FOR M. Tech (REGULAR)DEGREE COURSE

Applicable for the students of M. Tech (Regular) Course from the Academic Year 2016-17 onwards
The M. Tech Degree of Jawaharlal Nehru Technological University Kakinada shall be conferred on candidates who are admitted to the program and who fulfil all the requirements for the award of the Degree.

1.0 ELIGIBILITY FOR ADMISSIONS

Admission to the above program shall be made subject to eligibility, qualification and specialization as prescribed by the University from time to time.

Admissions shall be made on the basis of merit/rank obtained by the candidates at the qualifying Entrance Test conducted by the University or on the basis of any other order of merit as approved by the University, subject to reservations as laid down by the Govt. from time to time.

2.0 AWARD OF M. Tech DEGREE

2.1 A student shall be declared eligible for the award of the M. Tech Degree, if he pursues a course of study in not less than two and not more than four academic years.
2.2 The student shall register for all 80 credits and secure all the 80 credits.
2.3 The minimum instruction days in each semester are 90.

3.0 ATTENDANCE

3.1 A student shall be eligible to write University examinations if he acquires a minimum of 75% of attendance in aggregate of all the subjects.
3.2 Condonation of shortage of attendance in aggregate up to 10% (65% and above and below 75%) in each semester shall be granted by the College Academic Committee.
3.3 Shortage of Attendance below 65% in aggregate shall not be condoned.
3.4 Students whose shortage of attendance is not condoned in any semester are not eligible to write their end semester examination of that class.
3.5 A prescribed fee shall be payable towards condonation of shortage of attendance.
3.6 A student shall not be promoted to the next semester unless he satisfies the attendance requirement of the present semester, as applicable. They may seek readmission into that semester when offered next. If any candidate fulfills the attendance requirement in the present semester, he shall not be eligible for readmission into the same class.

4.0 EVALUATION

The performance of the candidate in each semester shall be evaluated subject-wise, with a maximum of 100 marks for theory and 100 marks for practical's, on the basis of Internal Evaluation and End Semester Examination.

4.1 For the theory subjects 60 marks shall be awarded based on the performance in the End Semester Examination and 40 marks shall be awarded based on the Internal Evaluation. The internal evaluation shall be made based on the **average** of the marks secured in the two Mid Term-Examinations conducted-one in the middle of the Semester and the other immediately after the completion of instruction. Each midterm examination shall be conducted for a total duration of 120 minutes with 4 questions (without choice) each question for 10 marks. End semester examination is conducted for 60 marks for 5 questions to be answered out of 8 questions.
4.2 For practical subjects, 60 marks shall be awarded based on the performance in the End Semester Examinations and 40 marks shall be awarded based on the day-to-day performance as Internal Marks.

4.3 There shall be two seminar presentations during III semester and IV semester. For seminar, a student under the supervision of a faculty member, shall collect the literature on a topic and critically review the literature and submit it to the department in a report form and shall make an oral presentation before the Project Review Committee consisting of Head of the Department, Supervisor and two other senior faculty members of the department. For each Seminar there will be only internal evaluation of 50 marks. A candidate has to secure a minimum of 50% of marks to be declared successful.

4.4 A candidate shall be deemed to have secured the minimum academic requirement in a subject if he secures a minimum of 40% of marks in the End semester Examination and a minimum aggregate of 50% of the total marks in the End Semester Examination and Internal Evaluation taken together.

4.5 In case the candidate does not secure the minimum academic requirement in any subject (as specified in 5.4) he has to reappear for the End semester Examination in that subject. A candidate shall be given one chance to re-register for each subject provided the internal marks secured by a candidate are less than 50% and has failed in the end examination. In such a case, the candidate must re-register for the subject(s) and secure the required minimum attendance. The candidate's attendance in the re-registered subject(s) shall be calculated separately to decide upon his eligibility for writing the end examination in those subject(s). In the event of the student taking another chance, his internal marks and end examination marks obtained in the previous attempt stand cancelled. For re-registration the candidates have to apply to the University through the college by paying the requisite fees and get approval from the University before the start of the semester in which re-registration is required.
4.6 In case the candidate secures less than the required attendance in any re registered subject (s), he shall not be permitted to write the End Examination in that subject. He shall again re-register the subject when next offered.
4.7 Laboratory examination for M. Tech. courses must be conducted with two Examiners, one of them being the Laboratory Class Teacher or teacher of the respective college and the second examiner shall be appointed by the university from the panel of examiners submitted by the respective college.

5.0 EVALUATION OF PROJECT/DISSERTATION WORK

Every candidate shall be required to submit a thesis or dissertation on a topic approved by the Project Review Committee.

5.1 A Project Review Committee (PRC) shall be constituted with Head of the Department and two other senior faculty members.
5.2 Registration of Project Work: A candidate is permitted to register for the project work after satisfying the attendance requirement of all the subjects, both theory and practical.
5.3 After satisfying 6.2, a candidate has to submit, in consultation with his project supervisor, the title, objective and plan of action of his project work for approval. The student can initiate the Project work, only after obtaining the approval from the Project Review Committee (PRC).
5.4 If a candidate wishes to change his supervisor or topic of the project, he can do so with the approval of the Project Review Committee (PRC). However, the Project Review Committee (PRC) shall examine whether or not the change of topic/supervisor leads to a major change of his initial plans of project proposal. If yes, his date of registration for the project work starts from the date of change of Supervisor or topic as the case may be.
5.5 A candidate shall submit his status report in two stages at least with a gap of 3 months between them.
5.6 The work on the project shall be initiated at the beginning of the II year and the duration of the project is two semesters. A candidate is permitted to submit Project Thesis only after
Successful completion of theory and practical course with the approval of PRC not earlier than 40 weeks from the date of registration of the project work. The candidate has to pass all the theory and practical subjects before submission of the Thesis.

5.7 Three copies of the Project Thesis certified by the supervisor shall be submitted to the College/School/Institute.

5.8 The thesis shall be adjudicated by one examiner selected by the University. For this, the Principal of the College shall submit a panel of 5 examiners, eminent in that field, with the help of the guide concerned and head of the department.

5.9 If the report of the examiner is not favorable, the candidate shall revise and resubmit the Thesis, in the time frame as decided by the PRC. If the report of the examiner is unfavorable again, the thesis shall be summarily rejected. The candidate has to re-register for the project and complete the project within the stipulated time after taking the approval from the University.
5.10 If the report of the examiner is favorable, Viva-Voce examination shall be conducted by a board consisting of the Supervisor, Head of the Department and the examiner who adjudicated the Thesis. The Board shall jointly report the candidate's work as one of the following:

A. Excellent

B. Good

C. Satisfactory

D. Unsatisfactory

The Head of the Department shall coordinate and make arrangements for the conduct of Viva-Voce examination.

5.11 If the report of the Viva-Voce is unsatisfactory, the candidate shall retake the Viva-Voce examination only after three months. If he fails to get a satisfactory report at the second Viva-Voce examination, the candidate has to re-register for the project and complete the project within the stipulated time after taking the approval from the University.

6.0 AWARD OF DEGREE AND CLASS

After a student has satisfied the requirements prescribed for the completion of the program and is eligible for the award of M. Tech. Degree he shall be placed in one of the following four classes:

Class Awarded	**% of marks to be secured**
First Class with Distinction	70% and above (Without any Supplementary Appearance)
First Class	Below 70% but not less than 60% 70% and above (With any Supplementary Appearance)
Second Class	Below 60% but not less than 50%

The marks in internal evaluation and end examination shall be shown separately in the memorandum of marks.

