BANGALORE UNIVERSITY

REGULATIONS, SCHEME AND SYLLABUS

For the course

MASTER OF SCIENCE IN COMPUTER SCIENCE
(M Sc (CS))

I to IV Semesters

(Choice Based Credit System –Y2K14 Scheme)

Revised w.e.f.

Academic Year 2014-15 and onwards

MCA PROGRAMME
JNANABHARATHI CAMPUS
BANGALORE UNIVESITY, BANGALORE

SCHEME OF STUDY AND SCHEDULE OF EXAMINATION

1. Title of the course: Computer Science, M.Sc.
2. Duration of the course : 2 years (4 semesters) 3. Eligibility:
    a) B.Sc. (Computer Science) or BCA with Mathematics as one of the subject and atleast 50% aggregate marks of all optional subjects (throughout 3 years B.Sc. / BCA course), B.Sc. PCM with PG Diploma / Certificate in Computer Science of duration one year.
    b) The minimum requirement for SC / ST candidates are relaxed in accordance with University regulations.
4. Intake: 15 + supernumerary quota as per University regulations. Total number of students including payment seats not to exceed – 35. Payment seat fee is as per university guidelines.
5. Admission: A category wise merit list will be prepared with marks obtained in all optionals in all the three years.
6. Attendance: As per regulations of the University for P.G. courses.
7. Medium of instruction: English.
8. Scheme of study: Each semester is of 4 months duration I to III semester: Theory papers 4, Practicals 2 in each semester. IV semester: Project, seminar and viva-voce, theory papers 4.
9. Scheme of examination: There shall be a University examination at the end of each semester.
    a) Appearance for the examination: As per regulations of the University for P.G. Courses.
    b) Provision for repeaters: As per regulations of the University for P.G. Courses.
    c) Dissertation and viva-voce examination: The period of dissertation work is on full semester (4th semester). A student has to select a guide from the department in consultation with the chairperson of the department.
10. Result declaration: As per regulations of the University for P.G. Courses.
11. Miscellaneous:
    a) It is recommended that tutorial work be provided for all theory and practical papers.
    b) Internal assessment: Attendance - 10
       Seminars and Assignments - 10
       Mid-semester exam - 10
    c) Lectures from experts in the field from R&D institutions are highly desirable.

Any other issue not envisaged above shall be resolved by the Vice-Chancellor in consultation with the appropriate bodies of the University, which shall be final and binding.

SCHEME OF STUDY AND EXAMINATION FOR MASTER OF SCIENCE IN COMPUTER SCIENCE (M Sc (CS))

Semester	Paper Code	Title of the paper	Hours / Week	Marks			Credits	
				IA	Exam	Total	Subject	Semester
I	MSC101T	File Structures	4	30	70	100	4	26
	MSC102T	Advanced Database Management Systems	4	30	70	100	4	
	MSC103T	Theory of Computation	4	30	70	100	4	
	MSC104T	Advanced Architecture	4	30	70	100	4	
	MSC105P	File Structures Lab	8	30	70	100	4	
	MSC106P	Advanced DBMS Lab	8	30	70	100	4	
	MSC107T	Soft Core – Quantitative, Teaching and Research Aptitude	3	30	70	100	2	
II	MSC201T	Object Oriented Analysis and Design using UML	4	30	70	100	4	26
	MSC202T	Advanced Java Programming	4	30	70	100	4	
	MSC203T	Artificial Intelligence	4	30	70	100	4	
	MSC204T	Quantitative Techniques	4	30	70	100	4	
	MSC205P	Object Oriented Design using UML Lab	8	30	70	100	4	
	MSC206P	Advanced Java Programming Lab	8	30	70	100	4	
	MSC207T	Soft Core – Soft Skill and Personality Development	3	30	70	100	2	
III	MSC301T	Advanced WEB Programming	4	30	70	100	4	24
	MSC302T	Advanced Algorithms	4	30	70	100	4	
	MSC303T	Cryptography and Network Security	4	30	70	100	4	
	MSC304T	Open Elective	4	30	70	100	4	
	MSC306P	Web Programming Lab	8	30	70	100	4	
	MSC307P	Advanced Algorithms Lab	8	30	70	100	4	
IV	MSC401T	Research Methodology	4	30	70	100	4	24
	MSC402T	Elective – 1	4	30	70	100	4	
	MSC403T	Elective – 2	4	30	70	100	4	
	MSC404T	Scilab Lb	4	30	70	100	4	
	MSC405P	Main Project	16	50	150	200	8	

FIRST SEMESTER MSc

MSC101T: FILE STRUCTURES

Total Teaching Hours: 52 No. of Hours / Week: 04

UNIT – I [12 Hours]

Introduction: File Structures, The Heart of the file structure Design, A Conceptual Toolkit; Fundamental File Operations: Physical Files and Logical Files, Opening Files, Closing Files, Reading and Writing, Seeking, Special Characters, The Unix Directory Structure, Physical devices and Logical Files, File-related Header Files, UNIX file System Commands; Buffer Management, Input /Output in UNIX. Fundamental File Structure Concepts, Managing Files of Records: Field and Record Organization, Using Classes to Manipulate Buffers, Using Inheritance for Record Buffer Classes, Managing Fixed Length, Fixed Field Buffers, An Object-Oriented Class for Record Files, Record Access, More about Record Structures, Encapsulating Record Operations in a Single Class, File Access and File Organization.

UNIT – II [10 Hours]

Organization of Files for Performance, Indexing: Data Compression, Reclaiming Space in files, Internal Sorting and Binary Searching, Key sorting; Index: Introduction, A Simple Index for Entry- Sequenced File, Object-Oriented support for Indexed, Entry-Sequenced Files of Data Objects, Indexes that are too large to hold in Memory, Indexing to provide access by Multiple keys, Retrieval Using Combinations of Secondary Keys. Consequential Processing and The Sorting of Large Files: A Model for Implementing Consequential Processes, Application of the Model to a General Ledger Program, Extension of the Model to include Multi-way Merging, A Second Look at Sorting in Memory, Merging as a Way of Sorting Large Files on Disk.

UNIT – III [10 Hours]

Multilevel indexing and B-Trees: The invention of B-Tree, Statement of the problem, Indexing with Binary Search Trees; Multi-Level Indexing, B-Trees, Example of Creating a B-Tree, An Object-Oriented Representation of B-Trees, B-Tree Methods; Nomenclature, Formal Definition of B-Tree Properties, Worst-case Search Depth, Deletion, Merging and Redistribution, Redistribution during insertion; B* Trees.

UNIT – IV [10 Hours]

Indexed Sequential File access and Prefix B+ Trees: Indexed Sequential Access, Maintaining a Sequence Set, Adding a Simple Index to the Sequence Set, The Content of the Index: Separators Instead of Keys, The Simple Prefix B+ Tree and its maintenance, Index Set Block Size, Internal Structure of Index Set Blocks: A Variable-order B- Tree, Loading a Simple Prefix B+ Trees, B-Trees, B+ Trees and Simple Prefix B+ Trees in Perspective.

UNIT – V [10 Hours]

HASHING: Introduction, A Simple Hashing Algorithm, Hashing Functions and Record Distribution, Collision resolution by progressive overflow, Buckets. How Extendible Hashing Works, Implementation, Deletion, Extendible Hashing Performance, Alternative Approaches.

Reference

1. *Michael J. Folk, Bill Zoellick, Greg Riccardi, "File Structures-An Object Oriented Approach with C++ ", 3rd Edition, Addison-Wesley, 1998.*
2. *Raghu Ramakrishan and Johannes Gehrke, "Database Management Systems", 3rd Edition, McGraw Hill, 2003.*
3. *Robert L. Kruse, Bruce P. Leung, Clovis L.Tondo, "Data Structures and Program Design in C", 2nd Edition, Prenctice Hall India, 2001.*

MSC102T: THEORY OF COMPUTATION

Total Teaching Hours: 52 No. of Hours / Week: 04

UNIT – I [12 Hours]

Review of Mathematical Terms and Theory: Basic Mathematical Notations and Set Theory, Logic Functions and Relations, Language Definitions, Mathematical Inductions and Recursive Definitions. Finite Automata: Deterministic and Non Deterministic Finite Automata, U-Transitions, Conversion from NFA to DGA, Kleene's Theorem, Regular and Non Regular Languages.

