Proposal for

Revised Syllabus

For

Master of Science (MSc) – Computer Science

M.Sc. Computer Science Programme Structure (Proposal)
(2014 - Admission Onwards)

This document contains the draft of the proposed syllabus for MSc Computer Science for the academic year 2014-15 in University Centres and affiliated colleges. The complete draft syllabus is enclosed herewith for your information and review.

Please thoroughly go through the scheme\contents and inform me your feedbacks at the earliest. This will help me to take necessary steps for incorporating the required changes before placing it in the Board of Studies meeting.

The copy of your feedbacks may also be copied to: abdulhaleem@farookcollege.ac.in

Expecting a prompt response in this regard.

NB: Please circulate this information to all concerned faculty members, students and subject experts in your contact list.

With warm regards,

Dr.LAJISH.V.L
Assistant Professor & Head (i/c) &
Chairman, PG Board of Studies in CS & Applications.
Department of Computer Science
University of Calicut, Kerala-673635, INDIA
E-mail : lajish@uoc.ac.in
Tel : +91-494-24017325 (Office)
: +91-9495793094 (Cell)
Web : http://www.universityofcalicut.info

1. Course should be helpful for the students
    a. To enable them to contest for competitive examinations like NET/GATE/JEST etc.
    b. to take up a research oriented higher course such as MPhi/PhD, Or
    c. to help them to secure a position in the industry.

2. These objectives are desired by giving the students choices for their "specialization" through a set of electives right from the Semester II.

3. You are kindly request to contribute your ideas in the following aspects:
    a. Are there any papers to be included/excluded in/from the core papers section?
    b. Can even the core papers be made industry relevant by adding topics relevant to the present. If yes, portions to be added to each of such subjects.
    c. Are the broad range of topics proposed under Electives (I, II, III and IV) are sufficient to enable the student to decide his specialization? – these subjects can be conventional/industry related/ or a mix of both.

4. Kindly go through this document and check the following:
    a. Whether the course has a logical flow right from the first semester to the last semester?
    b. Does the syllabus proposed for each course reflect the intended objectives (overall and course wise)?
    c. Are the syllabus contents realistic – can they be covered within one semester?
    d. Are they any overlapping of contents in the syllabus (for eg, between Theory of Computation and Discrete Structures)?
    e. Does the syllabus reflect the contents of the reference/text books listed?
    f. Are the text books readily available?
    g. Practical
        i. Is the list of experiments and corresponding theory portions have a correspondence each other?
        ii. Is the list sufficient?
        iii. Are there any experiments that are not "viable" or not suitable?

5. In case, if you have a modification to be proposed for any of the subjects, please use the attached format to specify your proposals (Proposal For Modification of Existing Subjects.doc).
6. If you would like to propose a new course, that can also be done in the attached format (Proposal For Subject.doc).

Course Structure, Scheme of Evaluation

Semester I

Course No	Subject Code	Subject Title	Instructional Hours/week			Marks		
			Theory	Practical	Total	Internal	Final	Credits
1	S1.1	Discrete Mathematical Structures	4	0	4	25	75	4
2	S1.2	Advanced Data Structures	4	0	4	25	75	4
3	S1.3	Theory of Computation	4	0	4	25	75	4
4	S1.4	The Art of Programming Methodology	4	0	4	25	75	4
5	S1.5	Computer Organization & Architecture	4	0	4	25	75	4
6	S1.6	Practical 1 (1.2 & 1.4) - (Departmental)	0	4	4	25	75	4
							Total Credits	**24**

Semester II

Course No	Subject Code	Subject Title	Instructional Hours/week			Marks		
			Theory	Practical	Total	Internal	Final	Credits
7	S2.1	Design and Analysis of Algorithms	4	0	4	25	75	4
8	S2.2	Operating System Concepts	4	0	4	25	75	4
9	S2.3	Computer Networks	4	0	0	25	75	4
10	S2.4	Computational Intelligence	4	0	4	25	75	4
11	S2.5	Elective I	4	0	4	25	75	4
12	S2.6	Practical 2 (S2.2 & S2.3) (University)	0	4	4	25	75	4
13	S2.7	Seminar	0	1	1	75	0	1
							Total Credits	**25**

Elective II S2.5 - List of Courses		
Course No	**Subject Code**	**Subject Title**
11a	S2.5a	Computer Graphics
11b	S2.5b	Introduction to Soft Computing
11c	S2.5c	Web Technology
11d	S2.5d	Bio Informatics
11e	S2.5e	Computer Optimization Techniques
11f	S2.5f	Numerical and Statistical Methods

Semester III

Course No	Subject Code	Subject Title	Instructional Hours/week			Marks		
			Theory	Practical	Total	Internal	Final	Total
14	S3.1	Advanced Data Base Management System	4	0	4	25	75	4
15	S3.2	Principles of Compilers	4	0	4	25	75	4
16	S3.3	Object Oriented Programming Concepts	4	0	4	25	75	4
17	S3.4	Elective II	4	0	4	25	75	4
18	S3.5	Elective III	4	0	4	25	75	4
19	S3.6	Practical 3 (S3.1 & S3.3) (Departmental)	0	4	4	25	75	4
							Total Credits	**24**

Elective II S3.4 – List of Courses		
Course No	**Subject Code**	**Subject Title**
17a	S3.4a	Pattern Recognition
17b	S3.4b	Wireless and Mobile Networks
17c	S3.4c	Cryptography & Network Security
17d	S3.4d	Advanced Web Technology
17e	S3.4e	Virtualisation And Cloud Computing
17f	S3.4f	Data Warehousing and Data Mining

Elective III S3.5 – List of Courses		
Course No	**Subject Code**	**Subject Title**
18a	S3.5a	Data Compression
18b	S3.5b	Pervasive Computing
18c	S3.5c	System Security
18d	S3.5d	Molecular Simulation and Modeling
18e	S3.5e	Fundamentals of Big Data
18f	S3.5f	Web Engineering

Semester IV

Course No	Subject Code	Subject Title	Instructional Hours/week			Marks		
			Theory	Practical	Total	Internal	Final	Total
19	S4.1	Elective IV	4	0	4	25	75	4
20	S4.5	Major Project + SE & RM (Duration of the Project = 16 Weeks) (University)	Duration of the Project = 16 Weeks Total Theory Hours for SE & RM = 20 Hours			300	100	8
							Total Credits	**12**

Elective IV S4.1 – List of Courses		
Course No	**Subject Code**	**Subject Title**
19a	S4.1a	Digital Image Processing
19b	S4.1b	Advanced Topics in Database Design
19c	S4.1c	Software Development for Portable Devices
19d	S4.1d	Storage Area Networks
19e	S4.1e	Semantic Web
19f	S4.2f	Advanced Java Programming

M.Sc. Computer Science

SYLLABUS

First Semester

S1.1 Discrete Mathematical Structures

Course Number: 1

L	P	C
4	0	4

Prerequisites/ Exposure: None

Objectives: To introduce discrete mathematics concepts necessary to understand basic foundation of computer science.

Unit 1: Sets and Mathematical Logic: Set Theory- Types of sets, Set operations, Principles of Inclusion Exclusion. Mathematical Logic-Propositional Calculus-Statement, Connectives, Conditional and Biconditional, Equivalence of Formula, Well Formed Formula, Tautologies, Duality Law, Functionally complete sets of connectives, Normal Forms, Theory of Inference for the Statement Calculus, Predicate Calculus-Statement Functions, Variables and Quantifiers, Free and Bound Variables, Theory of Inference for the Predicate Calculus.

Unit II: Functions and Relations: Functions – Types of Functions, Composition of Functions and Inverse Functions. Relations - Relations and Their Properties, Functions as relations, Closure of Relations, Composition of relations, Equivalence Relations and Partitions. Partial Ordering, Hasse Diagram. The Pigeon Hole principle.

Unit III: Lattices and Boolean Algebra-Lattices and Algebraic Systems, Principles of Duality, Basic properties of Algebraic systems defined by lattices, Distributive Lattices and Complemented Lattices. Boolean Lattices and Boolean Algebras. Boolean Functions and Boolean Expressions.

Unit IV: Group Theory – Definition and Elementary Properties- Permutation Groups, Cyclic Groups- Subgroups- Cosets and Lagrange's Theorem, Semigroup and Monoid. Homeomorphism and Isomorphism. Rings, Integral Domains and Fields. Prim's and Kruskal's Algorithm – Shortest Path Problem – Dijkstra's Algorithm.

Unit V: Graph Theory- Paths ,Cycles and Connectivity, Subgraphs, Types of Graphs, Representation of Graphs, Graph Isomorphism, Bipartite Graphs, Subgraphs, Eulerian and Hamiltonian Graphs. Trees – Spanning Trees, Cayley's theorem. Prims and Kruskals Algorithm – Shortest Path Problem – Dijktra's Algorithm.

References:

1. J.K. Sharma, Discrete Mathematics, Macmillan India Ltd.
2. Alan Doerr and Kenneth Levassur, Applied Discrete Structure for Computer Science, Galgotia Publication
3. C.L.Liu, Elements of Discrete Mathematics, McGraw–Hills Publications
4. Trembley J.P. & Manohar R.P, Discrete Mathematical Structures with Application to Computer Science, Mc.Graw Hill, 2007.

S1.2 Advanced Data Structures

Course No: 2

L	P	C
3	1	4

Prerequisites/Exposure: None

Objectives: To introduce basic and advanced data structures dealing with algorithm development and problem solving.

Unit I: Data structure – definition - types & operations, characteristics of data structures - Abstract Data Type (ADT) – algorithms – concepts – definition - objectives of algorithms - quality of an algorithm - space complexity and time complexity of an algorithm.

Unit II: Linear data structures - Arrays – records – representation - data structure operations - traversing, inserting and deleting - sorting and searching- sorting algorithms - linear search & binary search – complexity. Linked lists – operations and implementations, Stack - operations and its implementations(both array and linked list) – Applications- parsing arithmetic expressions, conversion and evaluating expressions - recursion-characteristics of recursion, types of recursion - applications of recursion in algorithms - comparison of recursive and non-recursive algorithms, queue - operations and its implementations (both array and linked list) – circular queue – dequeue - priority queues, recursive lists, heterogeneous lists, deterministic skip lists, doubly linked lists and circular lists - sparse matrix- representation.

Unit III: Non-linear Data Structures - trees – terminology - tree traversals algorithms - Binary trees - threaded binary trees – binary search trees - traversals and operations on BST – heap Tree - balanced trees - M-way trees – B and B+ trees, Red Black Tree, Digital Search Tree, Tries, Treaps, Huffman algorithm for extended binary tree - operationsand their implementation. Graphs - representation of graphs - operations - traversals and their implementation.

Unit IV: Hashing - overview of hashing – hash tables – hash functions and their computations – open addressing – linear probing - quadratic probing - double hashing algorithms and their implementations – rehashing – extendable hashing - separate chaining - hashing efficiency – heaps - overview of heaps - implementation and operations.

Unit V: Heap structures - Min-Max heaps - Deaps - leftist heaps - binomial heaps - Fibonacci heaps -binary heaps - skew heaps - pairing heaps – applications - amortized analysis - an unrelated puzzle - Binomial queues - skew heaps - Fibonacci heaps - Splay trees.

References:

1. Alfred V.Aho, John E.Hopcroft and Jeffrey D.Ullman, Data structures and Algorithms, Pearson Education Asia,2002.
2. Horowitz E & Sahni S, Fundamentals of data structures, Computer Science press, 1978.
3. Richard F. Gilberg & Behrouz A. Forouzan, Data Structure . A Pseudocode Approach with C ", Thomson Brooks/Cole Publications, 2004
4. Tanenbaum Andrew S, Y Langsam and M. J. Augenstein, Data Structure using C, Prentice- Hall, India, Reprint, 2007.
5. Robert Kruse, Tondo C L and Bruce Leung, Data Structures & Program Design in C, Pearson Education, 2nd Edition, 2004.
6. U.A. Deshpande & O. G. Kakde , Data Structures and Algorithms, ISTE Learning Materials Centre, New Delhi, 2003.
7. Thomas H Cormen, Charles E Leiserson, Ronald L Rivest and Clifford Stein, Introduction to Algorithms,Third Edition, PHI,2010.
8. Seymour Lipschutz and GAV Pai, Data Structures, Indian Adapted Edition, Schaum's Outlines Series, TMH, 2006.
9. Cormen, Leiserson and Rivest, Introduction to Algorithms, 3rd Edition, PHI.
10. Robert Kruse, C. L. Tondo , Bruce Leung, Data Structures and Program Design in C (Second Edition), Pearson Education. September 2007
11. Tremblay & Sorenson, Introduction to data structures with applications, TMH (Second Edition), McGraw Hill Book Company, 1998.

S1.3 Theory of Computation

Course No: 3

L	P	C
4	0	4

Prerequisites/Exposure: None

Objectives: To provide students with an understanding of basic concepts in the theory of computation.

Unit I: Preliminaries - Introduction to formal proof and inductive proofs- The central concepts of Automata Theory - Alphabets, Strings, Languages – Introduction to automata and grammar - Deterministic Finite Automata, Non-deterministic Finite Automata – Equivalence of Deterministic and Nondeterministic Finite Automata - Finite Automata with Epsilon Transitions - Equivalence of NFA with and without epsilon moves.

Unit II: Regular Expressions, Finite Automata and Regular Expressions, Properties of regular Languages - Pumping lemma and proof for existence of non regular languages, Closure properties, homomorphism, substitution - Decision Properties - Equivalence and Myhill Nerode and DFA state minimization – Regular Grammar.

Unit III: Context free Languages - Equivalence of CFG and PDA – Normal forms (CNF and GNF) – Closure properties of CFL's – DCFL's and their properties – Decision procedures – CYK algorithm – Pumping lemma and proof for existence of non context-free languages – Context sensitive languages: Equivalence of LBA and CSG.

Unit IV: Turing machines - TM computations – Equivalence of standard TM with multi tape and non deterministic TM's – Turing acceptable, Turing decidable and Turing enumerable language classes - Equivalence of type 0 grammars with TM's – Church' thesis – Chomsky hierarchy - Closure properties of recursive and recursively enumerable languages.

Unit V: Computability and Decidability – halting problem – reductions – post correspondence problem. Computational complexity - Time and space bounded simulations – Classes P and NP – NP completeness – Cook's theorem.

References

1. J.E Hopcroft and J.D Ullman, Rajeev Motwani, Introduction to Automata Theory , Languages of Computation , Narosa.
2. H.R Lewis and C.H Papadimitriou, Elements of Theory of Computation , Prentice Hall.
3. Linz P, An Introduction to formal Languages and Automata, Narosa.
4. Martin J.C, Introduction to Languages and Theory of Computation, Tata McGraw Hill.
5. J. E. Sagage, Models of Computation, Exploring the power of Computing, Addison Wesley, 1998.
6. Michael Sipser, Introduction to theory of Computation, Cenage Learning, Indian Edition.

