Jyoti Nivas College Autonomous

Post Graduate Centre
Hosur Road, Bangalore -95

Phone No: 255012143 Fax: 25533886
e-mail:info@jyotinivas.org / jncpgcourses@gmail.com
Website: www.jyotinivas.org

CHOICE BASED CREDIT SYSTEM (CBCS)

Proposed Syllabus of

Master of Computer Applications (MCA)

Academic Year 2018-19 Onwards

[Including Lateral Entry Scheme]

Higher Education has undergone quite a lot of transformation over the years. Academic freedom, under autonomy, helped colleges to innovate new curricula, design relevant courses, frame new syllabi and introduce new evaluation methods. But the required flexibility for the students to have a greater choice of courses appropriate to their interests, needs and long-term goals is not available; rather a rigid and compartmentalized system is perpetuated.

Choice based credit system (CBCS) the solution for this type of transformation from the traditional teacher oriented education to a student-centered education. Taking responsibility for their own education in this way, students can benefit the most from all the available resources. The Choice Based Credit System (CBCS) offers wide ranging choice for students to opt for courses based on their aptitude and their career goals.

As per the University Grants Commission, New Delhi recommendations, the College has decided to introduce Choice Based Credit System in its Post-graduate Programmes, with effect from the academic year 2015-16. For multifaceted development of students, curriculum emphasizes on wide variety of courses to enhance their knowledge in several core courses. Thus the present post-graduate programmes in subjects have been restructured to implement the Choice Based Credit Semester Scheme and the system makes the product of the college at par with the global practices in terms of academic standards and evaluation strategies.

The Salient Features of the Choice Based Credit System

The prominent features of the choice based credit system are a process of continuous evaluation of a student's performance/progress and flexibility to allow a student to progress at an optimum pace, suited to her ability or convenience, subject to fulfilling the minimum requirements for continuation.

A student's performance/progress is measured by the number of credits that she has earned, i.e. completed with a pass grade. It permits students to

- Student-centric learning
- Choose electives from a wide range of elective courses offered by other departments
- Undergo additional courses and acquire more than the required number of credits
- Adopt an inter-disciplinary approach in learning
- Make best use of the expertise of available faculty

MCA Course

The Master of Computer Applications (MCA) is a Post Graduate Programme started in the year 2003, affiliated to Bangalore University, and approved by All India Council of Technical Education (AIC). The MCA Programme is a three year course organized around a semester based credit system of study. The syllabi and course content are designed to be flexible and wide-ranging, incorporating the cutting edge as well as ensuring a firm grasp of core fundamentals. The first and second semesters will comprise of core subjects needed to build a base for the subjects to follow in pursuance of the objectives of the course. The third to sixth semesters will be mostly divided into core subjects, electives and open electives, which offer the student the opportunity to build expertise in fields based on personal choice. The last semester includes industry project work / internship, testing the students' application in the fields studied so far and provide them with valuable experience before they step into the professional world of information technology.

Structure of the Programme

- The MCA Course has a curriculum with syllabi consisting of (i) Core courses, (ii) Softcore iii) Elective Courses (iv) Open Electives and iv) Project work / internship.
- In addition, add-on courses, group activities, soft skill and value added training are provided based on the need.
- The total number of credits assigned to the course is 148 and the credits per paper are distributed as follows:
    - 4 credits per hard core and elective courses with 4 lecture hours per week
    - 3 credits per softcorepaper with 3 lecture hours per week
    - 4 credits per open elective paper with 4 lecture hours per week
    - 3 credits per 6 practical hours per week respectively
    - 12 credits for M.C.A. Industry project / Internship
- Core courses are compulsory subjects offered by the department. Total credits for Core Courses including theory, practical papers, softcore and projects are 134.
- Elective courses may be chosen by the student from the list of Electives offered by the department. Total credits for Elective Courses are 12.
- Open Elective has to be taken from other post graduate disciplines. Credit for open Elective is 4.

The Proposed Course Structure

Semester	Paper		No. of hrs/week	Marks		Credits
				CIA	ESE	
I / II	Core	5T	5 x 4	5 x 50	5 x 100	5 x 4
		2P	2 x 6	2 x 50	2 x 100	2 x 3
	Total					26 / sem
III	Hard Core	3T	3x 4	3x 50	3 x 100	3 x 4
		2P	2 x 6	2 x 50	2 x 100	2 x 3
	Open Elective	1T	1*4	1*50	1*100	1*4
	Elective I	1T	1 x 4	1 x 50	1 x 100	1 x 4
	Total					26 / sem
IV	Core	4T	4*4	4*50	4*100	4*4
		2P	2*6	2*50	2*100	2*3
	Soft Core I	1T	1*3	1*50	1*100	1*3
	Total					25/sem
V	Core	3T	3 x 4	3 x 50	3 x 100	3 x 4
		2P	2 x 6	2 x 50	2 x 100	2 x 3
	Elective - II	1T	1 x 4	1 x 50	1 x 100	1 x 4
	Soft Core II	1T	1 x 3	1 x 50	1 x 100	1 x 3
	Total					25 / sem
VI	Elective - III	1T	1 x 4	1 x 50	1 x 100	1 x 4
	Elective - IV	1T	1 x 4	1 x 50	1 x 100	1 x 4
	Industry Project		24	100	200	12
	Total					148

Course Matrix

Semester	Paper Code	Title of the Paper	Hours / Week	Marks			Credits	
				CIA	ESE	Total	Paper	Semester
I	1MCA1	Computer Organization and Architecture	4	50	100	150	4	26
	1MCA2	Programming in C	4	50	100	150	4	
	1MCA3	Web Technologies	4	50	100	150	4	
	1MCA4	Operating Systems	4	50	100	150	4	
	1MCA5	Discrete Mathematics	4	50	100	150	4	
	1MCA6	C Programming Lab	6	50	100	150	3	
	1MCA7	Web Technologies Lab	6	50	100	150	3	
II	2MCA1	Data Structures	4	50	100	150	4	26
	2MCA2	Computer Networks	4	50	100	150	4	
	2MCA3	Database Management Systems	4	50	100	150	4	
	2MCA4	Object Oriented Concepts using Java	4	50	100	150	4	
	2MCA5	Probability and Statistics	4	50	100	150	4	
	2MCA6	Java Programming Lab	6	50	100	150	3	
	2MCA7	Data Structures Lab	6	50	100	150	3	
III	3MCA1	Software Engineering Methodologies	4	50	100	150	4	26
	3MCA2	Design and Analysis of Algorithms	4	50	100	150	4	
	3MCA3	Linux Programming	4	50	100	150	4	
	3MCA4	Linux Programming Lab	6	50	100	150	3	
	3MCA5	Database Management Systems Lab	6	50	100	150	3	
	3MCA6	Elective I	4	50	100	150	4	
	3MCA7	Open Elective	4	50	100	150	4	

	4MCA1	Distributed Systems	4	50	100	150	4	25
	4MCA2	Exploring Python	4	50	100	150	4	
	4MCA3	Mobile Application Development	4	50	100	150	4	
	4MCA4	Information Security	4	50	100	150	4	
	4MCA5	Python Programming Lab	6	50	100	150	3	
	4MCA6	Mobile Apps Development Lab	6	50	100	150	3	
	4MCA7	Soft Core I	3	50	100	150	3	
V	5MCA1	Cloud Computing	4	50	100	150	4	25
	5MCA2	Data Mining and Business Intelligence	4	50	100	150	4	
	5MCA3	Artificial Neural Networks	4	50	100	150	4	
	5MCA4	Elective II	4	50	100	150	4	
	5MCA5	Soft Core II	3	50	100	150	3	
	5MCA6	Mini Project Lab	6	50	100	150	3	
	5MCA7	Research based Project Lab	6	50	100	150	3	
VI	6MCA1	Elective III	4	50	100	150	4	20
	6MCA2	Elective IV	4	50	100	150	4	
	6MCA3	Industry Project Work / Internship	24	100	200	300	12	

The Total Credits (1- 6 semesters):148

List of Soft Core Papers

- Agile Software Engineering
- Visual Programming
- Arduino Programming
- Big Data Analytics
- Advanced Java
- Financial Accounting and Management

List of Electives

A. Advanced DBMS Concepts
B. Theory of Computation
C. Digital Image Processing
D. Software Testing
E. Natural Language Processing
F. Information Storage and Management
G. Multimedia Systems
H. Research Methodology
I. TCP / IP
J. Mobile Computing
K. Embedded Systems
L. Machine Learning Techniques
M. System Programming
N. Computer Graphics using OpenGL

Certificate Courses

- Logic N Life
- Soft Skills & Communication Skills
- Academic Writing
- Responsive Web Designing
- Recent Technologies

Proposed Open Electives offered by the Department of MCA to other Discipline Students

- Cyber Security
- Big Data Analytics

END SEMESTER EXAMINATION

1. End Semester Examination will be conducted for the total marks of 100and the time duration is 3 hours.
2. The Syllabus is divided in to five major units; the questions should be equally distributed among all the units.

Question Paper Pattern

Part A
A student has to answer 10 questions out of 12. Each question carries 4 marks

Part B
A student has to answer 6 questions out of 10. Each question carries 10 marks.

CONTINOUS INTERNAL ASSESMENT (CIA) PATTERN

CIA for Theory

Component	**Marks**
Mid Term Examination	20
Assignment / Presentation / Case Study / Mini Project	20
Group Activity	5
Attendance	5
Total	**50**

CIA for Practical

Component	**Marks**
Mid Term Examination	20
Program Implementation during class	20
Documentation	5
Attendance	5
Total	**50**

Semester I

1MCA 1: COMPUTER ORGANIZATION AND ARCHITECTURE (60 Hrs)

Objective: This Paper focuses on the fundamental concepts of digital logic such as number systems, combinational and sequential circuits, interaction between hardware and software components, structure and behaviour of computer systems, instruction sets and formats.

Learning Outcomes

After completion of this course, the student will be able to:

- Experiment different digital circuits
- find the applications of different logic circuits
- to write set of micro-operations

Unit I: Digital Circuits: Introduction ,Digital computers, Logic gates, Boolean algebra, Demorgan's theorem, map simplification, Data representation: Data types, Complements, fixed point representation, Floating-point representation, other binary codes and Error detection codes, Combinational Circuits: Half adder, Full adder.

(12)

Unit II: Flip- flops:- SR, D, JK, T, Edge Triggered Flip flops, Sequential Circuits:-Flip-flop Input Equations, state table , state diagram. Digital Components:-Decoders, Encoders, Multiplexer, Registers:-,Registers with parallel load, Shift Registers-Bidirectional shift register with Parallel load, types of shift registers, Binary Counters-Binary counters with parallel load. Memory Unit-RAM, ROM, types of ROMs

(12)

Unit III: Register Transfer and Micro operations:- Register transfer Language, Register transfer, Bus and Memory transfers, Arithmetic Micro operations, Logic Micro operations, Shift Micro operations, Arithmetic Logic Shift Unit.

(12)

Unit IV: Basic Computer Organization:- Instruction Codes, Computer registers, Instructions, Timing and Control, Instruction Cycle, Memory-Reference instructions, Input-Output and interrupt, Design of Accumulator Logic.

(12)

Unit V: Central Processing Unit:- Introduction, general register organization, stack organization, instruction formats, Addressing modes, Data transfer and manipulations, program control, reduced instruction set computer, overlapped register windows.

(12)

Reference Books:

1. Morris Mano, Computer *System Architecture*, Prentice Hall of India- 3rd Edition-2007.

2. John.P.Hayes, *Computer Architecture and Organization*, Mcgraw Hill International Editions, 3rd Edition, 2012.
3. V.Carl Hamachar,*Computer Organization*, Mcgraw Hill ,5th editon 2002.
4. William Stallin, *Computer Organization and Architecture*, Pearson 8th edition, 2012.

1MCA 2. PROGRAMMING IN C [60 Hrs]

Objective: The first programming language a student learns in a computer science course is C. This teaches them the programming logic, syntax and semantics. This also creates a platform to learn other programming languages like C++, Java etc.

Learning Outcomes

After completion of this course, the student will be able to:

- Design C programs for problems
- Write and execute C programs for simple applications

Unit I: Introduction: Basics of programming – Algorithm, flowchart, pseudo code. Introduction to C - development of C, features, constants, variables, data types, operators and expressions, library functions. I/O statements - formatted and unformatted I/O, scanf(), printf(), getchar(), putchar() functions.

Control structures and Arrays: Control structures: Conditional and unconditional – if, for, while and do-while, switch, break and continue, go to statement. **(12)**

Unit II: Arrays: Single and Multidimensional arrays, Strings and string functions. Functions: Definition, different types, advantages, calling a function, passing parameters, call by reference and call by value. Local and global variables, recursive functions.**(12)**

Unit III: Pointers: Introduction, Pointer features, declaration, operations. Pointer and arrays, pointers of pointers, pointers to pointers, pointer and strings, void pointers. **(12)**

Unit IV: Structures and union: Defining a structure, classification, union, user-defined data types, pointer to a structure and structure as an argument to a function.

Macros and storage classes: Definition, preprocessor, macro classification, #if, #elseif, #endif, #define. Storage classes different types, enumerated data type. **(12)**

Unit V: Files: Introduction: Streams and file types, steps for file operations, file I/O structure read and write, file functions.

Bitwise Operators: Introduction, Bitwise AND, OR, Exclusive OR, Bitwise shift Operators and Bit Fields. **(12)**

Reference Books

1. Ashok N. Kamthane, *Programming with ANSI and TURBO C*, Pearson Education, 2002.
2. Herbert Schildt, *The Complete Reference*, Tata Mc-Graw Hill, 6th edition, 2005.
3. YeswanthKanitkar,*Let us C*, BPB Publications, 10th Edition, 2010.
4. E.Balagurusami, *ANSI C*, Tata McGraw Hill, revised edition, 2010.

1MCA3. WEB TECHNOLOGIES [60 Hrs]

Objective: The objective of this paper is to familiarize the students with creating client side and server side programming and enable them to develop efficient web pages. At the end of the course, student should be able to, understand the basics of Web, Building static web pages using HTML, Perform client side scripting using JavaScript and Create dynamic Internet application using PHP components.

Learning Outcomes

After completion of this course, the student will be able to:

- Understand various web technology tools used to develop web pages.
- Enhance the students to develop their own Web page and how to host own web site on internet.

