV **12 Hours**

Host and target systems, cross compilers, linkers, locators for embedded systems. Getting embedded software into the target system. Debugging techniques. Testing on host machine, Instruction set emulators, logic analyzers In-circuit emulators and monitors.

REFERENCE BOOKS

1. David A .Simon, An Embedded Software Primer, Pearson Education.
2. Daniel W. Ewis, Fundamentals of Embedded Software Where C and Assembly Meet, Pearson Education.
3. Oliver H. Baileg, Embedded System: Desktop Integrations, Wordware Publishing Inc.
4. Tammy Noergaard Newnes, Embedded System Architecture.

MCAS 511 : ARTIFICIAL INTELLIGENCE

Hours/Week: 4 **I.A. Marks: 30**
Credits : 4 **Exam. Marks: 70**

Course Outcomes:

CO1: Demonstrate knowledge of the building blocks of AI as presented in terms of intelligent agents.

CO2: Analyze and formalize the problem as a state space, graph, design heuristics and select amongst different search or game based techniques to solve them.

CO3: Develop intelligent algorithms for constraint satisfaction problems and also design intelligent systems for Game Playing

CO4: Attain the capability to represent various real life problem domains using logic based techniques and use this to perform inference or planning.

CO5: Formulate and solve problems with uncertain information using Bayesian approaches.

CO6: Apply concept Natural Language processing to problems leading to understanding of cognitive computing

UNIT-I **12 Hours**

Introduction: Artificial Intelligence: Its scope history and applications; AI as Representation and Search- The Predicate calculus-inference rules. Logic based financial adviser; Structures and strategies for state space search– Graph theory, Strategies for search, Using state space to represent reasoning with the predicate calculus.

UNIT-II **12 Hours**

Heuristic search: Heuristic Search: An algorithm for Heuristic Search, Admissibility, Môn tonicity and Informed Heuristics in games, Complexity issues. Control and Implementation of state space research- Recursion based search, Pattern directed search, Production systems, Predicate calculus and Planning, The black board architecture for Problem solving. **Knowledge based systems:** Knowledge-Intensive problem solving: Overview of Expert System technology, Rule-based Expert systems, Model-based reasoning, and Case-based reasoning. The knowledge Representation Problem; Reasoning with uncertain or incomplete information – The Statistical approach to uncertainty, Non-monotonic systems, Reasoning with Fuzzy sets.

UNIT-III **12 Hours**

Knowledge presentation and lisp: Knowledge representation languages, Issues in Knowledge representation, A survey of network representation. Conceptual graphs: A Network representation language, Structured representations. Further issues in knowledge representation; Introduction to LISP–Search in LISP: A functional approach to Farmer, Wolf, Goat, and Cabbage problem. Higher order functions and procedural abstraction, Search strategies in LISP, A Recursive Unification function, Interpreters and Embedded languages. Logic programming in LISP, Streams and delayed evaluation. An expert systems hell in LISP.

UNIT-IV **12 Hours**

Automated reasoning: Automated Reasoning: Weak methods in Theorem proving, The general problem solver and difference tables, Resolution Theorem proving, Further issues in Automated Reasoning; Machine Learning: Connectionist–Foundations for Connectionist Networks, Perceptions learning, Black- propagation learning, Competitive learning, Hebbian Coincidence learning, Attract or Networks or Memories. Machine Learning: Social and Emergent–modes, The Genetic algorithm, Classifier systems and Genetic programming,

Artificial life and Society based learning.

REFERENCE BOOKS

1. G.F. Lugerand W.A. Stubblefield, Artificial Intelligence – Structures and Strategies for Complex Problem Solving, Addison-Wesley, 1998, Third Edition.
2. P.H. Winston, Artificial Intelligence, Addision-Wesley, 1992, Third Edition.
3. E. Richand Knight, Artificial Intelligence, TataMcGraw Hill,1991, SecondEdition.
4. Nils J. Nilsson, Artificial Intelligence, A New Synthesis, Morgan Kaufmann, 2000.

MCAP512 : Data Mining Lab

Course Outcomes:

CO1: The data mining process and important issues around data cleaning, pre-processing and integration.
CO2: The principle algorithms and techniques used in data mining, such as clustering, association mining, classification and prediction.
CO3: Design a data mart or data warehouse for any organization
CO4: Develop skills to write queries using DMQL
CO5: Extract knowledge using data mining techniques
CO6: Adapt to new data mining tools.