S1.4 The Art of Programming Methodology

Course No: 4

L	P	C
2	2	4

Prerequisites/Exposure: None

Objectives:

- To learn the art of designing algorithms and flowcharts.
- To introduce the concept of algorithmic approach for solving real-life problems.
- To develop competencies for the design, coding and debugging of computer programs.
- To learn designing program with advanced features of C.

Unit I: Part A - Problem Solving - Three Methods of Describing a Program - Flow Charts for Structured Programming – Computer Model – Procedures and Environments – Executing Procedure Calls and Returns – Global and Local Variables. Interfacing Procedures – Introduction – Reference Parameters – Automatic Protection of Arguments – Expression as Arguments in a Procedure Call – Function Procedures – Name Parameters – Parameters that Stand for Procedures and Functions – Recursion. **Part B** - Algorithm Design – Problem Solving Aspect – Top Down Design - Implementation of Algorithms – Fundamental Algorithms (Discuss the Design of Algorithms only). **Part C** - Program, Characteristics of a good program - Modular Approach - Programming style - Documentation and Program Maintenance - Compilers and Interpreters - Running and Debugging Programs - Syntax Errors- Run-Time Errors - Logical Errors - Concept of Structured Programming.

Unit II:. Introduction to C Programming, overview and importance of C, C Program Structure and Simple programs, Creation and Compilation of C Programs under Linux and Windows Platforms. Elements of C Language and Program constructs: - structure of C program - character set, tokens, keywords, identifier - Data types, constants, symbolic constants, variables, declaration, data input and output, assignment statements. Operators in C - arithmetic operators, relational operators, logical operators, assignment operators, increment and decrement operators, conditional operators, special operators, precedence of operators - arithmetic expressions – evaluation of expressions, type conversion in expressions – precedence and associativity - mathematical functions - I/O operations.

Unit III: Decision making – IF statement, IF ELSE statement, Nesting of IF ELSE and ELSE IF Ladder, SWITCH statement, BREAK statement, CONTINUE statement, GOTO statement, return statement – Looping - WHILE, DO-WHILE, and FOR loops, nesting of loops, skipping & breaking loops - Arrays - single dimension arrays - accessing array elements - initializing an array, two dimensional & multi dimensional arrays - memory representation - strings – processing of strings - string manipulation functions.

Unit IV: The Concept of modularization - defining function - types of functions – User

defined functions - function prototype and definition – arguments - passing parameters - call by reference - call by value – returning - nesting of functions and recursion - passing arrays & strings to function - returning multiple values - recursion – scope and life time of variables storage class specifiers - automatic, extern, static storage, register storage - Structures & Union definition , giving values to members, structure initialization, comparison of structure variables, arrays of structures, arrays within structures, structures within arrays, structures and functions, Unions, bit-fields.

Unit V: Pointer - pointer operator - pointer expression - declaration of pointer - initializing pointer - de-referencing - pointer to pointer, constant pointer, array of pointers, pointer to function. Files **-** file handling - defining & opening a file - closing a file - Input/output operations on files – error handling , random access to files, command line arguments – dynamic memory allocation - preprocessor directives: macro substitution directives - simple macros - macros with arguments - nesting of macros, compiler control directives.

References

1. Elliot I Organick, Alexandra L Forsythe and Robert P Plummer, Programming Language Structures, Academic Press New York (Unit I Part A).
2. R G Dromey, How to Solve by Computer, Pearson Education, Fifth Edition 2007 (Unit I Part B).
3. J.B Dixit, Computer Fundamentals and Programming in C, Laxmi Publications (Unit I Part C).
4. E Balagruswamy, Programming in ANSI C, TMH, Third Edition 2005.
5. Gottfried, Programming with C, Schaums Outline Series, TMH Publications.
6. Kernighan & Ritchie, C Programming Language.
7. Kanetkar, Let Us C, BPB Publications.
8. Mahapatra, Thinking in C, PHI Publications.
9. Kernighan & Ritchie, C Programming Language.

S 1.5 Computer Organization & Architecture

Course No: 5

L	P	C
4	0	4

Prerequisites/Exposure: None

Objective: To familiarize with the digital fundamentals, computer organization, computer architecture and assembly language programming.

Unit I: Number systems and Conversions, Boolean Algebra - Truth Tables - Logic gates and Map simplification - flip-flops - design of combinational and sequential circuits - examples of digital circuits – adders, multiplexers, decoders, counters, shift registers - register transfer language and micro operations - data representation - data types, sign and magnitude, complements, fixed-point representation, floating-point representation, other binary codes, error detection codes.

Unit II: Basic computer organization – machine instructions – classification, function, addresses, size, addressing modes – instruction cycle - instruction sequencing. fundamental concepts – registers, register transfers, performing arithmetic or logic operations, memory read and write, execution of a complete instruction, branch instruction, Single bus, two bus, three bus organization, a complete processor - Control unit: - hardwired control, microprogrammed control, micro instructions-types.

Unit III: Arithmetic & Logic Unit - addition of positive numbers – fast adders – signed addition and subtraction - addition/subtraction logic unit – multiplication of positive numbers – array multiplier, sequential multiplier - signed number multiplication - multiplication using Booth's algorithm - fast multiplication – bit pair recording of multiplication, division-restoring and non restoring algorithms, floating point numbers and operations.

Unit IV: Main Memory - memory hierarchy – main memory – RAM,ROM- memory cells -cell organization - working – performance considerations - cache memory – virtual memory - memory management requirements - secondary storage – memory interleaving. Input / Output Organization - Accessing I/O devices – programmed I/O, interrupt I/O - interrupts - interrupt processing – hardware interrupts – programmable interrupt controller – vectored interrupts - interrupt nesting - daisy chaining - direct memory access (DMA) - DMA operations & DMA Controller, Introduction to I/O interfaces, I/O channels, IO Processors.

Unit V: Architecture - General 8-bit microprocessor and its architecture - 8085 - Functional block diagram-architecture functions of different sections - architecture of 8086 CPU. Instruction Sets - Instruction format - addressing modes - instruction set of 8085 CPU - Instruction cycle-timing diagrams - different machine cycles - fetch and execute operations - estimation of execution time - estimation of execution time. Intel 8051 Micro controller – Architecture - basic instructions-basic assembly language programs- peripherals: interrupts, timers, parallel port, serial port.

References:

1. V C Hamacher, Computer Organization, Mc-Graw Hill International Edition, Fifth Edition.
2. Morris Mano, Digital logic and Computer design, Prentice Hall of India, 2004.
3. M Morris Mano, Computer System Architecture, Prentice Hall, Third Edition.
4. William Stallings, Computer Organization and Architecture, Fifth Edition.
5. Andrew S Tanenbaum, Structured Computer Education, Prentice Hall, Fourth Edition.
6. Floyd and Jain , Digital Fundamentals, Pearson Education, Eighth Edition.
7. Albert Paul Malvino, Donald P Leach, Digital Principles and Applications, McGraw Hill, Fourth Edition.

8. Thomas C Bartee, Digital computer Fundamentals, McGraw Hill, Sixth Edition.
9. Ramesh. S. Gaonkar, Microprocessor Architecture, Programming, and Applications With the 8085, Wiley Eastern Ltd, New Delhi.
10. Mohamed Rafiquzzaman, Introduction to Microprocessors and Microcomputer Based System Design, 2nd edition, CRC Press
11. Muhammad Ali Mazidi and Janice Gillispie Mazidi, The 8051 Microcontroller and Embedded Systems, Pearson Education Asia, Fifth Indian Reprint 2003.

S 1.6 Practical – Programming and Data Structures using C

Course No: 6

L	P	C
0	2	4

Prerequisites/Exposure: None
Objective: To practically implement the techniques learned from course no 2 and 4.

Unit I: C Programming

1. Simple C Programs like area of a circle, checking whether a given number is odd or even.
2. Implementation of programs using Loops (pyramid printing, factorial computation, number reversing, checking for Armstrong numbers, finding first N or Nth Prime numbers etc.).
3. Use of 1D and 2D Arrays (searching, sorting and vector operations, matrix addition, matrix multiplication).
4. String Manipulations.
5. Structures and Unions (like addition of Two Complex numbers, student record creation and manipulation etc.)
6. Writing functions.
7. Implementation of recursion (recursive function to compute a factorial, reverse string etc)
8. Command line arguments.
9. Pointers - simple programs to learn concept of pointers, array operation using pointers etc.
10. File operations – file and structures.

Unit II: Data Structures and Algorithms

1. Implement stacks using arrays.
2. Implement queues, circular queue using arrays.
3. Implement sequential search and binary search techniques.
4. Implement linked lists and operations (add, insert, delete, search) on linked lists.
5. Implement stacks using linked list.
6. Implement queues using linked list.
7. Implement doubly linked lists.

8. Implement circular linked lists.
9. Implement binary tree and traversals
10. Implement Binary search trees and perform the operations on BST.
11. Implement various sorting algorithms.
12. Convert an infix expression to the postfix form using stacks.
13. Write a program to evaluate a postfix expression.
14. Implement Graphs and graph traversals.
15. Implement Heap tree and operations.

Second Semester

S 2.1 Design and Analysis of Algorithms

Course No: 7

L	P	C
4	0	4

Objective:

- To introduce the concept of algorithmic approach for solving real-life problems.
- To teach basic principles and techniques of computational complexity.
- To familiarize with parallel algorithms and related techniques.

Unit I: Efficiency of Algorithms - RAM model – cost estimation based on key operations - Analysis of Algorithms, Time and Space complexity, Asymptotic Notations, Average case analysis of simple programs like finding of a maximum of n elements. Recursion and its systematic removal. Quicksort - non recursive implementation with minimal stack storage.

Unit II: Master's theorem – solution to recurrence relations with full history - probabilistic analysis – linearity of expectations – worst and average case analysis of quick sort, merge sort, heapsort, binary search, hashing algorithms – lower bound proofs for the above problems - amortized analysis – aggregate, accounting and potential methods – analysis of Knuth-Morris-Pratt algorithm –amortized weight balanced trees.

Unit III: Design of algorithms - Divide and conquer - General methods - binary search - Min Max - Greedy Method - Elements of greedy strategy - 0-1-knapsack problem - Graph Algorithms – Breadth First Search, Depth First Search, Minimum Spanning Trees, Single Source Shortest Path.

Unit IV: Complexity - complexity classes – P, NP, Co-NP-Hard and NP-complete problems – Cook's theorem – NP completeness reductions for clique, vertex cover, subset sum, Hamiltonian cycle and TSP.

Unit V: Dynamic Programming – all pairs shortest path. Backtracking, Branch and Bound – TSP problem. Deterministic and non deterministic algorithms.

References

1. Thomas H Cormen, Charles E Leiserson, & Ronald L Rivest, Introduction to Algorithms, 3rd Edition, Prentice Hall of India Private Limited, New Delhi, 2001.
2. S. Basse, Computer Algorithms: Introduction to Design and Analysis, Addison Wesley, 1998.
3. U. Manber, Introduction to Algorithms : A creative approach, Addison Wesley, 1989.
4. Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, The design and Analysis or Computer Algorithms, Addison Wesley, 1974.
5. Gilles Brassard and Paul Bratley, Fundamentals of Algorithmics, Prentice-Hall of India, 2007.
6. Goodman S E and Hedetniemi, Introduction to the Design & Analysis of Algorithms, Mcgraw Hill, 2002.
7. Horowitz E & Sahni S, Fundamentals of Computer Algorithms, Galgotia Publications Pvt. Ltd, 2004.
8. Sahni, Data Structures, Algorithms and Applications in C++, Tata Mcgraw Hill.
9. Levitin, Introduction to the Design and Analysis of Algorithms, 1st Edition.

S 2.2 Operating System Concepts

Course No: 8

L	P	C
4	0	4

Prerequisites/Exposure: S1.2 Advanced Data Structures, S 1.5 Computer Organization & Architecture.

Objectives:

- Introduce the underlying principles of an operating system.
- Exposure of multi programming, virtual memory and resource management concepts.
- Case study of public and commercially available operating systems.

Unit I: Operating System Overview - Objectives and functions – Evolution of Operating System – Major Achievements – Process Description and Control – Process, Creation & Termination of Processes, Five State Model, Suspended Process, Process Description, Process Control – Modes of Execution, Process Creation, Process and Mode Switching. Threads – Processes Vs Threads, Multithreading, Thread States, Types of Threads, Multi Core and Multithreading. Case Study - Unix SVR4 Process Management, Linux Process and Thread Management.

Unit II: Concurrency – Principles, Race Condition, Operating System Concerns, Process Interaction, Completion for Resources, Cooperation by Sharing. Mutual Exclusion - , Requirements, Hardwire Support, Semaphores, Producer Consumer Problem, Monitors, Message Passing, Readers/Writers Problem. Deadlock – Principles, Prevention,

Avoidance, Detection, Recovery, Dining Philosophers Problem. Case Study: Unix Concurrency Mechanisms.

Unit III: Memory Management, Address binding, Logical Vs Physical address space, Dynamic Loading, Dynamic Linking and Shared Libraries, Overlays, Swapping, Contiguous Memory allocation, Paging, Segmentation, Virtual memory, Demand paging, Page replacement, Thrashing. Case Study: Windows Memory Management. Uniprocessor Scheduling – types, scheduling algorithms – criteria, nonpreemptive, preemptive, FCFS, SJF, Priority, RR, Multilevel, Feedback Queue. Multiprocessor Scheduling – Classification, Granularity, Design Issues, Process Shceduling, Thread Scheduling. Real Time Scheduling - Background, Characteristics of Real Time OS, Scheduling, Deadline Scheduling, Rate Monotonic Scheduling, Priority Inversion. Case study: Linux Scheduling.

Unit IV: Embedded Operating Systems - Embedded Systems, Characteristics of Embedded OS. eCoS - Configuration, Components, Kernel, I/O System, Scheduler, Thread Synchronization. TinyOS – Goals, Components, Scheduler, Example Configuration, Resource Interface.

Unit V: Client/Server Computing – Definition, Applications, Classes, TThree-Tier Client/Server Architecture, Middleware. Service-Oriented Architecture – Distributed Message Passing - Remote Procedure Calls -Clusters. Case study - iOS and Android - Architecture and SDK Framework - Media Layer - Services Layer - Core OS Layer – File System.

References:

1. William Stallings, Operating Systems, Internals and Design Principles, Seventh Edition, Pearson.
2. Abraham Silberschatz; Peter Baer Galvin; Greg Gagne, Operating System Concepts, Seventh Edition, John Wiley & Sons, 2004.
3. Ann McIver McHoes, Ida M. Flynn, Understanding Operating Systems, 6th Edition, Cengage Learning, 2010.
4. Mukesh Singhal and Niranjan G. Shivaratri, Advanced Concepts in Operating Systems – Distributed, Database, and Multiprocessor Operating Systems, Tata McGraw-Hill, 2001.
5. Neil Smyth, iPhone iOS 4 Development Essentials – Xcode, Fourth Edition, Payload media, 2011

S 2.4 Computer Networks

Course No: 9

L	P	C
4	0	4

Prerequisites/Exposure: None

Objectives:

- To provide the student with a top down approach of networking starting from the application layer.
- To introduce computer networking in the back drop of Internet protocol stack.
- Be conversant with primitives of network application programming.