UNIT – II [10 Hours]

Context Free Grammar: Introduction to CFG, CFG and Known Languages, Unions, Concatenations and *'s Notations and CFL, Derivatives of Trees and Ambiguity and Unambiguous CFG and Algebraic Expressions, Normal Forms and Simplified Forms.
Pushdown Automata, CFL and NFL: Introduction to PDA, Definition, DPDA, PDA Corresponding to CFG, CFG Corresponding to PDA, Introduction to CFL, Intersections and Complements of CFL, Decisions Problems and CFL.

UNIT – III [10 Hours]

Turing Machines, Recursive Language: Model of Computation and Church Turning Thesis, Definitions of Turing Machine, TM and Language Acceptors, Variations of TM, Non Deterministic TM, Universal TM, Enumerable and Language, Recursive and Non Recursive Enumerable.

UNIT – IV [10 Hours]

Computation Functions, Measuring, Classifications And Complexity: Primitive Recursive Functions, Halting Problem, Recursive Predicates and Some Bounded Operations, Unbounded Minimizations and μ-Recursive Functions, Godel Numbering, Computable Functions and μ-Recursive, Numerical Functions.

UNIT – V [10 Hours]

Tractable and Intractable Problems: Growth Rate and Functions, Time and Speed Complexity, Complexity Classes, Tractable and Possibly Intractable Problems, P and Np Completeness, Reduction of Time, Cook's Theorem, Np-Complete Problems.

Reference

1. *John E. Hopcroft, Rajeev Motwani, Jeffrey D.Ullman, "Introduction to Automata Theory, Languages and Computation", 3rd Edition, Pearson Education, 2011.*
2. *John C Martin, "Introduction to Languages and Automata Theory", 3rd Edition, Tata McGraw-Hill, 2007.*
3. *Daniel I.A. Cohen, "Introduction to Computer Theory", 2nd Edition, John Wiley and Sons, 2009.*
4. *Thomas A. Sudkamp, "An Introduction to the Theory of Computer Science, Languages and Machines", 3rd Edition, Pearson Education, 2006.*

MSC103T: ADVANCED DATABASES

Total Teaching Hours: 52 No. of Hours / Week: 04

UNIT – I [12 Hours]

Database Concepts: Characteristics of Database Approach – Data Models – Schemas- Three Schema Architecture and Data Independence; Database Design: ER Modelling – ER diagrams; Normalization; Relational Model and Query Processing

UNIT – II [10 Hours]

TRANSACTION PROCESSING AND CONCURRENCY CONTROL: Definition of Transaction and ACID properties; Concurrency Control Techniques: Lock based Concurrency control -Optimistic Concurrency Control – Timestamp based Concurrency Control, Deadlocks; Database Security: Security Issues – Control Measures- Discretionary, mandatory and role based access control; Database Recovery Techniques: Recovery Concepts- Deferred Update and Immediate Update techniques – Shadow Paging – ARIES – Database backup and recovery

UNIT – III [10 Hours]

OBJECT ORIENTED, PARALLEL AND DISTRIBUTED DATABASES: Concept of Object Database: Object Definition Language ODL- Object Query Language; Object Database conceptual Design: Difference between ODB and RDB. Database System Architectures: Centralized and Client-Server Architectures - Parallel Systems- Distributed Systems

UNIT – IV [10 Hours]

Parallel Databases: I/O Parallelism – Inter and Intra Query Parallelism – Inter and Intra operation Parallelism – Design of Parallel Systems; Distributed Database Concepts - Distributed Data Storage – Distributed Transactions – Commit Protocols – Concurrency Control – Distributed Query Processing.

UNIT – V [10 Hours]

EMERGING DATABASE TECHNOLOGIES: Multimedia Databases ; Spatial Databases ; XML and Web Databases ; Mobile Databases: Location and Handoff Management - Effect of Mobility on Data Management -Location Dependent Data Distribution - Mobile Transaction Models -Concurrency Control -Transaction Commit Protocols; Data Warehousing Data Mining; Text Mining.

Reference

1. *R. Elmasri, S.B. Navathe, "Fundamentals of Database Systems", Fifth Edition, Pearson Education/Addison Wesley, 2008.*
2. *Henry F Korth, Abraham Silberschatz, S. Sudharshan, "Database System Concepts", Sixth Edition, McGraw Hill, 2011.*
3. *C.J.Date, A.Kannan, S.Swamynathan, "An Introduction to Database Systems", Eighth Edition, Pearson Education, 2006.*

MSC104T: ADVANCED COMPUTER ARCHITECTURE

Total Teaching Hours: 52 No. of Hours / Week: 04

UNIT – I [12 Hours]

Fundamentals of Computer design: Instruction set principles and examples- classifying instruction set - memory addressing- type and size of operands - addressing modes for signal processing-operations in the instruction set- instructions for control flow- encoding an instruction set. Overview of Parallel Processing and Pipelining Processing Necessity of high performance, Constraints of conventional architecture, Parallelism in uniprocessor system, Evolution of parallel processors, Architectural Classification, Applications of parallel processing

UNIT – II [10 Hours]

Parallel Computer methods: Multiprocessor and multi computers – Shared-Memory multiprocessors, Distributed-Memory Multiprocessors. Multi-vector and SIMD computers. PRAM and VLSI models - Architectural development tracks - Multiple Processor Tracks, Multi-vector and SIMD Tracks, Multi-threaded and Dataflow Tracks. Program and Network properties: Conditions of parallelism - Program partitioning and scheduling - Program flow mechanism - System interconnect architecture. Principles of Scalable Performance: Performance metrics and measures - Speedup performance laws - Scalability analysis and approaches

UNIT – III [10 Hours]

Processors and Memory Hierarchy: Advanced processor technology - Super scalar and vector processors - Memory hierarchy technology - Virtual memory technology. Bus, Cache and Shared Memory: Bus System-Cache memory organizations-Shared memory organization-Sequential and weak consistency models.

UNIT – IV [10 Hours]

Instruction level Parallelism & Data Parallel Architectures: Instruction level parallelism (ILP)- over coming data hazards- reducing branch costs –high performance instruction delivery- hardware based speculation- limitation of ILP - ILP software approach- compiler techniques- static branch protection- VLIW approach- H.W support for more ILP at compile time- H.W verses S.W solutions - SIMD Architectures – Associative and Neural Architectures – Data-Parallel Pipelined and Systolic Architectures – Vector Architectures

UNIT – V [10 Hours]

Multiprocessors and Thread level parallelism: Multi-threaded Architectures, Distributed Memory MIMD Architectures, Shared Memory Architectures. Architecture of Multithreaded processors, Latency hiding techniques, Principles of multithreading, Issues and solutions. Synchronization and Multiprocessing modes – Shared-Variable program structures, Message Passing program development, Mapping programs onto Multicomputers.