Unit I: Basic Web Concepts: WWW, Web Server, Web Browser, Static and Dynamic web pages.

HTML: Basic Tags, Formatting Text, Linking Documents, Using Colors and Images, Image Maps, Lists, Tables, Frames, Forms.

CSS: Introduction, Internal, external and embedded style sheets, style sheet classes, DIV and SPAN tags. **(12)**

Unit II: JavaScript : Introduction, Operators and Expressions, JavaScript Programming Constructs, Document Object Model, Working With Objects – Window object, Array Object, String Object, Form Object, IFrame Object, Image Object, Handling Events, Processing Forms, JavaScript Regular expressions.

(**14)**

Unit III: PHP: Introduction to Server Side Scripting, PHP Basic syntax, Embedding PHP Code in the Web pages, control structures, functions, Arrays, strings and Regular Expressions. **(12)**

Unit IV: PHP: File Handling**,** Working with forms, working with Cookies and Sessions and headers. **(12)**

Unit V: MySQL: Introduction to MySQL, CRUD statements, creating database/tables, Using MySQL from PHP, Building dynamic Applications. **(10)**

Reference Books

1. Jon Duckett, *Beginning HTML, XHTML, CSS*2nd Edition, Wiley Publishing, 2010.
2. Chris Bates, *Web Programming – Building Internet Applications*, Wiley Dreamtech India, 3rd Edition, 2009.
3. John Pollock, *Java Script, A Beginner's Guide*, 3rd Edition, Tata McGraw Hill, 2010.
4. Ivan Bayross, *Web Enabled Commercial Application Development on HTML, DHTML, JavaScript, Perl CGI, 3rd* Edition, 2012.
5. Jason Gilmore, *Beginning PHP and MySQL*, 3rd Edition, Apress Publications, 2008.

6. *Timothy*Boronczyk, Elizabeth Naramore, Jason Gerner, Yann Le Scouarnec, *Jeremy Stolz,*MichaelK.Glass, Beginning *PHP6, Apache, MySQL, Web Development,* Wiley India, 2009.
7. Ellie Quigley, Marko GargentaE.Quigley, *PHP and MySQL by Example***,** Prentice Hall, 2006.

1MCA4. OPERATING SYSTEMS [60 Hrs]

Objective: Operating system is essential part of any computer system. This paper focuses on the core structure, functions and design principles of operating system.

Learning Outcomes

After completion of this course, the student will be able to:

- Get the knowledge of process
- To implement Scheduling algorithms
- Handle Deadlock issues

Unit I: Introduction to Operating System - Definition, Types of operating systems, Distributed Systems, Special purpose systems, computing environments.

Operating system structures: services, user operating system interface, system calls, types of system calls, system programs, operating system design and implementation, operating system structure, virtual machines, OS debugging, OS generations and system boot. **(12)**

Unit II: Process Management: Processes, Process concept, Process scheduling, Operations on processes, Inter process communication.

CPU Scheduling: Basic concepts –Scheduling criteria - Scheduling algorithms -Multiple processor scheduling - Real time scheduling - Algorithm evaluation. **(12)**

Unit III: Process synchronization: background, the critical section problem, Peterson's solution, semaphores, classic problems of synchronization,

Deadlocks: System model - Deadlock characterization - Methods for handling deadlocks - Deadlock prevention - Deadlock avoidance - Deadlock detection - Recovery from deadlock. **(12)**

Unit IV: Memory management: Background – Swapping - Contiguous memory allocation – Paging – Segmentation - Segmentation with paging.

Virtual memory: Background - Demand paging - Process creation - Page replacement - Allocation of frames - Thrashing. **(12)**

Unit V: Storage Management: File system interface: file concept, access methods, directory and disk structure, file system mounting, file sharing, protection.
Mass storage structure: overview, disk structure, disk attachment, disk scheduling, disk management, swap space management, RAID structure.

Protection: Goals of protection - Domain of protection - access matrix - Implementation of access matrix - Revocation of access rights - Language based protection. **(12)**

Reference Books

1. Silberschatz, Galvin, *Operating System Concepts*, WSE, 9th Edition, 2010.
2. Andrew S. Tanenbaum, *Modern Operating Systems*, 3rd edition, 2008.
3. William Stallings, *Operating Systems*, 4th Edition, Pearson Education, 2001.

1MCA5. DISCRETE MATHEMATICS [60 Hrs]

Objective: Since the advent of electronic computers, discrete mathematics and numerical methods has become one of the most important theoretical courses for a Master's level program in computer science and applications. This provides tools for analysis and design of computer hardware and computer software. This will enable the students to gain, Mathematical reasoning, combinatorial analysis and Algorithmic thinking. Successful applications of the concepts help in making better logical conclusions.

Learning Outcomes

After completion of this course, the student will be able to:

- Implementing graphs and trees to solve problems
- Develop their reasoning and computational skills

Unit I: Logic, propositional equivalences, predicates and quantifiers, nested quantifiers, methods of proof. Matrices. **(10)**

Unit II: Induction and recursion, sequences and summations, mathematical introduction, Recurrence relations and its solutions. Counting: Basics of counting, Pigeonhole principle, permutation and combinations, generalized permutations and combinations . **(10)**

Unit III: Sets and Set operations, Inclusion –Exclusion and its applications.
Relations: Introduction, n-ary relations and applications, representing relations, Closure of relations, Equivalence relations, Partial ordering. **(10)**

Unit IV: Graphs: Introduction, representing graphs & graph Isomorphism, Connectivity,Euler and Hamilton Paths, shortest path problems, Planar Graphs, Graph coloring.
Trees: Introduction, spanning trees, minimum spanning trees. **(15)**

Unit V: Lattices and Boolean Algebra: Partially ordered set,relations, maps, order relations, recursion and Introduction,Poset,closures,orderings,totalorderings,Hassediagrams.Includes important theorems on lattices. **(15)**

Project work
Each student has to submit a project work (application in computer science) with minimum 10 pages on any one of the above topic.

Reference books

1. Kenneth H. Rosen, *Discrete mathematics and its applications*, 5th edition, Tata McGraw-Hill, 2003. ISBN 0-07-242434-6.

2. C.L.Liu,*Elements of Discrete Mathematics*, 2nd edition, Tata McGraw-Hill, 7th reprint, 2002, ISBN 0-07-043476-x
3. Trembly& R. Manohar, *Discrete Mathematical Structures with its applications to Computer Science*, Tata McGraw-Hill, ISBN 0-07-065142-6.
4. Edward R. Scheinerman, *Mathematics- A Discrete Introduction*, 1st reprint 2001, ISBN 981-240-092-3, Thompson Learning.
5. M.K.Jain, S.R.K.Iyengar, R.K Jain, Numerical Methods for Scientific and Engineering Computation: New Age International (P) Limited. Publishers, Fourth Edition 2003. ISBN: 81 – 224 – 1461 – 3.

1MCA6 C Programming Lab

Demonstration and implementation of C programming concepts learnt in Theory.

1. Using arithmetic expression
2. Branching
3. Looping
4. Nesting of loops
5. Arrays (Single dimensional and Two Dimensional)
6. String manipulation
7. Macros
8. Functions
9. Recursion
10. Pointers
11. Structures
12. Files

1MCA7 Web Technologies Lab

1. Static web page creation using HTML
2. Using style sheets
3. Client Side Scripting using Java Script
4. Developing Dynamic Pages Using PHP
5. Database handling using PHP & MySQL
6. Creating Simple dynamic web application.

Semester II

2MCA1. DATA STRUCTURES [60 Hrs]

Objective: Computer science is primarily concerned with the study of data. It is important to introduce the student to these aspects of data and data structures which are required in modular programming. The basic algorithms related to handle data like stack, queue, tree and graphs are introduced in this subject.

Learning Outcomes

After completion of this course, the student will be able to:

- Understand different data structures and its application.
- It will provide them an ability to write programs for different Data Structures and Algorithms.

Unit I: Introduction: Definition of: Data, data objects, data types, data structure. Purpose of a data structure and implementation of data structure. Introduction to algorithms, properties of algorithms.
Arrays: Array as data structure, Storage representation of arrays, polynomial representation using arrays, sparse matrix representation. Applications of arrays. **(12)**

Unit II: Searching: Introduction, Searching Techniques: linear and binary search. Hashing, Hash functions, collision and collision Resolution.
Sorting: Introduction, insertion sorts, selection sorts, bubble sorts, quick sort, radix sort, enumeration sort and merge sort. **(12)**

Unit III: Lists:Linear list concepts-single and doubly linked list, circular lists, applications, operations on linked list.
Stacks: Introduction, operations on stack, static and dynamic implementation, application of stack: recursion, prefix, infix, and postfix expressions and matching parenthesis in an expression. **(12)**

Unit IV: Queues: Introduction, operations on Queue, static and dynamic implementation, Types of Queue: circular, priority, dequeue. Application of Queue: job scheduling.
Trees: Tree terminology, Binary tree, Binary tree representation, introduction to binary search tree (BST). **(12)**

Unit V:Operation on BST: creation, traversal, insertion and deletion in binary tree, binary tree sort.Balanced Tree: Introduction, AVL tree, height balance in AVL trees. B-tree, insertion and deletion into a B-tree. Expression tree.
Graphs: Terminology, representation, traversal, operations and applications: shortest path problem, minimum spanning tree. **(12)**

Reference Books

1. Robert L. Kruse, Bruce P. Leung, Clovis L. Tondo, *Data structures and program design in C,* BPB Publications, 2010.

2. Mark Allen Weiss, *Data structures and Algorithm Analysis in C,* Tata McGraw hill Publishing, 2nd edition, 2006.
3. Jean Paul Trembly and Soberson, *Introduction to data structures,* Tata McGraw hill Publishing, 2001.
4. Gilberg R.F, Forouzan B.A and Cengage, *Data Structures: A pseudocode approach withc++,* 2nd edition, 2005.

2MCA2. COMPUTER NETWORKS [60 Hrs]

Objectives: The main purpose of the subject is to provide the broad coverage of communications and computer networks. At the end of the semester students will be knowing about the layered protocols, Congestion control algorithms, Routes and routing algorithms, Design applications and standards, Internetworking and applications, Resource sharing and reliability

Learning Outcomes

After completion of this course, the student will be able to:

- Students will be able to understand the concept of networks, different topologies and network devices
- Discuss the objectives and functionalities of the layers of OSI model.
- Describe how the available methods and algorithms are implemented in the real-time networks such as Ethernet, Bluetooth and internet protocols.
- Understand the working of few application protocols such as SMTP, POP and HTTP.

Unit I: Introduction - Uses of Computer Networks - Network Hardware, Network Topology, Types of Networks, Network Software: Protocol Hierarchies, Design Issues for the layers, Connection-oriented and Connectionless Services, Service Primitives, OSI Reference Model, Introduction to internetworking, TCP/IP Model
Physical Layer: Analog and Digital Signals, Digital Transmission – Line Coding, Block Coding, Sampling and Transmission Mode, Modulation Techniques, Transmission Media – Twisted pair, Coaxial cable, Fiber-Optics, Switching Techniques. **(12)**

Unit II: Data link Layer: Design issues, Framing Methods, Error detection and correction, sliding window protocols.
MAC Sub Layer: Static and dynamic channel allocation, LAN protocols - CSMA, Collision Free, Limited Contention and Wireless LAN Protocols. IEEE Standards for LAN - Ethernet, Token rings (FDDI) and Bluetooth – Switching Devices. **(12)**

Unit III: Network Layer: Design Issues, Static and Dynamic Routing, Routing Algorithms – Adaptive, non-Adaptive, Broadcast and Multicast Routing; Routing in Mobile Networks, Congestion Control and Quality of Services – Requirements and Techniques. Overview of IP protocol. **(12)**

Unit IV: Transport Layer: Design Issues, Transport Service Primitives and Service Management, Elements of Transport Protocols – Addressing, Connection Establishment and Release, Flow Control and buffering, Multiplexing and Crash Recovery. Overview of TCP, UDP protocols.
Session Layer: Design Issues, Remote Procedure Calls. **(12)**

Unit V: Presentation Layer: Design Issues, Data Compression Techniques – Run-length and Huffman Encoding, Need for security.

Application layer: Electronic mail – Architecture and Services, the user agent, message formats, Message Transfer and Final Delivery, WWW – Architectural Overview, Static and Dynamic Web Documents, HTTP Protocol, Performance Enhancements and The Wireless Web. **(12)**

Reference Books

1. Andrew S. Tanenbaum,*Computer Networks*,5th Edition, Pearson Education, 2012**.**
2. Behrouz A. Forouzan, *Data communications and Networking*, Tata McGraw-Hill, Third Edition, 2006.
3. *Data communications and computer Networks*, 1st Edition, 5th Reprint, PHI, 2009.

2MCA3. DATABASE MANAGEMENT SYSTEMS [60 Hrs]

Objective: This paper sets strong foundation for design of data base application. Also, this paper helps the students to acquire the basics of databases, different data models and familiarize the relational and object-oriented database technology.

Learning Outcomes

After completion of this course, the student will be able to:

- Create Database , Tables, Stored Procedures, Triggers
- Implement DDL and DML queries and implementing normalization techniques
- Comoare and contrast various Database systems

Unit I: Purpose of database systems, view of data - data abstraction, instances and schemas, data models, Database languages -DDL & DML, Database architecture – two tier & three tier, Data mining and information retrieval. **(12)**

Unit II: Storage structure and file organization: Primary and secondary storage devices, sequential, indexed sequential, random file access, hashing techniques.
E-R data model: Entities, attributes and relationships, different types of attributes, Drawing E-R diagrams. **(12)**

Unit III: Relational Data model: Relation, Integrity constraints-domain, entity and referential integrity constraints, Relational algebra, select, project and join operations, Normalization concepts, first, second, third normal forms, Boyce-Codd normal form, Projection-Join normal form. **(12)**

Unit IV: SQL: Data Definition & Data Manipulation Commands, Sub queries, Correlated sub-query, Concept of a view.

Data Administration: Introduction, security issues, different methods of protecting the data base, database encryption, types of database failure, recovery and concurrency control, Locking Mechanism. **(12)**

Unit V: Object Oriented Database Development: Introduction, defining a class, attribute, user structures, operations, range for an attribute, relationships, defining an abstract class, creating object instances, object query language. **(12)**

Reference Books

1. Elmasri & Navathe, Addison Wesley, *Fundamentals of database systems***,** Pearson Education, 6th Edition, 2010.

2. Fred R McFadden, Jeffrey A Hoffer and Mary B Prescott, *Modern database management system*, 8th edition, 2009.
3. C J Date, *An Introduction to database systems***,** Pearson Education, 8th Edition, 2009.
4. Abraham Silberschatz, Henry F.Korth, S.Sudarshan, *Database System Concepts*, Tata McGraw Hill Publication, 6th Edition.