.

SWARNANDHRA

COLLEGE OF ENGINEERING AND TECHNOLOGY
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CBCS

COURSE STRUCTURE (PG)

For

M.TECH

COMPUTER SCIENCE AND ENGINEERING

M.Tech. POST GRADUATE COURSE

(Applicable for the batches admitted from 2016-17)

SWARNANDHRA COLLEGE OF ENGINEERING & TECHNOLOGY
Seetharampuram,Narsapur – 534 280, W.G.Dt.
Andhra Pradesh

(AUTONOMOUS)

COMPUTER SCIENCE AND ENGINEERING
COURSE STRUCTURE – PG(CBCS)
SEMESTER-I

S. No.	Subject Code	Subject Title	L	T	P	C	I	E	Tm
1	**16CM1T01**	Advanced Data Structures	3	-	-	3	40	60	100
2	**16CM1T02**	Computer Networks	3	-	-	3	40	60	100
3	**16CM1T03**	Database Management Systems	3	-	-	3	40	60	100
4	**16CM1T04**	Software Engineering	3	-	-	3	40	60	100
5	**Elective-I**	**16CM1E01:** Scripting Languages	3	-	-	3	40	60	100
		16CM1E02: Computer Organization							
		16CM1E03: Soft Computing							
6	**Elective-II**	**16CM1E04:** Operating Systems	3	-		3	40	60	100
		16CM1E05: Web Technologies							
		16CM1E06: Artificial Intelligence							
7	**16CM1L01**	ADS And DBMS Lab	-	-	4	2	40	60	100
		Total	18		4	**20**	**280**	**420**	**700**

SEMESTER-II

S. No.	Subject Code	Subject Title	L	T	P	C	I	E	Tm
1	**16CM2T01**	Information Security	3		-	3	40	60	100
2	**16CM2T02**	Design & Analysis of Algorithms	3	-	-	3	40	60	100
3	**16CM2T03**	Software Testing Methodologies	3	-	-	3	40	60	100
4	**16CM2T04**	Wireless Networks	3	-	-	3	40	60	100
5	**Elective-III**	**16CM2E01:** Cloud Computing	3	-	-	3	40	60	100
		16CM2E02: Advanced Computer Architecture							
		16CM2E03: Data Ware housing & Data Mining							
6	**Elective-IV**	**16CM2E04:**Machine learning	3	-	-	3	40	60	100
		16CM2E05: Mobile Application Development							
		16CM2E06: Big Data Analytics							
7	**16CM2L01**	Software Testing Methodologies Lab	-		4	2	40	60	100
		Total	**18**		**4**	**20**	**280**	**20**	**700**

COMPUTER SCIENCE AND ENGINEERING
COURSE STRUCTURE – PG
SEMESTER-III

S. No.	Subject Code	Subject Title	L	T	P	C	I	E	Tm
1	**16CM3S01**	Seminar				2	50	-	50

2	**16CM3P01**	Project Work Phase-I				18	-	-	-
Total						20	50		50

SEMESTER-IV

S. No.	**Subject Code**	**Subject Title**	**L**	**T**	**P**	**C**	**I**	**E**	**Tm**
1	**16CM4Q01**	Technical Paper Write-up				2	50		50
2	**16CM4P01**	Project Work Phase-II				18			
Total						20	50		50

ADVANCED DATA STRUCUTES
M.TECH R16 I SEMESTER

OBJECTIVES: The objectives of the course are
1. To allow to assess how the choice of data structures and algorithm design methods impacts the performance of programs.
2. To choose the appropriate data structure and algorithm design method for a specified application.
3 To learn the systematic way of solving problems, various methods of organizing large amounts of data.
4. To solve problems using data structures such as linear lists, stacks, queues, binary trees, binary search trees, and graphs and writing programs for these solutions.
5. To efficiently implement the different data structures and solutions for specific problems.

COURSE OUTCOMES
1. Basic ability to analyze algorithms and to determine algorithm correctness and time efficiency class.
2. Master a variety of advanced abstract data type (ADT) and data structures and their Implementations.
3. Master different algorithm design techniques (brute-force,divide and conquer,greedy)

UNIT - I
Introduction, Analysis of Algorithms: Time Complexity & Space Complexity.
Linked Lists: Single, Double And Circular Lists. Linked Stacks & Linked Queues.

UNIT - II
Searching: Linear Search, Transpose Sequential Search, Binary Search and Fibonacci Search.
Sorting: Bubble Sort, Insertion Sort, Selection Sort, Merge Sort, Shell Sort, Quick Sort, Heap Sort.

UNIT - III
Trees: Binary Trees, representation of binary trees, tree traversals, expression trees (infix, prefix, and postfix), and threaded binary trees.
Graphs: Basic terminologies, representation, traversals (BFS, DFS).

UNIT - IV
Dictionaries, ADT, The list ADT, Stack ADT, Queue ADT, Hash table representation, hash function, collision resolution-separate chaining, and open addressing- linear probing, double hashing.

UNIT - V
Priority Queues: Definition, ADT, Realizing a Priority queue using Heaps, definition, insertion, deletion.
Search Trees: Binary search trees, Definition, ADT, Implementation.AVL Trees, Definition, Height of an AVL tree, Introduction to Red-Black and Splay trees, B- Trees, height of a B-tree, operations-searching, insertion and deletion

Text book:

1. Data Structures and Algorithms- GAV Pai, TMH Publications.

Reference Books:

1. Classic Data Structures- Debasis Samantha, 2/e, PHI Learning.
2. Data Structures and Algorithm Analysis- Mark Allen Weiss,2/e, Pearson Education.
3. Introduction to Algorithms – Thomas H Cormen, Charles E Leiserson, Ronald L Rivest & Clifford Stein 3/e, PHI Learning.

ADVANCED DATA STRUCTURES LAB
M.TECH R16 I SEMESTER

COURSE OBJECTIVES:

1. Develop program to implement linked lists, stacks, queues and binary search trees.
2. Implement searching and sorting algorithms.

LEARNING OUTCOMES:

Students are able to:

1. Demonstrate the implementation of linked lists.
2 .Implement stack and queue using arrays and linked lists.
3. Demonstrate applications of stack.
4. Demonstrate the implementation of binary search trees.
5. Implement different searching and sorting algorithms.

Write a c program for the following

Exercise – I

1. Use both recursive and non-recursive functions to perform Linear search for a Key value in a given list.
2. Use both recursive and non-recursive functions to perform Binary search for a Key value in a given list.
3. Implement following techniques to sort a given list of integers in ascending order (i) Insertion sort (ii) Bubble sort (iii) Selection sort

Exercise - II

3. Implement following techniques to sort a given list of integers in ascending order. (i) Quick sort (ii) Merge sort

Exercise - III

4. Implement stack (its operations) using arrays.
5. To evaluate postfix expression.
6. To Convert infix expression into postfix expression.

Exercise – IV

7. Implement queue (its operations) using arrays.
8. Use functions to (i) create a singly linked list.
(ii) Insert an element into a singly linked list.
(iii)Delete an element from a singly linked list.

Exercise - V

9. Implement stack (its operations) using Linked list
10. Implement Queue (its operations) using linked lists.

Exercise - VI

11. To reverse elements of a single linked list.
12. Use functions to
(i) Create a circular linked list.
(ii) Insert an element into a circular linked list.
(iii)Delete an element from a circular linked list.

Exercise –VII

13. Use functions to
(i) Create a Doubly linked list.
(ii) Insert an element into a doubly linked list.
(iii)Delete an element from a doubly linked list.