Reference

1. *Dezso Sima, Terence Fountain, Peter Kacsuk, "Advanced Computer Architectures – A Design Space approach", Pearson Education, 2009*
2. *Kai Hwang, "Advanced Computer Architecture – Parallelism, Scalability, Programmability", Tata McGraw-Hill, 2008.*
3. *John L. Hennessey and David A. Patterson, "Computer architecture – A quantitative approach", Morgan Kaufmann / Elsevier Publishers, 5th Edition*

MSC105P: FILE STRUCTURES LAB

1. Write a C++ Program to read series of names, one per line, from standard input and write these names spelled in reverse order to the standard output using I/O redirection and pipes. Repeat the exercise using an input file specified by the user instead of the standard input and using an output file specified by the user instead of the standard output.
2. Write a C++ program to read and write student object with fixed length records and the fields delimited by "|". Implement pack(), unpack(), modify(), and search() methods.
3. Write a C++ program to read and write student objects with Variable-Length records using any suitable record structure. Implement pack(), unpack(), modify(), and search() methods.
4. Write a C++ program to read and write student objects with Variable-Length records using any suitable record structure and to read from this file a student record using RRN.
5. Write a C++ program to implement simple index on primary key for a file of student objects. Implement add(), search(), delete() using the index.
6. Write a C++ program to implement index on secondary key, the name, for a file of student objects. Implement add(), search(), delete() using the secondary index.
7. Write a C++ program to read two lists of names and then match the names in the two lists using sequential Match based on a single loop. Output the names common to both the lists.
8. Write a C++ program to read k Lists of names and merge them using k-way merge algorithm with k = 8.
9. Write a C++ program to implement B-Tree for a given set of integers and its operations insert() and search(). Display the tree.
10. Write a C++ program to implement B+ Tree for a given set of integers and its operations insert() and search(). Display the tree.
11. Write a C++ program to store and retrieve student data from file using hashing. Use any collision resolution techniques.
12. Write a C++ program to reclaim the free space resulting from the deletion of records using liked list.

MSC106P: ADVANCED DATA BASE MANAGEMENT SYSTEMS LAB

1. Database Customization
2. Creating Databases/Table spaces
3. Create Objects
4. Moving Data
5. Recovery
6. Locking
7. Preparing Applications for Execution using a front end tool
8. Application Performance Tool

The students are supposed to practice and develop a mini application for above mentioned lab. The students can do the activity in a group (team) consisting of not more than 2 students.

The entire application to be submitted by each team should be done with all the above activities. The examiner may ask to perform any of the above acts.

MSC 107T (Soft Core): QUANTITATIVE, TEACHING AND REASERCH APTITUDE

Total Teaching Hours: 48 No. of Hours / Week: 03

UNIT – I [8 Hours]

Numbers Property – Simplification – Divisibility – HCF and LCM – Decimal Fractions – Square roots and Cube Roots – Logarithms – Antilogarithms - Surds and indices - Permutation and Combination – Probability – Odd man out series - Number series - letter series – codes – Relationships – classification.

UNIT – II [10 Hours]

Time and work – Problems on Ages – Calendar – Clock – Pipes and Cistern – Time and Distance – Problems of Train – Boats and Streams. Area – Volume and surface Areas – Heights and Distances – Data Interpretation: Tabulation – Bar Graphs – Pie Charts – Line Graphs. Data Interpretation - Sources, acquisition and interpretation of data; Quantitative and qualitative data; Graphical representation and mapping of data.

UNIT – III [10 Hours]

Simple Interest – Compound Interest – Stocks and Shares – True Discount – Banker's discount. Averages – Percentage – Profit and Loss - Ratio and Proposition – Partnership – Allegation and mixture – Chain rule. Understanding the structure of arguments; Evaluating and distinguishing deductive and inductive reasoning; Verbal analogies: Word analogy Applied analogy; Verbal classification; Reasoning Logical Diagrams: Simple diagrammatic relationship, multidiagrammatic relationship; Venn diagram; Analytical Reasoning.

UNIT – IV [10 Hours]

Teaching: Nature, objectives, characteristics and basic requirements; Learner's characteristics; Factors affecting teaching; Methods of teaching; Teaching aids; Evaluation systems. Research Aptitude: Meaning, characteristics and types; Steps of research; Methods of research; Research Ethics; Paper, article, workshop, seminar, conference and symposium; Thesis writing: its characteristics and format. Reading Comprehension: A passage to be set with questions to be answered. Communication: Nature, characteristics, types, barriers and effective classroom communication.

UNIT – V [10 Hours]

Higher Education System: Governance, Polity and Administration; Structure of the institutions for higher learning and research in India; formal and distance education; professional/technical and general education; value education: governance, polity and administration; concept, institutions

Reference

1. *R.S. Aggarwal, Quantitative Aptitude, S. Chand & Company, New Delhi, 2012*
2. *Govind Prasad Singh and Rakesh Kumar, Text Book of Quickest Mathematics (for all Competitive Examinations), Kiran Prakashan, 2012.*
3. *R.S. Aggarwal, Objective Arithmetic, S. Chand & Company, New Delhi, 2005.*
4. *Dr. Lal, Jain, Dr. K. C. Vashistha, "U.G.C.- NET/JRF/SET Teaching & Research Aptitude", Upkar Prakashan, 2010.*
5. *"UGC NET/SLET: Teaching & Research Aptitude", Bright Publications, 2010.*

SECOND SEMESTER M.Sc

MSC201T: OBJECT ORIENTED ANALYSIS AND DESIGN USING UML

Total Teaching Hours: 52 No. of Hours / Week: 04

UNIT – I [12 Hours]

Introduction: An overview - Object basics - Object state and properties, Behavior, Methods, Messages. Object Oriented system development life cycle, Benefits of OO Methodology. Overview of Prominent OO Methodologies: The Rumbaugh OMT, The Booch methodology, Jacobson's OOSE methodologies, Unified Process, Introduction to UML, Important views & diagram to be modelled for system by UML. Factional View (models): Use case diagram - Requirement Capture with Use case - Building blocks of Use Case diagram - actors, use case guidelines for use case models - Relationships between use cases - extend, include, generalize. Activity diagram - Elements of Activity Diagram - Action state, Activity state, Object, node, Control and Object flow, Transition (Fork, Merge, Join) - Guidelines for Creating Activity Diagrams - Activity Diagram - Action Decomposition (Rake) - Partition - Swim Lane.

UNIT – II [10 Hours]

Static structural view (Models): Classes, values and attributes, operations and methods, responsibilities for classes, abstract classes, access specification (visibility of attributes and operations). Relationships among classes: Associations, Dependencies. Inheritance - Generalizations, Aggregation. Adornments on Association: association names, association classes, qualified association, n-ary associations, ternary and reflexive association. Dependency relationships among classes, notations. Notes in class diagram, Extension mechanisms, Metadata, Refinements, Derived, data, constraint, stereotypes, Package & interface notation. Object diagram notations and modeling, relations among objects (links).

UNIT – III [10 Hours]

Class Modeling and Design Approaches: Three approaches for identifying classes - using Noun phrases, Abstraction, Use Case Diagram - Comparison of approaches - Using combination of approaches - Flexibility guidelines for class diagram: Cohesion, Coupling, Forms of coupling (identity, representational, subclass, inheritance), class Generalization, class specialization versus aggregation. Behavioral (Dynamic structural view): State diagram - State Diagram Notations, events (signal events, change events, Time events) - State Diagram states (composite states, parallel states, History states), transition and condition, state diagram behaviour (activity effect, do-activity, entry and exit activity), completion transition, sending signals.

UNIT – IV [10 Hours]

Interaction diagrams: Sequence diagram - Sequence diagram notations and examples, iterations, conditional messaging, branching, object creation and destruction, time constraints, origin of links, Activations in sequence diagram - Collaboration diagram - Collaboration diagram notations and examples, iterations, conditional messaging, branching, object creation and destruction, time constraints, origin of links, activations in sequence diagram. Approaches for developing dynamic systems: Top - down approach for dynamic systems - Bottom - up approach for dynamic systems - Flexibility Guidelines for

Behavioral Design - guidelines for allocating and designing behaviors that lead to more flexible design.

UNIT – V [10 Hours]

Architectural view: Logical architecture: dependency, class visibility, sub systems - Hardware architecture: deployment diagram notations, nodes, object migration between node - Process architecture: what are process and threads and their notations in UML, object synchronization, invocation schemes for threads (UML notations for different types of invocations). Implementation architecture: component diagram notations and examples. Reuse: Libraries, Frame works components and Patterns: Reuse of classes, Reuse of components, Reuse of frameworks, black box framework, white box frame, Reuse of patterns: Architectural pattern and Design pattern.