2MCA4. OBJECT ORIENTED CONCEPTS USING JAVA [60 Hrs]

Objective: The objective of this paper is to teach the fundamentals of the Java language. Java is a pure object-oriented language, language of the Internet and is often used to create applets and other Web-based applications. By studying this core Java, the students will have a solid foundation that will enable them to develop software using J2EE.

Learning Outcomes

After completion of this course, the student will be able to:

- Design problem solutions using Object Oriented Techniques
- Apply the concepts of data abstraction ,encapsulation and inheritance for problem solutions
- Use the control structures of Java appropriately
- Design Java programs for problems
- Write and execute Java programs for simple applications

Unit I: Introduction to object-oriented programming: procedural approach Vs object oriented approach, principles of OOP: encapsulation, inheritance and polymorphism.
Concepts of OOP: Abstraction, overloading, reusability, extensibility, dynamic binding, message passing. **(12)**

Introduction to Java: Importance and features of java, keywords, constants, variables and data types, Operators and expressions, Decision making, branching and looping statements: if..else, switch,?: operator, while, do, for statements, labeled loops, jump statements: break, continue, return. Introducing classes, objects and methods: defining a class, adding variables and methods, creating objects, constructors. **(12)**

Unit II: Arrays and strings: creating an array, one and two dimensional arrays, string array and methods, String and String Buffer classes, Wrapper classes.
Inheritance: Basics types, using super, Multilevel hierarchy, abstract and final classes.

Packages and interfaces: Basics, Access protection, Extending Interfaces and packages.
Exception Handling: Fundamentals, exception types, uncaught exceptions, try-catch block, throw, throws, finally, built in exceptions, creating your own exceptions. **(12)**

Unit III: Multithreaded Programming: Fundamentals, single and multiple thread creation, priorities, synchronization, Thread class, Thread class methods, Runnable interface, inter thread Communication, suspending, resuming and stopping threads.

Input/Output: Basics, Streams, Byte and Character stream, Reading and writing from console and files, Using Standard Java Packages. **(12)**

Unit IV: Event Handling: Different mechanism, the Delegation Event Model, Event Classes, Event Listener Interfaces, Adapter and Inner Classes, Working with windows,

graphics and text, using AWT controls, Layout managers and menus, handling Image, Java Applet. (**12)**

Unit V: Java Swing: Creating a Swing Applet and Application, Programming using Panes, Pluggable Look and feel Labels, Text fields, Buttons, Toggle buttons, Checkboxes,Radio
Buttons,TabbedPanes,ScrollPane,List,Combobox,Trees,Table,scrollbars, Menus and toolbar.

JDBC: The connectivity Model, JDBC/ODBC Bridge, java.sql package, connectivity to remote database, navigating through multiple rows retrieved from a database. (**12)**

Reference Books

1. Patrick Naughton and HerbertzSchildt, *Java-2 The complete Reference***,** seventh edition, 2010.
2. Richard A. Johnson, *An Introduction to Java Programming and Object Oriented Application Development*, CENGAGE Learning India Pvt. Ltd., New Delhi, 1st Edition, 2007.
3. E. Balagurusamy, *Programming with Java: A Primer*, McGraw-Hill Education (India), New Delhi, 3rd Edition, 2008.
4. Harvey M. Deitel& Paul J. Deitel, *Java How to Program*, PHI Learning Pvt. Ltd., New Delh, 8th Edition, 2009.
5. Mahesh P. Bhave& Sunil A. Patekar, *Programming with Java*, Pearson Education, Inc. New Delhi, 1st Edition, 2009.
6. P. Radha Krishna, *Object-Oriented Programming through Java*, Universities Press (India) Pvt. Ltd., Hyderabad, 1st Edition, 2007.

2MCA5. PROBABILITY AND STATISTICS [60 Hrs]

Objective: Statistics is very widely used for data collection and analysis. The advancement in computing technology, particularly in relation to changes in science and business has increased the need for more people to understand the techniques used to examine the large amount of data that is collected. Hence there is a need for studying statistics as an applied science. Modern statistical analyses are computer based and hence there is a need for knowing the basics and applications of Probability and statistics.

Learning Outcomes

After completion of this course, the student will be able to:

- Implement probability and statistics techniques in various applications
- Learn and observe about data distribution

Unit I : Introduction to statistics and basic statistical concepts. Analysis of bivariate data – Karl Pearson's correlation coefficient and its properties, Spearman's rank correlation coefficient.Linear regression - properties of regression lines and regression coefficients**.** **(10)**

Unit II : Multiple regression and correlation, Introduction to discriminant and cluster analysis. **(10)**

Unit III: Probability - Basic definitions, additive law (with proof), independence of events conditional probability, multiplicative law through definition of conditional probability, Bayes' theorem and its applications. **(10)**

Unit IV: Random variables - random variables - definition, discrete and continuous random variables, probability mass function, probability density function, Expectation and variance and their properties. Bivariate distributions - Joint, marginal and conditional functions for discrete and continuous random variables.
Covariance and correlation coefficient.Independence of random variables Addition and multiplication theorems of expectation.
Standard discrete and continuous distributions - Bernoulli, binomial, Poisson distributions - statement of properties.Normal distribution and its properties. **(15)**

Unit V: Sampling distributions - Distribution of sample mean and standard error. Statistical inference - statistical hypothesis, 2 types of errors, level of significance, size and Power of a test.
Test for normal mean, equality of two means for large small samples, small sample test for proportions.
Confidence intervals for means and proportions. Chi square test for goodness of fit and independence of attributes in a contingency table. Analysis of variance for one way and two way classified data. **(15)**

Project work
Each student has to submit assignment on data analysis using tools (studied in the above syllabus)

Reference Books

1. *Probability & Statistics for engineers and scientists*, Ronald E Walpole, Raymond H Myers, Sharon L Myers, sixth edition, Prentice Hall.
2. *Probability and statistics with computer applications*, Trivediet. Al, Tata McGraw Hill.
3. *Fundamentals of Mathematical statistics*, Gupta and Kapoor, Sultan chand& Co.

2MCA6. Java Programming Lab

Demonstrate and Implement the OOPS concepts and Applets using JAVA

2MCA7. DATA STRUCTURES Lab

The practical implementation of data structure studied in the 2nd semester will be done by students.

Lab Work includes:

1. Implementing different types of search.
2. Implementing different sorting algorithms.
3. Demonstrating the linked list operations.
4. Stack implementation
5. Queue implementation
6. Demonstrating Stack application
7. Demonstrating Queue application
8. Implementing Binary search tree and demonstrating its various operations
9. Demonstrating graph search algorithms.

Semester III

3MCA1.SOFTWARE ENGINEERING METHODOLOGIES [60 Hrs]

Objective: To give students a detailed understanding of processes and techniques for building large software systems, develop skills to evolve systems from analysis, to implementation, and to understand best ways of building software.

Learning Outcomes

After completion of this course, the student will be able to:

- Analyze a software problem
- Prepare Requirement analysis and preparing ER and Data flow diagram
- Preparing Test cases and perform Testing

Unit I: Introduction to Software Engineering: Software Engineering development, Software Life Cycle Models, Comparison of various models, Introduction to Agile Principles and Scrum (12)

Unit II:Requirements in various methodologies: Introduction to Object Oriented Methodology, Overview of Requirements Elicitation, Requirements Model, Action & Use cases, Requirements Elicitation Activities, Managing Requirements Elicitation, Requirements management in Agile Projects, Role of Product owner and Product backlog creation and grooming. (12)

Unit – III: Clean code – Introduction -–differentiate good and bad code, functions-comments -formatting-objects and data structures-error handling (12)

Unit – IV: Clean Code - boundaries-unit tests-classes-systems-Emergence, Test driven development (12)

Unit –V: Engineering Tools & Orchestration - continuous integration - Continuous deployment -–deployment pipelines, zero downtime releases, Continuous testing-reliable software releases (12)

References Books

1. Ian Sommeuville, Software Engineering, 5th edition, 2003.
2. Ivar Jacobson, Object Oriented Software Engineering, Pearson, 2004.
3. Grady Booch, James Runbaugh, Ivar Jacobson, The UML User Guide, Pearson Education, 2004.
4. Stephen R. Scach, Classical& Object Oriented Software Engineering with UML and Java, McGraw Hill, 1999.
5. Richard C. Lee, William M. Tepenhard, UML and C++, A Practical guide to object – oriented Development, Pearson Education.
6. Ken Schwaber,MikeBeedle, Agile Software Development with Scrum -Pearson; First Edition edition (2014)

7. Clean Code - A Handbook of Agile Software Craftsmanship - Robert C.Martin Series - Prentice Hall (unit - III & Unit- V)
8. Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation -Jez Humble and David Farley-Addison Wesley Signature Series (Unit-V)

Recommended Reading:

1. The Phoenix Project: A novel about IT, DevOps, and helping your Business Win- Gene Kim, Kevin Behr and George Spafford

3MCA2. DESIGN AND ANALYSIS OF ALGORITHM [60Hrs]

Objective: An algorithm is composed of a finite set of steps, each of which may require one or more operations. This, paper helps the students to learn the different paradigms used in design of an algorithm. This creates strong foundation for problem solving skills in student.

Learning Outcomes

After completion of this course, the student will be able to:

a) Understand the basic concepts of algorithms and analyze the performance of algorithms.
b) Understand various algorithm design techniques for developing algorithms.
c) Understand various searching, sorting and graph traversal algorithms.

Unit I: Introduction to algorithms, analysis of efficiency of algorithms - space and time complexity, asymptotic notations. A simple example of Design - Insertion sort.
Horner's method of evaluating a polynomial at a given point, finding maximum and minimum for a given set of numbers: straight max, straight min, combinations for max and min. Analysis of linear and binary search algorithms. **(15)**

Unit II: Divide and conquer algorithms, sorting, multiplication of two long integers, Strasen's matrix multiplication.
Greedy approach, Optimum scheduling, fractional Knapsack problem, Minimum spanning trees, Single source shortest path problem. **(15)**

Unit III: Dynamic Programming, All pairs shortest path problem, Traveling salesman problem. Optimal- parameterisation for product of sequence of matrices. **(10)**

Unit IV: Back tracking and Brach and bound methods, Least cost method, 4-Queens problem using back tracking, Traveling salesman problem using branch and bound method. **(10)**

Unit V: Limitations of algorithm, lower bound algorithm, decision trees. Introduction to P, NP, NP complete problems, NP hard problems, Approximation Algorithms : Introduction, vertex cover. **(10)**

Reference Books

1. E. Horowitz and Sahani, Fundamentals *of computer algorithms*, Galgotia Publications.
2. Sara Baase and Allen Van Gelder, *Computer algorithms, Introduction to Design & Analysis*, Pearson Education, 3rd edition, 2000.
3. SartajSahni, *Data structures, Algorithms and Applications in C++,* University Press, 2nd edition, 2005.

4. Aho, Hopcroft and Ullman, *The Design and Analysis of Computer Algorithms*, Pearson Education, 2000.
5. D.E. Knuth, *Fundamental algorithms: The Art of Computer Programming (Vol 1)*, Pearson Education, 3rd edition, 1998.

3MCA3 LINUX PROGRAMMING [60 hrs]

Objectives: The course provides comprehensive understanding of the layered architecture of LINUX operating system and its file system. It also focuses on acquiring skills needed to make effective use of Linux utilities, writing shell scripts, process and signal management and inter process communication.

Learning Outcomes

After completion of this course, the student will be able to:

1) Make use of Linux utilities
2) Design shell programs for different applications
3) Design signal generation and handling programs
4) Implement interprocess communication

Unit I: Linux Utilities-File handling utilities, Security by file permissions, Process utilities, Disk utilities, Networking commands, Filters, Text processing utilities and Backup utilities, sed – scripts, operation, addresses, commands, applications, awk – execution, fields and records, scripts, operation, patterns, actions, functions, using system commands in awk. **(12)**

Unit II : Working with shell(bash): Introduction, shell responsibilities, pipes and input Redirection, output redirection, here documents, running a shell script, the shell as a programming language, shell meta characters, file name substitution, shell variables, command substitution, shell commands, the environment, quoting, test command, control structures, arithmetic in shell, shell script examples, interrupt processing, functions, debugging shell scripts. **(12)**

Unit III : Files: File Concept, File System Structure, Inodes, File Attributes, File types, Library functions, the standard I/O and formatted I/O in C, stream errors, kernel support for files, System calls, file descriptors, low level file access – File structure related system calls(File APIs), file and record locking, file and directory management – Directory file APIs, Symbolic links & hard links. **(12)**

Unit IV : Process – Process concept, Kernel support for process, process attributes, process control - process creation, waiting for a process, process termination, zombie process, orphan process, Process APIs. Signals– Introduction to signals, Signal generation and handling, Kernel support for signals, Signal function, unreliable signals, reliable signals, kill, raise , alarm, pause, abort, sleep functions. **(12)**

Unit V : Interprocess Communication : Introduction to IPC, Pipes, FIFOs, Introduction to three types of IPC-message queues, semaphores and shared memory. **(12)**

Reference Books

1. Beginning Linux Programming, 4th Edition, N.Matthew, R.Stones,Wrox, Wiley India Edition.

2. Linux System Programming, Robert Love, O'Reilly, SPD.
3. Unix Concepts and Applications, 4th Edition, Sumitabha Das, TMH.

3MCA4 LINUX PROGRAMMING LAB [60 Hrs]

1. Usage of file & directory related commands
2. Usage of simple filters ((head, tail, sort, uniq& diff)
3. Demonstrate grep, awk and sed filters
4. Simple shell scripts
5. Demonstrate fork(), kill(), sleep() system calls
6. Demonstrate process synchronization
7. Create a file and read, write operations using different child process
8. Demonstrate Semaphores
9. Demonstrate Threads
10. Demonstrate Interprocess communication

3MCA5 DATABASE MANAGEMENT SYSTEMS LAB [60 Hrs]

Implementation of simple Query, nested queries, joints and storage procedures, using Relational databases such as MySQL / MS SQL Server/Oracle

Semester IV

4MCA1. DISTRIBUTED SYSTEMS [60 Hrs]

Objectives: The field of distributed systems covers many different subjects. This paper will guide the students to enrich their knowledge about, concept of distributed systems and their goals, Homogeneous/heterogeneous multicomputer systems, Distributed/ network operating systems, How the processes are internally organized, Organizing a distributed application in terms of clients and servers.