Unit I: Introduction to Computer networks – introduction – topology - categories of networks – Internetwork – Internet - network models - layered model - OSI and TCP/IP Models - Transmission media - Wired and unwired media. Computer networks and Internet - the network edge - the network core - network access - delay and loss - protocol layers and services – history of computer networking and Internet.

Unit II: Application layer protocols – principles – the web and HTTP – FTP – Email in Internet – DNS. Socket programming – building a Web server - content distribution.

Unit III: Transport layer services – introduction – relationship between Transport and Network layer – UDP – reliable data transfer – TCP - congestion control - Network layer services – routing – IP - routing in Internet - router - IPV6 - multicast routing – mobility.

Unit IV: Link layer services - error detection and correction - multiple access protocols – LAN address – ARP – Ethernet – hubs – bridges – switches - wireless links – PPP-ATM.

Unit V: Security in Networks – Principles of Cryptography – Authentication – Integrity – Key Distribution and Certification – Firewalls – Attacks and Counter Measures.

References

1. J. F. Kurose and K . W. Ross, Computer Networking: A Top-Down Approach Featuring Internet, 3/e, Perason Education, 2005.
2. Data Communications and Networking, Fourth Edition by Behrou A Forouzan, McGraw-Hill reprint, 2011.
3. Peterson L.L. & Davie B .S., Computer Networks, A Systems Approach, 3/E, Harcourt Asia, 2003.
4. Keshav S., An Engineering Approach to Computer Networking, Pearson Education, 2000.
5. Andrew S. Tanenbaum, Computer Networks, 3/E, PHI, 1996
6. Herbert Scheldt, Java Complete Reference, Tata McGraw Hill edition.

S 2.4 Computational Intelligence

Course No: 9

L	P	C
4	0	4

Prerequisites/Exposure: None

Objectives: Introduce concepts of artificial intelligence and machine learning.

Unit I: Introduction - Artificial Intelligence- problems, scope and applications, Problem space and search- Production system- characteristics- the predicate calculus, Inference rules, Structures and strategies for state space search, strategies for space search, using state space to represent reasoning with the predicate calculus.

Unit II: Heuristics Search: Control and implementation of state space search, Generate and test, Hill climbing, Best–first search, Problem Reduction, Constraint Satisfaction, Means-ends analysis, Heuristic in games, Complexity issues.

Unit III: Knowledge representation issues, representation and mappings, Representing simple facts in logic, Representing instances and ISA relationships, Computable functions and Predicates, Resolution, Natural deduction, Knowledge representation using rules, logic programming, forward versus backward reasoning, Symbolic reasoning under uncertainty- Nonmonotonic reasoning, Depth first search, Breadth first search.

Unit IV: Game Playing – The Minimax search procedure, adding Alpha-beta cutoffs, Additional refinement, Iterative deepening, Planning system and its components, Understanding, Understanding as constrained satisfaction. Slot and filler structures: Semantic nets, frames, conceptual dependency, scripts. Definition and characteristics of Expert System, representing and using domain knowledge, Expert system shells. Knowledge Engineering, knowledge acquisition, expert system life cycle & expert system tools, CYCIN & DENDRAL examples of expert system.

Unit V: Machine Learning – rote learning, learning by taking advice, learning in problem solving, learning from examples, Explanation based learning, Analogy, formal learning theory, Connectionist models- Hopfiled networks, learning in neural networks, back propagation, The genetic algorithm, classifier systems and genetic programming, artificial life and society based learning.

Text Books:

1. E. Rich, K. Knight and S.B.Nair, Artificial Intelligence, 3rd Edn. TMGH, New Delhi, 2009.
2. Foundations of Artificial Intelligence and Expert System - V S Janakiraman, K Sarukesi, & P Gopalakrishanan, Macmillan Series.
3. Stuart Russell, Peter Norvig, Artificial Intelligence: A Modern Approach, 3rd

Edition.

References:

1. G.F. Luger and W.A Stubblefield, Artificial Intelligence – Structures and Strategies for Complex Problem Solving, Addison-Wesley-1998.
2. P.H Winston – Artificial Intelligence, Addison-Wesley-1992.
3. Nils J. Nilsson , Artificial Intelligence , A New Synthesis, Morgan Kauf 2000.

S 2.5a Computer Graphics (Elective I)

L	P	C
4	0	4

Course No: 11a

Prerequisites/Exposure:

Objectives:

- To understand the fundamentals of the modern computer graphics.
- To pipeline the mathematics of affine transformations in three dimensions.
- To understand the common data structures to represent and manipulate geometry, colour and light representation and manipulation in graphics systems.
- To have an exposure to programming in Open GL.

Unit I: Introduction - application and output devices for computer graphics - raster and random scan display, CRT, color CRT, flat panel, LCD, LED, DVST. Adapters - monochrome display adapter (MDA), CGA, hercules graphics card, enhanced graphics adapter, Professional graphics adapter, VGA, SVGA. Graphics software - GKS, PHIGS, OpenGL. Scan conversion - Points & lines, line drawing algorithms - DDA algorithm, Bresenham's line algorithm. Circle generation algorithm - Mid-point circle algorithm, Ellipse generation.

Unit II: Filling, Clipping & Transformation (2D&3D) - Area scan conversion, seed fill algorithm, scan line polygon fill algorithm, Inside Outside test, Boundary fill algorithm, Flood fill algorithm. Character generation - Anti-aliasing - Clipping operations - Cohen Sutherland line clipping, Liang Barsky line clipping, Nicholl Lee line clipping, polygon clipping, Sutherland Hodgeman & Weiler Atherton polygon clipping, Text clipping. Transformation: Geometric & coordinate transformation, Inverse transformation, Composite transformation, Translation, rotation, scaling, shearing, reflection.

Unit III: Projection - 3D concepts & viewing pipeline, coordinate system, window to viewport coordinate transformation, parallel & perspective projection, projection matrix, view volume. 3D object representation - wireframe model, visible surface detection methods, depth comparison, Z-buffer algorithm, back face detection, BSP tree method, printer's algorithm, depth cueing.

Unit IV: Curves & Fractals - curve representation, surfaces, designs, spline representation, Bezier curves, cubic spline, beta spline, B-spline curves. Fractal's geometry, fractal generation procedure, classification of fractal, fractal dimension, fractal construction methods.

Unit V: Color & shading Models - Introduction, modelling light intensities and sources, diffuse reflection, Lambert's cosine law, specular reflection, half-toning, dithering, color model - XYZ,RGB,YIQ,CMY & HSV, shading algorithm & model, illumination model, gouraud shading, phong shading. OpenGL programming - Introduction, primitives drawing, colouring, transformation, filling, curve.

Reference:

1. Donald Hearn and M. Pauline Baker, Computer Graphics, Prentice Hall, 1997.
2. D.Hearn and M. P. Baker, Computer Graphics with Open GL, 3rd Ed., Prentice Hall, 2004.
3. FS Hill, JR, Computer Graphics using OpenG,L, Second Edition, Prentice Hall of India Private Ltd.-New Delhi, 2005
4. Dave, Mason Woo, Jackie, Tom Davis, Open GL Programming Guide, 6th Edition, Person.
5. OpenGL Redbook Version 1.1 (Online)
6. Shreiner and Angel, Interactive Computer Graphics: A Top-Down Approach with Shader-Based OpenGL, Pearson Education.

S2.5b Introduction to Soft Computing (Elective I)

Course No: 11b

L	P	C
4	0	4

Prerequisites/Exposure:

Objectives:

- To give students knowledge of soft computing theories fundamentals.
- To expose the fundamentals of non-traditional technologies and approaches to solving hard real-world problems.

Unit I: Introduction - introduction to statistical ,syntactic and descriptive approaches - features and feature extraction - learning - Bayes Decision theory - introduction - continuous case - 2-category classification - minimum error rate classification - classifiers - discriminant functions - and decision surfaces – error probabilities and integrals - normal density - discriminant functions for normal density.

Unit II: Introduction to Genetic Algorithm, Genetic Operators and Parameters, Genetic Algorithms in Problem Solving, Theoretical Foundations of Genetic Algorithms, Implementation Issues – systems

Unit III: Neural Model and Network Architectures, Perceptron Learning, Supervised

Hebbian Learning, Back-propagation, Associative Learning, Competitive Networks, Hopfield Network, Computing with Neural Nets and applications of Neural Network.

Unit IV: Introduction to Fuzzy Sets, Operations on Fuzzy sets, Fuzzy Relations, Fuzzy Measures, Applications of Fuzzy Set Theory to different branches of Science and Engineering.

Unit V: Advanced Topics - Support Vector Machines, Evolutionary computation (EC) - Evolutionary algorithms, Harmony search, Swarm intelligence.

References:

1. J.S.R.Jang, C.T.Sun and E.Mizutani, Neuro-Fuzzy and Soft Computing, Pearson Education, 2004.
2. M. Mitchell, An Introduction to Genetic Algorithms, Prentice-Hall, 1998.
3. D. E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, 1989.
4. S. V. Kartalopoulos, Understanding Neural Networks and Fuzzy Logic: Basic Concepts and Applications, IEEE Press - PHI, 2004.
5. S. Rajasekaran & G. A. Vijayalakshmi Pai, Neural Networks, Fuzzy Logic and Genetic Algorithms: Synthesis & Applications, PHI, 2003.

S2.5c Web Technology (Elective I)

Course No: 11c

L	P	C
4	0	4

Prerequisites/Exposure:

Objectives: Introduction tools for creating and maintaining websites – content development (HTML), client side scripting (JavaScript), web server (Apache), server side scripting (PHP), content management system (Joomla).

Unit I: Introduction to Web programming – Introduction to SGML features – HTML, XHTML, DHTML, XML – HTML Vs XML – Creating XML documents – Parsing an XML document – Writing well formed documents – Organizing elements with namespaces – Defining elements in a DTD – Declaring elements and attributes in a DTD. Overview of HTML - basic formatting tags - heading, paragraph, underline break, bold, italic, underline, superscript, subscript, font and image. Attributes - align, color, bgcolor, font face, border, size. Navigation Links using anchor tag - internal, external, mail and image links. Lists - ordered, unordered and definition, Table tag, HTML Form controls - form, text, password, textarea, button, checkbox, radio button, select box, hidden controls,

Frameset and frames. CSS.

Unit II: Client side programming – Introduction – popular client side scripting languages - Java Script - Introduction, Identifiers, Operators, Functions, Event handling, Classes, objects, Array, math, string, window object, Navigator DHTML Font, Text, Image change, Table expansion. JavaScript's object model; Strengths and weaknesses of JavaScript; Building and extending objects in JavaScript; Events in JavaScript; Event-handlers; Creating interactive forms; Introduction to cookies; using cookies in JavaScript & storing users choices in cookies. Encoding cookies; Browser objects: Object hierarchy, Creating Browser objects, Working with window, Document, History & location; Browser detection, Java to JavaScript communication.

Unit III: Web server – role - Apache Web Server – Introduction – Architecture – Features - Apache's Role in the Internet – LAMP – WAMP - Installation and Configuration - Build and Install Apache Web Server - Verify Initial Configuration Start, Stop, and Status the Apache Server Process. Configure Apache Core Modules Security - Basic Security with Apache - Host-based Authentication - User-based Authentication - Secure Sockets Layer (SSL) - Delivering Dynamic Web Content - Apache's Role in the Dynamic Web - Server Side Includes (SSIs) - Configure Apache Web Server to Support CGI – CGI Alternative Technologies. Virtual Hosts, Redirection, Indexing – Virtual Hosting with Apache, Virtual Host Configuration Redirection, Directory Indexing. Proxy Servers and Firewalls - Apache Proxy Configuring, Proxy Services Firewalls and Apache, Firewall Architecture Models Monitoring Apache Web Server - Error Logs, Logging HTTP Access ,Web Server Status and Server information, User Tracking - Proxy Caching.

Unit IV: Server side programming – server side scripts – PHP – Designing dynamic web pages using PHP - Defining PHP variables – variable types – operators – control flow constructs in PHP – passing form data between pages - Establishing connection with MySQL database – managing database.

Unit V: Overview of content management system - coding for reusability (header.php) – User Management - Article Publishing - Additional CMS features – Web site development using Joomla.

References :

1. Thomas A. Powell, The Complete Reference HTML
2. E. Stephen Mack & Janan Platt, HTML 4.0 - No experience required.
3. Robert W. Sebesta, Programming with World Wide Web, 4th edition, Pearson Education, 2009.
4. Xue Bal et. al, The Web Warrior Guide to Web programming, Thomson Learning.
5. Chris Bates, Web Programming: Building Internet Applications, 3rd ed, Wiley Academic Catalog.
6. H.M. Deitel, P.J. Deitel and A.B. Goldberg, Internet and World Wide Web: How to

Program, 3rd edition, Pearson Education.
7. Wagner and R. Allen Wyke, Javascript, SAMS.
8. Ye huda Shiran and Tomer Shiran, Learn Advanced JavaScript Programming.
9. Richard Bowen Ken Coar, Ken A Coar, Matthew Marlowe. Apache Server Unleashed.
10. Elizabeth Naramore, Jason Gerner, Yann Le Scouarnec, Jeremy Stolz, Michael K Glass, Beginning PHP5, Apache, and MySQL Web Development, Wrox , 2005.
11. Dan Squier, David Mercer, Allan Kent, Steven Nowicki, Clark Morgan, Wankyu Choi, Beginning PHP5 (Programmer to Programmer) (Paperback), Wrox, 2004.

S2.5d Bio Informatics (Elective I)

Course No: 11d

L	P	C
4	0	4

Prerequisites/Exposure:

Objectives:

- Expose students to the popular genomic and proteomic databases and to impart knowledge in processing and analyzing genomic data.
- Introduce advanced topics in bioinformatics.

Unit I: Introduction to Bioinformatics - Nature and scope of Computational Biology and Bioinformatics. Cells- Prokaryotes and Eukaryotes - DNA double helix - central dogma – RNA, Amino acids, Proteins - String representations. A glossary of Bioinformatics terms - File format for bio-molecular sequences, Sequence Alignment, Phylogeny, Gene finding, Microarray Analysis, Homology and evolutionary relationships.

Unit II: Basic Algorithms in Computational Biology - Exhaustive search methods and their applications in Computational Biology - String matching Algorithms. Motif finding - Tandem repeats – concept of Dynamic Programming - Graph Algorithms - Clustering Algorithms.