Exercise - VIII

14. To create a Binary Search Tree of integers, insert, delete and search integers into (from) Binary search tree.
15. Use recursive functions to traverse a binary search tree in preorder, in order and post order.

COMPUTER NETWORKS
M.TECH R16 I SEMESTER

COURSE OBJECTIVES:
1. Study and understand various network reference models such as OSI, TCP/IP and ATM.
2. Study and analyze various data link layer protocols
3. Analyze various routing algorithms and congestion control algorithms and Transport layer protocols
4. Introduction to Emerging trends in Computer networks like MANET

COURSE OUTCOMES
1. Ability to differentiat**e** network reference models such as OSI, TCP/IP and ATM and various multiplexing techniques
2.Ability to classify various Data Link Layer s such as elementaryProtocol,sliding window,HDLC,PPP
3.An ability to distinguish various MAC sub Layer Protocols and Its Applications
4.An ability to understand various routing algorithms and application layer protocols such as www And HTTP

UNIT - I
Introduction: OSI, TCP/IP and other networks models, Examples of Networks: Arpanet, Internet, Network Topologies
Physical Layer: Transmission media copper, twisted pair wireless, switching and encoding, narrow band, broad band ISDN and ATM.

UNIT-II
Data link layer: Design issues, framing, error detection and correction, CRC, Elementary Protocol-stop and wait, Sliding Window, Data link layer in HDLC, Internet,
Medium Access sub layer: A LOHA, MAC addresses, Carrier sense multiple access. IEEE 802.X Standard Ethernet, wireless LANS. Bridges
UNIT-III

Network Layer: Network layer design issues: store and forward packet switching, services provided to transport layer, implementation of connectionless service, implementation of connection oriented service, comparison of virtual circuit and datagram subnets.
Routing algorithm: Shortest path routing algorithm, flooding, distance vector routing, link state routing, hierarchical routing, broadcast routing, multicast routing, routing for mobile hosts, routing in adhoc networks.
Congestion control algorithms- load shedding, congestion control in datagram subnet.

UNIT – IV
Transport Layer: Transport Services, Connection management, Process to Process delivery**,** client/server paradigm, multiplexing and Multiplexing, connectionless versus connection oriented services, reliable versus unreliable.
UDP: well-known ports for UDP, user datagram, checksum, UDP operation, and uses of UDP.
TCP: TCP services, TCP features, segment, A TCP connection, flow control, error control, congestion control.

UNIT-V
Emerging trends in Computer networks:
Motivation for mobile computing: protocol stack issues in mobile computing environment, mobility issues in mobile computing, data dissemination security issues, Mobile networks.

Mobile adhoc networks: application of adhoc networks, challenges and issues in MANETS, MAC layer issues, routing protocols in MANET, transport layer issues.

TEXT BOOKS:

1. Computer Networks - Andrew S Tanenbaum, 4th Edition. Pearson Education/PHI
2. Data Communications and Networking, 4th Edition, Behrouz A Fourzan, TMH
3. Computer Networks, Mayank Dave, Cengage.

REFERENCES:

1. An Engineering Approach to Computer Networks-S.Keshav,2nd Edition, Pearson Education
2. Understanding communications and Networks,3rd Edition, W.A.Shay, Thomson

DATA BASE MANAGEMENT SYSTEMS
M. TECH (R16) I SEMESTER

COURSE OBJECTIVES:

1. To understand the basic concepts and the applications of database systems.
2. To master the basics of SQL and construct queries using SQL.
3. To understand the relational database design principles.
4. To become familiar with the basic issues of transaction processing and concurrency Control.
5. To become familiar with database storage structures and access techniques.

COURSE OUTCOMES:

1. To understand the different issues involved in the design and implementation of a Database system
2. Ability to design entity relationship and convert entity relationship diagrams into RDBMS and formulate SQL queries on the respect data.
3. An ability to use different concurrency control techniques while implementing real Time applications
4. An ability to Show how various kind of secondary storage devices to store data

UNIT I

Database System Applications, database system vs file systems, View of Data, Data Abstraction, Instances and Schemas, Data Models the ER Model, Relational Model, Other Models, Database Languages – DDL, DML. Database Access for Applications Programs, Database Users and Administrators, Transaction Management, Database system structure, storage manager, the query processor. Database Design and ER Diagrams, Entities, Attributes and Entity Sets, Relationships and Relationship sets ,Additional features of the ER model, conceptual design with the ER model,

UNIT-II

Relational Model-Introduction to the Relational Model, Integrity Constraints Over Relations, Enforcing Integrity constraints, Querying relational data, Logical data base Design, Introduction to Views –Altering Tables and Views. Relational Algebra-selection, projection set operations, renaming,joins.Relational calculus-Tupla relational calculus, Domain relational calculus. Basic SQL Queries, Nested Queries, Complex Integrity Constraints in SQL, Triggers

UNIT III

Introduction to Schema Refinement – Problems Caused by redundancy, Decompositions – Problem related to decomposition. Functional Dependencies , Reasoning about FDS, Normal Forms – FIRST, SECOND, THIRD Normal forms ,BCNF .Properties of Decompositions- Loss less- join Decomposition, Dependency preserving Decomposition. Schema Refinement in Data base Design, Multi valued Dependencies, FOURTH Normal Form, Join Dependencies, FIFTH Normal form.

UNIT IV

Transaction Management- The ACID Properties, Transactions and Schedules, Concurrent Execution of Transactions, Lock Based Concurrency Control.

Concurrency Control- Serializability, and recoverability – Introduction to Lock Management, Lock Conversions, Dealing with Dead Locks, Specialized Locking Techniques, Concurrency Control without Locking.

Crash recovery-Introduction to Crash recovery, Introduction to ARIES, the Log, and Other Recovery

related Structures, the Write-Ahead Log Protocol, Check pointing, recovering from a System Crash.

UNIT V

Overview of Storage and Indexing: Data on External Storage, File Organization and Indexing – Clustered Indexes, Primary and Secondary Indexes. Index data Structures – Hash Based Indexing, Tree based indexing.

Storing data: Disks and Files: -The Memory Hierarchy – Redundant Arrays of Independent Disks. Tree Structured Indexing: Intuitions for tree Indexes, Indexed Sequential Access Methods (ISAM) B+ Trees: A Dynamic Index Structure, Search, Insert and Delete.

TEXT BOOKS:

1. Data base Management Systems, Raghu Ramakrishnan, Johannes Gehrke, TMH, 3rd Edition, 2003.
2. Data base System Concepts, A.Silberschatz, H.F. Korth, S.Sudarshan, McGraw hill, VI edition, 2006.
3. Fundamentals of Database Systems 5th edition. Ramez Elmasri, Shamkant B.Navathe, Pearson Education, 2008.

SOFTWARE ENGINEERING
M.TECH (R16) I SEMESTER

COURSE OBJECTIVES:

1. To make the students learn about the basic concepts on software engineering methods And practices and their appropriate application in software industry.
2. To develop an understanding of software process models and Software Development Life Cycle.
3. To provide an idea on software testing techniques.
4. To teach an understanding role of the different aspects of Software Project Management.
5. To develop an approach on ethical and professional issues those are important for Software Project Management.

COURSE OUTCOMES:

1. Capabilities to identify, formulate, and solve software engineering problems.
2. be able to elicit, analyze and specify software requirements with various stakeholders of a Software development project.
3. Ability to participate in design, development, deployment and maintenance of a medium Scale software development project.
4. Knowledge to convey technical material through oral presentation and interaction with an Audience.
5. Ability to evaluate the impact of potential solutions to software engineering problems in a Global society, using the knowledge of models, tools, and techniques.