Reference

1. *Charles Richter, "Designing Flexible Object Oriented systems with UML" , Macmillan Technical, 1999*
2. *Jackson, Burd Thomson, "Object Oriented Analysis & Design", Thomson Course Technology, 2004*
3. *James Rumbaugh. Micheal Blaha, Object oriented Modeling and Design with UML. Pearson, second edition, 2005.*
4. *Grady Booch, James Rumbaugh, Ivar Jacobson, "The Unified Modeling Language User Guide", Pearson Education, 1999.*
5. *James Rumbaugh, "Object Oriented Modeling and Design", Prentice Hall, 1991.*
6. *Joseph Schmuilers, "Teach Yourself UML in 24 Hours", Sams publication, 2004.*
7. *Mike O'Docherty, "Object-Oriented Analysis and Design: using UML", Wiley Publication, 2005.*

MSC202T: ADVANCED JAVA PROGRAMMING

Total Teaching Hours: 52 No. of Hours / Week: 04

UNIT – I [12 Hours]

Introduction: Data Types, Operators, Classes, Inheritance, Packages and Interfaces. Exception Handling, Concurrency and Multithreaded programming, Enumerations, Autoboxing, Annotations, I/O, Generics, String handling

UNIT – II [10 Hours]

JVM: Java Class file, Class Loader, Linking model, Garbage collection, Type conversion, Floating Point Arithmetic, Method Invocation and Return, Thread synchronization. Java I/O: Closeable, Flushable Interfaces, The Stream classes, Bytes Streams, Character Streams, Console Class, Serialization. Java Networking - Networking Classes and Interfaces, TCP/IP Sockets, Datagrams

UNIT – III [10 Hours]

Event Handling: Event Classes, Event Listener Interfaces, Adaptor Classes, Inner Classes. Comparable and Comparator. Java Sandbox security model, Applets. Server side programming - Java Servlets, JSP, Java XML library - JAXP, XML Parsing - DOM, SAX, Stax. Java Web Services - RESTful Web Services, SOAP Web Services

UNIT – IV [10 Hours]

Java Design patterns: Singleton, Observer, Adaptor, Proxy, Decorator, Factory, AbstractFactory, Fascade, Command, Template Method patterns, MVC .

UNIT – V [10 Hours]

Spring and Hibernate framework, Spring Flow, Hibernate Flow.

Reference

1. *Herbert Schildt, "Java The Complete Reference", 7th addition, 2006.*
2. *Ken Arnold, James Gosling, David Holmes, "The Java TM Programming Language", Addison-Wesley, 2006*
3. *Bill Venners, "Inside the Java 2 Virtual Machine", McGraw-Hill, 2nd edition, 2000.*
4. *Santhosh, "Spring and Hibernate", Tata McGraw-Hill, 2009.*

MSC203T: ARTIFICIAL INTELLIGENCE

Total Teaching Hours: 52 No. of Hours / Week: 04

UNIT-I [12 Hours]

Introduction to Artificial Intelligence: Definition. AI Applications, AI representation. Properties of internal Representation, Heuristic search techniques. Best first search, mean and end analysis, A* and AO* Algorithm. Minimize search procedure, Alpha beta cutoffs, waiting for Quiscence, Secondary search.

UNIT-II [10 Hours]

Knowledge representation using predicate logic: predicate calculus, Predicate and arguments, ISA hierarchy, frame notation, resolution, Natural deduction. Knowledge representation using non monotonic logic: TMS (Truth maintenance system), statistical and probabilistic reasoning, fuzzy logic, structure knowledge representation, semantic net, Frames, Script, Conceptual dependency.

UNIT-III [10 Hours]

Planning: block world, strips, Implementation using goal stack, Non linear planning with goal stacks, Hierarchical planning, list commitment strategy. Perception: Action, Robot Architecture, Vision, Texture and images, representing and recognizing scenes, waltz algorithm, Constraint determination, Trihedral and non trihedral figures labeling.

UNIT-IV [10 Hours]

Learning: Learning as induction matching algorithms. Failure driver learning, learning in general problem solving concept learning. Neural Networks: Introduction to neural networks and perception-qualitative Analysis only, neural net architecture and applications.

UNIT-V [10 Hours]

Natural language processing and understanding and pragmatic, syntactic, semantic, analysis, RTN, ATN, understanding sentences. Expert system: Utilization and functionality, architecture of expert system, knowledge representation, two case studies on expert systems.

Reference

1. *E. Charnaik and D. McDermott," Introduction to artificial Intelligence", Pearson Education, 1992.*
2. *Dan W. Patterson, "Introduction to Artificial Intelligence and Expert Systems", PHI, 2003.*
3. *E. Rich and K. Knight," Artificial Intelligence", Tata McGraw Hill, 2003.*
4. *Nils J. Nilson, "Principles of Artificial Intelligence", Narosa Publishing Co. 2002.*

MSC204T: OPTIMIZATION TECHNIQUES

Total Teaching Hours: 52 No. of Hours / Week: 04

UNIT - I [12 Hours]

Introduction to Operations Research: Basics definition, scope, objectives, phases, models and limitations of Operations Research. Linear Programming Problem – Formulation of LPP, Graphical solution of LPP. Simplex Method, Artificial variables, big-M method, two phase method, degeneracy and unbound solutions.

UNIT - II [10 Hours]

Transportation Problem: Formulation, Solution, Unbalanced Transportation Problem. Finding Basic Feasible Solutions – Northwest corner rule, least cost method and Vogel's approximation method. Optimality test: the stepping stone method and MODI method. Assignment Model: Formulation. Hungarian method for optimal solution. Solving unbalanced problem. Traveling salesman problem and assignment problem.

UNIT - III [10 Hours]

Network Models: Definition, Minimum Spanning Tree algorithm, Shortest Route problem, Maximum flow problem. CPM & PERT: Network representation, Critical Path Computations, Linear Programming formulation of CPM, PERT Networks.

UNIT - IV [10 Hours]

Dynamic programming: Characteristics of dynamic programming. Dynamic Programming approach for Priority Management employment smoothening. Games Theory. Competitive games, rectangular game, saddle point, minimax (maximin) method of optimal strategies, value of the game. Solution of games with saddle points, dominance principle. Rectangular games without saddle point – mixed strategy for 2 X 2 games.

UNIT - V [10 Hours]

Queuing System: Elements of Queuing model, Pure birth and death models, Generalized Poission Queuing model, specialized poission. Queues: Steady-state Measure of performance, single sever models, Multiple server models, Matching serving model.

Reference

1. *J K Sharma., "Operations Research Theory & Applications , 3e", Macmillan India Ltd, 2007.*
2. *P. Sankara Iyer, "Operations Research", Tata McGraw-Hill, 2008.*
3. *P. K. Gupta and D. S. Hira, "Operations Research", S. Chand & co., 2007.*
4. *N.V.S. Raju, "Operations Research", HI-TECH, 2002.*

MSC205P: OBJECT ORIENTED DESIGN USING UML LAB

1. The student should take up the case study of Unified Library application which is mentioned in the theory, and Model it in different views i.e. Use case view, logical view, component view, Deployment view, Database design, forward and Reverse Engineering, and Generation of documentation of the project.

2. Student has to take up another case study of his/her own interest and do the same what ever mentioned in first problem. Some of the ideas regarding case studies are given in reference books, which were mentioned in theory syllabus, can be referred for some idea.

MSC206P: ADVANCED JAVA PROGRAMMING LAB

1. Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -,*, % operations. Add a text field to display the result.
2. Write a Java program that creates three threads. First thread displays “Good Morning” every one second, the second thread displays “Hello” every two seconds and the third thread displays “Welcome” every three seconds.
3. Write a java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green. When a radio button is selected, the light is turned on, and only one light can be on at a time No light is on when the program starts
4. Write a Java Program to execute select query using JDBC
5. Write a Java Program to Create Thread using Interface and class.
6. Write a Java Program to Implement Producer and Consumer problem using Threads.
7. Write a Java Program to Implement DOM parser.
8. Write a Java Program to Implement SAX parser.
9. Write a Java Program to Implement Singleton design pattern using java.
10. Write a Java Program to Implement Factory and Abstract Factory design pattern using java.
11. Write a Java Program to Implement Observer Design pattern method using java.
12. Write a Java Program to Implement Adapter design design pattern using java
13. Write a Java Program to Implement proxy design pattern using java 14. Write a Java Program to Implement Helloworld program using servlets.
15. Write a JSP Program using Expression, Scriplet and Directive.

