16. 1.10

Placed at the meeting of
Academic Council
held on 28.03.2016

APPENDIX - K

MADURAI KAMARAJ UNIVERSITY

(University with Potential for Excellence)

Revised Syllabus for M.Sc. Computer Science
(CBCS - Semester Pattern)

This will be effective from the academic year 2016-17

REGULATIONS :

1. **Course Objective:**
   To prepare the students to manage the software components in a computer independently and to be a Programmer/Project Leader. To motivate the students to take up Academic Research in Computer Science and other streams.
2. **Eligibility for Admission:**
   Students who studied B.Sc. (CS) and BCA..
3. **Duration of the Course:**
   The students shall undergo the prescribed course of study for a period of not less than three academic year (Six semesters).
4. **Medium of Instruction :** English
5. **Subjects/ Structure of Course Study :** See Appendix – PCS1
6. **Scheme of Examinations/ Structure of Question Paper:** : See Appendix - PCS2
   Detailed Syllabus: See Appendix – PCS3
7. **Eligibility for the Degree:**

   i) A Candidate shall be eligible for the award of the degree on completion of the prescribed course of study and passing all the prescribed external examinations.

   ii) Attendance progress, internal examinations, conduct certificate from the Head of the Institution shall be required for taking the external examination.

   iii) The passing minimum and the ranking are as per the existing rule of the Choice Based Credit System for the affiliated college of the University.

Appendix – PCS1
(Subject/Structure of Course Study

Semester	Subjects						Total Hours	Total Credits
I	CS11(5) [4]	CS12(5) [4]	CS13(5) [4]	CS14(5) [4]	CS15(5) [3]	CS16(5) [3]	30	22
II	CS21(5) [4]	CS22(5) [4]	CS23(5) [4]	ES1(5) [4]	CS24(5) [3]	CS25(5) [3]	30	22
III	CS31(5) [5]	CS32(5) [4]	ES2(5) [4]	NME(4) [4]	CS33(5) [3]	CS34(6) [3]	30	22
IV	CS41(5) [4]	CS42(5) [4]	ES3(5) [4]	CP (15) [12]			30	24
	Total							90

Abbreviations:

()	–	Number of Hours	**[ ]**	–	Number of Credits
CS	-	Core Subject	**NME**	-	Non Major Elective
ES	-	Elective Subject	**CP**	-	Course Project

I SEMESTER

S No	CODE	Subject	Hours	Credits	Internal Marks	External Marks
1	CS 11	Discrete Mathematical Structures	5	4	25	75
2	CS 12	Advanced C Programming	5	4	25	75
3	CS 13	Data Structures	5	4	25	75
4	CS 14	Computer Algorithms	5	4	25	75
5	CS 15	**Lab1:** Advanced C Programming	5	3	40	60
6	CS 16	Lab2: Data Structures and Algorithms Lab	5	3	40	60
			30	22		

II SEMESTER

S No	CODE	Subject	Hours	Credits	Internal Marks	External Marks
1	CS 21	Advanced Java Programming	5	4	25	75
2	CS 22	Compiler Design	5	4	25	75
3	CS 23	Operating System Design Principles	5	4	25	75
4	ES1	1.Embedded Systems 2.Advanced Software Engineering 3.Distributed Systems	5	4	25	75
5	CS 24	**Lab 3:** Advanced Java Programming	5	3	40	60
6	CS 25	**Lab 4:** Operating System	5	3	40	60
			30	22		

III SEMESTER

S No	CODE	Subject	Hours	Credits	Internal Marks	External Marks
1	CS 31	Digital Image Processing	5	4	25	75
2	CS 32	Soft Computing	5	4	25	75
3	ES2	1. Information Retrieval 2. Network Security 3. Internet of Things	5	4	25	75
4	NME	Information Technology and Data Compression	4	4	25	75
5	CS 33	**Lab 5**: Soft Computing	5	3	40	60
6	CS 34	**Lab 6:**Image Processing	6	3	40	60
			30	22		

IV SEMESTER

S No	CODE	Subject	Hours	Credits	Internal Marks	External Marks
1	CS 41	Pattern Recognition	5	4	25	75
2	CS 42	Advanced System Architecture	5	4	25	75
3	ES 3	1.Big Data Analytics 2.Wireless Sensor Networks 3.Cloud Computing	5	4	25	75
4	CP	Project Work & Viva voce	15	12	40	60
			30	24		

Non-Major Elective Course to be offered by the Department of Computer Science to other Departments

NME: Information Technology and Data Computation

Appendix – PCS2

Scheme of Examination /Question Paper Pattern

Scheme of Evaluation

Theory Subjects:

Question Paper Pattern:

Time: 3 Hours **Max. Marks: 75**

Part – A

Answer all the questions **(10*1=10)**

Ten Questions, two questions from every UNIT: *Multiple Choice Questions*

Part – B

Answer all the questions **(5*7=35)**

Five Questions, one question set from every UNIT: *Either ...Or... type*

Part – C

Answer any three questions **(3*10=30)**

Five Questions, one question from every UNIT

The following list of parameters taken into account for the evaluation of the Practical examination and Project work.

For Practical Subjects:

Parameters:

i.	Aim, Procedure / Algorithm and Program:	15
ii.	Coding and Compilation :	10
iii.	Debugging :	15
iv.	Results :	10
v.	Viva:	10
	Total	**60**

Note: The External Examiner can fix other exercises also, other than those found in the list *(Syllabus)* in consultation with the Internal Examiner without violating the scope of the prescribed syllabus.

For Project Work:

Total Marks: 100 (Internal: 40 marks, External: 60 Marks)

Parameters:

For Internal Marks (40):

Start-up Review	: 5 Marks
Design Review	: 7.5 Marks
Implementation and Validation Review	:7.5 Marks
Final Review	: 10 Marks
Overall Performance	: 10 Marks

For External Marks (60):

Project Report	: 20 Marks
Project work, Demo & Presentation	: 30 Marks
Viva-Voce	:10 Marks

Appendix – PCS3
(Detailed Syllabus)
CS 11: DISCRETE MATHEMATICAL STRUCTURES
(5 Hours – 4 Credits)

UNIT I:

Mathematical Logic: Statements and Notation – Connectives – Negation – Conjunction – Disjunction – Statement Formulas and Truth Tables – Logical Capabilities of Programming Languages – Conditional and Bi-conditional – Well-formed Formulas – Duality Law – Tautological Implications – Formulas with Distinct Truth Tables – Functionality Complete Sets of Connectives – Other Connectives – Two-state Devices and Statement Logic – Normal Forms – Disjunctive Normal Forms – Conjunctive Normal Forms – Principal Disjunctive Normal Forms – Principal Conjunctive Normal Forms – Ordering and Uniqueness of Normal Forms – Completely Parenthesized Infix Notation and Polish Notation – The Theory of Inference for the Statement Calculus – Validity Using Truth Tables – Rules of Inference – Consistency of Premises and Indirect Method of Proof – Automatic Theorem Proving – The Predicate Calculus – Predicates – The Statement Function, Variables, and Quantifiers – Predicate Formulas – Free and Bound Variables – The Universe of Discourse – Inference Theory of the Predicate Calculus

– Valid Formulas and Equivalences – Some Valid Formulas over Finite Universes – Special Valid Formulas Involving Quantifiers – Theory of Inference for the Predicate Calculus – Formulas Involving More Than One Quantifier.

UNIT II:
Relations and Ordering: Relations – Properties of Binary Relations in a Set – Relation Matrix and the Graph of a Relation – Partition and Covering of a Set – Equivalence Relations – Compatibility Relations – Composition of Binary Relations – Partial Ordering – Partially Ordered Set: Representation and Associated Terminology.

UNIT III:
Lattices and Boolean Algebra: Lattices as Partially Ordered Sets – Definition and Examples – Some Properties of Lattices – Lattices as Algebraic Systems – Sublattices, Direct Product, and Homomorphism – Some Special Lattices – Boolean Algebra – Definitions and Examples – Subalgebra, Direct Product and Homomorphism – Boolean Functions – Boolean Forms and Free Boolean Algebras – Values of Boolean Expressions and Boolean Functions – Representation and Minimization of Boolean Functions – Representation of Boolean Functions – Minimization of Boolean Functions.

UNIT IV:
Graph Theory: Basic Concepts of Graph Theory – Basic Definitions, Paths, Reachability and Connectedness – Matrix Representation of Graphs, Trees - Storage Representation and Manipulation of Graphs – Trees-Their Representation and Operations – List Structures and Graphs.

UNIT V:
Introduction to Computability Theory: Introduction – Finite-state Acceptors and Regular Grammars – Turing Machines and Partial Recursive Functions.

Text Book:

Discrete Mathematical Structures with Applications to Computer Science, J.P. Tremblay and R.Manohar, McGraw Hill Book Company, New York, 1988.

UNIT I – Chapter 1 – 1.1, 1.2, 1.3, 1.4, 1.5, 1.6 (Exercises Excluded)
UNIT II – Chapter 2 – 2.3 (Exercises Excluded)
UNIT III –Chapter 4 – 4.1,4.2,4.3,4.4 (Exercises Excluded)
UNIT IV –Chapter 5 – 5.1 & 5.2 (Exercises Excluded)
UNIT V – Chapter 6 – 6.1 & 6.2 (Exercises Excluded)

Reference Books:

1. Discrete Mathematics for Computer Scientists – John Truss – II Edition Addison Wesley – 2000.
2. Introduction to Automata Theory, Languages and Computation – John E.HopCroftR.Motwani, Jeffery D.Ullman – III Edition Pearson Education – 2008.

3. Discrete Mathematics with Graph Theory – GoodaireParmenter – Prentice Hall Inc., - 1998.
4. Discrete and Combinational Mathematics – Ralph P.Grimaldi – Fourth Edition Pearson Education – 1999.
5. Discrete Mathematics and Graph Theory – Satyanarayana – PHI Pvt. Ltd., - 2009.