Unit III: Sequence Alignment - Pair-wise sequence alignment, Need of scoring schemes - Penalizing gaps, Scoring matrices for amino acid sequence alignment, PAM Probability matrix and Log odds matrix, BLOSUM, Dot-plot visualization, Needleman-Wunsch algorithm- effect of scoring schemes –evalues - BLAST and FASTA, Smith – Waterman algorithm for local alignment.

Unit IV: Multiple Sequence Alignment - Sequence alignment using dynamic programming, N-dimensional dynamic programming. Tools for MSA - Muscle and T-Coffee. Phylogenetic Algorithms - Evaluation of phylogenetic trees, significance.

Unit V: Introduction to the Major Resources - NCBI, EBI and ExPASy - Nucleic acid sequence databases - GenBank, EMBL, DDBJ – Protein sequence databases - SWISS-PROT, TrEMBL, PIR_PSD - Genome Databases at NCBI, EBI, TIGR, SANGER – Procedures to access these databases and to make use of the tools available.

Text Books:

1. Mount D, Bioinformatics: Sequence & Genome Analysis, Cold spring Harbor press.
2. Dan Gusfiled, Algorithms on Strings Trees and Sequences, Cambridge University Press.
3. Pevzner P A, Computational Molecular Biology: An Algorithmic Approach, MIT Press , Cambridge, MA, 2000.
4. Jeremy J. Ramsden, Bioinformatics: An Introduction, Springer.
5. Sushmita M and Tinku A, Data Mining Multimedia, soft computing and Bioinformatics, John Wiley & Sons, Inc., 2003

References:

1. Richard M. Karp, Mathematical challenges from genomics and molecular biology, Notices of the American Mathematical Society, vol. 49, no. 5, pp. 544-553
2. Glyn Moody, Digital Code of Life: How Bioinformatics is Revolutionizing Science, John Wiley & Sons, Inc.
3. Tao Jiang, Ying Xu and Michael Q. Zhang, Current Topics in Computational Molecular Biology, Ane Books.
4. Andrzej K. Konopka and M. James C. Crabbe, Compact Handbook of Computational Biology, CRC Press.
5. Bellman R E, Dynamic Programming, Princeton University Press.
6. Needleman S B and Wunsch C D, A general method applicable to the search for similarities in the amino acid sequence of two proteins, J. Mol. Biol., 48 (1970) 443–453.
7. Smith T F and Waterman M S, Identification of Common Molecular Subsequences, J. Mol. Bio. 147 (1981) 195–197.
8. Watson J D and Crick F H C, A Structure for Deoxyribose Nucleic Acid, Nature, 171 (1953) 737–738
9. Pevzner P A and Waterman M S, Open Combinatorial problems in computational molecular biology, Proc. Third Israel Symp. Theo. Comp. Syst. IEEE Computer Society Press, (1995) 158 – 173.

S2.5e Computer Optimization Techniques (Elective I)

Course No: 11e

L	P	C
4	0	4

Prerequisites/Exposure:

Objectives:

- To give an exposure for the student to the area of modeling techniques, numerical methods and algorithms.
- To realize the importance of various aspects of optimization techniques in industries like manufacturing and IT.
- To implement the knowledge of optimization techniques in real life problems.

Unit I: Linear Programming and Sensitivity Analysis - Two-variable LP-model, graphical and algebraic LP solutions, some LP applications, the Simplex Method and sensitivity analysis, primal-dual relationships and economic interpretation, dual simplex and generalized simplex algorithms and post-optimal analysis.

Unit II: Transportation and Network Models - The transportation models and algorithm, the assignment and transshipment models, minimum spanning tree algorithm, shortest-route problem, maximum flow and min-cost models, critical path method and algorithms for matching.

Unit III: Advanced Linear Programming and Applications - Simplex method fundamentals, revised simplex method and computational considerations, bounded variables algorithm, duality, parametric linear programming, goal programming formulations and algorithms.

Unit IV: Integer Linear Programming - Illustrative applications, integer programming algorithms, unimodularity and cutting-plane methods, traveling salesperson problem.

Unit V: Dynamic Programming and its Application - Recursive nature of computations in DP, forward and backward recursion, selected DP applications, problem of dimensionality, branch and bound method and dynamic programming, some deterministic inventory models. Nonlinear Programming - Convex programming problems, unconstrained problems and algorithms, constrained problems and algorithms.

References

1. H. A. Taha, Operations Research: An Introduction, Pearson Prentice Hall
2. C. H. Papadimitriou, K. Steiglitz, Combinatorial Optimization: Algorithms and Complexity, Prentice Hall India.

S2.5f Numerical and Statistical Methods (Elective I)

Course No: 11f

L	P	C
4	0	4

Prerequisites/Exposure:

Objectives:

- To provide the student with basic concepts in Statistics, Probability that can be applied for mathematical modeling of computer applications.

Unit I: Approximation And Errors in Computing - Introduction, Significant Digits - Inherent Errors – Numerical Error - Modeling Errors - Blunders - Absolute and Relative Errors - Conditioning and Stability. Roots Of Non-Linear Equations: Introduction - Iterative methods – Bisection - False position – Newton - Raphson's, Secant and Bairstow's methods.

Unit II: Introduction Solution Of Linear Equations - Gauss Elimination - Gauss-Jordan method - Jacobi Iteration method - Gauss-Seidal methods. Interpolation: Linear Interpolation - Newton's forward backward & divided difference interpolation methods – Lagrange's method.

Unit III: Integration - trapezoidal rule, simpson's 1/3, & 3/8 rules. Differential equations: heunn's polygon, range-kutta fourth order, milne-simpson and adams-base forth-moulton methods.

Unit IV: Classical definition of probability – statistical definition of probability – axiomatic approach to probability – addition and multiplication theorem on probability - compound and conditional probability – independence of events – Bayes theorem Random variables – Discrete and continues – pmf, pdf and distribution functions.

Unit V: Introduction Linear programming – Mathematical formulation – graphical method of solution – Simplex method – duality – Dual Simlex – Transportation – Assignment problems.

References:

1. E. Balagurusamy, Numerical Methods, 1999 Tata Mcgraw-Hill.
2. S.G. Gupta and V.K. Kapoor, Fundamentals Of Mathematical Statistics, 9th Edition, Sultan Chand & Sons. (Reprint 1999)
3. Computer Oriented Numerical Methods – V.Rajaraman, 3rd Edition, Prentice Hall Of India, 1993
4. Gupta S.C Kapoor V.K Fundamental Of Mathematical Statistics Sultan Chand & Sons
5. Mital Sethi, Linear Programming Pragathi Prakashan

S2.6 Practical 2 (S2.2 & S2.3)

Course No: 12

L	P	C
0	2	4

Prerequisites/Exposure: S2.2 & S2.3

Objectives:

- To practically implement the theory parts covered in subjects S2.2 and S2.3

Unit I: Operating System

1. Shell programming: creating a script, making a script executable, shell syntax (variables, conditions, control structures, functions, commands).
2. Implement process creation using Process System Calls (fork (), wait(), exec(), stat(), readdir())
3. File System Calls (open(), read(), write())
4. Command simulation (ls, grep, cp, rm)
5. Process Scheduling (FCFS, SJF, Priority, Round robin) Interprocess Communication (Fibonacci & Prime nos, who | wc –l, Chat Messaging, Shared Memory, Producer-Consumer problem)
6. Memory Management (First Fit, Best Fit, FIFO Page Replacement, LRU Page Replacement
7. File Alloaction (Contiguous Allocation)
8. Semaphore: programming with semaphores (use functions semctl,semget, semop, set_semvalue, del_semvalue, semaphore_p, semaphore_v).

Unit II: Computer Networks

1. Design a LAN with a given set of requirements. The design should include topology, hardware and software requirements like cable, connectors, hubs/switches/bridges, interface cards along with a budget for the LAN. (Faculty in charge should give the requirements to the students).
2. Write a program to implement TCP Echo Client
3. Write a program to implement TCP Echo Server
4. Write a Program to check the Date and Time in TCP Date Time Client
5. Write a Program to check the Date and Time in TCP Date Time Server
6. Write a program to transfer a File using TCP.
7. Write a program to transfer Files using UDP.
8. Write a program to simulate the sliding window protocol.
9. Study of Network Simulators like NS2 / Glomosim

S2.7 Seminar

Course No: 13

L	P	C
0	1	1

The aim of this course is to introduce the student to research, and to acquaint him/her with the process of presenting his/her work through seminars and technical reports.

The student is expected to do an extensive literature survey and analysis in an area related to computer science, chosen by him/her, under the supervision of a faculty member from the department. The study should preferably result in a critical review of the present works/design ideas/designs/algorithms/theoretical contributions in the form of theorems and proofs/new methods of proof/new techniques or heuristics with analytical studies/implementations and analysis of results.

The student should give a seminar on his/her work, during the semester, and submit a technical report.

References:

Articles from ACM / IEEE Journals / Conference Proceedings and/or equivalent documents, standard textbooks and web based material, approved by the supervisor.

Third Semester

S3.1 Advanced Data Base Management System

Course No: 14

L	P	C
4	0	4

Prerequisites/Exposure:

Objectives:

- To understand the relational model, and know how to translate requirements captured in an Entity-Relationship diagram into a relational schema.
- To reason about dependencies in a relational schema.
- To understand normal form schemas, and the decomposition process by which normal forms are obtained Use relational query languages such as SQL.
- To familiarize with advanced SQl statements.
- To understand advanced features of database technologies.

Unit I: Introduction: Purpose of Database Systems, Views of Data –Data Abstraction, Instances and Schemas, Data Independence ,Data Models – Hierarchical Data Model, Network Data Model, Relational Data Model, ER Data Model. Database Languages-DDL,

DML, Transaction Management, Storage Management, Database Administrator, Database Users, Overall System Structure. Relational Data Model-Relational Model concepts, keys, Integrity constraints--Domain Constraints, Key Constraints, Entity Integrity Constraints , Referential Integrity Constraints. ER Data Model - Basic Concepts, Constraints, Keys, Design Issues, Entity Relationship Diagram, Weak Entity Sets, Extended ER Features, Design of an ER Database Schema, Reduction of an ER Schema to Tables. Relational Algebra and Calculus: Relational Algebra-Selection and Projection, Set operations, Renaming, Joins, Division. Relational Calculus: Tuple Relational Calculus, Domain Relational Calculus. Expressive power of Algebra and Calculus.

Unit II: Relational Database Design - Anomalies in a Database – Functional Dependency – Lossless Join and Dependency- Preserving Decomposition – normalization - normal forms – First, Second and Third Normal Form – Boyce Codd Normal Form – Multivalued, Dependency – Fourth Normal Form – Join Dependency – Project Join Normal Form – Domain Key Normal Form.

Unit III: Relational Database Query Languages - Basics of QBE and SQL. Data Definition in SQL - Data types, Creation, Insertion, Viewing, Updation, Deletion of tables, Modifying the structure of the tables, Renaming, Dropping of tables. Data Constraints – I/O constraints, Primary key, foreign key, unique key constraints, ALTER TABLE command - Database Manipulation in SQL - Computations done on table data - Select command, Logical operators, Range searching, Pattern matching, Grouping data from tables in SQL, GROUP BY, HAVING clauses, Joins – Joining multiple tables, Joining a table to itself. DELETE – UPDATE - Views - Creation, Renaming the column of a view, destroys view - Program with SQL - Data types: Using set and select commands, procedural flow, if, if /else, while, goto, global variables, Security - Locks, types of locks, levels of locks. Cursors - Working with cursors, Error Handling, Developing stored procedures, create, alter and drop, passing and returning data to stored procedures, using stored procedures within queries, building user defined functions, creating and calling a scalar function, implementing triggers, creating triggers, multiple trigger interaction.

Unit IV: Transaction Management, Concurrency Control and Query Processing - Concept, Definition and States of Transactions , ACID properties – Concurrency Control, Serializability –Conflict Serializability, View Serializability, Recoverability-Recoverable Schedules, Non-cascading Schedules, Strict Schedules. Concurrency Control Schemes-Locking-Two-Phase Locking, Deadlock, Granularity, Timestamp Ordering Protocol. Basics of Query Processing.

Unit V: Object Oriented Database Management Systems: Concepts, Need for OODBMS, Composite objects, Issues in OODBMSs, Advantages and Disadvantages of OODBMSs. Distributed databases-Motivation for Distributed Databases, Distributed Database Concepts, Types of Distribution, Architecture of Distributed Databases, The Design of

Distributed Databases, Distributed Transactions, Commit Protocols for Distributed Databases.

References:

1. Elmasri and Navathe, Fundementals of Database systems, 5th Edition, Pearson, 2009.
2. Abraham Silbersehatz, Henry F. Korth and S.Sudarshan, Database system concepts,6th Edition, Tata McGraw-Hill 2010.
3. CJ Date, Introduction to Database Systems, Addison Wesley.
4. Ramakrishnan and Gehrke, Database Management Systems, 3rd Edn, Mc Graw-Hill, 2003
5. Alexis Leon, Mathews Leon, Database Management Systems, Leon Vikas.
6. Vikram Vaswani, MySQL The complete Reference,1st Edition, Tata McGraw-Hill, 2004.
7. Paul DuBois, MySQL Cookbook, 2nd Edition, O'Reilly Media, 2006

S3.2 Principles of Compilers

Course No: 15

L	P	C
4	0	4

Prerequisites/Exposure:

Objectives:

- To introduce the concept of different phases of compiler.

Unit I: Introduction to Compiling - Definition of Compiler, Translator, Interpreter, Analysis of the source program, The phases of a compiler, Compiler Construction tools- Applications of Compiler technology – programming language basics - Lexical Analysis – Role of lexical analyser – Input Buffering - Specification of tokens – Recognition of tokens using Finite Automata - Regular Expressions and Finite Automata - From NFA to DFA - Regular Expression to an NFA - Design of a Lexical Analyser Generator.

Unit II: : Syntax Analysis – Role of Parser – Error handling and recovery – definitions of Parsing, Top-down parsing and Bottom-up Parsing - Context Free Grammars – derivations - parse tree – ambiguity – associativity and precedence of operators - writing a grammar – top-down parsing – recursive descent parsing - FIRST and FOLLOW – LL (1) Grammars – Recursive Predictive parsing - Bottom Up parsing – reductions – handle pruning – Shift reduce parsing - Operator precedence parsing, Simple LR parsing.

Unit III: Intermediate code generation **–** DAG – Three Address Code – Addresses and Instructions – quadruples – triples – Static Simple-Assignment Form – Types and

Declarations – Type Expressions - Type Equivalences – Declarations – Type Checking – Rules – Type Conversion – Function and Operator Overloading – type inference and polymorphic functions **–** Control Flow – Boolean Expressions – Short Circuit Code – Flow-Control Statements – Control-Flow translation for Boolean Expressions – Break Continue and Goto Statements**.**

Unit IV: Run-time Environments – Storage Optimization – Static Vs Dynamic Allocation – Stack Allocation of Space - Activation Trees and Records – Calling Sequences – Access to Non local Data on the Stack – Data Access Without Nested Procedures – Issues with Nested Procedures – Heap Management – The Memory Manager – The Memory Hierarchy – Locality in Programs – Reducing Fragmentation - Manual Deallocation Requests.