UNIT I:

Introduction to Software Engineering: The evolving role of software, Attributes of good software, Changing Nature of Software, Software myths.

A Generic view of process: Software engineering- A layered technology, a process framework, The Capability Maturity Model Integration (CMMI), process assessment.

UNIT II:

Process models: The waterfall model, Incremental process models, Evolutionary process models, The Unified process.

Software Requirements: Functional and non-functional requirements, User requirements, System requirements, Interface specification, the software requirements document.

UNIT III:

Requirements engineering process: Feasibility studies, Requirements elicitation and analysis, Requirements validation, Requirements management.

System models: Context Models, Behavioral models, Data models, Object models, structured methods.

UNIT IV:

Design Engineering: Design process and Design quality, Design concepts, the design model.

Object-Oriented Design: Objects and object classes, An Object-Oriented design process, Design evolution.

UNIT V:

Testing Strategies: A strategic approach to software testing, test strategies for conventional software, Black-Box and White-Box testing, Validation testing, System testing, the art of Debugging.

Quality Management: Quality concepts, Software quality assurance, Software Reviews,

Formal technical reviews, Statistical Software quality Assurance, Software reliability, The ISO 9000 quality standards.

TEXT BOOKS:

1. Software Engineering, A practitioner's Approach- Roger S. Pressman, 6th edition. McGraw-Hill International Edition.
2. Software Engineering- Somerville, 7th edition, Pearson education.

REFERENCES:

1. Software Engineering- K.K. Agarwal & Yogesh Singh, New Age International Publishers
2. Software Engineering, an Engineering approach- James F. Peters, Witold Pedrycz, John Wiely.
3. Systems Analysis and Design- Shely Cashman Rosenblatt,Thomson Publications.
4. Software Engineering principles and practice- Waman S Jawadekar, The McGraw-Hill Companies.

SCRIPTING LANGUAGES
M.TECH R16 I SEMESTER (ELECTIVE-1)

COURSE OBJECTIVES:
1. The study of the principles of scripting languages
2. The study of usage of scripting languages in IC design flow
3. The study of scripting languages such as PERL, TCL/TK , Python and BASH
4. The of the tools and the scripting languages necessary for design and development of applications

COURSE OUTCOMES:
1. Ability to understand the differences between scripting languages.
2. Ability to apply your knowledge of the weaknesses of scripting languages to select implementation.
3. Ability to create and run scripts using Perl / TCL / Python in IC design flow
4. Master an understanding of python especially the object oriented concepts.

UNIT-I

Introduction to PERL and Scripting: Scripts and Programs, Origin of Scripting , Scripting Today, Characteristics of Scripting Languages, Uses for Scripting Languages, Web Scripting, and the universe of Scripting Languages. PERL- Names and Values, Variables, Scalar Expressions, Control Structures, arrays, list, hashes, strings, pattern and regular expressions, subroutines.

UNIT-II

Advanced pen: Finer points of looping, pack and unpack, file system, eval, data structures, packages, modules, objects, interlacing to the operating system, Creating Internet ware applications, Dirty Hands Internet Programming, security Issues.

UNIT-III

PHP Basics: PHP Basics- Features, Embedding PHP Code in your Web pages, Outputting the data to the browser, Data types, Variables, Constants, expressions, string interpolation, control structures . Function, Creating a Function, Function Libraries, Arrays, strings and Regular Expressions.

Advanced PHP Programming: PHP and Web Forms, Files, PHP Authentication and Methodologies -Hard Coded, File Based, Database Based, IP Based, Login Administration, and Uploading Files with PHP, Sending Email using PHP, and PHP Encryption Functions, the M crypt package, and Building Web sites for the Work.

UNIT -IV

TCL: TCL Structure, syntax, Variables and Data in TCL, Control Flow, Data Structures, input/output, procedures , strings, patterns, files, Advance TCLe val, source, exec and up level commands, Name spaces, trapping errors, event driven programs, making applications Internet aware, Nuts and Bolts Internet Programming, Security Issues, C Interface. Tk-Visual Tool Kits, Fundamental Concepts of Tk, Tk by example, Events and Binding, PerI-Tk.

UNIT-V

Python: Introduction to Python language, python-syntax, statements, functions, Built-in-functions and Methods, Modules in python, Exception I Handling. Integrated Web Applications in Python-Building Small, Efficient Python Web Systems, Web Application Framework.

TEXTBOOKS

1. The World of Scripting Languages, David Barron,Wiley Publications.
2. Python Web Programming, Steve Holden and David Beazley, New Riders Publications.
3. Beginning PHP and MySQL, 3rd Edition, Jason Gilmore, Apress Publications (Dream tech.).

REFERENCE BOOKS

1. Open Source Web Development with LAMP using Linux, Apache, MySQL, Pen and PHP, J.Lee and B.Ware(Addison Wesley) Pearson Education.
2. Programming Python,M.Lutz,SPD.
3. PHP 6 Fast and Easy Web Development, Julie Meloni and Matt Telles, Cengage Learning Publications.
4. PHP 5.1, l.Bayross and S.Shah, The X Team, SPD.
5. Core Python Programming, Chun, Pearson Education.

COMPUTER ORGANIZATION
M.TECH R16 I SEMESTER (ELECTIVE-1)

COURSE OBJECTIVES:

1. To impart the basic knowledge of computer system including the analysis and design of Components of the system.
2. To understand the register transfer language, micro operations and design of basic Components of the system.
3. To outline the description of different parameters of a memory system, organization and Mapping of various types of memories.
4. To understand the concept of Data Transfer in various components of a computer system.

COURSE OUTCOMES:

1. Graduates will have fundamental knowledge about structure of computers.
2. Graduates will be able to choose appropriate addressing modes and instructions for writing programs.
3. Graduates will understand the need for using Peripheral devices for efficient operation of system.
4. Graduates will gain basic ability to analyze Micro operations such as Arithmetic micro operations, Shift micro operations, Logic micro operations and data transfer in various components of a computer system.

UNIT I

Basic Structure of Computers:
Basics of computer, Von Neumann Architecture, Generation of Computer, Types of Compute, Functional unit, Basic Operational Concepts and Bus Structures.
Computer Arithmetic: Addition and Subtraction, multiplication algorithms, Division Algorithms.

UNIT II

Register Transfer Language and Micro Operations: Register Transfer language. Register Transfer Bus and memory transfers, Arithmetic Micro operations, logic micro operations, shift micro operations, Arithmetic logic shift unit.
Basic Computer Organization and Design: Instruction codes, Computer Registers, Computer Instructions, Timing and control, Instruction Cycle, Memory – Reference, Input – Output and Interrupt Instructions. Design of basic computer, Design of Accumulator logic.

UNIT III

Central Processing Unit: General Register Organization, STACK organization. Instruction formats. Addressing modes. DATA Transfer and manipulation, Program control, Reduced Instruction Set Computer.
Micro Programmed Control: Control Memory, Address sequencing, micro program example, design of control unit.

UNIT IV

Input- Output Organization: Peripheral Devices, Input-Output Interface, Asynchronous data transfer, Modes of Transfer, Priority Interrupts, Direct memory Access.
The Memory System: Memory Hierarchy, Main Memory, Auxiliary memory, Associative Memory, Cache Memory and Virtual Memory.

UNIT V

Parallel Processing and Vector Processing

Pipeline and Vector Processing: Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline, Vector Processing, Array Processors.