CS 12: ADVANCED C PROGRAMMING
(5 Hours – 4 Credits)

UNIT I :

Introduction to pointers: The & and * operators – Pointers Expression – char, int and float pointers – passing addresses to functions – functions returning pointers.

Pointers and array: What are arrays? – passing array elements to a function – pointers and arrays – passing an array to a function – more than one dimension – pointers and two dimensional arrays – pointer to an array –passing two dimensional array to a function – three dimensional array – passing 3Darray to a function –returning array from function – returning 3D array from a function – dynamic memory allocation – array of pointers.

UNIT II :

Pointers and Strings: What are strings? – Standard library functions – pointers and strings – const qualifier –returning const values – two dimensional array of characters – array of pointers to string – limitations of array of pointers to string – limitations of array of pointers to strings.

Pointers and Structures: An array of structure – more about Structures – Structure pointers – offsets of Structure elements – linked list – Stacks and Queue –Doubly linked list.

UNIT – III:

Pointers and Data Structure: Merging of linked list – linked list and polynomials –sorting a linked list – circular linked list – Trees – binary tree – traversal of a binary tree –deletion from a binary tree – threaded binary tree – Graphs.

UNIT IV:

Operations on Bits: Bitwise operators – one's complement operator – right shift operator – left shift operator – bitwise AND – bitwise OR – bitwise XOR operator –showbit() function.

Advanced Concepts in C: Types of pointers – which pointers to use? Pointers and typecasting – physical address to segment: offset – the dancing dolls – capslocked – how much memory do you have? ROM-BIOS philosophy – CPU registers – interrupts and Interrupt vector table – invoking ROM-BIOS functions – int86() function – finding memory size – using the declarations in dos.h – positioning cursor on the screen – interrupts to Access ROM-BIOS/DOS services- few more examples – wri8ting directly to VDU memory.

UNIT – V:

Elementary TSR: How a TSR works? – Pointers to function – The first TSR – interrupt function modifier – variables in TSRs – Time Bound TSRs – Tinkering with the keyboard – functioning of the keyboard – how the pointer works – your printer is virused – printer jam! – TSR clock.

Some more TSRs: What is a stack? – the interrupt modifier revisited – TSR to write protect the Hard disk – Interrupt OX21 – A gateway to the DOS kingdom – delete which doesn't work – increase your monitor life – DOS key initiated.

Text Books:

1. Understanding Pointers in C, YashavantP.Kanetkar BPB publications., 2009.
   UNIT I : - **Chapter** 1, 2
   UNIT II : - **Chapter** 3, 4
   UNIT III : - **Chapter 5**
2. Let us C - **YashavantP.Kanetkar** (sixth revised and updated Edition), BPB, 2013
   UNIT IV : - **Chapter 18**
3. Writing TSRs Through C –YashavantP.Kanetkar, BPB Publications, 1995
   UNIT V : - Chapter 1, 2, 3

Reference Book:

1. TSR: Introduction in C – Ramesh B, 2006.
2. Write TSRs now with Borland's Turbo C/C++, Turbo Pascal – Jim Ras, Plano, Tex. : Wordware Pub., 1993.

CS 13: DATA STRUCTURES
(5 Hours – 4 Credits)

UNIT I:

Introduction and Overview: Definitions – Concept of Data Structures – Overview of Data structures – Implementation of Data Structures **Tables:** Rectangular Tables – Jagged Tables – Inverted Tables – Hash Tables – Hashing Techniques – Collision Resolution Technique – Closed Hashing – Open Hashing – Comparison of Collision Resolution Techniques.

UNIT II:

Trees: Basic Terminologies – Definition and Concepts – Binary Trees – Properties of a Binary Tree – Representations of Binary Tree – Linear Representation of a Binary Tree – Linked Representation of a Binary Tree – Physical Implementation of a Binary Tree in Memory – Operations on a Binary Tree – Insertion – Deletion – Traversals – Merging together Two Binary Trees-Types of Binary Trees – Expression Tree – Binary Search Tree – Threaded Binary Trees – Splay Tree – Decision Trees – B Trees – B Tree Indexing – Operations on a B Tree – Lower and Upper Bounds of a B Tree - B+ Tree Indexing – Trie Tree Indexing – Trie Structure.

UNIT III:

Sorting: Sorting – Basic Terminologies – Sorting Techniques – Sorting by Selection – Straight Selection Sort – Tree Selection Sort – Tree Selection Sort – Sorting by Exchange – Bubble Sort – Shell Sort – Radix Sort – Bucket Sort - **Searching** – Fibonacci Search – Interpolation Search.

UNIT IV:

Graphs – Introduction – Graph Terminologies - Representation of Graphs – Set Representation – Linked Representation – Matrix Representation – Operations on Graphs – Operations on

Linked List Representation of Graphs - Operations on Matrix Representation of Graphs - Application of Graph Structures – Shortest Path Problem – Topological Sorting – Minimum Spanning Trees – Connectivity in a Graph – Euler's and Hamiltonian Circuits – BDD and its Applications – Conversion of Decision Tree into BDD – Applications of BDD.

UNIT V:

Definition and Terminologies – Representation of Sets – Linked List Representation of Set – Hash Table Representation of Sets – Bit Vector Representation of Sets – Tree Representation of Sets – Operation of Sets – Operation on List Representation of Set – Operation on Hash Table Representation of Sets– Operation on Bit Vector Representation of Sets – Operation on Tree Representation of Sets – Applications of Sets – Spelling Checker – Information System using Bit Strings – Client-Server Environment.

Text Book:

Classic Data Structures – Debasis Samanta – Second Edition – PHI Pvt Limited- 2009.

UNIT I: Chapter 1(1.1 to 1.5),Chapter 6 (6.1 to 6.5)
UNIT II: Chapter 7(7.1,7.2,7.3.1,7.4.1,7.5.1,7.5.2,7.5.4,7.5.7,7.5.9,7.7.1,7.8,7.9.1)
UNIT III: Chapter 10 (10.1,10.2,10.4.1,10.4.2,10.5.1,10.5.3,10.6.1,10.6.2),11 (11.2.5,11.2.6)
UNIT IV: Chapter 8(8.1 to 8.7)
UNIT V: Chapter9 (9.1 to 9.5)

Reference Books:

1. Data Structures and Algorithms Alfred V. Aho, Jeffrey D. Ullman, John E. Hopcroft. AddisonWesley, 1983.
2. Data Structure and Algorithm Analysis in C, Mark Allen Weiss , Second Edition, Addison Wesley publishing company, 1997.
3. C and C++ Programming concepts and Data Structures, P.S.Subramanyam, BS Publications, 2013.

CS 14: COMPUTER ALGORITHMS
(5 Hours – 4 Credits)

UNIT I:

Introduction: What is an algorithm?-Fundamentals of Algorithmic Problem Solving: Understanding the Problem - Ascertaining the Capabilities of a Computational Device -Choosing between Exact and Approximate Problem Solving - Deciding on Appropriate Data Structures - Algorithm Design Techniques - Methods of Specifying an Algorithm - Proving an Algorithm's Correctness - Analyzing an Algorithm - Coding an Algorithm. Important Problem Types: Sorting - Searching - String Processing - Graph Problems - Combinatorial Problems - Geometric Problems - Numerical Problems. Fundamental Data Structures: Linear Data Structures - Graphs - Trees - Sets and Dictionaries.

Fundamentals of the Analysis of Algorithm Efficiency: Analysis Framework: Measuring an Input's Size - UNITs for Measuring Running Time-Ordering of Growth – Worst Case, Best Case, and Average Case Efficiencies - Recapitulation of the Analysis Framework. Asymptotic Notations and Basic Efficiency Classes: Informal Introduction - O-notation - Ω-notation - Θ-notation - Useful Property Involving the Asymptotic Notations - Using Limits for Comparing Orders of Growth - Basic Efficiency Classes. Mathematical analysis of non-recursive Algorithms - recursive Algorithms.

UNIT II:

Divide-and-Conquer: Merge sort – Quick sort - Binary Search - Multiplication of Large Integers and Strassen's Matrix Multiplication: Multiplication of Large Integers - Strassen's Matrix Multiplication.

Decrease-and-Conquer: Insertion Sort-Depth-First Search and Breadth-First Search: Depth-First Search-Breadth-First search-Topological Sorting.

UNIT III:

Transform-and-Conquer: Gaussian Elimination: LU Decomposition and Other Applications-Computing a Matrix Inverse-Computing a Determinant. Balanced Search Trees: AVL Trees-2-3 Trees. Heaps and Heap sort: Notion of the Heap – Heap sort.

Dynamic Programming: Computing a Binominal Coefficient - Warshall's and Floyd's Algorithms: Warshall's Algorithm-Floyd's Algorithm for the All Pairs Shortest Paths Problem - Optimal Binary Search Trees - The Knapsack Problem and Memory Functions: Memory Functions.

UNIT IV:

Greedy Technique: Prim's Algorithm-Kruskal's Algorithm: Disjoint subsets and Union-Find Algorithms. Dijkstra's Algorithm - Huffman Trees.

Coping with the Limitations of Algorithm Power: Backtracking: n-Queens Problem - Hamiltonian Circuit Problem - Sum of Subset Problem.

UNIT V:

Coping with the Limitations of Algorithm Power: Branch and Bound: Assignment Problem - Knapsack Problem - Traveling Salesman Problem.

Limitations of Algorithm Power:*P*, *NP*, and *NP*-complete Problems: *P* and *NP* Problems - NP - Complete Problems.

Coping with the Limitations of Algorithm Power: Approximation Algorithms for NP-hard problems: Approximation Algorithms for the Traveling Salesman Problem - Approximation Algorithms for the Knapsack Problem.

Text Book:

Introduction to the Design & Analysis of Algorithms, Anany Levitin, Pearson, second edition, 2012.

Reference Books:

1. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein.,McGraw-Hill, 2001.
2. Essential References ALGORITHMS IN C++ by Robert Sedgewick (Pearson Education)2008.
3. Algorithms ,Kenneth A. Berman and Jerome L. Paul, Cengage learning India Edition, New Delhi, 2002.
4. Computer Algorithms – Introduction to Design & Analysis,Sara Baase and Allen Van Gelder, Third Edition, Pearson Education, New Delhi, 2000.