Learning Outcomes

After completion of this course, the student will be able to:

- Understand the basic requirement from hardware & software, the different protocols required to build a distributed system is covered in this syllabus.
- Enable students to understand and build application on distributed system like internet.

Unit 1: Introduction: Definition, Goals, Hardware concepts, Software concepts, client-server model.
Communication: Layered protocols, Remote procedure call, Remote object invocation, message-oriented communication, stream-oriented communication. **(12)**

Unit 2: Process: Threads, Clients, Servers, Code Migration, and Software Agents. **(12)**

Unit 3: Naming: Naming entities, Locating mobile entities, Removing unreferenced entities. (**12)**

Unit 4: Synchronization: Clock synchronization, logical clocks, global state, election algorithms, mutual exclusion, distributed transactions. **(12)**

Unit 5: Consistency and Replication: Introduction, Data-centric consistency models, Client-Centric Consistency models, Distribution protocols, Consistency Protocols. **(12)**

Reference Books

1. 1. Andrew S. Tanenbaum and Maarten van Steen,*Distributed Systems Principles and Paradigms*, Prentice-Hall, India Pvt. Ltd.
2. Pradeep K and Sinha, *Distributed Operating systems Concepts and Design*, Prentice-Hall India Pvt., Ltd.

3. Weijia Jia and Wanlei Zhou, *Distributed Network Systems; from concepts to implementation*, Springer International Edition.

4MCA2. EXPLORING PYTHON [60 Hrs]

Objective: This course introduces the student to Python programming language through its core language basics and program design techniques suitable for modern applications. To understand the wide range of programming facilities available in Python and to utilize high-performance programming constructs available in Python to develop solutions in real life scenarios.

Learning Outcomes

After completion of this course, the student will be able to:

- To understand the wide range of programming facilities available in Python and develop solutions to simple computational problems
- To utilize high-performance programming constructs available in Python to represent and process compound data
- To develop solutions for real-time projects in areas that includes data analytics, visualization, image processing, artificial intelligence and machine learning

UNIT I: Introduction to Python: Getting Started: Python- an interpreted high-level language, interactive mode and script mode. Variables, Expressions and Statements, Conditional and Looping Constructs.

Functions & Modules: Built-In Function, invoking built in functions, Module (Importing entire module or selected objects using from statement), Functions from math, random, time & date module. User Define Function: Defining, invoking functions, passing parameters (default parameter values, keyword arguments), Scope of variables, void functions and functions returning values**.**

(12 hrs)

UNIT II: Data Structures: Strings: Creating, initializing and accessing the elements, String operators, String built in functions, Strings constants defined in string module, Regular Expression and Pattern Matching.

Lists – Concept of mutable lists, creating, initializing and accessing the elements of list, List operations, List comprehensions, List functions & method. **(12 hrs)**

UNIT III: Dictionaries – Concept of key-value pair, creating, initializing and accessing the elements in a dictionary, Traversing, appending, updating and deleting elements, Dictionary functions & Methods.

Tuples – Immutable concept, creating, initializing and accessing the elements in a tuple.

(12 hrs)

UNIT IV: I/O and File Handling: Formatting Output, Reading and Writing Files, File related modules in Python, File modes and permissions, Reading & Writing data from a

file, redirecting output streams to files, Working with directories, CSV files and Data Files.

Introduction to Object Oriented concepts: Objects, Python Scopes and Namespaces, Classes, Class Objects, Instance Objects, Method Objects, Class and Instance Variables, Inheritance.

(12 hrs)

Unit V: Arrays and Matrices: The NumPy Module, Creating Arrays and Matrices, Copying, Arithmetic Operations, Cross product & Dot product, Saving and Restoring, Matrix inversion, Vectorized Functions.
Data Visualization: Introduction to Matplotlib Module, Creating simple 2D and 3D plots. Overview of different Python libraries / packages and their applications.

Reference Books:

1. Paul Gries, Jennifer Campbell, Jason Montojo, Practical Programming: An introduction to Computer Science Using Python, second edition, The Pragmatic Bookshelf, 2013.
2. Allen Downey, Jeffrey Elkner, 2Learning with Python: How to Think Like a Computer Scientist Paperback –2015.
3. Hans Fangohr, Introduction to Python for Computational Science and Engineering (A beginner's guide), 2015.
4. Timothy A. Budd, Exploring Python, Mc Graw Hill Education, 2009.
5. Mark Lutz, Learning Python, Fourth Edition, O'Reilly publication, 2009.

4MCA3. MOBILE APPLICATION DEVELOPMENT [60 Hrs]

Objective: This course aimed at helping students to learn about mobile applications. It helps students to learn about Android operating system and also to develop applications on Android open-source platform. The course explains the setup of the Android Studio-based development tools, the Android SDK, all essential features, as well as the advanced capabilities and APIs

Learning Outcomes

After completion of this course, the student will be able to:

- Design activities, Fragments
- Sending email , messages through app
- Develop Apps for various applications

Unit – I: Introduction to Android- an open platform for mobile application development - android sdk features - introducing the open handset alliance - why develop for android? - Get to know the required tools - introducing the development framework-android application life cycle. **(12)**

Unit-II: Activities and Android User Interface, Understanding Activities, linking Activities using intents, fragments, calling Built-in Applications using Intents, Displaying Notifications, Understanding the components of a screen, Adapting to Display and Screen Orientation utilizing the Action Bar, Creating the user Interface programmatically, listening for UI Notifications **(12)**

Unit - III: **Designing with views and Displaying Pictures and Menus**: Using Basic Views, Using Picker views, Using List views to display lists, Understanding specialized fragments. Using Image Views to Display Pictures, Using Menus with Views, Some Additional Views. **(12)**

Unit - IV: **Data persistence and Content Providers**, Saving and Loading user preferences, creating and Using Databases with help of DBAdapter class. Sharing Data in Android, Using content provider, creating your own content providers **(12)**

Unit - IV : **Messaging, Location based services and Networking** - SMS Messaging, Sending E-mail, Displaying Maps, Getting Location Data, Monitoring a Location. Consuming Web Services using HTTP and consuming JSON Services. (**12)**

REFERENCES

1. Reto Meier, *Professional Android 4 application Development*, Wiley Publications, 2012
2. Jerome (J.F) DiMarzio, *Android -A programmer's Guide*, Tata Mcgraw Hill, 2010.
3. Paul Deitel, Harvey Deitel, Abbey Deitel, Michael Morgano, *Android for Programmers An App-Driven Approach*, Pearson Education Inc., 2012.
4. Wei-Meng Lee, *Beginning android 4 application Development*, John Wiley & sons, Inc, 2012.

4MCA4. INFORMATION SECURITY [60 Hrs]

Objectives: This subject aims to help students to get through knowledge about network and information security. At the end of the semester the students should have knowledge about various cryptographic algorithms and comparative study of the algorithms, about email security and secured electronic transactions and brief knowledge about intruders and viruses.

Learning Outcomes

After completion of this course, the student will be able to:

- Understand the principles and practices of cryptography and information security, Security architecture, threats and vulnerabilities
- Appreciate the role played by cryptographic techniques in enhancing network and information security.
- Identify and explain the concepts, protocols and technologies associated with a secure communication and transaction across the Network and the Internet.
- Understand various Security practices and System security standards
- Discuss the objectives of authentication, intrusion detection and access control methods and describe how the available methods are implemented in the defense of a network.
- Apply the different cryptographic operations of symmetric and public-key cryptographic algorithms

Unit I: Introduction - Security Vulnerabilities and Threats, Classification of Security Services, Foundations of modern cryptography – perfect security, information theory, product cryptosystem, cryptanalysis.

Cryptography: Encryption Principles, Conventional Encryption techniques – substitution and transposition, Symmetric cryptography, Block cipher principles, Modes of Operations, DES, AES Algorithms, Key Distribution.

(12)

Unit II: Public Key Cryptography – RSA cryptosystem, Key distribution and Management; Diffie -Hellman key-exchange, ElGamal cryptosystem, Introduction to Elliptic curve cryptography.

Message Authentication and Integrity: Message Authentication (MAC), Hash Function, Message Digests, MD5, SHA, Digital signature and authentication protocol - Digital Signature Standard (DSS), Non-Repudiation Process and Delivery.

(12)

Unit III: Entity Authentication – Password-Based Authentication, Address-Based Authentication, Certificates, Authentication Services - Kerberos, X.509.

Legal and Ethical Issues: Cybercrime and Computer Crime, Intellectual Property, Privacy Ethical Issues. **(12)**

Unit IV: Internet and – Electronic Mail security – PGP, S/MIME – IP Security - IPSec protocol, Transport Layer Security - Secure Socket Layer, Secure Electronic Transactions, Web Security – Threats; Web Security Requirements.

(12)

Unit V: System Security – Intruders, Access Control and Management, Malicious software, viruses, Firewalls. **(12)**

Reference Books:

1. William Stallings, "Cryptography and Network Security – Principles and Practice", 7th Edition, Pearson Education, 2017.
2. Mark Stamp, "Information Security: Principles and Practice", Wiley Publications, Paperback Edition, 2018.
3. Behrouz A. Forouzan, Debdeep Mukhopadhyay, "Cryptography and Network Security", McGraw Hill Education, 3rd Edition, 2015.
4. Atul Kahate, "Cryptographic and Network Security", The McGraw Hill Publications, 3rd Edition, 2013.
5. V K Pachghare, "Cryptography and Information Security", PHI Learning private limited, 2nd Edition, 2015.

4MCA5 PYTHON PROGRAMMING LAB [60 Hrs]

The students are expected to

- Demonstrate and implement the concepts learnt in Python Programming course.
- Develop a simple application / project based on python application-oriented package.
  The project can be a group project with a maximum of two students in a group.

4MCA6 MOBILE APPS DEVELOPMENT LAB [60 Hrs]

Demonstration and implementation of features of Android

1. Implementation of the features activities, intent and fragments.
2. User Interface and Designing with views.
3. Utilizing the Action Bar.
4. Listening for UI Notifications.
5. Implementation of list views to display list.
6. Usage of specialized fragments.
7. Implementation of content providers.
8. Using image views to display pictures.
9. Implementation of menus with views.

Finally Develop an Android App using the concepts Learned.

Semester V

5MCA1 : CLOUD COMPUTING [60 Hrs]

Objectives: This subject aims to help students to get through knowledge about Cloud computing. At the end of the semester the students should have knowledge about various cloud service providers and cloud models and security, Service oriented architecture with cloud and Benchmarking

Learning Outcomes

After completion of this course, the student will be able to:

- Understand different cloud service providers
- Identify the need of various cloud deployment models
- Get knowledge about various security issues in cloud environment

UNIT–I

Introduction: Essentials, Benefits and need for Cloud Computing-Business and IT Perspective-Cloud and Virtualization-Cloud Services Requirements-Cloud and Dynamic Infrastructure-Cloud Computing Characteristics Cloud Adoption.
Cloud Models: Characteristics-Measured Service-Cloud Models-Security in a Public Cloud - Public versus Private Clouds-Cloud Infrastructure Self Service **(12)**

UNIT II

Cloud as a Service: Gamut of Cloud Solutions-Principal Technologies-Cloud Strategy Cloud Design and Implementation using SOA-Conceptual Cloud Model-Cloud Service Defined
Cloud Solutions: Cloud Ecosystem-Cloud Business Process Management-Cloud Service Management-Cloud Stack-Computing on Demand (CoD)–Cloud sourcing**.** **(12)**

UNIT–III

Cloud Offerings: Information Storage, Retrieval, Archive and Protection-Cloud Analytics -Testing under Cloud-Information Security-Virtual Desktop Infrastructure-Storage Cloud.
Cloud Management: Resiliency–Provisioning-Asset Management-Cloud Governance-High Availability and Disaster Recovery-Charging models, Usage Reporting, Billing and Metering. **(12)**

UNIT IV

Cloud Virtualization Technology: Virtualization Defined-Virtualization Benefits-Server Virtualization-Virtualization for x86 Architecture-Hypervisor Management Software-Logical Partitioning (LPAR)-VIO Server-Virtual Infrastructure Requirements.
Cloud Virtualization: Storage virtualization-Storage Area Networks-Network-Attached storage-Cloud Server Virtualization-Virtualized Data Center. **(12)**

UNIT–V

Cloud and SOA: SOA Journey to Infrastructure-SOA and Cloud-SOA Defined-SOA and IaaS-SOA-based Cloud Infrastructure Steps-SOA Business and IT Services.

Cloud Infrastructure Benchmarking: OLTP Benchmark-Business Intelligence Benchmark-e-Business Benchmark-ISV Benchmarks-Cloud Performance Data Collection and Performance Monitoring Commands-Benchmark Tools**.** **(12)**

Text Books:

1.Cloud Computing– Insight into New Era Infrastructure, Dr. Kumar Saurabh, Wiley India

2.Michael Miller, Web based Application that change the way and collaborate online, cloud computing, Pearson publication ,2012

3.Anthony T volte, Toby J volte, Robert Elsenpeter, Cloud computing, A Practical Approach , Mcgraw Hill, 2010.

4.J Hurwitz , Cloud computing for Dummies, ISBN 978-0-470-484-8.

5MCA2. DATA MINING AND BUSINESS INTELLIGENCE [60 Hrs]

Objective: In this information age the challenge is to analyze the data to arrive at meaningful conclusion. The aim of this paper is to know data warehousing architecture and techniques, data preprocessing methods, data mining functionalities, different algorithms for classification and clustering and a brief introduction to business intelligence.