MCA207T: SOFT SKILLS AND PERSONALITY DEVELOPMENT

Total Teaching Hours: 48 No. of Hours / Week: 03

UNIT – I [10 Hours]

Introduction to Soft Skills and Hard Skills, Break the ice berg –FEAR, Self Development - Etiquette and Manners. The Self Concept: Attitude, The process of attitude formation, positive attitude, How to build a success attitude, You are the chief architecture of yourself. Self Management Techniques. Believe in yourself: Self Image and Self Esteem, Building Self Confidence, Environment we mix with, How to build self-image.

UNIT - II [10 Hours]

Meaning and definition of personality, Personal Planning and Success Attitude: Prioritizing, Creating the master plan, Active positive visualization and Spot analysis. Self-Motivation and Communication: Levels of motivation, power of irresistible enthusiasm, etiquettes and manners in a group, public speaking, Importance of listening and responding.

UNIT - III [10 Hours]

Motivation Skills & Personality Development, Goal Setting, Career Planning, Resume Building, Psychometric Test, Priority Management & Time Management, Positive Attitude and Self Confidence. Verbal Communication includes Planning, Preparation Delivery, Feedback and assessment of activities like: Public speaking, Group Discussion, Oral Presentation skills, Perfect Interview, Listening and observation skills, body language and use of Presentation aids.

UNIT - IV [8 Hours]

Written communication that includes project proposals, brochures, newsletters, articles. Etiquettes that include: etiquettes in social as well as office settings, email etiquettes, telephone etiquettes. Improving Personal Memory, study skills that include rapid reading, notes taking and creativity.

UNIT - V [10 Hours]

Problem Solving and Decision Making Skills, Perceptive, Conceptual, Creative, Analytical and Decisive. Leadership as a process: co-ordination while working in a team, Leadership styles, Leader and Team player, Management of conflict, Profiles of great and successful personalities, Role of career planning in personality development, negotiation, Motivating.

Reference

1. *Wallace: "Personality Development", 1st Edition, 2008 Cengage Learning India.*
2. *Richard Denny, "Succeed for your self", Kogan page India, 3rd Edition. www.vivagroupindia.com.*
3. *John Hoover & Angelo Valenti, "Unleashing Leadership", Jaico publishing House –WWW.JAICOBOOKS.COM*
4. *Kundu, C.L – "Personality development", Sterling Bangalore.*
5. *Sandra D. Collins, "Listening and Responding", Cengage Learning India, 2nd Edition, 2008.*
6. *David E. Rye, "1,001 ways to inspire your organization, your team and yourself", Jaico publishing house, Career Press, 1998.*

MSC301T: Advanced WEB Programming

Total Teaching Hours: 52 No. of Hours / Week: 04

UNIT – I [12 Hours]

Perl, CGI Programming: Origins and uses of Perl; Scalars and their operations; Assignment statements and simple input and output; Control statements; Fundamentals of arrays; Hashes; References; Functions; Pattern matching; File input and output; Examples. The Common Gateway Interface; CGI linkage; Query string format; CGI.pm module; A survey example; Cookies.

UNIT – II [10 Hours]

Servlets and Java Server Pages: Overview of Servlets; Servlet details; A survey example; Storing information on Clients; Java Server Pages. PHP: Origins and uses of PHP; Overview of PHP; General syntactic characteristics; Primitives, operations and expressions; Output; Control statements; Arrays; Functions; Pattern matching; Form handling; Files; Cookies; Session tracking.

UNIT – III [10 Hours]

Database Access through the Web: Relational Databases; An introduction to SQL; Architectures for Database access; The MySQL Database system; Database access with PERL and MySQL; Database access with PHP and MySQL; Database access with JDBC and MySQL.

UNIT – IV [10 Hours]

Introduction to Ruby, Rails: Origins and uses of Ruby; Scalar types and their operations; Simple input and output; Control statements; Fundamentals of arrays; Hashes; Methods; Classes; Code blocks and iterators; Pattern matching. Overview of Rails; Document requests; Processing forms; Rails applications with Databases; Layouts.

UNIT – V [10 Hours]

Introduction to Ajax: Overview of Ajax; The basics of Ajax; Rails with Ajax.

Reference

1. *Robert W. Sebesta: "Programming the World Wide Web", 4th Edition, Pearson Education, 2012.*
2. *M. Deitel, P.J. Deitel, A. B. Goldberg: "Internet & World Wide Web How to program", 3rd Edition, Pearson Education, 4th edition, PHI, 2011.*
3. *Chris Bates: "Web Programming Building Internet Applications", 3rd Edition, Wiley India, 2011.*
4. *Joyce Farrell, Xue Bai, Michael Ekedahl: "The Web Warrior Guide to Web Programming", 1st edition, Thomson, 2010.*

MSC302T: Advanced Algorithms

Total Teaching Hours: 52 No. of Hours / Week: 04

UNIT – I [12 Hours]

Design Paradigms: Overview: Overview of Divide and Conquer, Greedy and Dynamic Programming strategies. Basic search and traversal techniques for graphs, Backtracking, Branch and Bound. Max Flow Problem.

UNIT – II [10 Hours]

String Matching : Introduction to string-matching problem, Naïve algorithm, Rabin Karp, Knuth Morris Pratt, Boyer- Moore algorithms and complexity analysis. Theory of NP- Hard and NP-Complete Problems: P, NP and NP-Complete complexity classes; A few NP-Completeness proofs; Other complexity classes.

UNIT – III [10 Hours]

Approximation Algorithms Introduction, Combinatorial Optimization, approximation factor, PTAS, FPTAS, Approximation algorithms for vertex cover, set cover, TSP, knapsack, bin packing, subset-sum problem etc. Analysis of the expected time complexity of the algorithms.

UNIT – IV [10 Hours]

Parallel Algorithms: Introduction, Models, speedup and efficiency, Some basic techniques, Examples from graph theory, sorting, Parallel sorting networks. Parallel algorithms and their parallel time and processors complexity.

UNIT – V [10 Hours]

Probabilistic Algorithms & Randomized Algorithms: Numerical probabilistic algorithms, Las Vegas and Monte Carlo algorithms, Game-theoretic techniques, Applications on graph problems

Reference

1. *T. H Cormen, C E Leiserson, R L Rivest and C Stein: "Introduction to Algorithms", 3rd Edition, Prentice-Hall of India, 2011.*
2. *Mark Allen Weiss, Data Structures and Algorithm analysis in C++, 3rd edition, PEA, 2011.*
3. *Ellis Horowitz, Sartaj Sahni, S.Rajasekharan: "Fundamentals of Computer Algorithms", 1st edition, University Press, 2012.*

MSC303T: Cryptography and Network Security

Total Teaching Hours: 52 No. of Hours / Week: 04

UNIT – I [12 Hours]

Introduction to Information Security: Introduction; security, Critical characteristics of information; NSTISSC security model; Approaches to information security implementation; The Security System Development Life Cycle; Information Security Terminology. Planning for Security: Introduction; Information Security Policy, Standards, and Practices; The Information Security Blue Print.

UNIT – II [10 Hours]

Security Technology: Firewalls and VPNs: Introduction, Physical design, Firewalls, Protecting Remote Connections. Intrusion Detection, Access control and Other Security Tools: Introduction; Intrusion Detection Systems (IDS); Honey Pots, Honey Nets, and Padded cell systems; Scanning and Analysis Tools; Access Control Devices. Information Security maintenance: Introduction; Security Management Models; The Maintenance Model.