CS 15: LAB 1.ADVANCED C PROGRAMMING
(5 Hours – 3 Credits)

1. Write a C program to perform Binary search
2. Write a C program to add two numbers using typedef.
3. Write a C program to print the days of the week using enumeration
4. Write a C program to perform temperature conversion (Celsius to Fahrenheit) using command line argument.
5. Write a C program to convert a Roman numeral to its decimal equivalent.
6. Write a C program to prepare student marksheet using structure
7. Write a C program to create employee pay bill using array of structure
8. Write a C program to prepare student information with student name, subject and percentage as record1. With the help of record1 details, prepare record2 and print all the details using union.
9. Write a C program to perform the following using bitwise operators.
   a. Addition b. subtraction c. multiplication d. division
10. Write a C program to check whether a given number is palindrome or not using bitwise operators.
11. Write a C program to perform arithmetic operations using function (use pointer to function , refer operand using pointer)
12. Write a C program to perform matrix addition and subtraction using pointer
13. Write a C program to filter the alphabets in a sentence using pointers
14. Write a C program to arrange the given names in alphabetical order using pointers
15. Write a C program to create two files ODD and EVEN for the given set of numbers and print odd and even numbers in the corresponding files(use sequential file) .
16. Write a C program to create a stock details file with item number, item name, quantity-in- hand and price. Read the file and display all the details including calculated value of stock (use random file).
17. Write a C program to swap two numbers using macro
18. Write a C program to perform string concatenation using macro
19. Write a TSR program to permanently ON the capslock key
20. Write a TSR program to display digital clock

Note: The above are sample problems, Instructor can add more exercises on their requirements and to the technology

CS 16: LAB 2.DATA STRUCTURES AND ALGORITHMS
(5 Hours – 3 Credits)

1. Implementation of Stack
   a) Using Array
   b) Using Linked List
2. Implementation of Queue
   a) Using Array
   b) Using Linked List
3. Implementation of Heap Tree.
4. Implementation of Tree Traversal.
5. Implementation of BFS.
6. Implementation of DFS.
7. Implementation of Merge Sort using Divide and Conquer.
8. Implementation of Knapsack Problem using Dynamic Programming.
9. Implementation of Warshall's Algorithm using Dynamic Programming.
10. Implementation of Floyd's Algorithm using Dynamic Programming.
11. Implementation of Dijkstra's Algorithm using Greedy Technique.
12. Implementation of Prim's Algorithm using Greedy Technique.
13. Implementation of n-queens Problem using Backtracking.
14. Implementation of Assignment Problem using Branch and bound.

Note: The above are sample problems, Instructor can add more exercises on their requirements and to the technology

CS 21: ADVANCED JAVA PROGRAMMING
(5 Hours – 4 Credits)

UNIT I:

Applet and Graphics: Applet life cycle, Applet methods, Passing parameters to Applets, getDocumentBase() and getCodeBase(), Using images, Applet interfaces, Difference between Applet and Application Program, Drawing lines and different Shapes, Clipping.

UNIT II:

AWT and Event Handling: Introduction, Component, Frame, Button class, Layout Management, Insets, Canvas, Label, Text field, Check Box, Check Box Group, Choice, List, Menu, Event handling, Adapter class.

UNIT III:

HTML and Java Script: HTML Editors, Elements, Tags, Minimal HTML Document, Markup Tags, Java Script, Java Script and the Language, Java Script and Java, Strengths and Weakness of Java Script.

UNIT IV:

Servlet: Introduction, DHTML, CGI script, Java Servlet, Servlet Container, Servlet Life Cycle, Servlet Interface, Generic Servlet Class, HttpServlet Class, HttpServlet Interface, getOutputStream method, SetHeader() method, parameter passing to servlet, More about Servlet Owner, Java Web Server and Cookies.

UNIT V:

Networking and RMI: TCP/IP, UDP/IP, IP Address, DNS, Port, URL, Socket Programming using TCP/IP and UDP/IP, RMI packages, Programming using RMI.

Text Book:

Internet & Java Programming, 1e, R. Krishnamoorthy and S. Prabhu, Publishers: New Age International, 2013.

UNIT I:	Chapters	16 and 17
UNIT II:	Chapters	19
UNIT III:	Chapters	25
UNIT IV:	Chapters	22
UNIT V:	Chapters	18 and 21

Reference Book:

Java Complete Reference, Herbert Schildt, Tata McGraw Hill, 2013.

CS 22: COMPILER DESIGN
(5 Hours – 4 Credits)

UNIT I:

Compilers And Translators-Why Do We Need Translators?-The Structure Of A Compiler-Lexical Analysis-Syntax Analysis-Intermediate Code Generation-Optimization-Code Generation-Book Keeping-Error Handling-Compiler-Writing Tools-Getting started. The role of the lexical analyzer-Simple approach to design of a lexical analyzer-Regular Expressions-Finite Automata-From regular expression to finite automata-Minimizing the number of states of a DFA-A language for specifying lexical analyzer-Implementing a lexical analyzer- The scanner generator as Swiss army Knife.

UNIT II:

The Syntactic Specification of Programming Languages-Derivation and Parse Trees-Capability of context free Grammars. Parsers-Shift-reduce Parsing-Operator-precedence parsing-Top-down parsing-Predictive Parsers.

UNIT III:

LR parsers-The canonical collection of LR(0) items-constructing SLR parsing tables – constructing canonical LR parsing tables-constructing SLR parsing tables-constructing LALR parsing tables – Using Ambiguous grammars- An automatic parse generator Implementation of LR parsing Tables – constructing LALR set of items. Syntax directed translation schemes – Implementation if syntax directed schemes-Intermediate Code-Parse Tree and Syntax Trees – Three Address code, quadruples, and triples-Translation of assignment statements-Boolean Expression-Statements that alter the flow of control-postfix translations-Translation with a top-down parser.

UNIT IV:

The contents of a symbol tables-Data structure for a symbol table-Representing Scope information. Errors-Lexical-phase errors - syntactic-phase errors-Semantic errors. The principal sources of optimization-Loop optimization -The DAG representation of basic blocks-Value numbers and algebraic laws-Global data-flow analysis.

UNIT V:

Dominators-Reducible Flow graphs -Depth-first search-Loop-invariant computations –Induction variable elimination-Some other loop optimization. Code Generation-Object Programs- A machine Model- A simple code generator-Register allocation and assignment-Code generation from DAG's-Peephole Optimization.

Text Book:

Principles of Compiler Design, Alfred V.Aho and Jeffrey D.Ullman.25th Reprint, 2002.

UNIT I: Chapter 1,3
UNIT II: Chapter 4,5
UNIT III: Chapter 6,7
UNIT IV: Chapter 9,11,12
UNIT V: Chapter 13,15

Reference Books:

1. Compiler Design in C ,Allen I. Holub Prentice Hall of India, 2003.
2. Crafting a compiler with C ,C. N. Fischer and R. J. LeBlanc, Benjamin Cummings, 2003.
3. Introduction to Compiler Techniques, J.P. Bennet, Second Edition, Tata McGraw-Hill, 2003.

CS 23: OPERATING SYSTEM DESIGN PRINCIPLES

(5 Hours – 4 Credits)

UNIT I :

Computer system overview – basic elements - processor registers – instruction execution – interrupts – memory hierarchy – cache memory – I/O communication techniques. Operating system overview – operating system objectives and functions – evolution of operating systems – major achievements – developments leading to modern operating systems – Microsoft windows overview.

UNIT II:

Process description and control – what is a process? – process states – process description – process control – execution of operating system – security issues. Threads, SMP, Micro kernels – processes and threads – symmetric multiprocessing – micro kernels – windows vista thread and SMP management. Concurrency: Mutual exclusion and Synchronization - Principles of concurrency –mutual exclusion: hardware support – semaphores – monitors – message passing – reader/writer problem.

UNIT III:

Concurrency: Deadlock and Starvation – principles of deadlock – deadlock prevention –deadlock avoidance – deadlock detection – an integrated deadlock strategy – dining philosophers problem

– windows vista concurrency mechanisms. Memory management – memory management requirements – memory partitioning – paging – segmentation – security issues. Virtual memory – hardware and control structures – operating system software – windows vista memory management.

UNIT IV:
Uni processor scheduling – types of scheduling – scheduling algorithms. Multiprocessor and Real time scheduling – multiprocessor scheduling – real time scheduling – windows vista scheduling.

UNIT V:
I/O management and Disk scheduling – I/O devices – organization of I/O function – operating system design issues – I/O buffering – disk scheduling – RAID – disk cache – windows vista I/O. File management – overview – file organization and access – file directories – file sharing – record blocking –secondary storage management – file system security – windows vista file system.

Text book:

Operating Systems - Internals and Design Principles, William Stallings, Sixth Edition, Pearson Education Ltd, 2014

UNIT I : Chapter 1.1 to 1.7, 2.1 to 2.5
UNIT II : Chapter 3.1 to 3.6, 4.1 to 4.4, 5.1 to 5.6
UNIT III : Chapter 6.1 to 6.6, 6.10, 7.1 to 7.5, 8.1, 8.2, 8.5
UNIT IV : Chapter 9.1, 9.2, 10.1, 10.2, 10.5
UNIT V : Chapter 11.1 to 11.7, 11.10, 12.1 to 12.7, 12.10

Reference book:

1. Charles crowley, "Operating system – A design oriented approach", TMH, 2009

ES 1.1: EMBEDDED SYSTEMS

(5 Hours – 4 Credits)

UNIT I:
Introduction to 8051 microcontroller: Comparison between micro controller and general purpose microprocessor; different types of microcontrollers; Architecture of 8051; key features of 8051; I/O ports; memory organization; counters and timers; serial I/O ports; interrupts of 8051.

UNIT II:
8051 Instruction Set & Assembly Language programming:Addressing modes of 8051; instruction set; data move; arithmetic; logical; jump and call Instructions; Program for data transfer; Memory operations; arithmetic; logical; sorting.