Learning Outcomes

After completion of this course, the student will be able to:

- Apply data mining techniques and methods to large data structures
- Data Mining tools
- Compare and contrast the various classifiers

Unit 1: Introduction to Data warehouse: An overview – What is data warehouse? – A multidimensional data model – data warehouse architecture – data warehouse implementation. OLAP Technology – Need for OLAP, ROLAP, MOLAP and HOLAP, OLTP Vs OLAP. (12)

Unit 2: Data preprocessing: Why preprocess the data? – Data cleaning – Data integration and transformation – Data reduction – Data Discretization and concept hierarchy generation.From data warehouse to data mining**:** What is Data mining? – Types of databases – Data mining functionalities – Classification of data mining systems

(12)

Unit 3: Data mining task primitives – Integration of a data mining system with a database or data warehouse system– Major issues in data mining.Mining frequent patterns and Associations**:** Basic concepts – Market basket analysis – efficient and scalable frequent item set mining methods (12)

Unit 4: Classification and Prediction: What is classification? What is Prediction? – Issues regarding classification and prediction – Classification by decision tree induction – Bayesian classification – Rule-based classification – Prediction – Linear Regression – Non-Linear regression (12)

Unit 5: Cluster Analysis: What is cluster Analysis? – Types of data in cluster analysis – categorization of major clustering methods – partitioning methods (k-means, k-medoids) – hierarchical methods (AGNES, DIANA) – density-based methods (DBSCAN, OPTICS).

Business Intelligence:
Introduction to Business Intelligence: BI Architecture – spread sheets, concept of dashboard, decision engineering, BI Tools, BI Application in various domains, Deploying and supporting DW/BI system. (12)

REFERENCES

1. Jiawei Han and Micheline Kamber, *Data Mining Concepts and Techniques*, 2nd Edition, 2007, Morgan Kaufmann Publishers, An imprint of Elsevier.

2. Pang-Ning Tan, Michael Steinbach and Vipin Kumar, *Introduction to Data mining*, 4th Edition, 2009, Pearson Education.
3. Rajiv Sabherwal, Irma Becerra-Fernandez, *Business Intelligence: Practices, Technologies, and Management*
4. George M. Marakas, *Modern Data warehousing: Mining and Visualization*, PE, 2003
5. W.H. Inmon, *Building the Data Warehouse*, 3rd Edition, Wiley Dreamtech.
6. Arun Pujari, *Data mining Techniques*, University Press.

For Continuous Internal Assessment only:

Implementation of simple algorithms related to cluster analysis

5MCA3. ARTIFICIAL NEURAL NETWORKS [60 Hrs]

Objective: The course introduces neural networks, essentials of it, various architectures, its relation with mathematical models, and application arenas. This course would enable students to model networks pertaining to an application.

Learning Outcomes

After completion of this course, the student will be able to:

- Understand the learning and generalisation issue in neural computation.
- Understand the basic ideas behind most common learning algorithms for multilayer perceptrons, and Kohonen self-organising maps.
- Implement common learning algorithms using an existing package.
- Apply neural networks to classification and clustering problems.

Unit 1: Introduction to Neural Networks: Introduction, Organization of the Brain, Biological Neuron, Biological and Artificial Neuron Models, Characteristics of ANN, McCulloch-Pitts Model, Potential Applications of ANN. Essentials of Artificial Neural NetworksArtificial Neuron Model, Operations of Artificial Neuron, Types of Neuron Activation Function **(10)**

Unit 2: Neural network as directed graphs, ANN Architectures, Classification Taxonomy of ANN – Connectivity, Learning Strategy (Supervised, Unsupervised, Reinforcement), Learning Rules, learning tasks . **(10)**

Unit 3: Single Layer Feed Forward Neural Networks: Introduction, Adaptive filtering problem, Unconstrained optimization techniques, Least-mean-square algorithm, Perceptron, Perceptron Convergence Algorithm. Multilayer feed forward Neural Networks: introduction, Back propagation (BP) algorithm. **(16)**

Unit 4: Associative Memory Models: General Concepts of Associative Memory, Bidirectional Associative Memory (BAM) Architecture, BAM Training Algorithms, Architecture of Hopfield Network and training algorithm **(12)**

Unit 5: Self organizing maps – Competitive Learning: Kohonen network, training law, algorithm, Adaptive Resonance Theory: Introduction, different types, ART1 – architecture, methodology, limitations. Hebbian learning: introduction to principal component analysis. **(12)**

REFERENCES

1. Simon Haykin, *Neural Networks- A comprehensive foundation*, Pearson Education, 2005. *(Unit 1 and 2)*
2. Limin Fu, *Neural networks in Computer Intelligence*, Tata McGraw Hill, 2003. *(Unit 3 and 4)*
3. M. Ananda Rao and J. Srinivas, Narosa, *Neural Networks: Algorithms and Applications*, Publishing House, 2005. *(Unit 3 and 4)*
4. James A Freeman and Davis Skapura, *Neural Networks*, Pearson Education, 2002.
5. Satish Kumar, *Neural Networks: A class room Approach*, Tata McGraw Hill, 2004.

5MCA6 : MINI PROJECT LAB [60 Hrs]

Instructions:

1.The project can be done using any one of the .net packages learnt in the course
2.A group should not exceed 2 students

5MCA7 RESEARCH BASED PROJECT LAB [60 Hrs]

Instructions:

1. Select a recent topic for the presentation
2. Implement the work using any simulator
3. Present the paper with the results and conclusions

Semester VI

6MCA1 Elective III

6MCA2 Elective IV

6MCA3. Industry Project Work / Internship

Duration:

OBJECTIVE

- The objective of the project is to help the students know the environment and requirement of the industry.
- Any platform according to the need of the company
- This will also help them in getting jobs in the industry based on the projects they have done.
- To know the latest technologies used in the industry
- It provides experience to enter into IT industry.

ELECTIVES

A. ADVANCED DBMS CONCEPTS

OBJECTIVES:

- To understand the underlying principles of Relational Database Management System.
- To understand and implement the advanced features of DBMS.
- To develop database models using distributed databases.
- To implement and maintain an efficient database system using emerging trends.

Learning Outcomes

After completion of this course, the student will be able to:

- Work on various kinds of databases
- Identify the suitable database architecture based on the requirements
- Develop database models

UNIT I PARALLEL DATABASE 15

Centralized and Client-Server Architectures – Parallel Systems – Parallel Databases – I/O Parallelism – Inter- and Intra-Query Parallelism – Inter- and Intra-operation Parallelism – Inter-operation Parallelism – Query Optimization – Design of Parallel systems – Parallelism on Multicore Processors. **Distributed Database Concepts**: – Distributed Data Storage – Distributed Transactions- Heterogeneous Distributed Databases – Cloud-based Databases

UNIT II SPATIAl , TEMPORAL and MOBILE DATABASE 9

Spatial Databases: Representation of geometric information- Applications of geographic data-Spatial queries- Indexing of spatial data**:-** k-d trees, quad trees, R-trees. **Temporal Database Concepts:** Introduction, Time representation- calendars and Time Dimensions-Incorporating time in Relational Databases using and Object-Oriented databases-Temporal query language. **Mobile Databases:** Mobile Computing Architecture-Characteristics of mobile environments-Data Management issues

UNIT III XML DATABASES 12

XML Databases: Structure of XML Data – XML Document schema – Querying and Transformation – Storage of XML Data – XML Applications

UNIT IV MULIMEDIA DATABASES 12

Multidimensional Data Structures – Image Databases – Text/Document Databases – Video Databases – Audio Databases – Multimedia Database Design

UNIT V NoSQL DATABASE 12

Why NoSQL? – Aggregate Data Models: Aggregates, Key-Value and Document data models, Column-family stores, Graph databases – Schemaless database

REFERENCES

1. Thomas Cannolly and Carolyn Begg, "Database Systems, A Practical Approach to Design, Implementation and Management", Third Edition, Pearson Education, 2007.
2. Henry F Korth, Abraham Silberschatz, S. Sudharshan, "Database System Concepts", Sixth Edition, McGraw Hill, 2006.
3. Elmasri & Navathe, Addison Wesley, *Fundamentals of database systems***,** Pearson Education, 6th Edition, 2010.
4. V.S.Subramanian, "Principles of Multimedia Database Systems", Harcourt India Pvt. Ltd., 2001.
5. Pramod J. Sadalage , Martin Fowler, "NoSQL Distilled - A Brief Guide to the Emerging World of Polyglot Persistence", Addison-Wesley, 1st Edition, 2012.

B. THEORY OF COMPUTATION

Objectives: To explore the fundamental concepts of regular expressions, finite automata, types of grammars

Learning Outcomes:

After completion of this course students will be able to:

1) Implement Programming Techniques for Turing Machines
2) Understand languages of Pushdown Automata
3) Draw parse trees

UNIT I: Automata and Regular Expressions

Introduction to formal proof – Additional forms of proof – Inductive proofs –Finite Automata (FA) – Deterministic Finite Automata (DFA) – Non-deterministic Finite Automata (NDFA) – Finite Automata with Epsilon transitions.

Regular Expression – FA and Regular Expressions – Proving languages not to be regular – Closure properties of regular languages – Equivalence and minimization of Automata **(12)**

Unit II: Grammars

Grammar Introduction– Types of Grammar - Context Free Grammars and Languages— Parse Trees – Ambiguity in grammars and languages – Simplification of CFG – Elimination of Useless symbols - Unit productions - Null productions – Normal forms for CFG- Greiback Normal form – Chomsky normal form – Pumping Lemma for CFL **(12)**

Unit III: Push Down Automata

Definition of the Pushdown automata – Languages of a Pushdown Automata – Equivalence of Pushdown automata and CFG– Deterministic Pushdown Automata.**(12)**

Unit IV: Turing Machines

Turing Machines, Introduction – Formal definition of Turing machines – Instantaneous descriptions- Turing Machine as Acceptors – Turing Machine as Transducers- Programming Techniques for Turing Machines– Modifications of Turing Machines.**(12)**

UNIT V - COMPUTATIONAL COMPLEXITY

Undecidability- Basic definitions- Decidable and undecidable problems - Properties of Recursive and Recursively enumerable languages – Undecidable problems about Turing Machine – Introduction to NP-Hardness and NP-Completeness. **(12)**

REFERENCES

1. Hopcroft J.E., Motwani R. and Ullman J.D, "Introduction to Automata Theory, Languages and Computations", Third Edition, Pearson Education, 2008.

2. H.R. Lewis and C.H. Papadimitriou, *"Elements of the theory of Computation*", SecondEdition, Pearson Education,2003.

3. Thomas A. Sudkamp," *An Introduction to the Theory of Computer Science*,

Languages and Machines", Third Edition, Pearson Education, 2007.
4. Raymond Greenlaw an H.James Hoover, *" Fundamentals of Theory of Computation, Principles and Practice",* Morgan Kaufmann Publishers, 1998.
5. Micheal Sipser,"*Introduction of the Theory of Computation*", Thomson Brokecole,1997.
6. J. Martin, "*Introduction to Languages and the Theory of computation*" Third Edition, Tata Mc Graw Hill, 2007

C. DIGITAL IMAGE PROCESSING (60 hrs)

Objective: This course exposes the students to the basics of digital image processing and its applications. It covers a broad range of image processing techniques such as Image enhancement, restoration, segmentation and feature analysis. The course also provides develop on-hand experience in applying these techniques to process the images.

Learning Outcomes

After completion of this course, the student will be able to:

- Have a clear perceptive and practical scope of digital image processing, current technologies and issues that are specific to image processing systems.
- Understand the working of different image processing algorithms such as filtering, segmentation, morphological processing and image representation.
- Could implement basic image processing algorithms using image processing tools such as MATLAB or Python.

UNIT I: Fundamentals of Image Processing: Introduction, Components of image processing system, Elements of visual perception, Steps in Image Processing Systems, Image Acquisition, Sampling and Quantization, Pixel Relationships, Colour Fundamentals and RGB, CMY, HSV colourModels.
Mathematical Preliminaries: Vector algebra, Matrix operations, Fourier Transform **(12)**

UNIT II: Image Enhancement: Introduction to Spatial and Frequency domain, Image Operations, Arithmetic, Logical, Statistical and Spatial Operations, Convolution and Correlation, Enhancement in Spatial Domain – Gray level Transformations, Histogram Processing, Spatial Filtering, Smoothing and Sharpening. **(12)**

UNIT III: Filtering in Frequency Domain:Smoothing and Sharpening filters, Homomorphic Filtering.
Image Restoration: Noise models, Constrained and Unconstrained restoration models. **(10)**

UNIT IV:Image Segmentation: Detection of Discontinuities, Edge Operators, Edge Linking and Boundary Detection, Thresholding, Region Based Segmentation
Feature Analysis and Extraction: Image Features, Textures, Boundary representations and Descriptions, Component Labeling, Regional descriptors and Feature Selection Techniques. **(14)**

UNIT V: Image Morphology: Binary and Gray level morphology operations, Erosion, Dilation, Opening and closing operations, Distance transforms, Basic morphological operations.

Image Compression: Error criterion, Lossy Compression, Lossless Compression, Huffman Coding, Run length Coding, Block Coding, Quad Tree Coding, contour Coding. . **(12)**

REFERENCES

1. Rafael C.Gonzalez and Richard E.Woods, *Digital Image Processing*, Third Edition, Pearson Education, 2008
2. S. Sridhar, *Digital Image Processing*, Oxford University Press, 2011.
3. Jayaraman S, Veerakumar T, Esakkirajan S, *Digital Image Processing*, Tata McGraw-Hill Education, 2011
4. Rafael C. Gonzalez, Richard E. Woods , Steven L. Eddins , *Digital Image Processing Using Matlab 2E,* Gatesmark Publishing, 2009.

For Continuous Internal Assessment Only

Implementation of image processing concepts such as enhancement, restoration, segmentation and compression using Matlab / ImageJ / Similar tools

D. SOFTWARE TESTING (60 hrs)

Objective: This subject introduces the student to software testing and various stages of testing, from unit testing, till acceptance testing. This will also cover various techniques in testing including manual and automated testing and also expose the student to various tools and their benefits in testing.