MCAP513 : Python Programming Lab

Course Outcomes:

CO1: Describe the Numbers, Math functions, Strings, List, Tuples and Dictionaries in Python
CO2: Express different Decision Making statements and Functions
CO3: Interpret Object oriented programming in Python
CO4: Understand and summarize different File handling operations
CO5: Explain how to design GUI Applications in Python and evaluate different database operations
CO6: Design and develop Client Server network applications using Python

MCAP514 : Data Analytics Lab

Course Outcomes:

CO1: Implement statistical analysis techniques for solving practical problems.
CO2: Perform statistical analysis on variety of data.
CO3: Perform appropriate statistical tests using R and visualize the outcome
CO4: Design a data mart or data warehouse for any organization

MCAP515 : Android Applications Lab

Course Outcomes:

CO1: Experiment on Integrated Development Environment for Android Application Development.
CO2: Design and Implement User Interfaces and Layouts of Android App.
CO3: Use Intents for activity and broadcasting data in Android App.
CO4: Design and Implement Database Application and Content Providers.
CO5: Experiment with Camera and Location Based service.
CO6: Develop Android App with Security features.

MCAP516 : Machine Learning Lab

Course Outcomes:

CO1: Understand the implementation procedures for the machine learning algorithms
CO2: Design Java/Python programs for various Learning algorithms.
CO3: Apply appropriate data sets to the Machine Learning algorithms
CO4: Identify and apply Machine Learning algorithms to solve real world problems

MCAP517 : Artificial Intelligence Lab

Course Outcomes:

CO1: Design the building blocks of an Intelligent Agent using PEAS representation .
CO2: Analyze and formalize the problem as a state space, graph, design heuristics and select amongst different search or game based techniques to solve them.
CO3: Develop intelligent algorithms for constraint satisfaction problems and also design intelligent systems for Game Playing
CO4: Attain the capability to represent various real life problem domains using logic based techniques and use this to perform inference or planning.
CO5: Formulate and solve problems with uncertain information using Bayesian approaches.
CO6: Apply concept Natural Language processing and cognitive computing for creation of domain specific Chat-Bots.

VI SEMESTER M.C.A.								
Subject Code	**Subjects**	**Theory Hours/ Week**	**Practical Hours/ Week**	**Duration of exams (Hours)**	**Marks & Credits**			
					IA	**Exam**	**Total**	**Credits**
MCAH601	Major Project Internal Assessment Project Report Valuation	-	40	----- -----	120 --	-- 200	120 200	16
	Viva-Voce	-	-	-	--	80	80	-
Total			**40**		**120**	**280**	**400**	**16**

Preamble:

Project work has been made a part of the course to give students exposure in development of quality software solution. The primary emphasis of the project work is to understand and integrate the knowledge and the principles gained in the previous semesters of the programme with software engineering practices. As such, during the development of the project students shall involve themselves in all the stages of the software development life cycle (SDLC) like requirements analysis, systems design, Data Analysis, software development/coding, testing and documentation, with an overall emphasis on the development of reliable software systems. Since, the project work spans over the entire Sixth semester, the students shall be advised to take up projects for solving problems of software industry or any research organization or the real-life problems suggested by the faculty in-charge of Project work in the institutions. Topics thus selected, should be complex and large enough to justify as the course project. The project should be genuine and original in nature and should not be copied from anywhere else.

Hours/Week: 40 Major Project **I.A. Marks: 120**
Credits : 16 Internal Assessment **Exam. Marks: 280**
Project Report Valuation

Course Outcomes:

CO1: Graduates will have an ability to identify, formulate and implement computing solutions.
CO2: Graduates will have an ability to design and conduct experiments, analyze and interpret data related software development projects.
CO3: Graduates will be able to design a system, component or process as per needs and specification.
CO4: Graduates will have the skill to work on multidisciplinary tasks and will be aware of the new and emerging disciplines.
CO5: Graduates will demonstrate skills to use modern tools, software and equipment's to analyze problems
CO6: Graduates will have an ability to identify, formulate and implement computing solutions.
CO7: Graduates will be able to design a system, component or process as per needs and specification of the clients.
CO8: Graduates will have the skill to work on multidisciplinary tasks and will be aware of the new and emerging disciplines that will impact development.