Multi Processors and Multi Computers

Multiprocessors and Multi computers, Characteristics of Multiprocessors, Multiple Processor Organizations, Symmetric Multi-Processors, Cache Coherence.

Text Books

1. Computer System Organization, M.Moris Mano, 3rd Edition, Pearson / PHI
2. Computer Organization, Carl Hamacher, Zvonks Vranesic, SafeaZaky, 5th Edition, Hill.
3. Computer Organization, a quantitative approach, John L.Hennessy and David A.Patterson, Fourth Edition Elsevier

Reference Books

1. Computer Organization and Architecture - William Stallings Sixth Edition, Pearson / PHI
2. Structured Computer Organization - Andrew s. Tanenbaum, 4th Edition, PHI/ Pearson.
3. Fundamentals of Computer Organization and Design, - Sivaraama Dandamudi, Springer Int. Edition.

SOFT COMPUTING
M.TECH R16 I SEMESTER (ELECTIVE-1)

COURSE OBJECTIVES:

At the end of the course, student

1. Understanding the concepts of M/C Intelligence
2. Develop a Neural network for the proposed model
3. Design a genetic algorithm and implement various genetic operators
4. Ability to incorporate Fuzzy Logic and developing Neuro- fuzzy systems.

COURSE OUTCOMES:

Upon successful completion of the course, student can

1. Design, analyze the Neural network architectures
2. Develop the skills to design and implement Genetic programming solutions to various problems.
3. Applying Fuzzy Logic and the techniques of Neuro-fuzzy models.

UNIT-I

Introduction to Soft Computing and Neural Networks: Evolution of Computing, Soft Computing Constituents, From Conventional AI to Computational Intelligence, Machine Learning Basics.

UNIT -II

Genetic Algorithms:
Introduction to Genetic Algorithms (GA), Applications of GA in Machine Learning,
Machine, Learning Approach to Knowledge Acquisition.

UNIT-III Neural Networks:
Machine Learning Using Neural Network. Adaptive Networks, Feed forward Networks,
Supervised Learning Neural Networks, Radial Basis Function Networks, Reinforcement Learning, Unsupervised Learning Neural Networks, Adaptive Resonance architectures, Advances in Neural networks.

UNIT-IV: Fuzzy Logic:
Fuzzy Sets, Operations on Fuzzy Sets, Fuzzy Relations, Membership Functions, Fuzzy Rules and Fuzzy Reasoning, Fuzzy Inference Systems, Fuzzy Expert Systems, Fuzzy Decision Making.

UNIT -V Neuro-Fuzzy Modeling:
Adaptive Neuro, Fuzzy Inference Systems, Coactive Neuro, Fuzzy Modeling,
Classification and Regression Trees, Data Clustering Algorithms, Rule base Structure Identification, Neuro, Fuzzy Control, Case studies.

Text Books:

1. Iyh, Shlng Roger Jang, Chuen,Tsai Sun, EijiMizutani, "Neuro, Fuzzy and Soft Computing ", Prentice, Hall of India, 2003.
2. George J. Klir and Bo Yuan, "Fuzzy Sets and Fuzzy Logic, Theory and Applications ",Prentice Hall1995.

Reference Books:

1. James A. Freeman and David M. Skapura, "Neural Networks Algorithms, Applications, and Programming Techniques", Pearson Edn., 2003.
2. Mitchell Melanie, "An Introduction to Genetic Algorithm ", Prentice Hall, 1998.
3. David E. Goldberg, "Genetic Algorithms in Search. Optimization and Machine Learning ", Addison Wesley, 199

SWARNANDHRA

COLLEGE OF ENGINEERING AND TECHNOLOGY
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

OPERATING SYSTEMS

M.TECH R16 I SEMESTER (ELECTIVE-II)

COURSE OBJECTIVES:

1. To understand the fundamental concepts and techniques of Operating Systems.
2. To study the concepts in process management and concurrency control mechanisims
3. To understand the concepts in memory managements and deadlocks
4. To study on file management and storage structres

COURSE OUTCOMES:

1. An ability to understand basic concepts about operating system.
2. An ability to describe process management, scheduling and concurrency control mechanisims.
3. An ability to analyze memory management and deadlocks.
4. An ability to compare various file systems and its operating systems exmples.

UNIT-I:

Computer System and Operating System Overview: Overview of computer operating systems, operating systems functions, protection and security, distributed systems, special purpose systems, operating systems structures and systems calls, operating systems generation.

UNIT-II:

Process Management – Process concept- process scheduling, operations, Inter process communication. Multi Thread programming models. Process scheduling criteria and algorithms, and their evaluation.

UNIT-III:

Concurrency: Process synchronization, the critical- section problem, Peterson's Solution, synchronization Hardware, semaphores, classic problems of synchronization, monitors, Synchronization examples

UNIT-IV:

Memory Management: Swapping, contiguous memory allocation, paging, structure of the page table, segmentation

Virtual Memory Management:

Virtual memory, demand paging, page-Replacement, algorithms, Allocation of Frames, Thrashing

UNIT-V:

Principles of deadlock – system model, deadlock characterization, deadlock prevention, detection and avoidance, recovery form deadlock,

File system Interface- the concept of a file, Access Methods, Directory structure.

File System implementation- File system structure, allocation methods, free-space management **Mass-storage structure** overview of Mass-storage structure, Disk structure, disk scheduling.

TEXT BOOKS:

1. Operating System Concepts- Abraham Silberchatz, Peter B. Galvin,Gagne 7th Edition, John Wiley.
2. Operating Systems' – Internal and Design Principles Stallings, Sixth Edition–2005, Pearson Education

REFERENCE BOOKS:

1. http://nptel.iitm.ac.in/courses/Webcourse-contents/IISc-BANG/Operating%20Systems/New_index1.html
2. Operating systems- A Concept based Approach-D.M.Dhamdhere, 2nd Edition, TMH
3. Operating System A Design Approach-Crowley, TMH.
4. Modern Operating Systems, Andrew S Tanenbaum 3rd edition PHI.

SWARNANDHRA

COLLEGE OF ENGINEERING AND TECHNOLOGY
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

WEB TECHNOLOGIES

M.TECH R16 I SEMESTER (ELECTIVE-II)

COURSE OBJECTIVES:

1. This course is designed to introduce students with no programming experience to the programming languages and techniques associated with the World Wide Web.
2. The course will introduce web-based media-rich programming tools for creating interactive web pages.

COURSE OUTCOMES:

1. Analyze a web page and identify its elements and attributes.
2. Create web pages using XHTML and Cascading Styles sheets.
3. Build dynamic web pages.
4. Build web applications using PHP.
5. Programming through PERL and Ruby
6. Write simple client-side scripts using AJAX

SYLLABUS:

UNIT-I:

HTML tags, Lists, Tables, Images, forms, Frames. Cascading style sheets. Introduction to Java script. Objects in Java Script. Dynamic HTML with Java Script

UNIT-II:

Working with XML: Document type Definition, XML schemas, Document object model, XSLT, DOM and SAX.

UNIT-III:

AJAX A New Approach: Introduction to AJAX, Integrating PHP and AJAX. Consuming WEB services in AJAX: (SOAP, WSDL, UDDI)

UNIT-IV:

PHP Programming: Introducing PHP: Creating PHP script, Running PHP script. Working with variables and constants: Using variables, Using constants, Datatypes, Operators. Controlling program flow: Conditional statements, Control statements, Arrays, functions. Working with forms and Databases such as MySQL, Oracle and SQL Server.