On completion of this course, the student should be able to:

- Understand difference between manual and automated testing
- Understand different testing techniques and different testing approaches
- Testing in Agile software process development

Unit 1: Introduction: Fundamentals of Software Testing – Testing Principles – Need for Testing – Goal and purpose of Testing – Role of Testing in Software project – Testing Requirements – Testing Strategies – Testing Process – Techniques of Testing – Test Planning – Test Case and Test Execution. **(12)**

Unit 2: Levels of Testing: An overview – Unit Testing – Test Stubs and Test Drivers – Integration Testing – Integration Approaches – Regression Testing – System Testing – Acceptance Testing - Alpha and Beta Testing. **(12)**

Unit 3: Testing Techniques: What is Static Testing? – Stages in Static Analysis – Black Box Testing – White Box Testing – Main Metrics used for test coverage – Mutation Testing – Fault-based Testing – Smoke Testing – The view for Object Oriented Testing. **(12)**

Unit 4: Automated Testing and Tools: Benefits of Automation and Tools – Introduction of Test Tools – Test Tools: Viewers – Monitors – Drivers – stubs – Stress – Load Tools **(12)**

Unit 5: Software Test Automation: Programmed Macro – Fully Programmable Automated Test Tools - Random Testing – Realities of using Test Tools and Automation – Need of Test sharing. **(12)**

REFERENCES

1. S A Kelkar, *Software Qualityand Testing*: PHI, 1st Edition, 2012.(Unit 1-3)
2. Ron Patton, *Software Testing*: 2nd Edition, Pearson Education. (Unit 4)
3.Glenford J.Myers, *The Art of Software Testing*, John Wiley and Sons Publications, 2004.
4.Srinivasan D and Gopalswamy R, *Software Testing: Principles and Practices*, Pearson Edition, 2006.
5.Roger S. Pressman, *Software Engineering – A practitioner's approach*, 5th Edition, McGraw Hill.

E. NATURAL LANGUAGE PROCESSING (60 hrs)

Objective

To understand natural language processing and to learn how to apply basic algorithms in this field and also to get acquainted with the algorithmic description of the main language levels: morphology, syntax, semantics, and pragmatics, To conceive basics of knowledge representation, inference, and relations to the artificial intelligence.

At the end of the course, students will be able to:

Demonstrate accomplishments of knowledge and comprehension, application and analysis, and synthesis and evaluation

Unit I

Introduction Knowledge in Speech and Language Processing Ambiguity Models and Algorithms Language, Thought, and Understanding The State of the Art and the Near term Future Some Brief History.

Unit – II

Morphology and Finite-State Transducers – Survey of English Morphology – Inflectional Morphology – Derivational Morphology – Finite-State Morphological Parsing– The Lexicon and Morphotactics – Morphological Parsing with Finite-State Transducers – Orthographic Rules and Finite-State Transducers – Combining FST Lexicon and Rules – Lexicon-Free FSTs: The Porter Stemmer – Human Morphological Processing.

(12)

Unit – III

Word Sense Disambiguation and Information Retrieval – Selection Restriction-Based Disambiguation – Limitations of Selectional Restrictions – Robust Word Sense Disambiguation – Machine Learning Approaches – Dictionary-Based Approaches – Information Retrieval – The Vector Space Model – Term Weighting – Term Selection and Creation – Homonymy, Polysemy, and Synonymy – Improving User Queries – Other Information Retrieval tasks.

(12)

Unit – IV

Discourse – Reference Resolution – Reference Phenomena – Syntactic and Semantic Constraints on Conference – Preferences in Pronoun Interpretation – An Algorithm for Pronoun Resolution – Text Coherence – The Phenomenon – An Inference Based Resolution Algorithm – Discourse Structure.

(12)

Unit – V

Machine Translation – Language Similarities and Differences – The Transfer Metaphor – Syntactic Transformations – Lexical Transfer – The Interlingua Idea: Using Meaning – Direct Translation – Using Statistical Techniques – Quantifying Fluency – Quantifying Faithfulness – Search – Usability and System Development.

(12)

REFERENCES

1. Daniel Jurafsky and James H. Martin, Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition, Pearson Education, 2002.
2. Bharathi, A., Vineet Chaitanya and Rajeev Sangal. 1995. *Natural Language Processing-A Paninian Perspective*. Prentice Hall India, Eastern Economy Edition

F. INFORMATION STORAGE AND MANAGEMENT (60 hrs)

Objectives: To explore the fundamental concepts of Lifecycle Storage System Environment, intelligent storage management, networking technologies.

Learning Outcomes:

After completion of this course students will be able to:

1) Understand Storage Area Network
2) Implement RAID
3) Understand Fiber Channel Topologie

Unit I: Introduction to Information Storage and Management: Information Storage-Data, Types of Data, Information, Storage, Evolution of Storage Technology and Architecture, Data Center Infrastructure, Key Challenges in Managing Information, (12)

Unit II :Information Lifecycle Storage System Environment-Components of a Storage System Environment,Disk Drive Components, Disk Drive Components, Disk Drive Performance, Application Requirement and Disk Performance.
Data Protection: RAID: Implementation of RAID, RAID ArrayComponents, RAIDLevels, RAIDComparison, RAID Impact on Disk Performance.
(12)

Unit III: Intelligent Storagesystem: Components of an Intelligent StorageSystem, Intelligent Storage Array.
Introduction to Storage Area Network: SAN components and building blocks SAN servers, SAN storage,SAN Interface,SAN interconnects,Data access over SAN,**(12)**

Unit IV: Fibre Channel Basics: Overview of Fibre Channel Technology,Fibre Channel Topologies,Fibre Channel Layers,classes of service, Fibre Channel Products-FibreChannel Connectors,Hubs,Switches. **(12)**

UnitV: Storage Networking Technologies:Direct-Attached Storage and Introduction to SCSI. Network-Attached Storage. Content-Addressed Storage. Storage Virtualization. Storage security management: Securing and managing the storage infrastructure.
(12)

REFERENCES

1. Storage Professionals EMC Corporation, *Information Storage and Management*, Wiley, 2009
2. MeetaGupta, *Storage Area Network Fundamentals*, Pearson Education Limited, 2002.
3. Robert Spalding,*Storage Networks: The Complete Reference*,TataMcGraw Hill,Osborne, 2003.
4. Marc Farley,*Building Storage Networks*, TataMcGraw Hill, Osborne. 2001.

G. **MULTIMEDIA SYSTEMS (60 hrs)**

Objectives: To explore the fundamental concepts of image compression techniques, multimedia operating systems, and multimedia graphics**.**

Learning Outcomes:

After completion of this course students will be able to:

1) Develop animation videos
2) Make use of multimedia tools
3) Make use of video/audio conferencing techniques

UNIT-I Introduction: What are multimedia, multimedia application, Goal and objectives, Multimedia building blocks, multimedia and internet. **(8)**

UNIT-II Multimedia Configuration: Multimedia PC workstation components, multimedia platform, multimedia development tool, authoring tool, Interactivity, High end multimedia architectures. MULTIMEDIA OPERATING SYSTEM File system (File format: TIEF, BMP, PCX, GIF etc.) Process management, multimedia communication system, multimedia database management system. Multimedia Audio: Basic sound concepts, audio capture, music, speech sound processor, sound recovery technique,VOC4WAV file formats for sound. **(14)**

UNIT-III Multimedia graphics: 2D/3D animation fundamentals, color modules DIGITAL IMAGING: still and moving images; video capture animation video, Processing, video Recovery techniques, AVO, AVI file formats, NTSC, PAL, SECAM, HDTV, system video/audio conferencing techniques and standards, video streaming, motion of synchronization. **(14)**

UNIT-IV Image Compression techniques: LZW, DCT run length coding, JPEG, MPEG, standard hypertext MHEG, Hypertext and Hypermedia, document architecture ODA, MHEG. Augmented and virtual reality and multimedia: Concept, VR devices: hand Gloves, head mounted tracking system, V R Chair, CCD, VCR ,3D, sound system, Head Mounted Displays and rendering software setup, Virtual objects, VRML**.** **(12)**

UNIT-V Multimedia devices: Mass storage systems for multimedia requirements, Magnetic devices, Optical devices, CDROM, DVD. Scanners: Types and specifications. Windows support to Multimedia: Multimedia Databases (in Oracle), multimedia function calls, windows support for sound, animation, movies, music and midi controls. Multimedia and UNIX, Virtual Coffee house application. **(12)**

REFERENCES

1. Ralf Steinmetz & Klara Nahr Stedt, PHI Publications: Multimedia - Computing, Communications and Applications. 2003
2. Judith Jefcoate, Multimedia in Practice: Technology and Application PHI 2008.
3. Durano R Begault, Virual Reality and Multimedia, AP Professionals. 2003
4. Micheal J Young, Windows multimedia and animation with C++ programming for Win95, AP Professional. 2004

H. RESEARCH METHODOLOGY

Objective: To write a project/research report properly and to learn the linear programming, assignment and network models in detail.

Learning Outcomes

After completion of this course, the student will be able to:

- Format the research paper
- Organizing the results and reports
- Understand various network models

Unit 1: Research Methodology: Introduction – Meaning of Research – Objectives of Research – Motivation in Research – Types of Research - Research Process – Defining the Research Problem – Research Design.
Interpretation and Report Writing: Introduction – Interpretation Techniques – Steps in Writing Report – Layout – Types of Reports – Oral Presentation – Writing Research Reports. **(12)**

Unit 2: Introduction to Linear Programming, Construction of LP model, Graphical LP Solution: solution of a Maximization model and minimization model.
The simplex method: Transition from graphical to Algebraic solution, Simplex method: Iterative nature of the simplex method, computational details of the simplex algorithm, Procedure for Penalty method and Two-phase method. **(12)**

Unit 3: Definition of the Dual Problem and conversion of primal to dual problems.
Transportation Model: Definition of the Transportation model, nontraditional transportation models, The transportation Algorithm: Determination of the starting solution, Iterative computations of the transportation algorithm. **(12)**

Unit 4: The Assignment Model: The Hungarian method, Simplex explanation of the Hungarian method, Traveling Sales Person Problem and The Transshipment Model.**(12)**

Unit 5: Network Models: Definitions, CPM and PERT: Network representation, Critical Path (CPM) computations, Construction of the Time Schedule, Linear Programming Formulation of CPM and PERT networks. **(12)**

REFERENCES

1. *Operations Research: An Introduction*, Hamdy A. Taha. Prentice-Hall, India, Seventh Edition.
2. *Operations Research*, S D Sharma. Kedar Nath Ram Nath, Meerut-UP Dist., India, 15th Edition.
3. *Operations Research: Principles and Practice*, Ravindran, Phillips and Solberg. John Wiley & Sons, 2nd Edition.
4. *Research Methodology: Methods and Techniques*, C.R. Kothari, Second Edition, New Age International Publishers, 2004.

I. TCP/IP

Objective: TCP/IP is a protocol system – a collection of protocols that support network communications. The objective of this paper is to provide the theoretical and practical exposure of, Features and structures of TCP/IP, Classification of internet addresses, Internet protocol, ICMP, Routing algorithms, DNS, SNMP and structure IPV6

Learning Outcomes

After completion of this course, the student will be able to:

- Know about usage and suitability of different protocols
- Understand about different kinds of server
- Manage Network protocols

Unit 1: TCP/IP Fundamentals: Internetworking Concepts, Internet Protocols and Standards, OSI Model, TCP/IP Protocol Suite.
IP Protocol: Protocol Format, Classful Addressing, IPv6, Subnetting, Supernetting and CIDR, Address Resolution: ARP and RARP Protocols. **(15)**

Unit 2: IP Packets: Delivery, Forwarding and Routing, IP Design: ICMP protocol, PING and Trace Route Program. IP Routing: Routing Model, Routing Table, Types of Routing and Routing Algorithms: RIP, OSPF and BGP **(15)**

Unit 3: TCP/IP Name Resolution: The Domain Name system.Transport Protocols: UDP Protocol Format, UDP Operation, **(8)**

Unit 4: Transmission Control Protocol: TCP Services, Protocol Format, Connection Management, Transition Diagram, Flow Control, Error Control and Congestion Control.
TCP/IP Tools: File Transfer Protocols, Remote Login, Terminal Emulation, Dynamic Configuration using DHCP. **(10)**

Unit 5: Internet Services: Getting started with Internet, IIS Server Architecture and Services, Dial up Networking Concepts, VoIP, and Internet Security: IPsec and SSL.
Networking Management Concepts, Performance Monitoring, SNMP Protocol, Network Troubleshooting. **(12)**

REFERENCES

1. *TCP/IP Protocol Suite*, Behrouz A. Forouzan, Tata McGraw Hill, Third Edition, 2006.
2. *TCP/IP Illustrated* Volume I, W.Richard Stevens and G.Gabrani, Pearson Education, 2009 Edition.
3. *Internetworking with TCP/IP, Principles, protocols and architecture*, Douglas E. Comex, Fifth Edition, PHI, 2006.
4. *TCP/IP Unleashed,*Tim Parker, Mark A. Sportack, Techmedia, Edition 2000.

J. MOBILE COMPUTING (60 hrs)

Objectives:

The objective of this paper is to provide the comprehensive and in-depth knowledge of Mobile Computing concepts, technologies, architecture and Protocols by introducing and equipping with fundamental concepts, technologies, Data management issues and Security and also to expose the students to frontier areas of Mobile Computing by providing sufficient foundations to enable further exploration on their own.

Outcomes:

After studying the paper, the student would be possessing comprehensive knowledge in mobile computing concepts, technologies, architecture and protocols with the challenges on data management and security. This knowledge will enable them to develop their own mobile applications and explore the world of possibilities.

UNIT – I

Introduction, issues in mobile computing, overview of wireless telephony: cellular concept, GSM: air-interface, channel structure, location management: HLR-VLR, hierarchical, handoffs, channel allocation in cellular systems, CDMA, GPRS. **(12)**

UNIT –II

Wireless Networking, Wireless LAN Overview: MAC issues, IEEE 802.11, Blue Tooth, Wireless multiple access protocols, TCP over wireless, Wireless applications, data broadcasting, Mobile IP, WAP: Architecture, protocol stack, application environment, applications. **(12)**

UNIT – III

Data management issues, data replication for mobile computers, adaptive clustering for mobile wireless networks, File system, Disconnected operations. **(12)**

UNIT– IV

Mobile Agents computing, security and fault tolerance, transaction processing in mobile computing environment. **(12)**

UNIT– V

Ad Hoc networks, localization, MAC issues, Routing protocols, global state routing (GSR), Destination sequenced distance vector routing (DSDV), Dynamic source routing (DSR), Ad Hoc on demand distance vector routing (AODV), Temporary ordered routing algorithm (TORA), QoS in Ad Hoc Networks, applications. **(12)**

REFERENCES

1. J. Schiller, Mobile Communications, Addison Wesley, 2009.
2. A. Mehrotra, GSM System Engineering, Artech House, 1997.
3. M. V. D. Heijden, M. Taylor, Understanding WAP, Artech House, 2011.
4. Charles Perkins, Mobile IP, Addison Wesley, 2010.
5. Charles Perkins, Ad hoc Networks, Addison Wesley, 2009.