Unit V: Code Generation – Issues in the Design of a Code Generator – The Target Language – A Simple Target Machine Model – The Program and Instruction Costs – Address in the Target Code – Static Allocation – Stack Allocation – Run-Time Address for Names – Basic Blocks and Flow Graphs – Representation of Flow Graphs. Code Optimization - The principal sources of optimization – Data Flow Analysis – Abstraction – Data Flow Analysis Schema – Data Flow Schemas on Basic Blocks – Reaching Definitions – Live Variable Analysis – Available Expressions. Region Based Analysis – Regions – Region Hierarchies for Reducible Flow Graphs – Overview of a Region Based Analysis.

References:

1. V Aho, A.,Ravi Sethi, D Ullman,J. Compilers Principles,Techniques and Tools, Pearson education,2002.
2. W Appel, Andrew, Modern Compiler Implementation in C, Cambridge University Press,1997.
3. Tremblay, Sorenson, The Theory and Practice of Compiler Writing, BSP.
4. Torben Ægidius Mogensen, Basics of Compiler Design, Department Of Computer Science, University Of Copenhagen (Online Edition).

S 2.3 Object Oriented Programming Concepts using Java

Course No: 11

L	P	C
4	0	4

Unit I: Introduction to OOPS - Basic principles of Object Orientation (Objects , Attributes and Methods, Encapsulation and Information Hiding, State Retention, Object Identity, Messages, Class Hierarchy, Inheritance, Polymorphism, Genericity) Introduction to Java - History, Versioning, the Java Virtual Machine, Byte code, Features of Java, Language Components - Primitive Data Types, Comments, Keywords, literals, variables scope & declarations, Control structures - The for Statement, The if Statement, The while and do while Statements, The switch Statement, The break

Statement, The continue Statement, Operators - Casts and Conversions, Arrays.

Unit II: Object-Oriented Programming – Classes - Class Fundamentals - Declaring Objects - new operator – methods – parameter passing – Constructors - Parameterized Constructors - The this Keyword – finalize method. Overloading Methods and constructors, Access Controls, static and final, Nested and Inner Classes. Inheritance - extends, Member access and inheritance, super keyword, Polymorphism - Method Overriding, Dynamic Method Dispatch, Abstract Classes, Packages and interfaces.

Unit III: Exceptions, Threads & IO in Java - The File and Standard Streams, Stream classes and interfaces, Using Byte Streams and Character Streams, Threads - Threads vs. Processes, Creating Threads, Runnable interface, Thread Class, Inter thread communication, Synchronization. Exceptions - Basic of java Exception Handling, Hierarchy, Developing user defined Exception Classes.

Unit IV: Applets, AWT & Swing - Applet class, Types of applet, skeleton, Applet tag, passing parameters, Event Handling, Delegation event model, Event classes, Listeners, AWT classes and window fundamentals, Frames, Working with fonts, graphics and colors, AWT controls, layouts and Menus, Dialogue Boxes. Swings - Japplets, icon, labels, Buttons, Textbox, combo box, Tables and Panes.

Unit V: Database and Sockets – JDBC - introduction, architecture, Drivers, connections, statements, resultset and Meta data. Sockets: Introduction to networking, InetAddress, url, socket, server sockets, Datagrams.

Introduction to Unified Modeling Language, UML diagrams, Class diagrams, Object interaction diagrams, State and Activity diagrams, Component diagrams, Deployment diagrams. Introduction to Analysis Object Oriented System Analysis, Design and implementations.

References:

1. Herbert Scheldt, Java Complete Reference, Tata McGraw Hill edition.
2. E Balaguruswamy, Programming in Java.
3. David Flanagan, Jim Farley, William Crawford & Kris Mangnusson Java Enterprise in a nutshell, ,OReilly.
4. Grady Booch, James Rumbaugh, Ivar Jacobson, The Unified Modeling Language User Guide (2nd Edition).

S3.4a Pattern Recognition (Elective II)

Course No: 17a

Prerequisites/Exposure:

L	P	C
4	0	4

Objectives:

- To understand the concept of a pattern and the basic approach to the development of pattern recognition algorithms.
- To understand and apply methods for preprocessing, feature extraction, and feature selection to multivariate data.
- To understand supervised and unsupervised classification methods to detect and characterize patterns in real-world data.

Unit I: Introduction - introduction to statistical - syntactic and descriptive approaches - features and feature extraction - learning - Bayes Decision theory - introduction - continuous case - 2- category classification - minimum error rate classification - classifiers - discriminant functions - and decision surfaces – error probabilities and integrals - normal density - discriminant functions for normal density.

Unit II: Parameter estimation and supervised learning - maximum likelihood estimation - the Bayes classifier - learning the mean of a normal density - general Bayesian learning – nonparametric technique – density estimation - parzen windows - k-nearest neighbour estimation - estimation of posterior probabilities - nearest-neighbour rule - k-nearest neighbour rule.

Unit III: Linear discriminant functions - linear discriminant functions and decision surfaces – generalized linear discriminant functions - 2-category linearly separable case - non-separable behaviour - linear programming algorithms, support vector machines - multilayer neural networks – feedforward operation and classification, backpropagation algorithm, error surface, backpropagation as feature mapping.

Unit IV: Syntactic methods – stochastic search- Boltzmann learning – Nonmetric methods - decision trees – CART – other tree methods, grammatical methods, grammatical inference.

Unit V: Unsupervised learning and clustering – mixture densities and identifiability, maximum likelihood estimates, applications to normal mixtures, unsupervised Bayesian learning, data description and clustering.

References:

1. R.O.Duda, P.E.Hart and D.G.Stork, Pattern Classification, John Wiley, Second edition, 2006
2. Gonzalez R.C. & Thomson M.G., Syntactic Pattern Recognition - An Introduction,

Addison Wesley.
3. Fu K.S., Syntactic Pattern Recognition And Applications, Prentice Hall, Eaglewood cliffs
4. Rajan Shinghal, Pattern Recognition: Techniques and Applications, Oxford University Press, 2008.

S3.4b Wireless and Mobile Networks (Elective II)

Course No: 17b

L	P	C
4	0	4

Prerequisites/Exposure:

Objectives:

- To understand the fundamental concepts of wireless and mobile networks.
- To learn the basics of Wireless voice and data communications technologies.
- To build working knowledge on various telephone and satellite networks.
- To build skills in working with Wireless application Protocols to develop mobile content applications.
- To understand about the security aspects of Wireless Networks.
- To learn about Wireless Application Protocol and mobile operating systems.

Unit I: Introduction - Applications - Brief History of wireless communication – Open Research Problems – Wireless Transmission – Frequencies for Radio Transmission – Signals – Antennas – Signal Propagation – Multiplexing – Modulation – Spread Spectrum – Cellular Systems – Medium Access Control – Motivation – SDMA – FDMA – TDMA – CDMA – Comparison.
Unit II: Different Generations of Wireless Cellular Networks - 1G, 2G, 2.5G, 3G, 4G. Telecommunication Systems – GSM – DECT – TETRA – UMTS – IMT-2000. Wireless LAN – Infrared Vs Radio Transmission – Infrastructure Vs Adhoc Networks – IEEE 802.11 – HIPERLAN – Bluetooth.

Unit III: Mobile Network Layer - Mobile IP – Dynamic Host Configuration Protocol - Routing – DSDV – DSR – Alternative Metrics. Transport and Application Layers - Traditional TCP – Classical TCP improvements – WAP, WAP 2.0.

Unit IV: Wireless Network Security – IEEE 80211i Security – Wireless Transport Layer Security – Sessions and Connections – Protocol Architecture – WAP End-to-End Security.

Unit V: Java for Wireless Devices - Setting up the development environment - Basic Data types, Libraries (CLDC, MIDP) - UI Controls - Displayable and Display Image - Events and Event Handling - List and choice - Text box - Alerts - Persistent Storage - Record Stores – Records - Record Enumeration - Network MIDlets - The Connection Framework - Connection Interface - Making a connection using HTTP - Using datagram connection.

References:

1. Jochen Schiller, Mobile Communications, Pearson Education, 2nd Edition
2. Raj Kamal, Mobile Computing, Oxford Higher Education, 2007
3. William Stallings, Network Security Essentials Applications and Standards, Fourth Edition, Pearson, 2012.
4. Yu Feng and Dr Jun Zhu, Wireless Java Programming with J2ME, Techmedia Publications, 1st edition
5. William Stallings, Wireless Communications and Networks, Pearson Education Asia, 2002
6. Jochen Burkhardt, Dr. Horst Henn, Stefan Hepper, Klaus Rintdorff and Thomas Schack, Pervasive Computing Technology and Architecture of Mobile Internet Applications, Pearson Education, 2002.
7. Nishit Narang and Sumit Kasera, Mobile Networks GSM and HSCSD, Tata McGraw Hill
8. Asoke K Talukdar and Roopa R. Yavagal Mobile Computing, TataMcGrawHill

S3.4c Cryptography & Network Security (Elective II)

Course No: 17c

Prerequisites/Exposure:

L	P	C
4	0	4

Objectives: To be familiar with classical and modern encryption and decryption techniques and apply in the security system.

Unit I: Computer Security Concepts – Challenges – Security Attacks – Security Services – Security Mechanisms – A Model for Network Security. Cryptography – Symmetric Encryption Principles – Cryptography – Cryptanalysis – Feistal Cipher Strucuture. Symmetric Block Encryption Algorithms - DES – Triple DES – AES – Random and Pseudorandom Numbers – Stream Cipher and RC4 – Cipher Block Modes of Operation.

Unit II: Message Authentication – Approaches – MAC – One way Hash Function – Secure Hash Functions – Message Authentication Codes. Public Key Cryptography Principles – Algorithms – Digital Signatures.

Unit III: Network Security Applications – Symmetric Key Distributions using Symmetric Encryption – Kerberos Version 4 - Key Distributions using Asymmetric Encryption – X.509 Certificates - Public Key Infrastructure – Federated Identity Management. Transport Level Security – Web Security Considerations – Secure Socket Layer and Transport Layer Security – SSL Architecture – SSL Record Protocol – Change Cipher Spec Protocol – Handshake Protocol. Transport Layer Security - HTTPS – SSH. IP Security – Overview – Policy – Encapsulating Security Payload – Combining Security Associations – Internet Key Exchange.

Unit V: Intruders - Intruders, Intrusion Detection, Password management. Maclicious Software – Types, Viruses, Countermeasures, Worms, DDoS. Firewalls – Need –

Characteristics, Types, Firewall Basing, Location and Configuration – DMZ Networks, VPNs – Distributed Firewalls.

References:

1. William Stallings, Network Security Essentials Applications and Standards, Fourth Edition, Pearson.
2. William Stallings, Cryptography and Network Security, Fourth Edition, Prentice Hall, 2007.
3. Atul Kahate, Cryptography and Network Security, Tata McGraw-Hills, 2006.
4. Eric Maiwald, Information Security Series, Fundamental of Network Security, Dreamtech press, 2004.
5. Charlie Kaufman, Radia Perlman, Mike Speciner, Network Security: Private Communication in Public World, Prentice Hall, India, 2002.

S3.4d Advanced Web Technology (Elective II)

Course No: 17d

Prerequisites/Exposure:

L	P	C
4	0	4

Objectives: To be familiar with the design and development process for distributed system.

Unit I: Web 2.0 - Defnition, Characteristics, key features, client side technologies (Ajax and JavaScript frameworks - YUI Library, Dojo Toolkit, MooTools, jQuery, Ext JS and Prototype JavaScript Framework), server side technologies (PHP, Ruby, Perl, Python, Enterprise Java J2EE and Microsoft.NET Framework), Concepts (Rich Internet Application — Web-Oriented Architecture — Social Web), SLATES.

Unit II: Fundamentals of Web Services - Definition, Components, benefits, behavioral characteristics. Web Services Architecture - Web Service Roles, Web Service Protocol Stack, Service Transport. Web Services Components - XML-RPC, SOAP, WSDL, UDDI. Web Services Security (notions) - Confidentiality (XML-RPC and SOAP run on top of HTTP - support for Secure Socktes Layer (SSL) for HTTP - Encrypted Communication via SSL), Authentication (HTTP's built-in support for Basic and Digest authentication - SOAP Security Extensions - Digital Signature SOAP-DSIG - SAML).

Unit III: Introduction to Python – Installation – Python Interpreter – usage and customization – Editor setup – Variables, Expressions and Statements – Conditionals – Functions. Strings – Lists – List Comprehensions – Stacks – Queues – Tuples – Sequences – Sets – Dictionaries – Sets - Modules, I/O And Exception Handling - Modules – Search path – Compiled modules – Standard modules – Packages – Input and Output functions – Files – read and write – Exception – Handling and Raising – User defined Exceptions.

Unit IV: Server side programming using Python - Server side scripting - CGI - role of Web server – Apache Web Server – Python Server Side Script – Developing Python Server Side Pages (PSP) – capturing form data – validation – processing data – exchange of data between form and server.

Unit V: Python - SQLite integration - Features of SQLite, data types, Introduction to SQL commands - SELECT, DELETE, UPDATE, INSERT. Python functions for SQLite operations – database connection, database and table creation, selection, query, fetching results - Insertion and Deletion of data using Python - Displaying data from SQLite in webpage. Case study - Server MVC design pattern – Django/Plone (Choose any one of these).

References:

1. http://en.wikipedia.org/wiki/Web_2.0
2. http://www.tutorialspoint.com/webservices/
3. S. V. Subrahmanya and B. V. Kumar, Web Services: An Introduction, Tata McGraw-Hill.
4. Ron schmelzer et al, XML and Web Services, Pearson Education, 2002.
5. Sandeep Chatterjee and James Webber, Developing Enterprise Web Services: An Architect's Guide, Prentice Hall, 2004.
6. XML and Web Services, Ron schmelzer et al, Pearson Education, 2002.
7. The Python Tutorial available at http://docs.python.org/3.3/tutorial/
8. Peter Wentworth Jeffrey Elkner, Allen B. Downey, and Chris
9. Meyers How to Think Like a Computer Scientist: Learning with Python (3nd edition) Online Version: http://openbookproject.net/thinkcs/python/english3e/
10. Python Documentation available at http://www.python.org/doc/
11. Swarooph CH, A Byte of Python. Available at http://swaroopch.com/notes/python/
12. Wesley J Chun, Core Python Programming, Second Edition, Pearson.

S3.4e Virtualisation And Cloud Computing (Elective II)

Course No:17e

L	P	C
4	0	4

Prerequisites/Exposure:

Objectives:

- Understand the technical capabilities and business benefits of virtualization and cloud computing and how to measure these benefits.
- Describe the landscape of different types of virtualization and understand the different types of clouds.
- Illustrate how key application features can be delivered on virtual infrastructures.
- Explain typical steps that lead to the successful adoption of virtualization

technologies.