UNIT – III [10 Hours]

Introduction to Network Security: Attacks, Services, and Mechanisms; Security Attacks; Security Services; A model for Internetwork Security; Internet Standards and RFCs. Cryptography: Conventional Encryption Principles and Algorithms; Cipher Block Modes of Operation; Location of encryption devices; Key distribution; Approaches to message authentication; Secure Hash functions and HMAC; Public Key Cryptography Principles and Algorithms; Digital Signatures; Key management.

UNIT – IV [10 Hours]

Authentication Applications: Kerberos, X.509 Directory Authentication Service. Electronic Mail Security: Pretty Good Privacy (PGP), S/MIME. IP Security: IP Security Overview, IP Security Architecture, Authentication Header, Encapsulating Security Payload, Combining Security Associations, Key Management.

UNIT – V [10 Hours]

Web Security: Web security requirements, Secure Socket layer (SSL) and Transport layer Security (TLS), Secure Electronic Transaction (SET). Network Management Security: Basic concepts of SNMP, SNMPv1 community facility, SNMPv3.

Reference

1. *Michael E. Whitman and Herbert J. Mattord: "Principles of Information Security", 4th Edition, Thomson, 2012.*
2. *William Stallings: "Network Security Essentials Applications and Standards", 4th edition, Person Education, 2012.*
3. *Behrouz a Forouzan, Debdeep Mukhopadhyay: "Cryptography and Network Security", 2nd edition, Tata McGraw-Hill, 2011.*
4. *Deven N. Shah: Mark Stapms Information Security Principles & Practice, 1st edition, Wiley India, 2010.*

MSC305P: WEB Programming Lab

1. Develop and demonstrate a XHTML file that includes Javascript script to generate first n Fibonacci numbers.
2. Develop and demonstrate the usage of inline and external style sheet using CSS
3. Develop and demonstrate, using Javascript script, a XHTML document that collects the USN (the valid format is: A digit from 1 to 4 followed by two upper-case characters followed by two digits followed by two upper-case characters followed by three digits; no embedded spaces allowed) of the user. Event handler must be included for the form element that collects this information to validate the input. Messages in the alert windows must be produced when errors are detected.
4. Develop and demonstrate, using Javascript script, a XHTML document that contains three short paragraphs of text, stacked on top of each other, with only enough of each showing so that the mouse cursor can be placed over some part of them. When the cursor is placed over the exposed part of any paragraph, it should rise to the top to become completely visible.
5. Design an XML document to store information about a student in an engineering college affiliated to VTU. The information must include USN, Name, Name of the College, Brach, Year of Joining, and e-mail id. Make up sample data for 3 students. Create a CSS style sheet and use it to display the document.
6. Write a Perl program to display a digital clock which displays the current time of the server.
7. Write a Perl program to insert name and age information entered by the user into a table created using MySQL and to display the current contents of this table.
8. Write a PHP program to store current date-time in a COOKIE and display the 'Last visited on' date-time on the web page upon reopening of the same page.
9. Write a PHP program to read student data from an XML file and store into the MYSQL database. Retrieve and display.
10. Write a Perl program to keep track of the number of visitors visiting the web page and to display this count of visitors, with proper headings.
11. Write a CGI-Perl program to use a cookie to remember the day of the last login from a user and display it when run.
12. Write a Perl program to display various Server informations like Server Name, Server Software, Server protocol, CGI Revision etc.
13. Create a XHTML form with Name, Address Line 1, Address Line 2, and E-mail text fields. On submitting, store the values in MySQL table. Retrieve and display the data based on Name.
14. Write a Perl program to accept the User Name and display a greeting message randomly chosen from a list of 4 greeting messages.

MSC306P: Advanced Algorithms Lab

1. Write a Program to Implement recursive binary search and linear search and determine the time required to search an element. Repeat the experiment for different values of n, the number of elements in the list to be searched and plot a graph of the time taken versus n.
2. Write a Program to Sort a given set of elements using the Quicksort method and determine the time required to sort the elements. Repeat the experiment for different values of n, the number of elements in the list to be sorted and plot a graph of the time taken versus n. The elements can be read from a file or can be generated using the random number generator.
3. Write a Program to Sort a given set of elements using heap sort method and determine the time required to sort the elements. Repeat the experiment for different values of n, the number of elements in the list to be sorted and plot a graph of the time taken versus n.
4. Write a Program to Implement 0/1 Knapsack problem using Dynamic Programming.
5. Write a Program to From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra's algorithm.
6. Write a Program to obtain the Topological ordering of vertices in a given digraph. Compute the transitive closure of a given directed graph using Warshall's algorithm.
7. Write a Program to Find a subset of a given set S = {s1,s2,.....,sn} of n positive integers whose sum is equal to a given positive integer d. For example, if S= {1, 2, 5, 6, 8} and d = 9 there are two solutions {1,2,6} and {1,8}.A suitable message is to be displayed if the given problem instance doesn't have a solution.
8. Write a Program to solve the string matching problem using Boyer-Moore approach.
9. Write a Program to solve the string matching problem using naïve approach and the KMP algorithm and compare their Performances
10. Write a Program to Implement a parallelized Merge Sort algorithm to sort a given set of elements and determine the time required to sort the elements. Repeat the experiment for different values of n, the number of elements in the list to be sorted and plot a graph of the time taken versus n. The elements can be read from a file or can be generated using the random number generator (use OpenMP)
11. Write a Program to Implement All-Pairs Shortest Paths Problem using Floyd's algorithm. Parallelize this algorithm, implement it using OpenMP and determine the speed-up achieved.
12. Write a to implement a Monte Carlo algorithm to test the primality of a given integer and determine its performance

MSC401T: Research Methodology

Total Teaching Hours: 52 No. of Hours / Week: 04

UNIT – I [12 Hours]

Introduction: Definition and objectives of Research – Types of research, Various Steps in Research process, Mathematical tools for analysis, Developing a research question-Choice of a problem Literature review, Surveying, synthesizing, critical analysis, reading materials, reviewing, rethinking, critical evaluation, interpretation, Research Purposes, Ethics in research – APA Ethics code.

UNIT – II [10 Hours]

Quantitative Methods for problem solving: Statistical Modeling and Analysis, Time Series Analysis Probability Distributions, Fundamentals of Statistical Analysis and Inference, Multivariate methods, Concepts of Correlation and Regression, Fundamentals of Time Series Analysis and Spectral Analysis, Error Analysis, Applications of Spectral Analysis.

UNIT – III [10 Hours]

Tabular and graphical description of data: Tables and graphs of frequency data of one variable, Tables and graphs that show the relationship between two variables , Relation between frequency distributions and other graphs, preparing data for analysis

UNIT - IV [10 Hours]

Soft Computing: Computer and its role in research, Use of statistical software SPSS, GRETL etc in research. Introduction to evolutionary algorithms - Fundamentals of Genetic algorithms, Simulated Annealing, Neural Network based optimization, Optimization of fuzzy systems.

UNIT - V [10 Hours]

Structure and Components of Research Report, Types of Report, Layout of Research Report, Mechanism of writing a research report, referencing in academic writing.

Reference

1. *C.R. Kothari, Research Methodology Methods and Techniques, 2/e, Vishwa Prakashan, 2006*
2. *Donald H.McBurney, Research Methods, 5th Edition, Thomson Learning, ISBN:81-315-0047- 0,2006*
3. *Donald R. Cooper, Pamela S. Schindler, Business Research Methods, 8/e, Tata McGraw-Hill Co. Ltd., 2006.*
4. *Fuzzy Logic with Engg Applications, Timothy J.Ross, Wiley Publications, 2nd Edition, 2004.*
5. *Simulated Annealing: Theory and Applications (Mathematics and Its Applications, by P.J. van Laarhoven & E.H. Aarts[e], 1987.*
6. *Genetic Algorithms in Search, Optimization, and Machine Learning by David E. Goldberg, 1989.*

MSC404T: Scilab Lab

1. Program to draw 2-D and 3-D graph for some standard functions. e.g. x^2 , sin (x), exp(x), $x^3 + y^3$ etc.
2. Program to multiply the two matrices.
3. Program to find determinant and inverse of the matrix.
4. Program to find Eigenvalues and Eigenvectors.
5. Program to solve algebraic and transcendental equation by Newton Raphson method.
6. Program for Newton's forward interpolation.
7. Program for solving linear system of equations using Gauss Seidel methods.
8. Program for numerical integration using Simpson's 1/3rd rule.
9. Program for random number generation using various techniques.
10. Program for fitting of Binomial Distribution.
11. Program for fitting of Poisson Distribution.
12. Program to compute measures of skewness and kurtosis.
13. Program to compute Spearman's rank correlation coefficient.
14. Program to fit the curve by principle of least squares.
15. Program to Test based on students 't-test'
16. Program to Test based on Chi-square- Distribution.
17. Program to Solve LPP by Simplex method.