K. EMBEDDED SYSTEMS

Objectives: To introduce students to the embedded systems, its hardware and software, to introduce devices and buses used for embedded networking, to explain programming concepts and embedded programming in C and C++.

Learning Outcomes

After completion of this course, the student will be able to:

- Acquire knowledge about Micro-controllers
- Implementing programming concepts in Embedded systems
- Understand Embedded system concepts

Unit 1: Introduction to embedded systems 9: Definition and classification – Overview of Processors and hardware units in an embedded system – Software embedded into the system – Exemplary Embedded Systems – Embedded Systems on a Chip (SoC) and the use of VLSI designed circuits. Embedded Processor - 8051 Microcontroller. **(12)**

Unit 2: Devices and buses for devices network 9: I/O Devices - Device I/O Types and Examples – Synchronous - Iso-synchronous and Asynchronous Communications from Serial Devices - Examples of Internal Serial-Communication Devices - UART and HDLC - Parallel Port Devices - Sophisticated interfacing features in Devices/Ports (12)

Unit 3: Timer and Counting Devices - '12C', 'USB', 'CAN' and advanced I/O Serial high speed buses- ISA, PCI, PCI-X, cPCI and advanced buses.

Programming concepts and embedded programming in C, C++ : programming in assembly language (ALP) vs. High Level Language - C Program Elements, Macros and functions -Use of Pointers - NULL Pointers - Use of Function Calls – Multiple function calls in a Cyclic Order in the Main Function Pointers – Function Queues and Interrupt Service Routines Queues Pointers – Concepts of embedded programming in C++ - (12)

Unit 4: Objected Oriented Programming – Embedded Programming in C++, 'C' Program compilers – Cross compiler – Optimization of memory codes.

Real time operating systems – PART - 1 9: Definitions of process, tasks and threads – Clear cut distinction between functions – ISRs and tasks by their characteristics – Operating System Services- Goals – Structures- Kernel - Process Management – Memory Management – Device Management – File System Organization and Implementation – (12)

Unit 5 :I/O Subsystems – Interrupt Routines Handling in RTOS, REAL TIME Operating systems: RTOS Task scheduling models - Handling of task scheduling and latency and deadlines as performance metrics – Co-operative Round Robin Scheduling – Cyclic

Scheduling with Time Slicing (Rate Monotonics Co-operative Scheduling) – Preemptive Scheduling Model strategy by a Scheduler – Critical Section Service by a Preemptive Scheduler – Fixed (Static) Real time scheduling of tasks **(12)**

REFERENCES

1.Rajkamal,*Embedded Systems Architecture, Programming and Design***,** Tata McGraw-Hill, First reprint Oct. 2003

2.Steve Heath, *Embedded Systems Design*,Second Edition-2003.

3. David E.Simon, *An Embedded Software Primer*,Pearson Education Asia, First Indian Reprint 2000.

4. Wayne Wolf,*Computers as Components; Principles of Embedded Computing System Design*,Harcourt India, Morgan Kaufman Publishers, First Indian Reprint 2001.

5. Frank Vahid and Tony Givargis, *Embedded Systems Design – A unified Hardware /Software Introduction,* John Wiley, 2002.

L. MACHINE LEARNING TECHNIQUES (60 hrs)

OBJECTIVES

- To understand the concepts of machine learning
- To appreciate supervised and unsupervised learning and their applications
- To understand the theoretical and practical aspects of Probabilistic Graphical Models
- To appreciate the concepts and algorithms of reinforcement learning To learn aspects of computational learning theory

Learning Outcomes

After completion of this course, the student will be able to:

- Implementing supervised and unsupervised algorithms
- Writing Algorithms
- Understanding various graphical Models

UNIT I : MACHINE LEARNING 12

Machine Learning - Machine Learning Foundations –Overview – Design of a Learning system - Types of machine learning –Applications Mathematical foundations of machine learning - random variables and probabilities - Probability Theory – Probability distributions -Decision Theory- Bayes Decision Theory - Information Theory

UNIT II : **SUPERVISED LEARNING 12**

Linear Models for Regression - Linear Models for Classification – Naïve Bayes - Discriminant Functions -Probabilistic Generative Models -Probabilistic Discriminative Models - Bayesian Logistic Regression. Decision Trees - Classification Trees- egression Trees - Pruning. Neural Networks - Feed-forward Network Functions - Back- propagation. Support vector machines - Ensemble methods- Bagging- Boosting

UNIT III : UNSUPERVISED LEARNING 12

Clustering – K – Means – EM Algorithm – Mixture of Gaussians. The Curse of Dimensionality - Dimensionality Reduction - Factor analysis - Principal Component Analysis - Probabilistic PCA-Independent components analysis

UNIT IV: **PROBABILISTIC GRAPHICAL MODELS 12**

Graphical Models - Undirected graphical models - Markov Random Fields - Directed Graphical Models -Bayesian Networks - Conditional independence properties - Inference – Learning-Generalization - Hidden Markov Models - Conditional random fields(CRFs)

UNIT V: ADVANCED LEARNING 12

Sampling –Basic sampling methods – Monte Carlo. Reinforcement Learning- K-Armed Bandit-Elements - Model-Based Learning- Value Iteration- Policy Iteration. Temporal Difference Learning-Exploration Strategies- Deterministic and Non-deterministic Rewards and Actions Computational Learning Theory - Mistake

bound analysis, sample complexity analysis, VC dimension. Occam learning, accuracy and confidence boosting

REFERENCES:

1. Christopher Bishop, “Pattern Recognition and Machine Learning” Springer, 2007.
2. Kevin P. Murphy, “Machine Learning: A Probabilistic Perspective”, MIT Press, 2012.
3. EthemAlpaydin, “Introduction to Machine Learning”, MIT Press, Third Edition, 2014.
4. Tom Mitchell, "Machine Learning", McGraw-Hill, 1997.
5. Trevor Hastie, Robert Tibshirani, Jerome Friedman, "The Elements of Statistical Learning", Springer, Second Edition, 2011.
6. Stephen Marsland, “Machine Learning - An Algorithmic Perspective”, Chapman and Hall/CRC Press, Second Edition, 2014.

M. SYSTEM PROGRAMMING (60 hrs)

Objective: The objective is to learn procedures for the design of software systems.

Learning Outcomes

After completion of this course, the student will be able to:

- Acquire knowledge about different Architectures
- Get knowledge about Assemblers, Linkers, Loaders, Compilers

Unit 1: Machine Architecture**:** Introduction, System software, Machine Architecture, Simplified Instructional Computer (SIC) – SIC Machine Architecture , SIC / XE Machine Architecture , SIC Programming examples, Traditional CISC Machines- VAX Architecture, RISC Machines- Ultra SPARC Architectures, Cray T3E Architecture. **(12)**

Unit 2: Assemblers: Basic Assembler Function- A Simple SIC Assembler, Assembler Algorithm and Data Structures, Machine Dependent Assembler Features – Instruction Formats & Addressing Modes, Program Relocation, Machine Independent Assembler Features- Literals, Symbol – Definition Statements, Expression, Program Blocks, Control Sections and Programming Linking. Assembler Design Operations – One- Pass Assembler, MultiPass Assembler. **(12)**

Unit 3: 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 a Linking Loader, Machine-Independent Loader Features – Automatic Library Search, Loader Options, Loader Design Options – Linkage Editor, Dynamic Linkage, Bootstrap Loaders, Implementation Examples – MS-DOS Linker, Sun OS Linker, Cray MPP Linker. **(12)**

Unit 4: Macro Processor**:** Basic Macro Processor Functions – Macro Definitions and expansion, Macro Processor Algorithm and Data Structures. Machine Independent Macro Processor Features – Concatenation of Macro Parameters, Conditional Macro Expansion, Keyword Macro, Parameters, Recursive Macro Expansion. **(12)**

Unit 5: Compilers**:** Basic Compiler Function – Grammars, Lexical Analysis, Syntactic Analysis, Code Generation, Machine Dependent Compiler Features- Intermediate Form of the Program, Machine- Dependent Code Optimization, Machine Independent Code Optimization, Block Structured Languages, Compiler Design Options – Division in to Passes, Interpreters, P-Code Compilers, Compiler- Compilers. **(12)**

REFERENCES:

1.*System Software*, Leland. L. Beck. 3rd Edition, Addison Wesley, 1997.
Chapters: 1 (except 1.5.2), 2 (except 2.5.3) 3,4,5 (except 5.5).
2.*SDP*,John.R. Lvine, Tony Mason and Doug Brown, Lex and Yacc, O' Reilly, 1999.
Chapters: 1,2 (Page 27-42), 3 (Page 51-65)
3. *System Programming and Operating Systems*, D.M. Dhamdhere, 2nd Edition, 1999.

N. COMPUTER GRAPHICS USING OPEN GL [60 hrs]

Objective: Computer graphics has now become a common element in user interfaces, data visualization, television commercials, motion pictures and many other applications. The objective is to familiarize the students with the concepts of computer graphics like line, circle drawing algorithms, transformations, clipping, projection, 3D object representation. To make the students understand how to implement the computer graphics concepts using OpenGL.

Learning Outcomes

After completion of this course, the student will be able to:

- Know about Geometric transformations
- Implementing 2D and 3D transformations using C Editor
- Create graphics model

Unit I: Introduction, Overview of Graphics System: Video display devices, Raster Scan Systems, Random Scan Systems, Input Devices, Hard Copy Devices.Introduction to OpenGL. Graphics Output Primitives Coordinate Reference Frames, Two-Dimensional frame in OpenGL, Point Functions, Line Functions, Line-Drawing Algorithms : DDA, Bresenhams, Curve Functions, Midpoint Circle Algorithm, and Display-window reshape function, Area filling algorithm, Display lists **(12)**

.

Unit II: GeometricTransformations, Two Dimensional Geometric Transformations: Basic Transformations, Matrix Representations, and Homogeneous Co-ordinates, Composite Transformations, Three-Dimensional Geometry Transformations**:** Translation, rotation, scaling transformations, OpenGL geometric transformation functions. **(12)**

Unit III: Two Dimensional Viewing: The Viewing Pipeline, Viewing Co-ordinate Reference frame, Window to Viewport Co-ordinate Transformation, Clipping operation , point clipping, line clipping, polygon clipping, text clipping. **(12)**

Unit IV: Three Dimensional Concepts and Object Representations: Three Dimensional Display Methods, 3D Object Representations: Polygon Surfaces Curved Lines and Surfaces, Quadric Surfaces, Bezier Curves, B–Splines, Octrees and Fractal Geometry Methods. (**12)**

Unit V: Three Dimensional viewing**:** Viewing pipeline, Viewing Co-ordinates, Parallel and Perspective projection.
Visible Surface Detection Methods: Classification of Visible Surface Detection algorithms, Back-Face Detection, Depth-Buffer Method, and Scan-line Methods. **(12)**

Reference Books

1. D. Hearn, M. Pauline Baker, Computer Graphics with OpenGL. PHI, 3rd Edition,New Delhi, 2011.

2. Foley, Vandam&Feiner, Hughes, Computer Graphics Principles &Practice,Pearson Education , 6th Indian Reprint 2001.
3. Richard S Wright, Jr. Michael Sweet,Open GL Super Bible, 2nd Edition.

For CIA Only

Project: Each student has to develop a 3D model using the techniques learnt in OpenGL.

SOFTCORE PAPERS

Soft Core 1: AGILE SOFTWARE ENGINEERING [45 hrs]

Objective: This course introduces candidates to agile way of software development. This course takes the candidate through the software engineering lifecycle with an agile flavor. It introduces them to breaking down requirements, delivering minimum viable product each sprint, best practices to maintain code, importance for automation and tools needed to automate the software development process.

Learning Outcomes

- Learn tools applied on Agile projects

.

Unit I: Introduction to Agile &Requirements in SCRUM: Introduction to Agile, principles. How is agile different to other process Product Vision, Overview of Epics, stories, task. Product backlog creation, refinement, grooming **(5)**

Unit II: SCRUM in action: SCRUM in detail – Identifying SCRUM roles, preparing artifacts and performing ceremonies **(5)**

Unit III: Clean code concepts background and importance, recognize common sources of complexity (code smells), and write code to reduce code smells, write clean and maintainable code. Concept of cyclomatic complexity and technical debt. **(10)**

Unit IV :Dynamic Analysis of Code : Importance of unit testing and introduction to Junit, unit testing patterns, Working with Junit framework, how to use assertions, creating test cases using Junit, executing test cases (single vs batch), Code coverage &Test Driven development overview. **(10)**

Unit V: Continuous Integration – What is CI, benefits and importance.CI using Jenkins orchestration tool. Integrate build, static code analysis, unit test, code coverage using Jenkins (**15)**

Project: Based on concepts learned.

Reference Books

1. Robert C Martin, “Clean Code, A handbook of Agile software craftsmanship”, Prentice Hall, 2015
2. John Ferguson Smart Jenkins, The definitive guide, O’Reilly Publications 2014.
3. Mario E. Moreira Being Agile: Your roadmap to successful adoption of agile, Apress, 2013.

Soft Core 2: VISUAL PROGRAMMING [45 Hrs]

Objective: This paper provides a platform for developing and deploying Web based services in a simple secure and consistent manner.

Learning Outcomes

After completion of this course, the student will be able to:

- List the major elements of the .NET framework
- Explain how C# fits into the .NET platform
- Analyze the basic structure of a C# application
- Debug, compile and run a simple application
- Develop programs using C# on .NET
- Design and develop web based applications on .NET
- Discuss CLR

Unit I: Visual studio .NET: Understanding .NET, The .NET strategy, The origin of .NET technology, The .NET framework, The Common Language Runtime, Framework Base Class, Visual Studio .NET, .NET languages, benefits of .NET approach#

(9)

Unit II :C# Language Fundamentals : Introducing C#, Overview of C#, Literals, Variables, Data types, Operators, Expressions, Decision Making and Branching, Decision Making and Looping, Methods in C#, Handling Arrays

(9)

Unit III: Manipulating Strings, Structures and enumerations. Object Oriented Programming with C#: Classes and objects, Inheritance and polymorphism, Interface: Multiple Inheritance Operator overloading.