UNIT-V:

Introduction to PERL, Perl language elements, Interface with CGI- A form to mail program, Simple page search.
Introduction to Ruby, variables, types, simple I/O, Control, Arrays, Hashes, Methods, Classes, Iterators, Pattern Matching, Practical Web Applications

Text Books:

1. Programming the World Wide Web, Robet W Sebesta, 7ed, Pearson.
2. Web Technologies, Uttam K Roy, Oxford
3. The Web Warrior Guide to Web Programming, Bai, Ekedahl, Farrelll, Gosselin, Zak, Karparhi, Maclntyre, Morrissey, Cengage

Reference Books:

1.Ruby on Rails Up and Running, Lightning fast Web development, Bruce Tate, Curt Hibbs, Oreilly (2006)
2. Programming Perl, 4ed, Tom Christiansen, Jonathan Orwant, Oreilly (2012)
3. Web Technologies, HTML< JavaScript, PHP, Java, JSP,XML and AJAX, Black book, Dream Tech.
4. An Introduction to Web Design, Programming, Paul S Wang, Sanda S Katila, Cengage

SWARNANDHRA

COLLEGE OF ENGINEERING AND TECHNOLOGY
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

ARTIFICIAL INTELLIGENCE

M.TECH R16 I SEMESTER (ELECTIVE-II)

COURSE OBJECTIVES:

At the end of the course, student should be able to get

1. Introduced to students to advanced techniques within artificial intelligence (AI), with particular Focus on automated planning and multi - agent systems.
2. To explain, analyze and implement advanced AI techniques.

COURSE OUTCOMES:

Upon successful completion of the course, student can

1. Compare and assess the appropriateness of various AI techniques for solving a given concrete Problem
2. Combine different AI techniques in a theoretically sound and practically useful way
3. Apply a given AI technique to a given concrete problem
4. Create logical agents to do inference using first order logic.

UNIT-I

Introduction to artificial intelligence: Introduction, history, intelligent systems, foundations of AI, applications, tic-tac-tie game playing, development of AI languages, current trends in AI

UNIT-II

Problem solving: state-space search and control strategies: Introduction, general problem solving, characteristics of problem, exhaustive searches, heuristic search techniques, iterative-deepening a*, constraint satisfaction

Problem reduction and game playing: Introduction, problem reduction, game playing, alpha-beta pruning, two-player perfect information games

UNIT-III

Logic concepts: Introduction, propositional calculus, proportional logic, natural deduction system, axiomatic system, semantic tableau system in proportional logic, resolution refutation in proportional logic, predicate logic

UNIT-IV

Knowledge representation: Introduction, approaches to knowledge representation, knowledge representation using semantic network, extended semantic networks for KR, knowledge representation using frames **advanced knowledge representation techniques:** Introduction, conceptual dependency theory, script structure, cyc theory, case grammars, semantic web

UNIT-V

Expert system and applications: Introduction phases in building expert systems, expert system versus traditional systems, rule-based expert systems blackboard systems truth maintenance systems, application of expert systems, list of shells and tools

TEXT BOOKS:

1. Artificial Intelligence- Saroj Kaushik, CENGAGE Learning,
2. Artificial intelligence, A modern Approach , 2nd ed, Stuart Russel, Peter Norvig, PEA
3. Artificial Intelligence- Rich, Kevin Knight, Shiv Shankar B Nair, 3rd ed, TMH
4. Introduction to Artificial Intelligence, Patterson, PHI

REFERENCE BOOKS:

1. Artificial intelligence, structures and Strategies for Complex problem solving, -George F Lugar, 5th ed, PEA
2. Introduction to Artificial Intelligence, Ertel, Wolf Gang, Springer
3. Artificial Intelligence, A new Synthesis, Nils J Nilsson, Elsevier

INFORMATION SECURITY
M.TECH (R16) II SEMESTER

COURSE OBJECTIVES:

1. Learn fundamentals of cryptography and its application to network security.
2. Understand network security threats, security services, and countermeasures.
3. Background on well-known network security protocols such as IPsec, SSL, and WEP.
4. Understand vulnerability analysis of network security.
5. Acquire background on hash functions; authentication; firewalls; intrusion detection techniques.
6. Gain hands-on experience with programming and simulation techniques for security protocols.
7. Evaluate the security of communication systems, networks and protocols based on a multitude of security metrics.

COURSE OUTCOMES:

1. Understand and contrast various cryptographic algorithms.
2. Understand computational complexity aspects of cryptographic and crypto analytic methods.
3. Analyze attacks, such as man-in-middle on crypto systems.
4. Understand and analyze functionality and weakness of signature and authentication protocols.
5. Describe and analyze key exchange protocols.
6. Understand and analyze functionality and weakness of SSL and TLS protocols.
7. Relate hacking and intrusion techniques to network characteristics.
8. Relate current legal and social issues to cryptographic applications and usage.

UNIT I

Security Attacks (Interruption, Interception, Modification and Fabrication), Security Services (Confidentiality, Authentication, Integrity, Non-repudiation, access Control and Availability) and Mechanisms, A model for Internetwork security, Internet Standards and RFCs.

UNIT II

Conventional Encryption Principles, Conventional encryption algorithms, cipher block modes of operation, location of encryption devices, key distribution Approaches of Message authentication, Secure Hash Functions and HMAC.

UNIT III

Public key cryptography principles, public key cryptography algorithms: RSA, Diffie-Hellmen key exchange algorithms, digital signatures, digital Certificates, Certificate Authority and key management Kerberos, X.509 Directory Authentication Service. Email privacy: Pretty Good Privacy (PGP) and S/MIME.

UNIT IV

IP Security Overview, IP Security Architecture, Authentication Header, Encapsulating Security Payload, Combining Security Associations and Key Management. Web Security Requirements, Secure Socket Layer (SSL) and Transport Layer Security (TLS), Secure Electronic Transaction (SET).

UNIT V

Basic concepts of SNMP, SNMPv1 Community facility and SNMPv3. Intruders, Viruses and related threats. Firewall Design principles, Trusted Systems. Intrusion Detection Systems.
Learning Resource

Text Books:

1. Network Security Essentials (Applications and Standards) by William Stallings Pearson Education.

References:

1. Hack proofing your network by Ryan Russel, Dan Kaminsky, Rain Forest Puppy, Joe Grand, David Ahmad, Hal Flynn Ido Dubrawsky, Steve W. Manzuik and Ryan Permeh, Wiley Dreamtech.
2. Fundamentals of Network Security by Eric Maiwald (Dreamtech press)
3. Cryptography and network Security, Third edition, Stallings, PHI/Pearson
4. Principles of Information Security, Whitman, Thomson.5. Introduction to Cryptography, Buchmann, Springer.

DESIGN AND ANALYSIS OF ALGORITHMS
M.TECH (R16) II SEMESTER

COURSE OBJECTIVES:

1. To develop an understanding about basic algorithms and different problem solving Strategies
2. To improve creativeness and the confidence to solve non-conventional problems and expertise For analyzing existing solution
3. To design and implementation of various basic data structure

COURSE OUTCOMES:

1. Analyze worst-case running times of algorithms using asymptotic analysis.
2. Describe the divide-and-conquer paradigm and explain when an algorithmic design situation calls for it.
3. Describe the dynamic-programming paradigm and explain when an algorithmic design situation calls for it.
4. Describe the greedy paradigm and explain when an algorithmic design situation calls for it.
5. Analyze randomized algorithms, difference between a randomized algorithm and an algorithm with probabilistic inputs.