Electives
4E1: Software Testing

Total Teaching Hours: 52 No. of Hours / Week: 04

UNIT – I [12 Hours]

Basics of Software Testing and Examples: Basic definitions, Test cases, Insights from a Venn diagram, Identifying test cases, Error and fault taxonomies, Levels of testing. Examples: Generalized pseudocode, The triangle problem, The NextDate function, The commission problem, The SATM (Simple Automatic Teller Machine) problem. Decision Table-Based Testing: Decision tables, Test cases for the triangle problem, Test cases for the NextDate function, Test cases for the commission problem, Guidelines and observations. Data Flow Testing: Definition-Use testing, Slice-based testing, Guidelines and observations.

UNIT – II [10 Hours]

Levels of Testing: Traditional view of testing levels, Alternative life-cycle models, The SATM system, Separating integration and system testing. Integration Testing: A closer look at the SATM system, Decomposition-based, call graph- based, Path-based integrations, Case study. System Testing: Threads, Basic concepts for requirements specification, Finding threads, Structural strategies and functional strategies for thread testing, SATM test threads, System testing guidelines, ASF (Atomic System Functions) testing example.

UNIT – III [10 Hours]

Interaction Testing: Context of interaction, A taxonomy of interactions, Interaction, composition, and determinism, Client/Server Testing. Issues in Object-Oriented Testing: Units for object-oriented testing, Implications of composition and encapsulation, inheritance, and polymorphism, Levels of object-oriented testing, GUI testing, Dataflow testing for object-oriented software, Examples. Class Testing: Methods as units, Classes as units.

UNIT – IV [10 Hours]

Object-Oriented Integration Testing: UML support for integration testing, MM-paths for object-oriented software, A framework for object-oriented dataflow integration testing. GUI Testing: The currency conversion program, Unit testing, Integration Testing and System testing for the currency conversion program. Object-Oriented System Testing: Currency converter UML description, UML-based system testing, Statechart-based system testing. Exploratory Testing: The context-driven school, Exploring exploratory testing, Exploring a familiar example, Exploratory and context-driven testing observations.

UNIT – V [10 Hours]

Model-Based Testing: Testing based on models, Appropriate models, Use case-based testing, Commercial tool support for model-based testing. Test-Driven Development: Test-then-code cycles, Automated test execution, Java and JUnit example, Remaining questions, Pros, cons, and open questions of TDD, Retrospective on MDD versus TDD. A Closer Look at All Pairs Testing: The all-pairs technique, A closer look at NIST study, Appropriate applications for all pairs testing, Recommendations for all pairs testing. Software Testing Excellence: Craftsmanship, Best practice of software testing,

Top 10 best practices for software testing excellence, Mapping best practices to diverse projects.

Reference

1. *Paul C. Jorgensen: Software Testing, A Craftsman's Approach, 3rd Edition, Auerbach Publications, 2012.*
2. *Aditya P Mathur: Foundations of Software Testing, Pearson, 2008.*
3. *Mauro Pezze, Michal Young: Software Testing and Analysis – Process, Principles and Techniques, 1st edition, John Wiley & Sons, 2011.*
4. *Srinivasan Desikan, Gopalaswamy Ramesh: Software testing Principles and Practices, 1st Edition, Pearson, 2012.*
5. *Brian Marrick: The Craft of Software Testing, 1st edition, Pearson, 2012.*

4E2: e-Governance

Total Teaching Hours: 52 No. of Hours / Week: 04

UNIT – I [12 Hours]

Introduction to e- Governance, Different Stages of e-Governance, Advantages, Problems and Challenges of e-Governance, National Statues, International Status, Securities in e-Governance.

UNIT – II [10 Hours]

National e-Governance Plan, Government of India guidelines for websites, W3C guidelines, web 2.0, web 3.0

UNIT – III [10 Hours]

Different UN Survey on e-Governance, UN Survey on e-Governance – 2014, e-Government Act, 2002, Adhaar Bill, 2016, II Administrative Reforms Committee Report 11, Digital India Programme, IT Act, 2008 Section 1 to 11A, Section 43 and 66

UNIT – IV [10 Hours]

Workflow Management in e-Governance, Digital Divide, Mechanism to handle Digital Divide, Bridge the digital divide, M-Governance, e-Learning, Role of Social Media in e-Governance, Big data Analytics in e-Governance, Semantic web Analytics.

UNIT – V [10 Hours]

Case Study: Election Commission, Indian Railway Reservation, Addhar – UID, Income Tax, SAKALA, Bhoomi, e-Commission, CET admission, Centralized Admission, Student Scholarship Management.

Reference

1. *Mishra D.S (2007). E-Governance as reform strategy for combating corruption in delivery of public services. Indian Journal of Public Administration. LIII (3).*
2. *Bhogle Srinivas (2009). E-Governance. Selected Readings on Information Technology Management: Contemporary Issues ed. George Kelley. Information Science Reference, New York.*
3. *Bhuiyan H Shahjahan (2011). Modernizing Bangladesh public administration through e-governance: Benefits and challenges. 28, 54-65.*
4. *The World Wide Web Consortium (2008). Web Content Accessibility Guidelines (WCAG) 2.0. Downloaded on 10th January, 2012 from http://www.w3.org/*
5. *Government of India (2009). Guidelines for Indian Government websites. Downloaded on 15th January, 2012 from http://darpg.nic.in/*
6. *e-Government Act (2002). https://www.gpo.gov/fdsys/pkg/PLAW-107publ347/pdf/PLAW-107publ347.pdf*
7. *Digital India Programme. http://www.digitalindia.gov.in/*
8. *Information Technolgy Act, 2008. http://www.dot.gov.in/act-rules/information-technology-act-2000*

9. *Second Adminstrative Reforms Committee Report. Report 11: Promoting e-Governance: The SMART way Forwardhttp://arc.gov.in/*
10. *UN Survey on e-Governmen, 2014 (or latest). https://publicadministration.un.org/egovkb/portals/egovkb/documents/un/2014-survey/e-gov_complete_survey-2014.pdf*
11. *The Adhaar Bill, 2016. http://www.prsindia.org/billtrack/the-aadhaar-targeted-delivery-of-financial-and-other-subsidies-benefits-and-services-bill-2016-4202/*
12. *National e-Governance Plan website.*

4E3: Mobile Computing

Total Teaching Hours: 52 No. of Hours / Week: 04

UNIT – I [12 Hours]

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.

UNIT - II [10 Hours]

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.

UNIT– III [10 Hours]

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

UNIT– IV [10 Hours]

Mobile Agents computing, security and fault tolerance, transaction processing in mobile computing environment.

UNIT-V [10 Hours]

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.

Reference

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.*

4E4: Cloud Computing

Total Teaching Hours: 52 No. of Hours / Week: 04

UNIT – I [10 Hours]

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: Cloud Characteristics - Measured Service - Cloud Models - Security in a Public Cloud Public versus Private Clouds.

UNIT - II [6 Hours]

Cloud Infrastructure Self Service. Cloud as a Service: Gamut of Cloud Solutions - Principal Technologies - Cloud Strategy Cloud Design and Implementation using SOA - Conceptual Cloud Model - Cloud Service Defined.