UNIT III:
Programming and Debugging Using Keil C: Different types of Header files, declaration of variables, operators, Macro declaration; inclusion of files; I/O functions; String functions; Basic

debugging concept; Logic analyzer programming; Timer simulation; I/O port simulation and debug; Program for RPM counting; Program for PWM.

UNIT IV:

External Peripheral Interfacing: Interfacing switches; LEDs; Matrix Keyboard; Seven Segment Displays; 16 x 2 LCD; pulse measurement; analog to digital and digital to analog converters; interrupt programming; PC interfacing.

UNIT V:

Real Time Software Development: Architecture: Study of different architectures; simple Round Robin; Round Robin with Interrupt; Token passing method; Semaphores; Interrupt Latency; RTOS; RTOS applications;VxWorks RTOS study; RTC interfacing with RTOS; Selection procedure for Microcontrollers; SPI mode of operation.

Text Book:

The 8051 microcontroller Architecture programming and Applications, Kenneth J Ayala,.,Penram International Publishing Pvt. Ltd., 2005.

Reference Books:

1. Douglas V Hall, (2000). Microprocessor and Interfacing. (3rded.). Tata McGraw Hill.
2. Muhammad Ali Mazidi and Mazidi & McKinlayR.D, (2006).The 8051 Microcontroller and Embedded system.
3. Raj Kamal, (2005), Microcontroller Architecture programming Interfacing and system design,Pearson Education.
4. Ram.B, (2000), Fundamentals of Microprocessor & Microcomputer. DanpatRai Publication.

ES 1.2: ADVANCED SOFTWARE ENGINEERING

(5 Hours – 4 Credits)

UNIT I:

Phases in software development – Requirement Analysis – Software design – Coding – Testing– Maintenance – Effort Distribution with Phases – Error Distribution – Software Development Process Model: Waterfall model – Prototyping Interactive Enhancement – Spiral Model – Role of Management in Software Development, Metrics and Measurements – Software Requirements Specifications (SRS) – Role of SRS.

UNIT II:

Problem Analysis: Structuring Information – Data flow Diagram and Data Dictionary – Structured Analysis – Prototyping Requirements Specification Characteristics of an SRS – Specification Languages Structure of Requirements Document – Validation: Reading – construction scenarios – Requirement Review Automated cross Referencing – Prototyping – Metrics: Function Points – Number of Errors found – Change request frequency.

UNIT III:
Planning a Software project – Cost estimation – Uncertainties in cost estimation – Single variable Models: COCOMO Model – software size Estimation – Project Scheduling: Average Duration Estimation – Milestones, Staffing and Personnel planning – Raleigh Curve – Team structure – Software configuration Management configuration identification – configuration control – status accounting and auditing – software configuration and management – Quality assurance plans: verification and validation – Inspection and reviews – Output of a software development project – Project monitoring plans: Timesheets – Reviews – Cost – Schedule – Milestone Graph – Risk Management: Risk Management Activities – Risk Identification – Risk Analysis and Prioritization – Project planning and Risk management.

UNIT IV:
System Design: Design Objectives, Design Partitioning – Problem Partitioning – Abstraction, Top-Down and Bottom-Up strategies, Module Level Concepts – Coupling and Cohesion, Design Methodology – Structured Design – Structure Charts – Design Methodology – Transaction Analysis, Design Specification, Verification – Design Reviews – Automated Cross-Checking.

UNIT V:
Testing Fundamentals: Error Fault – Failures – Reliability – Levels of Testing – Test case and Test criteria – Test Oracle – Psychology of Testing – Top-Down and Bottom–Up Approaches– Functional Testing: Equivalence class portioning – Boundary value Analysis: case Effect Graphing – Test case Generations – Instrumentation for structural testing – Complexity Based Criteria – Mutation Testing – Combination Functional and structural Approaches, Testing Process – Test Plan – Test case Specification and Test case – Execution and Analysis, comparison of different V & V Techniques, Matrices, Reliability Assessment – Programmer Productivity – Error Removal Efficiency – Specifications for system testing – System Test Report – Error Report on a given problem.

Text Book:
An Integrated Approach to Software Engineering, Pankej Jalote, 2nd Edition, Narosa Publishing House, New Delhi 1997

Reference Books:
1. Richard E. Fairley, "Software Engineering – A practitioner's approach", McGraw Hill 1982
2. Martin L Shooman, "Software Engineering – Design, Reliability and Management" McGraw Hill 1983

ES1. 3: DISTRIBUTED SYSTEMS
(5 Hours – 4 Credits)

UNIT I:
Introduction to Distributed System: Goals, Hardware concepts, Software concepts, **and** Client-Server model. Examples of distributed systems. **Communication:** Layered protocols, Remote procedures call, Remote object invocation, Message-oriented communication, Stream-oriented communication.

UNIT II:

Processes: Threads, Clients, Servers, Code Migration, Software agent. **Naming:** Naming entities, Locating mobile entities, Removing un-referenced entities.

UNIT III:

Synchronization: Clock synchronization, Logical clocks, Global state, Election algorithms, Mutual exclusion, Distributed transactions. **Consistency and Replication:** Introduction, Data centric consistency models, Client centric consistency models, Distribution protocols, Consistency protocols.

UNIT IV:

Fault Tolerance: Introduction, Process resilience, Reliable client server communication, Reliable group communication. Distributed commit, Recovery. **Security:** Introduction, Secure channels, Access control, Security management.

UNIT V:

Distributed File System: Sun network file system, CODA files system. **Case Study:** CORBA, Distributed COM, Globe, Comparison of CORBA, DCOM, and Globe.

Text Book:

"Distributed Systems: Principles and Paradigms" A.S. Tanenbaum and M. van Steen, Pearson/Prentice-Hall, 2nd Edition, 2007.

UNIT I	-	Chapters 1,2 and 4
UNIT II	-	Chapters 3 and 5
UNIT III	-	Chapters 6 and 7
UNIT IV	-	Chapters 8 and 9
UNIT V	-	Chapters 11

Reference Books:

1. "Distributed Systems: Concepts and Design" G. Coulouris, J. Dollimore, and T. Kindberg, , 5th edition, Addison-Wesley is an imprint of Pearson, 2012.
2. "Advanced Concepts in Operating Systems" M. Singhal, N. Shivaratri, , McGraw-Hill Education (India) Pvt Limited, 2001.

CS 24: LAB3.ADVANCED JAVA PROGRAMMING

(5 Hours – 3 Credits)

1. Program to display life cycle of an applet
2. Program to display digital clock using applet
3. Program to display different graphical shapes in applet
4. Program to display graphical bar chart by passing parameters in applet
5. Program to find factorial value of N using AWT high level event handling
6. Program to illustrate window closing using AWT low level event handling
7. Program to illustrate TCP based network communication
8. Program to illustrate UDP based network communication

9. Program to find sum of digits using RMI
10. Program to find length of the given string using RMI
11. Program using HTML/Java script to find length of the given string
12. Program using HTML/Java script to find biggest element of an array
13. Program to compute factorial value of N using GenericServlet
14. Program to compute factorial value of N using HTTPServlet

Note: The above are sample problems, Instructor can add more exercises on their requirements and to the technology

CS 25:LAB 4. OPERATING SYSTEM LAB

(5 Hours – 3 Credits)

1. Write programs using the following system calls of UNIX operating system: fork, exec, getpid, exit, wait, close, stat, opendir, readdir
2. Write programs using the I/O System calls of UNIX operating system. (open, read, write, etc)
3. Write C Program to implement fork(), getpid() and wait().
4. Write C program to simulate UNIX command: ls.
5. Write C program to simulate UNIX command: grep.
6. Given the list of processes, their CPU burst times and arrival times. Display/print the Gantt chart for FCFS. Compute and print the average waiting time and average turnaround time.
7. Given the list of processes, their CPU burst times and arrival times. Display/print the Gantt chart for SJF. Compute and print the average waiting time and average turnaround time.
8. Given the list of processes, their CPU burst times and arrival times. Display/print the Gantt chart for Priority Scheduling. Compute and print the average waiting time and average turnaround time (2 sessions).
9. Given the list of processes, their CPU burst times and arrival times. Display/print the Gantt chart for Round robin. Compute and print the average waiting time and average turnaround time (2 sessions).
10. Develop Application using Inter-Process-Communication (Using shared memory, pipes or message queues).
11. Implement the Producer-Consumer problem using semaphores (Using UNIX system calls)
12. Implement some Memory management schemes like Paging and Segmentation.
13. Implement some Memory management schemes like FIRST FIT, BEST FIT & WORST FIT.
14. Implement any file allocation techniques(Contiguous, Linked or Indexed)
15. Write a shell script program to display the process attributes, to change the priority of processes and to change the ownership of processes.

<u>Example for exercises 12 & 13:</u>

Free space is maintained as a linked list of nodes with each node having the starting byte address and the ending byte address of a free block. Each memory request consists of the process-id and the amount of storage space required in bytes. Allocated memory space is again maintained as a linked list of nodes with each node having the process-id, starting byte address and the ending byte address of the allocated space. When a process finishes (taken as input) the appropriate node from the allocated list should be deleted and this free space should be added to the free space list (care should be taken to merge contiguous free blocks into one single block. This results in deleting more than one node from the free space list and changing the start and end address in the appropriate node). For allocation use first fit, worst fit and best fit.

Note: The above are sample problems, Instructor can add more exercises on their requirements and to the technology

CS 22: COMPILER DESIGN
(5 Hours – 4 Credits)

UNIT I:

Compilers And Translators-Why Do We Need Translators?-The Structure Of A Compiler-Lexical Analysis-Syntax Analysis-Intermediate Code Generation-Optimization-Code Generation-Book Keeping-Error Handling-Compiler-Writing Tools-Getting started. The role of the lexical analyzer-Simple approach to design of a lexical analyzer-Regular Expressions-Finite Automata-From regular expression to finite automata-Minimizing the number of states of a DFA-A language for specifying lexical analyzer-Implementing a lexical analyzer- The scanner generator as Swiss army Knife.