- Understand the similarities and difference between cloud computing and outsourcing.

Unit I: Introduction - Evolution of Cloud Computing – System Models for Distributed and Cloud Computing – NIST Cloud Computing Reference Architecture – Infrastructure as a Service (IaaS) – Resource Virtualization – Platform as a Service (PaaS) – Cloud platform & Management – Software as a Service (SaaS) – Available Service Providers.

Unit II: Virtualization - Basics of virtualization - Types of Virtualization - Implementation Levels of Virtualization - Virtualization Structures - Tools and Mechanisms - Virtualization of CPU, Memory, I/O Devices - Desktop virtualization – Server Virtualization – Linux KVM, Xen, Qemu, LXC, OpenVZ.

Unit III: Cloud Infrastructure - FOSS Cloud Software Environments - Eucalyptus, Open nebula, OpenStack – OpenStack Architecture – Compute, Object Storage, Image Service, Identity, Dashboard, Networking, Block Storage, Metering, Basic Cloud Orchestration and Service Definition.

Unit IV: Programming Model - Parallel and Distributed programming Paradigms – MapReduce, Twister and Iterative MapReduce – Mapping Applications - Programming Support – Apache Hadoop – HDFS, Hadoop I/O, Hadoop configuration, MapReduce on Hadoop.

Unit V: Security in The Cloud - Security Overview – Cloud Security Challenges – Software-as-a-Service Security – Security Governance – Risk Management – Security Monitoring – Security Architecture Design – Data Security – Application Security – Virtual Machine Security – Qubes – Desktop security through Virtualization.

References:

1. Kai Hwang, Geoffrey C Fox, Jack G Dongarra, Distributed and Cloud Computing, From Parallel Processing to the Internet of Things, Morgan Kaufmann Publishers, 2012.
2. John W. Rittinghouse and James F.Ransome, Cloud Computing: Implementation, Management, and Security, CRC Press, 2010.
3. Toby Velte, Anthony Velte, Robert Elsenpeter, Cloud Computing, A Practical Approach, TMH, 2013.
4. George Reese, Cloud Application Architectures: Building Applications and Infrastructure in the Cloud: Transactional Systems for EC2 and Beyond (Theory in Practice (O'Reilly), O'Reilly
5. James E. Smith, Ravi Nair, Virtual Machines: Versatile Platforms for Systems and Processes, Elsevier/Morgan Kaufmann, 2005.
6. Katarina Stanoevska - Slabeva, Thomas Wozniak, Santi Ristol, Grid and Cloud

Computing – A Business Perspective on Technology and Applications", Springer.
7. http://docs.openstack.org/ops/ – Open stack Operations Guide.
8. Tom White, Hadoop: The Definitive Guide, O'Reilly Media, 2009.

S3.4f Data Warehousing and Data Mining (Elective II)

Course No: 17f

Objectives:

- To provide the fundamentals on information retrieval and data mining techniques and focus on practical algorithms of textual document indexing, relevance ranking, web usage mining, text analytics, as well as their performance evaluations. , that lays foundations for the Data Analytics.
- To give an exposure to the fundamentals of Data Analytics.

Unit I: Data Warehouse – Definition – Operational Database Systems Vs Data Warehouses – Multidimensional Model – From Tables and Spreadsheets to Data Cubes – Schemas for Multidimensional Databases – Measures – Concept Hierarchies - OLAP Operations in the Multidimensional Data Model - DataWarehouse Architecture.

Unit II: Data Mining – Introduction – Definition - Data Mining Functionalities – Major Issues in Data Mining - Data Preprocessing – Data Cleaning – Data Integration and Transformation – Data Reduction – Data Discretization and Concept Hierarchy Generation. Association Rule Mining - Efficient and Scalable Frequent Item set Mining Methods – Mining Various Kinds of Association Rules – Association Mining to Correlation Analysis – Constraint-Based Association Mining.

Unit III: Classification and Prediction - Issues Regarding Classification and Prediction – Classification by Decision Tree Introduction – Bayesian Classification – Rule Based Classification – Classification by Back propagation – Support Vector Machines – Associative Classification – Lazy Learners – Other Classification Methods – Prediction – Accuracy and Error Measures – Evaluating the Accuracy of a Classifier or Predictor – Ensemble Methods – Model Section..

Unit IV: Cluster Analysis - Types of Data in Cluster Analysis – A Categorization of Major Clustering Methods – Partitioning Methods – Hierarchical methods – Density-Based Methods – Grid-Based Methods – Model-Based Clustering Methods – Clustering High-Dimensional Data – Constraint-Based Cluster Analysis – Outlier Analysis.

Unit V: Graph Mining - Mining Object, Spatial, Multimedia, Text and Web Data - Multidimensional Analysis and Descriptive Mining of Complex Data Objects – Spatial Data Mining – Multimedia Data Mining – Text Mining – Mining the World Wide Web.

References:

1. Jiawei Han and Micheline Kamber, Data Mining Concepts and Techniques, Second Edition, Elsevier, Reprinted 2008.
2. Alex Berson and Stephen J. Smith, Data Warehousing, Data Mining & OLAP, Tata McGraw – Hill Edition, Tenth Reprint 2007.
3. K.P. Soman, Shyam Diwakar and V. Ajay, Insight into Data mining Theory and Practice, Easter Economy Edition, Prentice Hall of India, 2006.
4. G. K. Gupta, Introduction to Data Mining with Case Studies, Easter Economy Edition, Prentice Hall of India, 2006.
5. Pang-Ning Tan, Michael Steinbach and Vipin Kumar, Introduction to Data Mining, Pearson Education, 2007.

S3.5a Data Compression (Elective III)

Course No: 18a

Prerequisites/Exposure:

L	P	C
4	0	4

Objectives:

- To understand the physical significance of some basic concepts of information theory including entropy, average mutual information and the rate distortion bound.
- To learn the design of entropy codes including Huffman codes, and arithmetic coding.
- To understand the operation of lossless compression schemes.
- To understand the operation of popular lossy compression schemes including delta modulation, differential pulse code modulation, transform coding, and vector quantization.

Unit I: Introduction to data compression - Basic Techniques - Runlength encoding, RLe Text compression, RLE image compression, Move-to-front coding, Scalar quantization.Statistical Methods - Information theory concepts, variable sixe codes, prefix codes, Shanon fanon coding, Huffman coding, Adaptive Huffman, Arithmetic coding.

Unit II: Dictionary methods - string compression, LZ77 sliding window, MZW, GIF images. Image Compression - Approaches to image compression, intuitive methods, image transform, test images, JPEG, Progressive image compression, Vector quantization.

Unit III: Wavelet Methods- Fourier transform, frequency domain, Fourier image compression, CWT and inverse CWT, Haar transform, filter bank, DWT, JPEG 2000. Video compression - analog video, Composite and component video, digital video, video compression, MPEG.

Unit IV: Audio Compression - Sound, digital audio, human auditory system, MPEG-1 audio layer. Fractal based compression - IFS. Comparison of compression algorithms. Implementation of compression algorithms.

References:

1. David Solomon, Data compression: The Complete Reference, 2nd edition, Springer-Verlag, New York. 2000.
2. Stephen Welstead, Fractal and Wavelet Image Compression Techniques, PHI, NewDelhi-1, 1999.
3. Khalid Sayood, Introduction to Data compression, Morgan Kaufmann Publishers, 2003 reprint

S3.5b Pervasive Computing (Elective III)

Course No: 18b

Prerequisites/Exposure:

L	P	C
4	0	4

Objectives:

- To provide a sound conceptual foundation in the area of Pervasive Computing aspects.
- To provide the students the ability to conceptualize, analyze and design select classes of pervasive computing systems.

Unit I: Introduction to Pervasive Computing - Past, present, future - the pervasive computing market, m-Business, Challenges and future of Pervasive Computing. Application Examples of Pervasive Computing: Retail, Airline Check-in and booking, Sales force automation, Healthcare, Tracking, Car Information Systems, Email Access via WAP and voice.

Unit II: Device Technology for Pervasive Computing - Hardware, Human-machine interfaces, Biometrics, Operating Systems, Java for pervasive devices, Outlook. Device Connectivity - Protocols, Security, Device Management.

Unit III: Web application concepts for pervasive computing - History, WWW architecture, Protocols, Trans-coding, Client Authentication via the Internet for pervasive computing. WAP and beyond - Introduction, Components of the WAP architecture, WAP infrastructure, WAP security issues, Wireless Markup Language, WAP push, Products, i-Mode, Outlook.

Unit IV: Web Voice Technology - Basics of Speech Recognition, Voice standards, Speech Applications, Speech and Pervasive Computing, Security Personal Digital Assistants - History, Device Categories, Personal Digital Assistant Operating Systems, Device Characteristics, Software Components, Standards, Mobile applications, Personal Digital Assistant Browsers. Server-side programming (Java) for pervasive computing -

Java 2 Enterprise Edition (Overview), Servlets, Enterprise Java Beans, Java Server Pages, Extensible Markup Language, Web Services, Model-View-Controller pattern.

Unit V: Pervasive Web application architecture - Background, Scalability & Availability - Development of pervasive computing Web Applications, Pervasive Application Architecture - Example Pervasive Application - Introduction, User Interface Overview, Architecture, Implementation. Access from PCs - Smart-card authentication via the Internet, Ordering goods. Access via WAP - WAP functionality, Implementation - Access from Personal Digital Assistants - Extending the example application to personal digital assistants, Implementation for synchronized devices, Implementation for intermittently connected devices, Implementation for connected devices - Access via Voice: Extending the example application to voice access, Implementation.

References:

1. Jochen Burkhardt, Horst Henn, Stefan Hepper, Thomas Schaec & Klaus Rindtorff, Pervasive Computing: Technology and Architecture of Mobile Internet Applications, Pearson Education, New Delhi, 2006.
2. Stefen Poslad, Ubiquitous Computing: Smart Devices, Environments and Interactions, Wiley, Student Edition, 2010.
3. Genco, S. Sorce, Pervasive Systems and Ubiquitous Computing, WIT Press, 2012.
4. Ajith Abraham (Ed.): Pervasive Computing, Springer-Verlag, 2012.
5. Guruduth S. Banavar, Norman H. Cohen, Chandra Narayanaswami, Pervasive Computing: An Application-Based Approach, Wiley Interscience, 2012.
6. Frank Adelstein, S K S Gupta, GG Richard & L Schwiebert: Fundamentals of Mobile and Pervasive Computing, Tata McGraw-Hill, New Delhi, 2005.

S3.5c System Security (Elective III)

Course No: 18c

L	P	C
4	0	4

Objectives:

An understanding of the differences between various forms of computer security, where they arise, and appropriate tools to achieve them.

Unit I: Notion of different types of securities - Information Security - Computer Security - Security Goals, Relation between Security, Confidentiality, Integrity, Availability and Authorization, Vulnerabilities - Principles of Adequate protection. Notions of Operating security, Database security, Program security, Network Security. Attacks - Threats, Vulnerabilities and controls. The kind of problems- Interception, Interruption, Modification, Fabrication. Computer Criminals - Amateurs, Crackers, Career Criminals. Methods of Defense - Control, Hardware Controls, Software Controls, Effectiveness of Controls.

Unit II: Program Security - Secure programs - Fixing Faults, Unexpected Behaviour, Types of Flaws. Non-malicious program errors - Buffer overflows, Incomplete Mediation.

Viruses and other malicious code - Why worry about Malicious Code, Kinds of malicious code, How viruses attach, How viruses gain control, Prevention, Control Example - The Brain virus, The Internet Worm, Web bugs. Targeted malicious code - Trapdoors, Salami Attack. Controls against program threats - Development Controls, Peer reviews, Hazard Analysis.

Unit III: Operating System Security - Protected objects and methods of protection - Memory address protection - Fence, Relocation, Base/Bounds Registers, Tagged Architecture, Segmentation, Paging. Control of access to general objects - Directory, Access Control List. File protection mechanism – Basics forms of Protection, Single Permissions. Authentication - Authentication basics, Password, Authentication Process Challenge-response, Biometrics. Trusted Operating systems - Security Policies for Operating Systems, Models of Security - Requirement of security systems, Multilevel Security, Access Security, Limitations of Security Systems. Trusted Operating System Design - Elements, security features, assurance, system flaws and assurance methods.

Unit IV: Database Security - Security requirements - Integrity of Database, Confidentiality and Availablity, Reliability and integrity, Sensitive data, Interface, Multilevel database, Proposals for multilevel security.

Unit V: Administrating Security - Security planning - Contents of a security Planning, Team members, commitment to a security plan, Business continuity Plans. Risk analysis - The nature of risk, steps of risk analysis. Arguments for and against risk analysis, Organizational security policies - Purpose and goals of Organizational Security. Audience, Characteristics of a Good Security Policy. Nature of security Policies- Data sensitivity policy, Government Agency IT security policy. Physical security- Natural Disaster, Human Vandals, Interception of Sensitive Information.

References:

1. C. P. Pfleeger, and S. L. Pfleeger, Security in Computing, Pearson Education.
2. Matt Bishop, Computer Security: Art and Science, Pearson Education.
3. William Stallings, Cryptography and Network Security, Fourth Edition, Prentice Hall, 2007.
4. Michael E. Whitman and Herbert J. Mattord, Principles of Information Security, Thomson.

S4.1d Molecular Modeling and Simulation (Elective III)

Course No: 18d

L	P	C
4	0	4

Objectives:

1. To understand application of simulation techniques to study molecular dynamics

and derive properties.
2. To learn and apply the statistical approaches and models for phylogenetic analysis and tree reconstruction.
3. To understand the basis and nature of protein-protein interactions.
4. To understand principles of docking simulations.

Unit I: Overview of molecular modeling - Molecular modeling methods - Semi-empirical method and empirical method. Model Type - static, dynamic and probabilistic models. Models of growth and decay.

Unit II: System Modeling - Concept, Principles of Mathematical modeling, static physical model, stochastic activities, continuous and discrete simulation. Discrete system simulation - Probability concepts in simulation, random number generations and their testing, stochastic variable generation, Model Execution - Event driven versus Time driven.

Unit III: Computational Gene Mapping - Genetic mapping, gene expression, gene prediction methods, gene prediction tools, mutational analysis, introduction to restriction mapping and map assembly, mapping with restriction fragment fingerprints, Lander-Waterman statistics. Software Packages for Phylogenetic Analysis - PHYLogeny Inference Package (Phylip), Phylogenetic Analysis using Parsimony (PAUP) and Phylogenetic Analysis by Maximum Likelihood (PAML). Microarray Technology - Techniques for Microarray Data Analysis, Microarray Databases. Scatter Plots, Principal Component Analysis, Cluster Analysis, Applications of Microarray Technology.