UNIT – III [14 Hours]

Cloud Solutions: Cloud Ecosystem - Cloud Business Process Management - Cloud Service Management - Cloud Stack - Computing on Demand (CoD) – Cloud sourcing. 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.

UNIT – IV [10 Hours]

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.

UNIT – V [12 Hours]

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.

Reference

1. *Cloud Computing – Insight into New Era Infrastructure, Dr. Kumar Saurabh, Wiley India, 2011.*
2. *Cloud Computing, Roger Jennings, Wiley India, 2009.*
3. *Cloud Computing Explained, John Rhoton, Recursive Press,2009.*
4. *Cloud Computing Bible, Barry Sosinsky, Wiley, 2011.*

5. *Cloud Computing: Principles and Paradigms, Rajkumar Buyya, James Broberg, Wiley, 2011.*
6. *Cloud Computing for Dummies, Judith Hurwiz, Wiley Publishing, 2009.*
7. *The Cloud at your service, Rosenberg and Matheos, Manning Publications, 2010.*

4E5: Data Mining

Total Teaching Hours: 52 No. of Hours / Week: 04

UNIT – I [10 Hours]

Data Warehousing: Overview, Definition, Delivery Process, Difference between Database System and Data Warehouse, Multi-Dimensional Data Model, Data Cubes, Stars, Snow Flakes, Fact Constellations, Concept hierarchy, Process Architecture, 3 Tier Architecture, Data Marting.

UNIT -II [10 Hours]

Aggregation, Historical information, Query Facility, OLAP function and Tools. OLAP Servers, ROLAP, MOLAP, HOLAP, Data Mining interface, Security, Backup and Recovery, Tuning Data Warehouse, Testing Data Warehouse.

UNIT – III [10 Hours]

Overview, Motivation(for Data Mining),Data Mining-Definition & Functionalities, Data Processing, Form of Data Preprocessing, Data Cleaning: Missing Values, Noisy Data,(Binning, Clustering, Regression, Computer and Human inspection),Inconsistent Data, Data Integration and Transformation. Data Reduction:- Data Cube Aggregation, Dimensionality reduction, Data Compression, Numerosity Reduction, Clustering, Discretization and Concept hierarchy generation.

UNIT– IV [10 Hours]

Concept Description:- Definition, Data Generalization, Analytical Characterization, Analysis of attribute relevance, Mining Class comparisions, Statistical measures in large Databases. Measuring Central Tendency, Measuring Dispersion of Data, Graph Displays of Basic Statistical class Description, Mining Association Rules in Large Databases, Association rule mining, mining Single-Dimensional Boolean Association rules from Transactional Databases– Apriori Algorithm, Mining Multilevel Association rules from Transaction Databases and Mining Multi-Dimensional Association rules from Relational Databases

UNIT – V [12 Hours]

Classification and Predictions: What is Classification & Prediction, Issues regarding Classification and prediction, Decision tree, Bayesian Classification, Classification by Back propagation, Multilayer feed-forward Neural Network, Back propagation Algorithm, Classification methods K- nearest neighbor classifiers, Genetic Algorithm. Cluster Analysis: Data types in cluster analysis, Categories of clustering methods, Partitioning methods. Hierarchical Clustering- CURE and Chameleon. Density Based Methods-DBSCAN, OPTICS. Grid Based Methods- STING, CLIQUE. Model Based Method –Statistical Approach, Neural Network approach, Outlier Analysis

Reference

1. *M.H.Dunham,"Data Mining:Introductory and Advanced Topics" Pearson Education,2013*
2. *Jiawei Han, Micheline Kamber, "Data Mining Concepts & Techniques" Elsevier, 2013.*

3. *Sam Anahory, Dennis Murray, "Data Warehousing in the Real World: A Practical Guide for Building Decision Support Systems, 1/e", Pearson Education. 2009.*
4. *Mallach,"Data Warehousing System", McGraw –Hill, 2008.*

4E6: DIGITAL IMAGE PROCESSING

Total Teaching Hours: 52 No. of Hours/ Week: 04

UNIT I DIGITAL IMAGE FUNDAMENTALS [12 Hours]
Origin of Digital Image processing –fundamental steps –Components of Image Processing system –Visual perception –Light and EM spectrum –Image sensing and acquisition –Image sampling and Quantization –relationship between pixels, Two-Dimensional Mathematical Preliminaries

UNIT II IMAGE ENHANCEMENT [10 Hours]
Spatial Domain: Gray level transformation –Histogram processing –Arithmetic / Logic operations- Spatial filtering –smoothing filters –sharpening filters Frequency Domain: Fourier transform –smoothing frequency domain filters –sharpening filters – Homographic filtering

UNIT III IMAGE RESTORATION [10 Hours]
Image Restoration - Degradation Model, Unconstrained Restoration - Lagrange multiplier and Constrained restoration, Inverse filtering-removal of blur caused by uniform linear motion, Wiener filtering, Geometric transformations-spatial transformations.

UNIT IV IMAGE SEGMENTATION [10 Hours]
Edge detection, Edge linking via Hough transform, Thresholding, Region based segmentation, Region growing, Region splitting and Merging, Segmentation by morphological watersheds, Basic Concepts, Dam Construction, Watershed segmentation algorithm.

UNIT V IMAGE COMPRESSION [10 Hours]
Need for data compression, Fundamentals –Image compression models Huffman, Run Length Encoding, Shift codes, Arithmetic coding, Vector Quantization, Transform coding, JPEG standard, MPEG.

Reference

1. *Rafael C. Gonzalez, Richard E. Woods, , Digital Image Processing', Pearson, Second Edition, 2004.*
2. *Anil K. Jain, , Fundamentals of Digital Image Processing', Pearson 2002.*
3. *Kenneth R. Castleman, Digital Image Processing, Pearson, 2006.*
4. *Rafael C. Gonzalez, Richard E. Woods, Steven Eddins,' Digital Image Processing using MATLAB', Pearson Education, Inc., 2004.*
5. *D,E. Dudgeon and RM. Mersereau, , Multidimensional Digital Signal Processing', Prentice Hall Professional Technical Reference, 1990*
6. *William K. Pratt, , Digital Image Processing' , John Wiley, New York, 2002*
7. *Milan Sonka et aI, 'IMAGE PROCESSING, ANALYSIS AND MACHINE VISION', Brookes/Cole, Vikas Publishing House, 2nd edition, 1999*

BANGALORE UNIVERSITY
MCA PROGRAMME
Open Elective: "Cyber Space"

Total Teaching Hours: 52 No. of Hours / Week: 04

Objectives:
To understand cyber space, social media in cyber space, advantages, disadvantages, IT Act 2000/2008, Digital Signature, Electronic Signature, e-commerce, and e-governance

UNIT-I [20 Hours]
Basics of internet, www, http, html, DNS, IP Address, electronic mail, web browsers, search engines, Social Media: Twitter, Facebook, Youtube, whatsapp, LinkedIn, advantages, disadvantages, privacy issues

UNIT-II [10 Hours]
e-commerce, advantages of e-commerce, survey on popular e-commerce sites

UNIT-III [10 Hours]
Introduction to e-governance, stages of e-governance, advantages, challenges, International Status, Indian status

UNIT-IV [12 Hours]
IT Act, 2000 salient features, digital signature, electronic signature, Cyber Appellate Tribunal, Adjudicator, offences, and penalties.

Reference

1. *Information Technology Amended Act, 2008, Ministry of Law and Justice, Government of India.*
2. *SrinivasBhogle, "E-Governance" Chapter III in Selected Readings on Information Technology Management: Contemporary Issues, Information Science reference, Hershey, New York, page no. 40-61.*
3. *Tom Huskerson. Social Media, the Good, Bad, and Ugly: Volume. 3. 2014*
4. *RitendraGoel. "e-commerce", New Age International Publishers, 2008*
5. *Dougals E Comer. Computer Network and Internet. Pearson, 2008*