UNIT II:

The Syntactic Specification of Programming Languages-Derivation and Parse Trees-Capability of context free Grammars. Parsers-Shift-reduce Parsing-Operator-precedence parsing-Top-down parsing-Predictive Parsers.

UNIT III:

LR parsers-The canonical collection of LR(0) items-constructing SLR parsing tables – constructing canonical LR parsing tables-constructing SLR parsing tables-constructing LALR parsing tables – Using Ambiguous grammars- An automatic parse generator Implementation of LR parsing Tables – constructing LALR set of items. Syntax directed translation schemes – Implementation if syntax directed schemes-Intermediate Code-Parse Tree and Syntax Trees – Three Address code, quadruples, and triples-Translation of assignment statements-Boolean Expression-Statements that alter the flow of control-postfix translations-Translation with a top-down parser.

UNIT IV:

The contents of a symbol tables-Data structure for a symbol table-Representing Scope information. Errors-Lexical-phase errors - syntactic-phase errors-Semantic errors. The principal sources of optimization-Loop optimization -The DAG representation of basic blocks-Value numbers and algebraic laws-Global data-flow analysis.

UNIT V:

Dominators-Reducible Flow graphs -Depth-first search-Loop-invariant computations –Induction variable elimination-Some other loop optimization. Code Generation-Object Programs- A machine Model- A simple code generator-Register allocation and assignment-Code generation from DAG's-Peephole Optimization.

Text Book:
Principles of Compiler Design, Alfred V.Aho and Jeffrey D.Ullman.25th Reprint, 2002.

UNIT I: Chapter 1,3
UNIT II: Chapter 4,5
UNIT III: Chapter 6,7
UNIT IV: Chapter 9,11,12
UNIT V: Chapter 13,15

Reference Books:

1. Compiler Design in C ,Allen I. Holub Prentice Hall of India, 2003.
2. Crafting a compiler with C ,C. N. Fischer and R. J. LeBlanc, Benjamin Cummings, 2003.
3. Introduction to Compiler Techniques, J.P. Bennet, Second Edition, Tata McGraw-Hill, 2003.

CS 31: DIGITAL IMAGE PROCESSING

(5 Hours – 4 Credits)

UNIT I:
Digital Image Processing: Origins of Digital Image Processing, Steps in Digital Image Processing, Digital Image Fundamentals: Elements of Visual Perception, Light and the Electromagnetic Spectrum, Image Sensing and Acquisition, Image Sampling and Quantization, Basic Relationships between Pixels, Mathematical Tools used in Digital Image Processing.

UNIT II:
Image Transformation & Filters: Basic Intensity Transformation Functions, Histogram Processing, Fundamentals of Spatial Filtering, Smoothing Spatial Filter, Sharpening Spatial Filters, Combining Spatial Enhancement methods, Fuzzy techniques for Intensity Transformation and Spatial Filtering. Filtering in the Frequency Domain: Preliminary Concepts, Sampling and the Fourier Transforms of Sampled Functions, The Discrete Fourier Transform (DFT), Properties of the 2-D DFT, Filtering in the Frequency Domain, Image Smoothing and Sharpening using Frequency Domain Filters, Selective Filtering.

UNIT III:
Image Restoration, Reconstruction and Image Segmentation: Image Degradation/Restoration process, Noise Models, Restoration in the presence of Noise only-Spatial Filtering, Periodic Noise Reduction by Frequency Domain Filtering, Linear, Position-Invariant Degradations, Estimating the Degradation Functions, Inverse Filtering, Wiener Square Error Filtering, Constrained Least Square Filtering, Geometric Mean Filter, Image Reconstruction from Projections. Image Segmentation: Point, Line and Edge Detection, Thresholding, Region-Based Segmentation, Segmentation Using Morphological Watersheds, Use of Motion in Segmentation.

UNIT IV:

Color Image Processing: Color Fundamentals, Color Models, Pseudocolor Image Processing, Full Color Image Processing, Color Transformation, Smoothing and Sharpening, Image Segmentation Based on Color, Noise in Color Images. Wavelets and Multiresolution Processing: Multiresolution Expansion, Wavelet Transforms in One Dimension, The Fast Wavelet Transforms, Wavelet Transforms in Two Dimensions, Wavelet Packets. Image Compression: Fundamentals, Basic Compression Methods, Digital Image Watermarking.

UNIT V:

Morphological Image Processing: Erosion and Dilation, Opening and Closing, The Hit-Or-Miss Transformation, Basic Morphological Algorithms, Gray-Scale Morphology. Object Recognition: Patterns and Pattern Classes, Recognition Based on Decision-Theoretic Methods, Structural Methods.

Text Book:

Digital Image Processing, Rafael C. Gonzalez, Richard E. Woods, 3rd Edition, Pearson Education, 2008.

UNIT I : Chapter 1 and 2
UNITII: Chapter 3 and 4
UNIT III: Chapter 5 and 10
UNIT 1V :Chapter 6 and 7
UNIT V : Chapter 9 and 12

Reference Books:

1. Digital Image Processing using MATLAB Rafael C. Gonzalez, Richard E. Woods,2nd Edition, Prentice Hall of India, 2002.

2. Fundamentals of Digital Image Processing A.Jain, Prentice Hall of India.

CS 32: SOFT COMPUTING
(5 Hours – 4 Credits)

UNIT I :

Introduction - Neural Networks – Application scope of Neural Networks – Fuzzy logic – Genetic Algorithm – Hybrid Systems – Soft Computing. Artificial Neural Network – An Introduction– Fundamental Concept – Evolution of Neural Networks – Basic models of Artificial Neural Network– Important terminologies of ANNs – McCulloch-Pitts Neuron – Linear Separability – Hebb Network.

UNIT II:

Supervised Learning Network – Introduction – Perceptron Networks – Adaptive Linear Neuron – Multiple Adaptive Linear Neurons – Back Propagation Network.

UNIT III:
Introduction to classical sets and Fuzzy Sets – Introduction – Classical sets (Crisp Sets) – Fuzzy Sets – Classical Relations and Fuzzy Relations – Introduction – Cartesian Product of Relation – Classical Relation – Fuzzy Relation.

UNIT IV:
Genetic Algorithm – Introduction – Basic Operators and Terminologies in Genetic Algorithms – Traditional Algorithm Vs Genetic Algorithm – Simple Genetic Algorithm – General Genetic Algorithm – The Schema Theorem – Classification of Genetic Algorithm – Holland Classifier Systems – Genetic Programming – Applications of Genetic Algorithm.

UNIT V:
Application of Soft Computing – Introduction – A fusion Approach of Multispectral Images with SAR (Synthetic Aperture Radar) Image for Flood Area Analysis – Optimization of Traveling Salesman Problem using Genetic Algorithm Approach – Genetic Algorithm based Internet Search Technique.

Text Book:
Principles of Soft Computing, S.N.Sivanadam & S.N.Deepa. First Edition , Wiley India, 2007.

UNIT I :Chapter 1, 2
UNIT II :Chapter 3
UNIT III :Chapter 7, 8
UNIT IV :Chapter 15
UNIT V :Chapter 16

Reference Book:
Soft Computing and Its Applications, R AAliev& R RAliev. Second Edition, world scientific, 2012.

ES2. 1: INFORMATION RETRIEVAL
(5 Hours – 4 Credits)

UNIT I:
Boolean retrieval: An example information retrieval problem- A first take at building an inverted index- Processing Boolean queries- The extended Boolean model versus ranked retrieval. **The term vocabulary and postings lists:** Document delineation and character sequence decoding- Determining the vocabulary of terms- Faster postings list intersection via skip pointers- Positional postings and phrase queries. **Dictionaries and tolerant retrieval:** Search structures for dictionaries- Wildcard queries- Spelling correction- Phonetic correction.

UNIT II:
Scoring, term weighting and the vector space model: Parametric and zone indexes- Term frequency and weighting- The vector space model for scoring- Variant tf-idf functions. **Computing scores in a complete search system:** Efficient scoring and ranking- Components of an information retrieval system- Vector space scoring and query operator interaction.

UNIT III:

Text classification and Naive Bayes: The text classification problem- Naive Bayes text classification- The Bernoulli model- Feature selection- Evaluation of text classification. **Vector space classification:** Document representations and measures of relatedness in vector spaces- Rocchio classification-k nearest neighbour- Classification with more than two classes- The bias-variance tradeoff.

UNIT IV:

Support vector machines and machine learning on documents: Support vector machines: The linearly separable case- Extensions to the SVM model- Issues in the classification of text documents- Machine learning methods in ad hoc information retrieval. **Flat clustering:** Clustering in information retrieval- Problem statement- Evaluation of clustering- K-means.

UNIT V:

Hierarchical clustering: Hierarchical agglomerative clustering- Single-link and complete-link clustering- Group-average agglomerative clustering- Centroid clustering- Divisive clustering- Cluster labeling.

Text Book:

Introduction to Information Retrieval, Christopher D. Manning, Prabhakar Raghavan and Hinrich Schutze, Cambridge University Press, 2014.

UNIT I: Chapters-1, 2, 3.
UNIT II: Chapters- 6, 7
UNIT III: Chapters-13, 14
UNIT IV: Chapters-15, 16
UNIT V: Chapters-17

Reference Books:

1. Information Retrieval- David A. Grossman and Ophir Frieder, Springer, 2003.
2. Modern Information Retrieval- Ricardo Baeza-Yates, Berthier Ribeiro-Neto, Pearson Edition-2003.

ES 2.2: NETWORK SECURITY
(5 Hours – 4 Credits)

UNIT I:

Introduction: Security Goals – Attacks – Services and Mechanism – Techniques. **Mathematics of Cryptography:** Integer Arithmetic – Modular Arithmetic – Matrices – Linear Congruence – Traditional Symmetric Key Ciphers: Instruction – Substitution Ciphers – Transposition Ciphers – Stream and Block Ciphers. **Introduction to Modern Symmetric Key Ciphers:** Modern Block Ciphers – Modern Stream Ciphers.