UNIT-I:

Introduction: Algorithm, Pseudo code for expressing algorithms, performance Analysis Space complexity, Time complexity, Asymptotic Notation- Big oh notation, Omega notation, Theta notation and Little oh notation, Probabilistic Analysis. Disjoint Sets - disjoint set operations, union and find algorithms, spanning trees, connected components and bi-connected components.

UNIT-II:

Divide and conquer: General method, applications-Binary search, Quick sort, Merge sort, Strassen's matrix multiplication. Greedy method: General method, applications-Job sequencing with deadlines, 0/1 knapsack problem, Minimum cost spanning trees, Single source shortest path problem.

UNIT-III:

Dynamic Programming: General method, applications-Matrix chain multiplication, Optimal binary search trees, 0/1 knapsack problem, All pairs shortest path problem, Travelling sales person problem, Reliability design.

UNIT-IV:

Backtracking: General method, applications-n-queen problem, sum of subsets problem, graph coloring, Hamiltonian cycles.

UNIT-V:

Branch and Bound: General method, applications - Travelling sales person problem,0/1 knapsack problem- LC Branch and Bound solution, FIFO Branch and Bound solution. NP Hard and NP-Complete problems: Basic concepts, non-deterministic algorithms, NP - Hard and NP Complete classes, Cook's theorem.

TEXT BOOKS:

1. Fundamentals of Computer Algorithms, Ellis Horowitz, Satraj Sahni and Rajasekharam, Universities Press.
2. The Algorithm Design Manual, 2nd edition, Steven S. Skiena, Springer.
3. Introduction to Algorithms, second edition, T.H.Cormen, C.E.Leiserson, R.L.Rivest and C.Stein, PHI Pvt. Ltd.

REFERENCE BOOKS:

1. Introduction to the Design and Analysis of Algorithms, AnanyLevitin, PEA
2. Design and Analysis of Algorithms, Parag Himanshu Dave, Himansu B Alachandra Dave, Pearson Education.
3. Introduction to Design and Analysis of Algorithms A strategic approach, R.C.T. Lee, S.S.Tseng, R.C.Chang and T.Tsai, McGraw Hill.
4. Design and Analysis of algorithms, Aho, Ullman and Hopcroft, Pearson education.

SWARNANDHRA

COLLEGE OF ENGINEERING AND TECHNOLOGY
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SOFTWARE TESTING METHODOLOGIES
M.TECH (R16) II SEMESTER

COURSE OBJECTIVES:

1. To study fundamental concepts in software testing, including software testing objectives, process, criteria, strategies, and methods.
2. To discuss various software testing issues and solutions in software unit test; integration, regression, and system testing.
3. To learn how to planning a test project, design test cases and data, conduct testing operations, manage software problems and defects, generate a testing report.
4. To expose the advanced software testing topics, such as object-oriented software testing methods, and component-based software testing issues, challenges, and solutions.
5. To gain software testing experience by applying software testing knowledge and methods to practice-oriented software testing projects.

COURSE OUTCOMES:

By the end of the course, the student should:

1. Have an ability to apply software testing knowledge and engineering methods.
2. Have an ability to design and conduct a software test process for a software testing project.
3. Have an ability to identify the needs of software test automation, and define and develop a test tool to support test automation.
4. Have an ability understand and identify various software testing problems, and solve these problems by designing and selecting software test models, criteria, strategies, and methods.

Syllabus:

UNIT I:

Software Testing: Introduction, Evolution, Myths & Facts, Goals, Psychology, Definition, Model for testing, Effective Vs Exhaustive Software Testing. Software Testing Terminology and Methodology: Software Testing Terminology, Software Testing Life Cycle, relating test life cycle to development life cycle Software Testing Methodology.

UNIT II:

Verification and Validation: Verification & Validation Activities, Verification, Verification of Requirements, High level and low level designs, How to verify code, Validation

Dynamic Testing I: Black Box testing techniques: Boundary Value Analysis, Equivalence class Testing, State Table based testing, Decision table based testing, Cause-Effect Graphing based testing, Error guessing

UNIT III:

Dynamic Testing II: White-Box Testing: need, Logic coverage criteria, Basis path testing, Graph matrices, Loop testing, data flow testing, mutation testing Static Testing: inspections, Structured Walkthroughs, Technical reviews

UNIT IV:

Validation activities: Unit testing, Integration Testing,. Function testing, system testing, acceptance testing Regression testing: Progressives Vs regressive testing, Regression testability, Objectives of regression testing, When regression testing done? Regression testing types, Regression testing techniques

UNIT V:

Software Quality Management: Software Quality metrics, SQA models

Debugging: process, techniques, correcting bugs, Basics of testing management tools, test link and Jira Automation and Testing Tools: need for automation, categorization of testing tools, selection of testing tools, Cost incurred, Guidelines for automated testing, overview of some commercial testing tools.

Testing Object Oriented Software: basics, Object oriented testing

Testing Web based Systems: Challenges in testing for web based software, quality aspects, web engineering, testing of web based systems, testing mobile systems

Text Books:

1. Software Testing, Principles and Practices, Naresh Chauhan, Oxford
2. Foundations of Software testing, Aditya P Mathur, 2ed, Pearson
3. Software Testing- Yogesh Singh, CAMBRIDGE

Reference books:

1. Software testing techniques - Baris Beizer, International Thomson computer press, second edition.
2. Software Testing, Principles, techniques and Tools, M G Limaye, TMH
3. Effective Methods for Software testing, Willian E Perry, 3ed, Wiley

SWARNANDHRA

COLLEGE OF ENGINEERING AND TECHNOLOGY
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SOFTWARE TESTING METHODOLOGIES LAB
M.TECH (R16) II SEMESTER

Lab Assignments

Problem Statement 01

Consider an automated banking application. The user can dial the bank from a personal computer, provide a six-digit password, and follow with a series of keyword commands that activate the banking function. The software for the application accepts data in the following form:

Area Code	Blank or three-digit number
Prefix	Three-digit number, not beginning with 0 or 1
Suffix	Four-digit number
Password	Six-character alphanumeric
Commands	"Check status", "Deposit", "Withdrawal"

Design adhoc test cases to test the system

Problem Statement 02

Consider an automated banking application. The user can dial the bank from a personal computer, provide a six-digit password, and follow with a series of keyword commands that activate the banking function. The software for the application accepts data in the following form:

Area Code	Blank or three-digit number
Prefix	Three-digit number, not beginning with 0 or 1
Suffix	Four-digit number
Password	Six-character alphanumeric
Commands	"Check status", "Deposit", "Withdrawal"

Design the test cases to test the system using following Black Box testing technique: BVA, Worst BVA, Robust BVA, Robust Worst BVA Equivalence class testing (Input/output domain)

Problem Statement 03

Consider an application that is required to validate a number according to the following simple rules:

1. A number can start with an optional sign.
2. The optional sign can be followed by any number of digits.
3. The digits can be optionally followed by a decimal point, represented by a period.
4. If there is a decimal point, then there should be two digits after the decimal.
5. Any number-whether or not it has a decimal point, should be terminated a blank.
6. A number can start with an optional sign.
7. The optional sign can be followed by any number of digits.
8. The digits can be optionally followed by a decimal point, represented by a period.
9. If there is a decimal point, then there should be two digits after the decimal.

10. Any number-whether or not it has a decimal point, should be terminated a blank. Generate test cases to test valid and invalid numbers.
(HINT) Use Decision table and cause-effect graph to generate test cases.

Problem Statement 04
Generate test cases using Black box testing technique to Calculate Standard Deduction on Taxable Income. The standard deduction is higher for tax payers who are 65 or older or blind. Use the method given below to calculate tax.