Unit IV: Structural Modeling - Use of sequence patterns for protein structure prediction. Prediction of protein secondary structure from the amino acid sequences. Prediction of three dimensional protein structures. Protein structure classification - Two major classification schemes - CATH and SCOP. Protein structure prediction – Steps Unit IV: Structural Modeling - Use of sequence patterns for protein structure prediction. Prediction of protein secondary structure from the amino acid sequences. Prediction of three dimensional protein structures. Protein structure classification - Two major classification schemes - CATH and SCOP. Protein structure prediction - Steps involved in homology modeling. Protein-Protein Interactions - Prediction methods for Protein-Protein interactions - Protein-protein interaction Databases - Computer Assisted Drug Design (CADD) - Protein based drug design cycle, Drug discovery pipeline.

Unit V: Molecular Visualization - Visualization of protein structure, Methods of studying proteins, Proteomics databases, Protein family databases, PDB file format. Software tools for 3D molecular graphic visualization - Rasmol - basic operations and steps in Rasmol to visualize the molecule, advantages of Rasmol, advantages of Swiss-PdbViewer. Docking Simulations - Rigid docking and Flexible docking.

Text Books:

1. Stephen Misener and Stephen A. Krawetz , Bioinformatics: Methods and rotocols, Publisher: Humana Press, 2000.
2. Gordan, Simulation and Modeling, PHI.
3. Tamar Schlick, Molecular Modeling and Simulation: An Interdisciplinary Guide, Springer, 2001
4. Narsingh Dev, System simulation and modeling, PHI
5. Leach, Andrew, Molecular Modelling: Principles and Applications, Prentice Hall. 2001.
6. Prakash S Lohar, Bioinformatics –MJP publishers, Chennai.

References:

1. Sharma, Munjal and Shanker, A text book of Bioinformatics, RASTOGI publications- New Delhi.
2. Des Higgins (Ed), Willie Taylor (Ed), Bioinformatics: Sequence, Structure and Databanks - A Practical Approach, Publisher: Oxford University Press, 2000.

S3.5e Fundamentals of Big Data (Elective III)

Course No: 18e

L	P	C
4	0	4

Prerequisites/Exposure:

Objectives:

- To cover the basics of big data
- To familiarize with big data technology and tools

Unit I: Introduction to Big Data – definition & importance of Big Data - four dimensions of Big Data - volume, velocity, variety, veracity – industry examples – terminologies – structured data, unstructured data, semi structured data, streaming data, real-time data, meta data, data at rest – relational databases and SQL – Non-Relational databases - big data sources that can change one's business - Integrating Big Data with traditional data - The role of the Data Scientist - Big Data Analytics in Industry Verticals.

Unit II: Key roles for a successful analytic project - Main phases of the lifecycle - Developing core deliverables for stakeholders.

Unit III: Big Data analytics – Introduction – Concepts - Storing Big Data - Analyzing your data characteristics - Selecting data sources for analysis - Eliminating redundant data - Open source technology for Big Data analytics - Predictive analytics - Crowdsourcing analytics - Computing platforms, limitations, and emerging technologies - Consumption of analytics - Modern analytic approaches - ensemble

modeling, commodity models, and text analysis.

Unit IV: Introduction to MapReduce/Hadoop for analyzing unstructured data - design patterns – Filtering Patterns - Join Patterns - Meta Patterns - Hadoop ecosystem of tools – In database Analytics - MADlib and Advanced SQL Techniques, NoSQL, JSON store, MDX.

Unit V: Introduction to learning and knowledge analytics - Rise of Big Data - Big Data From Technology Perspective – Hadoop - Components of Hadoop, Application Development in Hadoop , The Distributed File System - HDFS, GPFS, Hadoop Cluster Architecture, Batch Processing - Low Latency NoSQL.

References:

1. Michael Minelli, Michele Chambers and, Ambiga Dhiraj, Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses.
2. Noreen Burlingame, Little Book of Big Data, Ed. 2012
3. Tom White, Hadoop, The definitive guide, O'Reilly Media, 2010
4. Hurwitz, Alan Nugent, Fern Halper, Marcia Kaufman, Big Data For Dummies.
5. Faraz Rabbani, Ali Roghani, Big Data Analytics For Beginners.
6. Alex Holmes, Hadoop in practice, Manning Publications, 2012
7. Donald Miner, Map Reduce Design Patterns: Building Effective Algorithms and Analytics for Hadoop and Other Systems, O'Reilly Media, 2012
8. Nathan Marz , Big Data: Principles and best practices of scalable real-time data systems, Manning Publications, 2012
9. Big Data Now: Current Perspectives, O'Reilly Radar [kindle Edition], 2011.
10. Paul Zikopoulos et al., Harness the Power of Big Data The IBM Big Data Platform, McGraw-Hill, 2013
11. Bill Franks, Taming The Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics
12. Thomas H. Davenport, Big Data at Work: Dispelling the Myths, Uncovering the Opportunities
13. Foster Provost, Tom Fawcett, Data Science for Business: What you need to know about data mining and data-analytic thinking 1st edition, ISBN-13: 978-1449361327
14. Viktor Mayer-Schönberger, Kenneth Cukier, Big Data: A Revolution That Will Transform How We Live, Work, and Think

S3.5f Web Engineering (Elective III)

Course No: 18f

Prerequisites/Exposure:

L	P	C
4	0	4

Objectives:

- To understand the concepts, principles, strategies, and methodologies of web applications development.

- To understand and apply web development processes.

Unit I: Web Engineering (WE) – Introduction – Motivation – Categories & Characteristics of Web Applications – Product related, usage related and development related – Evolution of WE.

Unit II: Requirements Engineering (RE) for Web Applications – Introduction – Fundamentals – Where do requirements come from? – RE activities – RE specifications in WE - RE Principles for Web Applications – Adapting RE Methods for Web Applications Development – Requirement Types , Notations, Tools.

Unit III: Web Application Architecture – Introduction – Fundamentals – Definition of Architecture – Developing and Characterising Architectures – Components of a generic web application architecture – Layered Architecture – Database Centric Architecture - Architecture for Web Document Management – Architecture for Multimedia Data.

Unit IV: Modeling Web Applications – Introduction – Modelling Speicifics in WE – Levels – Aspects – Phases of Customizations – Modelling Requirements – Hypertext Modelling - Hypertext Strucuture Modelling Concepts – Access Modelling Concepts. Web Application Design – Web Design from an evolutionary perspective – Information Design – Software Design – Merging Information Design & Software Design – Problems and Restrictions in Integrated Web Design – A proposed Strucutural Approach – Presentation Design – Presentation of Nodes and Meshes – Device independent Development – Approaches – Interaction Design – User Interaction – User Interface Organization – Navigation Design – Deigning a link Representation – Designing Link Internals – Navigation and Orientation – Strucutural Dialog for Complex Activities – Interplay with Technology and Architecuture – Functional Design.

Unit V: Testing Web Applications – Introduction – Fundamentals – Terminology – Quality Characteristics – Test Objectives – Test Levels – Role of Tester – Test Specifics in WE – Test Approaches – Conventional, Agile - Test Schemes – Three Test Dimensions – Applying the Scheme to Web Applications – Test Methods and Techniques – Link Testing – Browser Testing – Usability Testing – Load, Stress and Continues Testing – Testing Security – Test-Driven Development. Web Project Development – Scope – Refining Frame Work Activities – Building an WebE team - Risk Management – Making Schedule – Managing Quality, Change – Project Tracking.

Text Books:

1. Gerti Kappel and Birgit Proll, Web Engineering, John Wiley and Sons Ltd, 2000.
2. Roger S Pressman and David Lowe, Web Engineering, Tata Macgraw Hill Publications, 2007.
3. Guy W Leeky-Thompson, Web Engineering, Cenagage Learning, 2008

References:

1. Moller, An Introduction to XML and Web Technologies, Pearson Education, New Delhi, 2009.
2. Chrits Bates, Web Programming: Building Internet Applications, Third Edition, Wiley India Edition, 2007.
3. John Pual Mueller, Web Development with Microsoft Visual Studio 2005, Wiley Dreamtech, 2006.

S3.6 Practical 3 (S3.1 & S3.3)

L	P	C
0	4	4

Unit I : Advanced DBMS

1. Creating database tables and using data types (Create table, Modify table, Drop table).
2. Data Manipulation (Adding data with Insert, Modify data with Update, Deleting records with Delete)
3. Implementing the Constraints (NULL and NOT NULL, Primary Key and Foreign Key Constraint, Unique, Check and Default Constraint).
4. Retrieving Data Using SELECT (Simple select, Where, IN, BETWEEN, Ordered By, Distinct and Group By).
5. Aggregate Functions (AVG, COUNT, MAX, MIN, SUM).
6. String functions.
7. Date and Time Functions.
8. Use of union, intersection, set difference.
9. Implement Nested Queries & JOIN operation.
10. Performing different operations on a view.
12. Implementing use of triggers, cursors & procedures.

Unit II : Java

1. Simple Java programs like computing formulas expressions etc.
2. Programs involving loops and decisions like generating Fibonacci, prime, strange series.
3. Programs involving arrays.
4. Programs involving class and objects.
5. Illustrate method overloading.
6. Illustrate single level inheritance.
7. Illustrate multiple inheritance using interface.
8. String sorting, pattern matching etc.
9. Illustrate threads and thread priorities.
10. Illustrate the use of Packages.
11. Exception handling (user-defined).

12. Abstract class.
13. Method overriding.
14. Illustrate usage of Applets like moving ball, face etc.
15. Create an AWT application for a simple calculator.
16. Frame application to illustrate the window events.
17. Frame application to illustrate mouse and keyboard event handling.
18. Swing applications.
19. Create a JDBC application to add the details of a student into a table.
20. Socket Programming.

S4.1a Digital Image Processing (Elective IV)

Course No: 19a

Prerequisites/Exposure:

L	P	C
4	0	4

Objective: To be familiar with processing of the images, recognition of the pattern and their applications.

Unit I: Introduction - digital image representation - fundamental steps in image processing - elements of digital image processing systems - digital image fundamentals - elements of visual perception – a simple image model – sampling and quantization - basic relationship between pixels – image geometry.

Unit II: Image transforms - introduction to Fourier transform - discrete Fourier transform (DFT) - properties DFT- other separable image transforms - Walsh, Hadamard and Discrete Cosine transforms. Hotelling transform.

Unit III: Image enhancement - basic grey level transformation - histogram equalization – image subtraction - Image averaging - spatial filtering - smoothing, sharpening filters – Laplacian filters. Enhancement in the frequency domain – frequency domain filters - smoothing, sharpening filters - homomorphic filtering.

Unit IV: Image restoration - model of Image degradation/restoration process - noise models – inverse filtering - least mean square filtering - constrained least mean square filtering. Edge detection - thresholding - region based segmentation - Boundary representation.

Unit V: Image compression - fundamental concepts of image compression - compression models - information theoretic perspective. Lossless compression - Huffman coding - arithmetic coding - bit plane coding - run length coding. Lossy compression - transform coding – Image compression standards.

References:

1. R.C. Gonzalez and R.E. Woods, Digital Image Processing – 3rd ed., Prentice Hall of India, New Delhi, 2008
2. B. Chanda and D.D. Majumder, Digital Image Processing and Analysis, PHI
3. A.K. Jain, Fundamentals of Digital Image Processing, PHI
4. W.K. Pratt, Digital Image Processing, John Wiley, 2006
5. M. Sonka, V. Hlavac and R. Boyle, Image Processing Analysis and Machine Vision, Brooks/colic, Thompson Learning, 1999.

S4.1b Advanced Topics in Database Design (Elective IV)

Course No: 19b

Prerequisites/Exposure:

L	P	C
4	0	4

Objective:

- To study the advanced database techniques beyond the fundamental database techniques.

Unit I: The Extended Entity Relationship Model and Object Model - The ER model revisited, Motivation for complex data types, User defined abstract data types and structured types, Subclasses, Super classes, Inheritance, Specialization and Generalization, Constraints and characteristics of specialization and Generalization, Relationship types of degree higher than two.

Unit II: Object-Oriented Databases - Overview of Object-Oriented concepts, Object identity, Object structure, and type constructors, Encapsulation of operations, Methods, and Persistence, Type hierarchies and Inheritance, Type extents and queries, Complex objects, Database schema design for OODBMS, OQL, Persistent programming languages, OODBMS architecture and storage issues, Transactions and Concurrency control, Example of ODBMS.

Unit III: Object Relational and Extended Relational Databases - Database design for an ORDBMS - Nested relations and collections, Storage and access methods, Query processing and Optimization, an overview of SQL3, Implementation issues for extended type; Systems comparison of RDBMS, OODBMS, ORDBMS.

Unit IV: Parallel and Distributed Databases and Client-Server Architecture - Architectures for parallel databases, Parallel query evaluation, Parallelizing individual operations, Sorting, Joins, Distributed database concepts, Data fragmentation, Replication and allocation techniques for distributed database design, Query processing in distributed databases, Concurrency control and Recovery in distributed databases. An

overview of Client-Server architecture.

Unit V: Object Databases on the Web and Semi Structured Data - Web interfaces to the Web, Overview of XML; Structure of XML data, Document schema, Querying XML data; Storage of XML data, XML applications; the semi structured data model, Implementation issues, Indexes for text data. Enhanced Data Models for Advanced Applications - Active database concepts; Temporal database concepts; Spatial databases Concepts and architecture; Deductive databases and Query processing; Mobile databases, Geographic information systems.

References:

1. Elmasri and Navathe, Fundamentals of Database Systems [4e], Pearson Education
2. Raghu Ramakrishnan, Johannes Gehrke, Database Management Systems [3e], McGraw-Hill
3. Korth, Silberchatz, Sudarshan , Database System Concepts, McGraw-Hill.
4. Peter Rob and Coronel, Database Systems, Design, Implementation and Management, Thomson Learning.
5. C.J.Date, Longman, Introduction to Database Systems, Pearson Education

S4.1c Software Development for Portable Devices (Elective IV)

Course No: 19f

L	P	C
4	0	4

Objectives:

- Explain the key differences between development of systems to run on mobile devices and on typical personal computing.
- Design effective applications for a mobile device by taking into consideration the underlying hardware-imposed restrictions such as screen size, memory size and processor capability.
- Identify potential security issues and suggest mechanisms to ensure the safety of applications on the mobile device.
- To critically analyse and communicate the differences in architecture and specialised topics such as eventhandling between applications on the mobile device and non-mobile platforms.