UNIT II:

Data Encryption Standard (DES): Introduction – DES Structure – DES Analysis – Multiple DES – Security of DES. **Advanced Encryption Standard (AES):** Introduction – Transformations – Key Expansion – Ciphers – Examples – Analysis of AES.

UNIT III :

Asymmetric Key Cryptography: Introduction – RSA Crypto System. **Message Integrity and Message Authentication:** Message Integrity – Random Oracle Model – Message Authentication.

UNIT IV:

Cryptographic Hash Functions: Introduction – SHA – 512 – WHIRLPOOL. **Digital Signature:** Comparison – Process – Services – Attacks on Digital Signature – Digital Signature Schemes.

UNIT V:

Entity Authentication: Introduction – Passwords – Challenge Response – Zero Knowledge – Bio Metrics. **Key Management:** Symmetric Key Distribution – Kerberos – Symmetric Key Agreement – Public Key Distribution.

Text Book:

Cryptography and Network Security – Behrouz A. Forouzan, TheMcGraw Hill, 2011.

UNIT I – Chapter 1,2,3,5
UNIT II – Chapter 6, 7
UNIT III – Chapter 10, 11
UNIT IV – Chapter 12, 13
UNIT V – Chapter 14, 15

Reference Book:

Cryptography and Network Security – William Stallings, PHI, 2008.

ES 2.3: INTERNET OF THINGS
(5 Hours – 4 Credits)

UNIT I:

Introduction to Internet of Things: Introduction – Physical Design of IoT – Logical Design of IoT – IoT Enabling Technologies – IoT & Deployment Templates. **Domain Specific IoTs:** Introduction – Home Automation – Cities – Environment – Energy – Retail – Logistics – Agriculture – Industry – Health & Life style.

UNIT II:

IoT and M2M : Introduction : M2M – Difference between IoT and M2M – SDN and NFV for IoT.

IoT System Management with NETCONF-YANG : Need for IoT Systems Management – Simple Network Management Protocol (SNMP) – Network Operator Requirements – NETCONF- YANG – IoT Systems Management with NETCONF_YANG.

UNIT III:

IoT Platforms Design Methodology: Introduction – IoT Design Methodology – Case Study on IoT System for Weather Monitoring – Motivation for using Python. **IoT Systems – Logical Design using Python:** Introduction – Installing Python – Python Data types & Data Structures – Control Flow – Functions – Modules – Packages – File Handling – Date/Time Operations – Classes – Python packages of Interest for IoT.

UNIT IV:

IoT Physical Devices & Endpoints: What is an IoT Device – Exemplary Device: Raspberry Pi – About the Board – Linux on Raspberry Pi – Raspberry Pi Interfaces – Programming Raspberry Pi with Python – Other IoT devices. **IoT Physical Servers & Cloud Offerings :** Introduction to Cloud Storage Models & Communication APIs – WAMP - AutoBahn for IoT – Xively Cloud for IoT – **Python Web application** Framework-Django – Designing a REST ful Web API – Amazon Web Services for IoT – **Skynet**IoT messaging platform.

UNIT V:

Case Studies Illustrating IoT Design: Introduction – Home Automation – Cities – Environment – Agriculture – Productivity applications. **Data Analytics for IoT :** Introduction – Apache Hadoop – Using Hadoop MapReduce for Batch Data Analysis – Apache Oozier – Apache Spark – Apache Storm – Using Apache Storm for Real-time Data Analysis.

Text Book:

1. Internet of Things, Arshdeep Bahga, Vijay Madisetti, Universities Press(INDIA) Private Ltd., 2015.

UNIT I : Chapters 1 &2
UNIT II: Chapters 3 & 4
UNIT III: Chapters 5 & 6
UNIT IV: Chapters 7 & 8
UNIT V: Chapters 9 & 10

Reference Books:

1. Getting Started with the Internet of Things, CunoPfister, O'Relly, 2011.
2. Designing the Internet of Things,AdrianMcewen, HakinCassimally, Willey,2015.
3. The Internet of Things in the Cloud: A Middleware Perspective,Honbo Zhou, CRC Press, 2012.
4. Architecting the Internet of Things, Dieter Uckelmann; Mark Harrison; Florian Michahelles, (Eds.) Springer, 2011.

NME: INFORMATION TECHNOLOGY AND DATA COMPUTATION

(4 Hours – 4 Credits)

UNIT I:

Introduction: History of computers – classification of computers – Basic organization: memory – RAM , ROM , PROM , EPROM, EEPROM and types– Secondary storage devices: magnetic – floppy – hard disks . Optical: CDROM , WORM, and types . Concept of virtual memory and cache memory and their needs – Computer arithmetic: Number systems – binary, octal. hexadecimal. Introduction to computer languages:- definition of assembler, complier, interpreters.

UNIT II:

Database Systems: Database – Database Management System – Classical File Based Systems – Modern DBMS approach – Significance of Database Systems. Modelling a Database: Representation of data – Data Models – Entity Relationship Model (E-R Model) – Logical Data Models – comparison of Logical Data Models- other data models.Modelling with E-R Diagram: E-R Based Modelling – symbolic representation of E-R components- Mapping between entities – Modelling with E-R Diagram – Advantages and Disadvantages of E-R Model.

Principles of Relational Database Management Systems: Foundations of Relational Model – Application areas of RDBMS – Advantages of RDBMS – RDBMS Packages and their Developers – CODD's rules for RDBMS – Relational operations.

Relational Database Design: Designing a Database –Functional Dependency-Relational Decomposition – Normalization- importance of Normalization- limitations of Normal Forms.

UNIT III:

Multimedia Applications: Text: Elements of Text- Text data Files-Hypertext. Graphics: Element of Graphics - Graphics Files and Application Formats. Digital Audio: Characteristics of sound and Digital Audio-MIDI-Audio File Formats. Digital Video and Animation: Background on Video - Characteristics of Digital video - Computer Animation

UNIT IV:

Computer Network – Overview of Network – Communication Medium – Types of Networks – Network Topology – Network Protocols.

UNIT V :

Introduction to SPSS: Starting SPSS – SPSS main menu – Working with data editor – SPSS viewer – Import and export the data. Basic statistical concepts: Research in behavioral science – Qualitative research – Quantitative variables – Reliability and validity- Hypothesis testing. Descriptive statistics: Basic concepts of descriptive statistics – Descriptive statistics using SPSS. Comparing means: One or two sample t-tests: Basic concepts of t-tests – implementing t-tests using SPSS. Comparing means: Analysis of variance: Basic concepts of ANOVA procedure – one way and two way ANOVA. Chi-square test for Independence discrete data: Basic concepts of chi-square test – implementing chi-square test using SPSS. Correlation Analysis: Basic concepts of correlation – Bivariate and Partial correlation using SPSS. Multiple regression: Basic concepts of regression coefficient, R-values and design issues- Standard multiple regression and hierarchical regression. Logistic regression: Basic concepts of logistic regression, fit indices and types of logistic regression – implementing logistic regression using SPSS. Data reduction and scale reliability: Factor analysis: Basic concepts of factor analysis – Factor analysis and scale reliability using SPSS.

Text Book:

1. Fundamentals of Information Technology, Alexis Leon, Mathews Leon, Leon Vikas Ltd, Second Edition, 2009.
2. Introduction to Information Systems, Alexis Leon, Mathews Leon, Vijay Nicole Imprints Pvt. Ltd, Second Reprint, 2009.
3. Database Management System, Malay K. Pakhira, PHI Learning Private Limited, New Delhi, 2013.
4. Multimedia Technology and Applications by David Hillman, Reprint 2012.
5. Statistical Methods for Practice and Research : A guide to data analysis using SPSS, AjaiS.Gaur and SanjayaS.Gaur, SAGE Publications, New Delhi, 2009.

Reference Book:

Database Management Systems, G.K. Gupta, TATA McGraw Hill Education Private Limited, New Delhi, 2011.

CS 33:LAB 5. SOFT COMPUTING
(5 Hours – 3 Credits)

Section - A (Fuzzy Logic)

1. a) Write a program (m.file) to calculate union, intersection, complement and difference of two fuzzy sets.
   b) Write a program (m.file) to calculate the Demorgan's Law.
2. Find whether the given matrix is (a) reflexive (b) tolerance and (c) transitivity matrix or not.
3. Find whether the given matrix is symmetry or not.
4. Find the fuzzy relation between two vectors *R* and *S*

   R =
   0.7 0.5
   0.8 0.4
   S =
   0.9 0.6 0.2
   0.1 0.7 0.5
   Using max–product and max-min method

5. a) Use command line commands to display the Gaussian membership function. Given *x* = 0–10 with increment of 0.1 and Gaussian function is defined between 0.5 and −5.
   b) Use command line commands to display the triangular membership function. Given *x* = 0–10 with increment of 0.2 triangular membership function is defined between [3 4 5]

6. Illustrate different types of generalized bell membership functions using a program
7. Using program find the crisp lambda cut set relations for $\lambda = 0.2$, the fuzzy matrix is given by

$$R= \begin{matrix} 0.2 & 0.7 & 0.8 & 1 \\ 1 & 0.9 & 0.5 & 0.1 \\ 0 & 0.8 & 1 & 0.6 \\ 0. & 0.4 & 1 & 0.3 \end{matrix}$$

8. Temperature control of the reactor where the error and change in error is given to the controller. Here the temperature of the reactor is controlled by the temperature bath around the reactor thus the temperature is controlled by controlling the flow of the coolant into the reactor. Form the membership function and the rule base using FIS editor.
9. Consider the water tank with following rules
   a) IF (level is okay) THEN (valve is no_change) (1)
   b) IF (level is low) THEN (valve is open_fast) (1)
   c) IF (level is high) THEN (valve is close_fast) (1)
   Using Mamdani method and max–min method for fuzzification and method of centroid for defuzzification method construct a FIS. Before editing that rules, membership functions must be defined with membership function editor.