(9)

Unit IV: Delegates and events, Managing Console I/O operations, Managing errors and exceptions, Multithreading in C#, Introduction to ASP.NET: Introduction to ASP.NET 3.5, ASP.NET tools and development Environment# and ASP.NET3.5, Doing more with C# and ASP.NET

(9)

Unit V :,HTML forms: A review, Standard web controls, CSS for ASP.NET 3.5 page formatting, Control events and event handlers, Validation controls,ASP.NET and database : A SQL primer.

(9)

Reference Books

1. *Programming in C#* , E. Balagurusamy, 3rd edition, McGrawHill, 2012.
2. *ASP.NET 3.5 Beginner's Guide*, William B Sanders, Tata McGraw Hill, 2009.
3. *ASP.NET The Complete Reference*, Mathew MacDonald, 2002.
4. *ASP.NET*, Black Book Dream Tech press 2011

Soft Core 3 : ARDUINO PROGRAMMING [45 Hrs]

Objective: To explore basics of Arduino programming and implement simple IOT projects

Learning Outcomes:
At the end of the course, students will be able to:
Develop small IoT applications.

Unit 1 : Basics of Microcontrollers- (5)

Microcontroller – Introduction – Microcontroller Overview & Basic Layout - Microcontroller Components/Peripherals - Comparison between Microprocessor and Microcontroller – Advantages of Microcontrollers – Disadvantages of Microcontrollers – Applications.

Unit 2 : Basics of Arduino (5)

Arduino – Overview – Board Types - Arduino boards based on ATMEGA32u4 microcontroller - Arduino boards based on ATMEGA2560 microcontroller - Arduino boards based on AT91SAM3X8E microcontroller – Arduino UNO Board Description - Arduino – Installation.

Unit 3 : Arduino - Program Structure (10)

Arduino Structure – Introduction – Data types – Variables – Constants – Operators – Control statements – Functions – Strings – String Object – Time, Arrays, I/O functions , Advanced I/O functions, Character functions, Libraries.

Unit 4 : Arduino – Simple Exercises (10)

Hello World – a blinking LED - Fading LED - LED Bar graph - Controlling keys in the keyboard, Controlling computer's onscreen cursor - Working with sensors – Humidity, Temperature, PIR, Ultrasonic, switch, DC motor.

Unit 5 Using Arduino to implement simple IOT Exercises (15)

Introduction to IoT – IoT components – IoT Communication Technologies – IoT Communication Modules – Simple exercises Using ESP8266 WIFI Serial module – temperature sensor, heartbeat sensor. Interfacing Arduino and Blynk via USB and ESP8266 – LED Blinking.

As part of this subject, students have to implement a small IoT project Using Arduino Microcontroller.

Soft Core 4 : BIG DATA ANALYTICS [45 hrs]

Objectives: To explore the fundamental concepts of big data analytics, To learn to analyze the big data using intelligent techniques. To understand the various search methods and visualization techniques. To learn to use various techniques for mining data stream. To understand the applications using Map Reduce and Spark Concepts.

Learning Outcomes

- Students will be able to analyze big data using tools
- Students will explore more about data visualization techniques

Unit I: Introduction to Big data , Challenges of Conventional Systems , Intelligent data analysis, Nature of Data , convergence of key trends, unstructured data , industry examples of big data, web analytics, big data and marketing, fraud and big data, risk and big data, credit risk management, big data and algorithmic trading, big data and healthcare, big data in medicine, advertising and big data, big data technologies, introduction to Hadoop, open source technologies, cloud and big data, mobile business intelligence, Crowd sourcing analytics, inter and trans firewall analytics

(12)

Unit II: Types of analytics, descriptive analytics diagnostic analytics, predictive analytics, prescriptive analytics, application of analytics, end,to,end analytics life cycle, key roles of successful analytics project, main phases of the life cycle, developing core deliverables for stakeholders, architecture of the analytics solution architecture, modern data analytic tools

(12)

Unit III: Unstructured data, aggregate data models, key,value and document data models schema less databases, materialized views, distribution models,Map reduce, partitioning and combining, composing map,reduce calculations.

(10)

Unit IV: Introduction to Hadoop:Data format, analyzing data with Hadoop, scaling out, Hadoop streaming, Hadoop pipes, design of Hadoop distributed file system (HDFS), HDFS concepts, Java interface, data flow, Hadoop I/O, data integrity, compression, serialization, Avro, file, based data structures. Map Reduce Applications: MapReduce workflows, unit tests with MRUnit,test data and local tests, anatomy of MapReduce job run, classic MapReduce, YARN, failures in classic MapReduce and YARN, job scheduling, shuffle and sort, taskexecution, MapReduce types, input formats, output formats. Introduction to Spark. **(14)**

Unit V: Related Tools:HBase, data model and implementations, HBaseclients, HBase examples praxis. Cassandra, Cassandra data model, Cassandra examples, Cassandraclients, Hadoop integration. Pig, Grunt, pig data model, PigLatin, developing

and testing Pig Latin scripts. Hive, data types and file formats,HiveQL data definition, HiveQL data manipulation, HiveQL queries.

(12)

References

1. Michael Minelli and Michele Chambers, Big Data Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses, Wiley Publications,2013
2. Zikopoulos, Paul, Chris Eaton, Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data, Tata McGraw Hill Publications, 2012
3. Tom White, "Hadoop: The Definitive Guide", Third Edition, O'Reilley, 2012.
4. VigneshPrajapati, Big data analytics with R and Hadoop, SPD 2013.
5. E. Capriolo, D. Wampler, and J. Rutherglen, "Programming Hive", O'Reilley,2012.
6. Lars George, "HBase: The Definitive Guide", O'Reilley, 2011.
7. Alan Gates, "Programming Pig", O'Reilley, 2011.

Soft Core 5 : ADVANCED JAVA [45 Hrs]

Objective: To learn the concept of advanced JAVA programming and to implement it in the enterprise level application developments. Also to learn the concepts of networking in Java package to get a hands on.

Outcomes : The students will and experience the capability of Java Enterprise Edition in building robust and networked enterprise level applications. The students will also learn the concepts of networking and will implement in simple applications. This experience will help them to confidently face the challenges in the industry related to this domain.

Unit – 1

Introduction: Java and Flavours (SE/EE/ME), significance of JAVA EE Application, XML fundamentals. Introduction to Web servers and containers, Web servers Vs Application Server, Client Server Architecture (2 tier, 3 tier). **(9)**

Unit – 2

Introduction to Server side Programming. Introduction to Java Servlets, CGI Vs Servlet, Servlet Architecture, Servlet life cycle. HTTP request, response, request dispatcher, scope, session tracking, cookies, listeners & Filters, Web.XML (DD) **(9)**

Unit – 3

JSP introduction, JSP Vs other technologies (HTML, ASP, JavaScript), JSP Lifecycle, servlet Vs JSP, implicit objects, declaration, expression, scriptlet, JavaBean, Expression Language (EL), MVC & usage. **(9)**

Unit – 4

Programs covered: reading from xml from html - Request Dispatcher using Servlets - Session using Servlets -Filter using servlets - javabean component – MVC - Expression Language in JSP **(9)**

Unit – 5

Web Services: Introducing web services, SOAP, WSDL, UDDI, web services architecture and usage with real time examples**.**

(9)

Reference Books

1. Marty Hall and Larry Brown, Core Servlets and JavaServer Pages, Volume 1: Core technologies, Pearson Publishing, 2nd Editon, 2011.
2. Couch and Steinberg, *J2EE 1.4 Bible*, Wiley Publishing Inc., 2004.
3. Cay S.Horstmann and Gary Cornell, "Core Java Volume II – Advanced Features", Eighth edition, PHI, 2008.
4. Kogent Solution Inc, "Java 6 Programming Black Book", Dreamtech Press, 2007.

5. Paul Deitel and Harvey Deitel, “Java How to Program”, 9th Edition, Prentice Hal, 2012.

Herbert Schildt , “Java The Complete Reference”, 8th Edition, Tata McGraw Hill, 2011

Soft Core 6: FINANCIAL ACCCOUNTING AND MANAGEMENT (45 Hrs)

Objective: This paper aims to enable students to acquire state of the art knowledge and understanding of the field of Finance and accounting. It provides an educational experience of different financial analysis and the essentials of budgeting.

Learning Outcomes

- Analysing and Evalating data using budget tools

Unit I: Fundamentals of Accounting: Introduction to Financial and Management Accounting, Simple Problems on Journal, Ledger and Trial Balance **(12)**

Unit II: Final Accounts with Adjustments - Problems with outstanding and prepaid expenses, outstanding and prepaid income, Bad Debts Calculation **(12)**

Unit III: Financial Analysis – Problems on Trend Analysis, Comparative Statements and Common size Statements, Ratio analysis – Problems on Current Ratio, Liquid Ratio, Absolute Liquid Ratio, Stock Turnover Ratio, Debtors Turnover Ratio, Creditors Turnover Ratio, Net Profit Ratio, Gross Profit Ratio, Operating Ratio, Debt Equity Ratio, Cash flow Statement (AS-3) – Simple Problems **(14)**

Unit IV: Cost Concepts: Basic concepts, Classification of costs, elements of cost and Simple problems on Cost Sheet

Marginal Costing: Meaning, cost volume profit analysis, Difference between Absorption costing and Marginal Costing, Advantages and Disadvantages of Marginal Costing, Problems on Marginal Costing **(10)**

Unit V: Budgeting and Budgeting Control: meaning, essentials, classification, types, merits, limitations, Problems on flexible budget and cash budget **(12)**

Reference Books

1. Jain &Narang and Simmi Agrawal, *Accounting for Managers,* Kalyani Publishers, 4th Edition, 2012.
2. S.N. Maheswari, *Financial and Management Accounting*, Sultan Chand & Sons, 5th edition, 2010.
3. M.Y. Khan & P.K. Jain, *Management Accounting*, Tata McGraw Hill, 5th Edition, 2011-12.
4. Shashi K. Gupta, R.K. Sharma, *Management Accounting*, Kalyani Publishers, 12th edition, 2011.

Open Electives

OPEN ELECTIVE1: CYBER SECURITY [60hrs]

Objectives: The proliferation of Internet has impacted the lives of people in all professions. Equally, they are also prone to get attacked by hackers and intruders and eventually lose their privacy. The objective of this course to understand the need for Cyber security and its related threats and attacks; Learn methods to become secure in the cyber world and securely communicate in the cyber world and become knowledgeable about the best practices related to cyber security.

Learning Outcomes:

After learning this paper, the students will understand the vulnerabilities of their online presence. They will also know the different threats and attacks that are possible online and will follow the best practises to counter them and be safe online.

Unit I: Basics of internet, www, http, html, DNS, IP Address, electronic mail, web browsers, search engines. Social Media: Twitter, Facebook, Youtube, whatsapp, LinkedIn, advantages, disadvantages. **(12)**

Unit II: Need For Cyber Security: Introduction to security, CIA triad, Case studies, security attacks, privacy and security issues related to social networking, Guidelines Methods to secure oneself in the cyber world, Reversible and Irreversible Cryptographic mechanisms, Applications of Digital Signature, Good password practices **(12)**

Unit III: Secure Transactions: e-commerce, advantages of e-commerce, Online banking security- Online shopping fraud, Guidelines and Recommendations, survey on popular e-commerce sites.
Introduction to e-governance, stages of e-governance, advantages, challenges, International Status, Indian status. **(12)**

Unit IV: Everyday Security: Connecting your laptop, mobile devices, PDAs to Internet, Managing your browser, Social Media Security, E-mail security, Safe guarding from Viruses: Antiviruses, Best practices and guidelines. **(12)**

Unit V: Cyber Security Laws: Indian IT Act, 2008 salient features, What is Cyber Forensics? , Functions of cybercrime cell, Responding to a cyber attack.**(12)**

REFERENCES

1. Information Technology Amended Act, 2008, Ministry of Law and Justice, Government of India. (deity.gov.in/sites/upload_files/dit/.../itact2000/it_amendment_act2008.pdf)

2. "Information Security Awareness Handbook, ISEA, Department of Electronics and Information Technology", Government of India, 2010
3. SrinivasBhogle, "E-Governance" Chapter III in Selected Readings on Information Technology Management : Contemporary Issues, Information Science reference, Hershey, New York, page no. 40-61.
4. Tom Huskerson. Social Media, the Good, Bad, and Ugly: Volume. 3. 2014

OPEN ELECTIVE 2: BIG DATA ANALYTICS [60hrs]

Objectives : To understand need for big data, To understand different types of analytics, To explore the fundamental concepts of big data analytics; to learn to analyze the big data using intelligent techniques, to understand the various search methods and visualization techniques.

Learning Outcomes

Unit I: Introduction to BigData Platform – Challenges of Conventional Systems - Intelligent data analysis – Nature of Data - Analytic Processes and Tools - Analysis vs Reporting **(12)**

Unit II: Types of Analytics- Descriptive Analytics Diagnostic Analytics- Predictive Analytics- Prescriptive Analytics- Application of Analytics, End-to-End Analytics Life Cycle- Key Roles of Successful Analytics Project- Main Phases of the Life Cycle, Developing Core Deliverables for Stakeholders- Architecture of the Analytics Solution Architecture , Modern Data Analytic Tools **(12)**

Unit III: Introduction To Streams Concepts – Stream Data Model and Architecture – Stream Computing - Sampling Data in a Stream – Filtering Streams – Counting Distinct Elements in a Stream – Estimating Moments – Counting Oneness in a Window – Decaying Window - Real time Analytics Platform(RTAP) Applications - Case Studies - Real Time Sentiment Analysis, Stock Market Predictions. **(12)**

Unit IV: Introduction to Big Data &Hadoop: Hadoop, Definition of Big Data, open source software related to Hadoop, Big Data solutions on the Cloud, Introducing to Cloud Computing: Cloud Computing, main players in the Cloud space **(12)**

Unit V: Applications on Big Data Using Pig and Hive – Data processing operators in Pig – Hive services – HiveQL – Querying Data in Hive - fundamentals of HBase and ZooKeeper - IBM InfoSphereBigInsights and Streams. Visualizations - Visual data analysis techniques, interaction techniques; Systems and applications **(12)**

REFERENCES

Michael Minelli and Michele Chambers, Big Data Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses

Michael Minelli (Author), Michele Chambers (Author), AmbigaDhiraj (Author) , Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses,Wiley Publications,2013

Zikopoulos, Paul, Chris Eaton, Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data, Tata McGraw Hill Publications, 2012

Michael Berthold, David J. Hand, “Intelligent Data Analysis”, Springer, 2007.