Unit I: Introduction to Mobile Web (HTML 5) - Semantic Elements – Structural Elements - Basic formatting tags - heading, paragraph, underline break, bold, italic, underline, superscript, subscript, font and image. Different attributes like align, color, bgcolor, font face, border, size. Navigation Links using anchor tag - internal, external ,mail and image links. Lists - ordered, unordered and definition, Table tag, HTML5 Form controls - form, input types – color, date, datetime, datetime-local, email, month, number, range, search, tel, time, url, week, text, password, textarea, button, checkbox,

radio button, select box, hidden controls, calendar, date, time, email, url, search. Datalist, keygen, output - HTML5 Form attributes for <form> and <input> - Element for 2D drawing <canvas> - Inline Scalable Vector Graphics (SVG) - elements for media playback audio and video - Geolocation - Drag and Drop - Support for local storage – localStorage – sessionStorage - Application Cache – HTML5Web Workers - Server-Sent Events – Multimedia support - HTML - Plug-ins – <object> – <embed> - <video> + <embed> - Playing YouTube Videos - New content-specific elements like <article>, <footer>, <header>, <nav>, <section> - CSS3.

Unit II: JQuery – Introduction - Adding jQuery to Your Web Pages – Downloading – Accessing from CDNs - jQuery Syntax - jQuery Selectors - Event Methods - ready(), click(), dblclick(), mouseenter(), mouseleave(), mousedown(), mouseup(), hover(), foucs(), blur() - Effects – Hide, Show, Fading, Sliding, Animation - Callback Functions – Chaining - methods for changing and manipulating HTML elements and attributes - adding new elements/content - append(), prepend(), after(), before() – Removing Elements - remove(), empty() - Manipulating CSS3 - dimensions of elements and browser window – Traversing – Ancestors, Descendants, Siblings - Web SQL Database - Opening Database - Executing queries.

Unit III: Introduction to android and smart phones, Android Architecture & Virtual Machine, Mobile Technology terminologies, setting up the environment, Setting up Emulators, android fundamentals - Activities and Applications Activity Life Cycles Activity Stacks, Activity States, introduction to manifest, resources & R.java , assets, Values – strings.xml - Form widgets, views, Layouts & Drawable Resources - XML Layouts, Linear Layouts, Relative layouts, Table Layouts, android Widgets, UI XML Specifications Events, Bundles & Intents- Explicit Intents Implicit Intents Event Broadcasting with Intents Event Reception with Broadcast Receivers, Adapters and Data Binding.

Unit IV: Files, Content Providers, and Databases - Saving and Loading Files, SQLite Databases - Android Database Design - Exposing Access to a Data Source through a Content Provider Content Provider Registration Native Content Providers, Android Debug Bridge(adb) tool, Linkify.

Unit V: Adapters and Widgets , Notifications , Custom components Threads running on UI thread, Worker thread Handlers & Runnable AsynTask(in detail), Playing Audio and Video, Recording Audio and Video, Using the Camera to Take and Process Pictures. Networking & Location based services - Live Folders, Using sdcards – Reading and writing, XML Parsing - JSON Parsing - Including external libraries in applications, Map-Based Activities, Maps via intent and Map Activity GPS, Location based Services configuration, Geocoding, Accessing Phone services(Call,SMS,MMS) Network connectivity services, Using Wifi & Bluetooth Action bar tabs and custom views on Action bars. Introduction to cross platform application development - ruby on rail, phone gap (notions only).

References:

1. Html5 Black Book:Covers Css3,Javascript,Xml,Xhtml,Ajax,Php And Jquery , Kogent Learning Solutions Inc.
2. KESSLER, Programming HTML 5 Applications, OReilly
3. Android wireless application development, second edition by shane conder, Lauren darcey – Addison - Welsey
4. Android Application Development by Rick rogers,John Lombardo – O'Reilly
5. Professional Android 2 application development by Reto Meier - Wrox

S4.1d Storage Area Networks

Course No: 19h

L	P	C
4	0	4

Unit I: Basic Networking Concepts and Topologies: OSI Reference Model, Common Network Devices, Network Topologies, MAC Standards - Need for Storage Networks – Storage Devices and Techniques Evolution and benefits of SANs - SAN Components and Building Blocks Fibre Channel Basics: Fibre Channel Topologies, Fibre Channel Layers, Classes of Service SAN Topologies.

Unit II: SANs Fundamentals: SAN Operating Systems Software and Hardware Types of SAN Technology: Technology and Configuration, High Scalability and Flexibility Standards Storage Management Challenges Networked Storage Implementation Challenges Storage Subsystems for Video Services..

Unit III: Storage Networking Architecture Storage in Storage Networking: Challenges, Cost, Performance Network in Storage Networking: Fibre Channel, Emerging SAN interconnect Technologies Basic Software Advanced Software Backup Software Implementation Strategies.

Unit IV: Storage Network Management In-Band management Out-of-Band Management-SNMPHTTP - TELNET Storage Network Management Issues Storage Resource Management Storage Management Storage, Systems, and Enterprise Management Integration.

Unit V: Designing and building a SAN- Design considerations Business requirements Physical layout Placement Storage pooling Data availability Connectivity scalability migration manageability fault tolerance and resilience - prevention of congestion routability- backup and restoration - SAN Security & iSCSI Technology Basic security guidelines implementing SAN security Backup and restoration iSCSI technology - Future of SANS.

References:

1. Meeta Gupta, Storage Area Network Fundamentals , Cisco Press, 2002.
2. John R. Vacca, The Essential Guide to Storage Area Networks , Prentice Hall, 2002.
3. Richard Barker, Paul Massiglia, Storage Area Network Essentials , John Wiley & Sons, Inc., 2002.
4. Tom Clark, Designing Storage Area Networks , Addison Wesley Pearson Education (Second Edition).
5. Alex Goldman, Storage Area Networks Fundamentals , Cisco Press 2002.
6. Christopher Poelker, Storage Area Networks for Dummies.

S4.1e Semantic Web

Course No: 19e

L	P	C
4	0	4

Objectives: To discover the capabilities and limitations of semantic web technology for different applications.

Unit I: Components – Types – Ontological Commitments – Ontological Categories – Philosophical background –Knowledge Representation Ontologies – TopLevel Ontologies – Linguistic Ontologies – Domain Ontologies – Semantic Web – Need – Foundation – Layers – Architecture.

Unit II: Languages for Semantic Web and Ontologies - Web Documents in XML – RDF - Schema – Web Resource Description using RDF - RDF Properties – Topic Maps and RDF – Overview – Syntax Structure – Semantics –Pragmatics -Traditional Ontology Languages – LOOM- OKBC – OCML - Flogic Ontology Markup Languages – SHOE – OIL – AML – OIL – OWL.

Unit III: Ontology Learning for Semantic Web - Taxonomy for Ontology Learning – Layered Approach – Phases of Ontology Learning – Importing and Processing Ontologies and Documents – Ontology Learning Algorithms – Evaluation.

Unit IV: Ontology Management and Tools - Overview – need for management – development process – target ontology – ontology mapping – skills management system – ontological class – constraints – issues. Evolution – Development of Tools and Tool Suites – Ontology Merge Tools – Ontology based Annotation Tools.

Unit V: Applications - Web Services – Semantic Web Services - Case Study for specific domain – Security issues – current trends.

References:

1. Asuncion Gomez-Perez, Oscar Corcho, Mariano Fernandez-Lopez, Ontological Engineering: with examples from the areas of Knowledge Management, e-Commerce and the Semantic Web, Springer, 2004.
2. Grigoris Antoniou, Frank van Harmelen, A Semantic Web Primer (Cooperative Information Systems), The MIT Press, 2004.
3. Alexander Maedche, Ontology Learning for the Semantic Web, Springer; 1 edition, 2002.
4. John Davies, Dieter Fensel, Frank Van Harmelen, Towards the Semantic Web: Ontology – Driven Knowledge Management, John Wiley & Sons Ltd., 2003.
5. Dieter Fensel (Editor), Wolfgang Wahlster, Henry Lieberman, James Hendler, Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential", The MITPress, 2002.

S4.2f Advanced Java Programming

Course No: 19f

Prerequisite:

L	P	C
4	0	4

Objectives: To learn the advanced features of Java programming language.

Unit I: RMI & Servlets Introduction, Architecture, defining remote objects, creating stubs and skeletons, serializable classes, Accessing remote objects, factory classes, dynamically loaded classes,RMI activation, registering remote objects. Servlets, generic servlet, servlets that access request headers, Develop servlets that manipulate response headers, HTTP servlets, Forms, HTTP Proctols - Configuring Tomcat Server, Servlet Context, servlet context listener, Servelet Chaining.

Unit II: JNDI & EJB - Architecture, context initial context class, objects in a context, binding objects, accessing directory services, attributes and attribute interface modifying directory entities, creating directories entities. EJB roles, architecture, container, implementing a basic EJB object, Implementing session beans, implementing Entity bean, Deploying an enterprise bean object.

Unit III: Java Server Pages: Developing JSP Pages, technology, syntax using scripting elements, syntax using the courier page directive, Create and use JSP error pages, Building Reusable Web Presentation, Components Describe how to build Web page layouts from reusable presentation components, JSP technology syntax using the include directive, JSP technology syntax using the jsp:include standard action ,Developing JSP Pages Using Custom Tags ,problem with JSP technology scriptlet code, Given an existing custom tag library, develop a JSP page using the library, developing a Simple Custom Tag , structure and execution of a custom tag in a JSP page, tag handler class for a simple empty custom tag ,custom tag that includes its body in the contour of the HTTP response, tag library description for a simple, empty custom tag.

Unit IV: Hybernate - ORM Overview - Hibernate Overview - Hibernate Architecture - Hibernate Environment - Hibernate Configuration - Hibernate Sessions - Hibernate Persistent Class - Hibernate Mapping Files - Hibernate Mapping Types - Hibernate Examples - Hibernate O/R Mappings - Hibernate Annotations - Hibernate Query Language - Hibernate Criteria - Queries - Hibernate Native SQL, Hibernate Caching, Hibernate Batch Processing, Hibernate Interceptors.

Unit V: Struts 2 - Basics - Basic MVC Architecture - Overview - Environment Setup - Architecture - Struts2 Configuration – Actions – Interceptors - Result Types - Value Stack/OGNL - File Uploads - Database Access - Sending Email – Validations – Localization - Type Conversion - Themes/Templates - Exception Handling – Annotations.

References:

1. Java Servlets - Tata McGraw Hill JSP - Java Sever Pages - IDG Books
2. Java Beans Developers Resource – PHI
3. Chuck Cavaness, Programming Jakarta Struts, 2nd Edition.
4. Madhusudhan Konda, Just Hibernate, Oreilly

20 S4.5 Major Project

Course No: 20

L	P	C
		8

Major project work is to be done individually by each student, under the guidance of a faculty member of the concerned department. Exposure to Software Engineering (SE) Principles and an insight to the Research Methodology (RM) is to be imparted to the student so that (s)he can proceed with the project work as per the underlying principles of SE/RM.

Students can either take up a real-life application oriented project work or research and development project. The student can formulate a project problem with the help of her/his Guide and submit the project proposal of the same. Approval of the project proposal is mandatory. If approved, the student can commence working on it, and complete it.

There shall be an evaluation committee (EC) for the internal evaluation of the work. EC should consists of HOD, at least two senior most faculty members and the guide of the student. EC can set a schedule for the evaluation of the work in different stages. For eg, a software development project can be evaluated in 5 stages – problem formulation, analysis, design, implementation and testing.

At the time of external evaluation, if the performance is below the mark, student will be given a chance to reappear within 3 months to present the work again, after incorporating the changes suggested by the external examiner. A certificate stating that the changes suggested by the external examiner are incorporated in the revised report

is to be attached with the revised report which is signed by the student, guide and the HOD.

Evaluation (Internal)

The internal evaluation will be done by the EC in periodic intervals. 5% weightage is to be given for the evaluation of the SE/RM course work and remaining 45% weightage is for the work.

Evaluation (External)

Guidelines for submission of report

The distinguishing mark of a dissertation is an original contribution to knowledge. The dissertation is a formal document whose sole purpose is to prove that you have made an original contribution to knowledge. Failure to prove that you have made such a contribution generally leads to failure.

It is a test of the student's ability to undertake and complete a sustained piece of independent research and analysis / application development, and to write up the work in a coherent form according to the rules and conventions of the academic community.

A satisfactory dissertation should not only be adequate in its methodology, in its analysis and in its argument, and adequately demonstrate its author's familiarity with the relevant literature; it should also be written in correct, coherent language, in an appropriate style, correctly following the conventions of citation. It should, moreover, have a logical and visible structure and development that should at all times assist the reader's understanding of the argument being presented and not obscure it. The layout and physical appearance of the dissertation should also conform to university standards.

The dissertation is to be prepared in tex format (either Latex or using an equivalent Windows tex variant such as MikeTex). The format of the report will be distributed shortly.

Syllabus for Fundamentals of Software Engineering & Research Methodology

Note:-

- This course is to be offered in the last semester for 20 hours. An evaluation test is to be conducted at the end of the session.

Unit I (4 Hours)
Software Engineering - Introduction - Software characteristics - Classification of Software - Phases in Software Engineering - Key challenges in Software Engineering. Waterfall Model – Agile Model – SDLC - Software Process, Project and Product - Components of Software Process- Process Framework - Process Assessment. Software Life Cycle Models

Unit II (4 Hours)
Requirements Engineering - Feasibility study - Types of Feasibility - Requirement Elicitation - Elicitation techniques. Requirement analysis - Structured Analysis – DFD - Object Oriented Modeling. Activity Diagram - Data Diagram- ER diagram - Use case Diagram.

Unit III (3 Hours)
Software Requirements Specification: Purpose of SRS, Structure of SRS, IEEE template of SRS. Software Design: Principles of Software Design- Software Design Concepts.

Unit IV (2 Hours)
Software Coding: Features of Software Code - Coding Guidelines- Coding Methodology. Code verification techniques.

Unit V (2 Hours)
Software Testing: Software Testing Basics - verification and validation- Guidelines of Software Testing - Steps involved in Test Plan- Software Testing Strategies.

Unit VI (2 Hours)
Introduction to Research Methods - Philosophy of Science, Evolutionary Epistemology, Scientific Methods, Hypotheses Generation and Evaluation, Code of Research Ethics, Definition and Objectives of Research, Various Steps in Scientific Research, Types of Research; Research Purposes - Research Design - Survey Research - Case Study Research.

Unit VI (4 Hours)

How to perform a literature review - Sampling Methods - Data Processing and Analysis strategies - Data Analysis with Statistical Packages - Hypothesis-testing - Generalization and Interpretation.

Unit VII (3 Hours)
Research Reports - Structure and Components of Research Report, Types of Report, Layout of Research Report, Mechanism of writing a research report

References:

1. Pankaj Jalote, An Integrated Approach to Software Engineering , Narosa publication.
2. Rohit Khurana, Software Engineering – Principles and Practices, Vikas publishing
3. Garg, B.L., Karadia, R., Agarwal, F. and Agarwal, U.K., 2002. An introduction to
4. Research Methodology, RBSA Publishers.
5. Kothari, C.R., 1990. Research Methodology: Methods and Techniques. New Age International. 418p.
6. Sinha, S.C. and Dhiman, A.K., 2002. Research Methodology, Ess Ess Publications. 2 volumes.