10. a) Form a fuzzy system, which approximates function *f*, when $x \in [-10, 10]$.Repeat the same by adding random, normally distributed noise with zero mean and UNIT variance.
    b) Simulate the output when the input is sin(*t*). Observe what happens to the signal shape at the output.
11. Use Fuzzy Logic Toolbox to model the tip given after a dinner for two, where the food can be disgusting, not good, bland, satisfying, good, or delightful, and the service can be poor, average, or good. To get started, you type fuzzy in a window. Then use the fuzzy inference system and membership function editors to define and tune your rules.

Section - B (Neural Network)

12. Design networks of McCulloch-Pitts neurons that implement logical NOT, AND and OR gates. Draw each network and label all the weight and threshold values.

13. Derive expressions for the weights and thresholds of a McCulloch-Pitts neuron that can compute the following input-output mappings:

in1	*in2*	*out*
0	0	1
0	1	0
1	0	0
1	1	0

Write code for the above ANN.

14. Investigation the use of back-propagation learning using a sigmoidal nonlinearity to achieve one-to-one mapping, as described here:

1. $f(x) = 1/x$` $1 \leq x \leq 100$
2. $f(x) = \log_{10} x$, $1 \leq x \leq 10$
3. $f(x) = \exp(-x)$, $1 \leq x \leq 10$
4. $f(x) = \sin x$, $0 \leq x \leq \pi/2$

For each mapping, do the following:

(a) Set up two sets of data, one for network training, and the other for testing.

(b) Use the training data set compute the synaptic weights of the network, assumed to have a single hidden layer.

(c) Evaluate the computation accuracy of the network by using the test data. Use a single layer but with a variable number of hidden neurons. Investigate how the network performance is affected by varying the size of the hidden layer.

15. The data presented in the Table P4.17 show the weights of eye lenses of wild Australian rabbits as a function of age. No simple analytical function can exactly interpolate these data, because we do not have a single valued function. Instead, we have a nonlinear least squares model of this data set, using a negative exponential, as described by $Y = 2.33.846(1 - \exp(-0.006042x)) + \in$

Where $\in$ is an error term.

Using the back- propagation algorithm, design a multiplayer perceptron that provides a nonlinear least-squares approximation to this data set. Compare your result against the least-sequence model described.

Table P4.17 Weights of Eye Lenses of Wild Australian Rabbits

Ages (days)	Weights (mg)	Ages (days)	Weights (mg)	Ages (days)	Weights (mg)	Ages (days)	Weights (mg)
15	21.66	75	94.6	218	174.18	338	203.23
15	22.75	82	92.5	218	173.03	347	188.38
15	22.3	85	105	219	173.54	354	189.7
18	31.25	91	101.7	224	178.86	357	195.31
28	44.79	91	102.9	225	177.68	375	202.63
29	40.55	97	110	227	173.73	394	224.82
37	50.25	98	104.3	232	159.98	513	203.3
37	46.88	25	134.9	232	161.29	535	209.7
44	52.03	142	130.68	237	187.07	554	233.9
50	63.47	142	140.58	26	176.13	591	234.7
50	61.13	147	155.3	258	183.4	648	244.3
60	81	147	152.2	276	186.26	660	231
61	73.09	150	144.5	285	189.66	705	242.4
64	79.09	159	142.15	300	186.09	723	230.77
65	79.51	165	139.81	301	186.7	756	242.57
65	65.31	183	153.22	305	186.8	768	232.12
72	71.9	192	145.72	312	195.1	860	246.7
75	86.1	195	161.1	317	216.41		

Section - C (Genetic Algorithm)

16. Write a program to implement Roulette wheel and ranking selection method.
17. Write a program to maximize a function
    $f(x,y)=x\sin(4\pi x) + y\sin(20\pi x)$ subject to
    $-3.0 \leq \leq x \leq 12.1$
    $4.1 \leq y \leq 5.8$

Note: The above are sample problems, Instructor can add more exercises on their requirements and to the technology

CS 34:LAB 6. IMAGE PROCESSING
(6 Hours – 3 Credits)

1. Implement the spatial image enhancement functions on a bitmap image – Mirroring (Inversion)
2. Implement the spatial image enhancement functions on a bitmap image – Rotation (Clockwise)
3. Implement the spatial image enhancement functions on a bitmap image – Enlargement (Double Size)
4. Implement (a) Low Pass Filter (b) High Pass Filter
5. Implement (a) Arithmetic Mean Filter (b) Geometric Mean Filter
6. Implement Smoothing and Sharpening of an eight bit color image
7. Implement (a) Boundary Extraction Algorithm (b) Graham's Scan Algorithm
8. Implement (a) Edge Detection (b) Line Detection
9. Display an image and its histogram
10. Write a Program to Perform shrinking, zooming and cropping of an image
11. Write a Program to Perform the experiment for histogram equalization.
12. Write a Program to Perform blurring and de-blurring on an image.
13. Write a Program to Remove salt and pepper noise in an image.
14. Write a Program to Perform Edge detection using Operators.
15. Write a Program to Perform 2-D DFT and DCT.
16. Write a Program to Perform DWT of images.
17. Implement a function for image segmentation.
18. Implement a function for image morphology that analyze the form and shape detail of image structures.
19. Implement a function for Image Restoration.
20. Models for representing the color and methods of processing the color plane

Note: The above are sample problems, Instructor can add more exercises on their requirements and to the technology

CS 41: PATTERN RECOGNITION

(5 Hours – 4 Credits)

UNIT I:

Pattern Classifier : Introduction - Data Sets for Pattern Recognition - Different Paradigms for Pattern Recognition - Representation -Data Structures for Pattern Representation - patterns as Vectors- Patterns as Strings - Logical Descriptions - Fuzzy and Rough Pattern Sets -Patterns as Trees and Graphs - Representation of Clusters - Proximity Measures - Distance Measure - Weighted Distance Measure - Non-Metric Similarity Function - Edit Distance -Mutual Neighbourhood Distance (MND) - Conceptual Cohesiveness - Kernel Functions - Size of Patterns - Normalisation of Data -Use of Appropriate Similarity Measures - Abstractions of the Data Set.

UNIT II:

Clustering : Introduction to Clustering- Hierarchical Algorithms - Divisive Clustering - Agglomerative Clustering - Partitional Clustering - k-Means Algorithm - Soft Partitioning- Clustering Large Data Sets - Possible Solutions - Incremental Clustering - Divide-and-Conquer Approach.

UINT III:

Linear Classifiers: Introduction - Linear Discriminant Functions and Decision Hyperplanes - The Perceptron Algorithm - Least Squares Methods –STOCHASTIC Approximation and LMS algorithm - Mean Square Error Estimation - Mean Square Error Regression -The Bias-Variance Dilemma -Separable Classes - Nonseparable Classes.

UNIT IV:

Hidden Markov Models And Support Vector Machine: Markov Models for Classification - Hidden Markov Models - HMM Parameters - Learning HMMs - Classification Using HMMs - Classification of Test Patterns.

Linear Discriminant Functions - Learning the Linear Discriminant Function - Learning the Weight Vector - Multi-class Problems - Generality of Linear Discriminants - SVM for Classification - Linearly Separable Case - Non-linearly Separable Case.

UNIT V:

Feature Selection and Extraction: Feature selection - Feature selection criteria - Search algorithms for feature selection -Suboptimal search algorithms - Linear feature extraction - Principal components analysis -Karhunen–Loeve transformation -Factor analysis- Multidimensional scaling - Classical scaling -Metric multidimensional scaling -Ordinal scaling- Algorithms -Multidimensional scaling for feature extraction.

Text Book:

1. Pattern Recognition, M. Narasimha Murthy and V. Susheela Devi, Springer 2011.
2. Pattern Recognition, S.Theodoridis and K.Koutroumbas, 4th Ed., Academic Press, 2009.
3. Stastical Pattern Recognition ,Andrew Webb, Arnold publishers, London,1999.

UNIT I : Chapters 1,2 (Text Book 1)
UNIT II : Chapters 9 (text book 1)
UNIT III : Chapters 3 (text book 2)
UNIT IV : Chapter 5,7(text book 1)
UNIT V : Chapter 9 (textbook 3)

Reference Books:

1. Pattern Recognition Statistical, Structural and Neural Approaches Robert J.Schalkoff, John Wiley & Sons Inc., New York, 1992.
2. Pattern Recognition and Machine Learning C.M.Bishop, Springer, 2006.
3. Pattern Classification ,R.O.Duda, P.E.Hart and D.G.Stork, John Wiley, 2001.
4. Statistical Pattern Recognition,Andrew Webb, Arnold publishers, London,1999.

CS 42: ADVANCED SYSTEM ARCHITECTURE

(5 Hours – 4 Credits)

UNIT I:

Introduction to Advanced Computer Architecture and Parallel Processing : Four Decades of Computing - Flynn's Taxonomy of Computer Architecture - SIMD Architecture - MIMD Architecture - Interconnection Networks **Multiprocessors Interconnection Networks** : Interconnection Networks Taxonomy - Bus-Based Dynamic Interconnection Networks - Switch-Based Interconnection Networks - Static Interconnection Networks - Analysis and Performance Metrics.

UNIT II:

Shared Memory Architecture : Classification of Shared Memory Systems - Bus-Based Symmetric Multiprocessors - Basic Cache Coherency Methods - Snooping Protocols - Directory Based Protocols - Shared Memory Programming - **Message Passing Architecture :**Introduction to Message Passing - Routing in Message Passing Networks - Switching Mechanisms in Message Passing - Message Passing Programming Models - Processor Support for Message Passing - Example Message Passing Architectures - Message Passing Versus Shared Memory Architectures.

UNIT III:

Abstract Models : The PRAM Model and Its Variations - Simulating Multiple Accesses on an EREW PRAM - Analysis of Parallel Algorithms - Computing Sum and All Sums - Matrix Multiplication - Sorting - Message Passing Model - Leader Election Problem - Leader Election in Synchronous Rings.

UNIT IV:

Parallel Programming in the Parallel Virtual Machine : PVM Environment and Application Structure - Task Creation - Task Groups - Communication Among Tasks - Task Synchronization - 6 Reduction Operations - **Message Passing Interface (MPI) :** Communicators - Virtual Topologies - Task Communication - Synchronization - Collective Operations - Task Creation - One-Sided Communication.

UNIT V:

Scheduling and Task Allocation: The Scheduling Problem - Scheduling DAGs without Considering Communication - Communication Models - Scheduling DAGs with Communication - The NP-Completeness of the Scheduling Problem - Heuristic Algorithms - Task Allocation - Scheduling in Heterogeneous Environments.

Text Book:

Advanced Computer architecture and parallel processing, Hesham El-Rewini and Mostafa Abo-El-Barr, , A John wiley & sons, publication, 2005.

UNIT I: Chapter 1.1,1.2,1.3,1.4,1.5& Chapter 2.1,2.2,2.3,2.4, 2.5
UNIT II : Chapter 4.1, 4.2,4.3,4.4,4.5,4.6 & Chapter 5.1,5.2,5.3,5.4,5.5,5.6,5.7
UNIT III: Chapter 6.1,6.2,6.3,6.4,6.5,6.6,6.7,6.8,6.9
UNIT IV: Chapter 8.1,8.2,8.3,8.4,8.5,8.6 & Chapter 9.1,9.2,9.3,9.4,9.5,9.6,9.7
UNIT V: Chapter 10.1,10.2,10.3,10.4,10.5,10.6,10.7,10.8

Reference Book:

1. Computer Architecture, Kai Hwang Kai Hwang & F. A. Briggs, "Computer Architecture and Parallel Processing", McGraw Hill, 2013
2. Advanced Computer Architectures – A Design Space approach , DezsoSima, Terence Fountain, Peter Kacsuk, Pearson Education, 2009
3. Kai Hwang, "Advanced Computer Architecture – Parallelism, Scalability, Programmability", Tata McGraw-Hill, 2008.
4. John L. Hennessey and David A. Patterson, "Computer architecture – A quantitative approach", Morgan Kaufmann / Elsevier Publishers, 5th Edition.

ES 3.1: BIG DATA ANALYTICS
(5 Hours – 4 Credits)

UNIT I:

Introduction to Big Data: Types of Digital Data: Classification of Digital Data, Introduction to Big Data: Characteristics of data-Evolution of Big data-Challenges of Big data-Other Characteristics of Data Which are not Definitional Traits of Big Data-Why Big Data?-Are we Just an Information Consumer or Do we also produce Information?-Traditional Business Intelligence (BI) versus Big Data – A Typical Data Warehouse Environment – A Typical Hadoop Environment – What is New Today? – What is changing in the Realms of Big Data?

UNIT II:

Analytics Basics: Big Data Analytics: Where do we Begin? – What is Big Data Analytics? – What Big Data Analytics Isn't? – Why this Sudden Hype Around Big Data Analytics? –

Classification of Analytics – Greatest Challenges that Prevent Business from capitalizing on Big Data – Top Challenges Facing Big Data – why is Big Data Analytics Important? – What kind of Technologies are we looking Toward to Help Meet the Challenges Posed by Big Data? – Data Science – Data Scientist... Your New Best Friend – Terminologies Used in Big Data Environments – Basically Available Soft State Eventual Consistency (BASE) – Few Top Analytics Tools.

UNIT III:

Big Data Technologies: The Big Data Technology Landscape: NoSQL (Not Only SQL) – Hadoop, Introduction to Hadoop: Introducing Hadoop – Why Hadoop? – Why not RDBMS? – RDBMS versus Hadoop – Distributed Computing Challenges – History of Hadoop – Hadoop Overview – Use Case of Hadoop – Hadoop Distributors – HDFS(Hadoop Distributed File System) – Processing Data with Hadoop – Managing Resources and Applications with Hadoop YARN(Yet another Resource Negotiator) – Interacting with Hadoop Ecosystem.

UNIT IV:

Introduction to MAPREDUCE Programming: Introduction – Mapper – Reducer – Combiner – Partitioner – Searching – Sorting – Compression, Introduction to Hive: What is Hive? – Hive Architecture – Hive Data Types – Hive File Format – Hive Query Language (HQL) – RCFile Implementation – SerDe – User – Defined Function (UDF).

UNIT V:

Analytical Algorithms: Introduction to Machine Learning: Introduction to Machine Learning – Machine Learning Algorithms.

Text Book:

Big Data and Analytics, SeemeAcharya, and Subhashini Chellappan, Wiley India Pvt.Ltd. First Edition-2015.

Unit I : Chapters-1,2
Unit II: Chapter 3
Unit III: Chapter 4,5
Unit IV: 8,9
Unit V: 12

Reference Books:

1. Big Data – Principles and best practices of scalable real-time data systems, Nathan Marz, and James Warren, Manning Publication cp., USA-2015.
2. Analytics in a Big Data World: The Essential Guide to Data Science and its Applications, Bart Baesens, Wiley India Pvt.Ltd-2015.
3. Big Data, Data Mining and Machine Learning, Jared Deamn, Willey India Pvt.Ltd-2015.

ES 3.2: WIRELESS SENSOR NETWORKS

(5 Hours – 4 Credits)

UNIT I :

Introduction: the vision, Networked wireless sensor devices, Applications, Key design challenges. **Network deployment:** Structured versus randomized deployment, Network topology, Connectivity, Connectivity using power control, Coverage metrics, Mobile deployment.

UNIT II:

Localization: issues & approaches, Coarse-grained & Fine-grained node localization, Network-wide localization, Theoretical analysis of localization techniques. **Synchronization:** Issues & Traditional approaches, Fine-grained clock synchronization, and Coarse-grained data synchronization.

UNIT III:

Wireless characteristics: Basics, Wireless link quality, Radio energy considerations, SINR capture model for interference. **Medium-access and sleep scheduling:** Traditional MAC protocols, Energy efficiency in MAC protocols, Asynchronous sleep techniques, Sleep-scheduled techniques, and Contention-free protocols.

UNIT IV:

Sleep-based topology control: Constructing topologies for connectivity, constructing topologies for coverage, Set K-cover algorithms. **Routing:** Metric-based approaches, Routing with diversity, Multi-path routing, Lifetime-maximizing energy-aware routing techniques, Geographic routing, Routing to mobile sinks.

UNIT V:

Data-centric networking: Data-centric routing, Data-gathering with compression, Querying, Data-centric storage and retrieval, The database perspective on sensor networks. **Reliability and congestion control:** Basic mechanisms and tunable parameters, Reliability guarantees, Congestion Control, Real-time scheduling.

Text Book:

Wireless Sensor Networks: Technology,KazemSohraby, Daniel Minoli, TaiebZnati ,
Protocols, and Applications, Wiley Inter Science, 2007.

Unit I	-	Chapters 1 and 2
Unit II	-	Chapters 3, 4 and 5
Unit III	-	Chapters 6 and 7
Unit IV	-	Chapters 8 and 9
Unit V	-	Chapters 10 & 11

Reference Books:

1. Wireless Sensor Networks: Architectures and Protocols Edgar H. Callaway, Jr. Auerbach Publications, CRC Press, 2003.
2. Wireless Sensor Networks: Edited by C.S Raghavendra, Krishna M, Sivalingam, TaiebZnati . Springer, 2005.
3. Networking Wireless Sensors, Bhaskar Krismachari, , Cambridge University Press, 2005
4. Distributed Sensor Networks: A Multiagent Perspective, Victor Lesser, Charles L. Ortiz, and MilindTambe, , Kluwer Publications, 2003.
5. Wireless Sensor Networks: An Information Processing Approach, Feng Zhao, Léonidas Guibas , Morgan Kaufmann Series in Networking, 2004.

ES 3.3: CLOUD COMPUTING
(5 Hours – 4 Credits)

UNIT I:

Cloud Architecture and Model: Technologies for Network-Based System – System Models for Distributed and Cloud Computing –NIST Cloud Computing Reference Architecture. Cloud Models:- Characteristics – Cloud Services – Cloud models (IaaS, PaaS, SaaS) – Public vs Private Cloud –Cloud Solutions - Cloud ecosystem – Service management – Computing on demand.

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 - Virtual Clusters and Resource management – Virtualization for Data-center Automation.

UNIT III:

Cloud Infrastructure: Architectural Design of Compute and Storage Clouds – Layered Cloud Architecture Development – Design Challenges - Inter Cloud Resource Management – Resource Provisioning and Platform Deployment – Global Exchange of Cloud Resources.

UNIT IV:

Programming Model: Parallel and Distributed Programming Paradigms – MapReduce , Twister and Iterative MapReduce – Hadoop Library from Apache – Mapping Applications - Programming Support - Google App Engine, Amazon AWS - Cloud Software Environments - Eucalyptus, Open Nebula, OpenStack, Aneka, CloudSim.

UNIT V:

Security in The Cloud: Security Overview – Cloud Security Challenges and Risks – Software-as-a-Service Security – Security Governance – Risk Management – Security Monitoring – Security Architecture Design – Data Security – Application Security – Virtual Machine Security - Identity Management and Access Control – Autonomic Security.

Text Book :

Distributed and Cloud Computing, From Parallel Processing to the Internet of Things Kai Hwang, Geoffrey C Fox, Jack G Dongarra, Morgan Kaufmann Publishers, 2012.

Unit I	-	Chapters 1 and 2
Unit II	-	Chapters 3 and 4
Unit III	-	Chapters 5 and 6
Unit IV	-	Chapters 7 and 8
Unit V	-	Chapters 9

Reference Books:

1. Cloud Computing: Implementation, Management, and Security,John W.Rittinghouse and James F.Ransome, “”, CRC Press, 2010.
2. Cloud Computing, A Practical Approach,Toby Velte, Anthony Velte, Robert Elsenpeter, TMH, 2009.
3. Cloud Computing – insights into New-Era Infrastructure, Kumar Saurabh,Wiley India,2011.
4. Cloud, George Reese, O'Reilly.
5. Virtual Machines: Versatile Platforms for Systems and Processes James E. Smith, Ravi Nair, Elsevier/Morgan Kaufmann, 2005.