1. The first factor that determines the standard deduction is the filing status. The basic standard deduction for the various filing status are:

Single	$4,750
Married, filing a joint return	$9,500
Married, filing a separate return	$7,000

2. If a married couple is filing separate returns and one spouse is not taking standard Deduction, the other spouse also is not eligible for standard deduction.

3.An additional $1,000 is allowed as standard deduction, if either the filer is 65 yrs. or the spouse is 65 yrs. or older (the latter case applicable when the filing status is “Married” and filing “joint”).

4. An additional $1,000 is allowed as standard deduction, if either the filer is blind or the spouse is blind (the latter case applicable when the filing status is
“Married” and filing “joint”).
(HINT):
From the above description, it is clear that the calculation of standard deduction depends on the following 3 factors:
1. Status of filing of the filer
2. Age of the filer
3. Whether the filer is blind or not
In addition, in certain cases, the following additional factors also come into play in calculating the standard deduction.
1. Whether spouse has claimed standard deduction
2. Whether spouse is blind
3. Whether the spouse is more than 65 years old

Problem Statement 05
Consider the following program segment:
1. int max (int i, int j, int k) 2. {
3. int max;
4. if (i>j) then
5. if (i>k) then max=i;
6. else max=k;
7. else if (j > k) max=j
8. else max=k
9. return (max);
10. }
a) Draw the control flow graph for this program segment
b) Determine the cyclomatic complexity for this program
c) Determine the independent paths

Problem Statement 06

Source code of simple insertion sort implementation using array in ascending order in c programming language

```
int
main(){
int i,j,s,temp,a[20]
Printf ("Enter total elements: "); Scanf ("%d",&s);
printf("Enter %d elements: ",s); for(i=0;i<s;i++) scanf("%d",&a[i]);
for(i=1;i<s;i++){ temp=a[i]; j=i-1; while((temp<a[j])&&(j>=0)){
a[j+1]=a[j];
j=j-1;
}
a[j+1]=temp;
}
printf("After
sorting: ");
for(i=0;i<s;i++)
printf("
%d",a[i]);
return 0;
}
```

HINT: for loop is represented as while loop

a) Draw the program graph for given program segment
b) Determine the DD path graph
c) Determine the independent paths
d) Generate the test cases for each independent path

Problem Statement 07

Consider a system having an FSM for a stack having the following states and transitions:

States
Initial: Before creation
Empty: Number of elements = 0
Holding: Number of elements > 0, but less than the maximum capacity Full: Number elements = maximum
Final: After destruction
Initial to Empty: Create
Empty to Holding, Empty to Full, Holding to Holding, Holding to Full: Add Empty to Final, Full to Final, Holding to Final: Destroy
Holding to Empty, Full to Holding, Full to Empty: Delete

Design test cases for this FSM using state table-based testing

Problem Statement 08

Given the following fragment of code, how many tests are required for 100% decision coverage? Give the test cases.

if width > length
then biggest_dimension = width if height > width
then biggest dimension = height end_if
else if biggest dimension = length

then if height > length
then biggest_dimension = height end_if
end_if end_if
Hint 04 test cases

Problem Statement 09

Given the following code, how much minimum number of test cases is required for full statement and branch coverage?

```
read p read q
if p+q> 100
then print "Large" endif
if p > 50
then print "p Large" endif
```

Hint 1 test for statement coverage, 2 for branch coverage

Problem Statement 10

Consider a program to input two numbers and print them in ascending order given below. Find all du paths and identify those du-paths that are not feasible. Also find all dc paths and generate the test cases for all paths (dc paths and non dc paths).

```
#include<stdio.h>
#include<conio.h>

1. void main ()
2. {
3 int a, b, t;
4.      Clrscr ();
5.      Printf ("Enter first number");
6. scanf ("%d",&a);
7. printf("Enter second number");
8. scanf("%d",&b);
9.      if (a<b){
10.     t=a;
11a=b;
12 b=t;
13}
14. printf ("%d %d", a, b);
15 getch ();
}
```

Problem Statement 11

Consider the above program and generate possible program slices for all variables. Design at least one test case from every slice.

Problem Statement 12

Consider the code to arrange the nos. in ascending order. Generate the test cases for relational coverage, loop coverage and path testing. Check the adequacy of the test cases through mutation testing and also compute the mutation score for each.

```
i = 0;
n=4; //N-Number of nodes present in
the graph While (i<n-1) do j = i + 1;
While (j<n) do
if A[i]<A[j] then swap (A[i],
A[j]); end do; i=i+1;
end do
```

SWARNANDHRA

COLLEGE OF ENGINEERING AND TECHNOLOGY
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

WIRELESS NETWORKS

M.TECH (R16) II SEMESTER

COURSE OBJECTIVES:

At the end of the course, student
1. realizes the benefits of “wireless communication" in mobile applications
2. Realizes the limitations of allocation of frequencies
3. Know the optimal utilization of bandwidth

COURSE OUTCOMES:

Upon successful completion of the course, students get
1. General knowledge of Wireless transmission in mobile Communication technology.
2. Brief history of type’s communication system and its working
3. Brief knowledge IEEE wireless standards.

SYLLABUS:

UNIT I

Introduction: Wireless transmission, Frequencies for Radio Transmission, Signals, Antennas, Signal Propagation, Multiplexing, Modulations, Spread Spectrum, MAC, SDMA, FDMA, TDMA, CDMA, Cellular Wireless Networks.

UNIT II

Telecommunication Systems: GSM, GPRS, Satellite Networks, Basics, Parameters and Configurations, Capacity Allocation, FAMA and DAMA, Broadcast Systems, CDMA and 3G.

UNIT III

Wireless LAN: IEEE 802.11, Architecture, Services, MAC-Physical Layer, IEEE 802.11a-802.11b Standards, Bluetooth enabled devices network, layers in Bluetooth protocol.

UNIT IV

MOBILE AD-HOC AND WIRELESS SENSOR NETWORKS: introduction to mobile ad-hoc network, MANET, security in ad-hoc network, wireless sensor networks, sensor network applications.

UNIT V

MOBILE WIRELESS SHORT RANGE NETWORKS: WAP (Wireless Application Protocol), WAP 2.0, IrDA, ZigBee protocols, 4G networks.

Text Books:

1. Raj Kamal, “Mobile Computing”, Oxford, 2009
2. William Stallings, “Wireless Communications and Networks”, Second Edition, Prentice Hall of India, Pearson Education, 2004.

Reference Books:

1. Asoke K Talukder,Roopa R Yavagal, “Mobile Computing”, TMH 2008
2. Jochen, M Schiller,“Mobile Communications, 2ndEdition Pearson Education, India, 2009.

Reference Books:

SWARNANDHRA

COLLEGE OF ENGINEERING AND TECHNOLOGY
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CLOUD COMPUTING

M.TECH (R16) II SEMESTER (ELECTIVE-III)

COURSE OBJECTIVES:

1. To understand the concepts of Cloud Computing.
2. To learn Taxonomy of Virtualization Techniques.
3. To learn Cloud Computing Architecture.
4. To acquire knowledge on Aneka Cloud Application Platform.
5. To learn Industry Cloud Platforms.

COURSE OUTCOMES:

1. Understand the concept of virtualization and how this has enabled the development of Cloud Computing.
2. Know the fundamentals of Cloud and cloud Architectures and types of services in cloud.
3. Understand scaling, cloud security and disaster management.
4. Design different Applications in cloud.
5. Explore some important cloud computing driven commercial systems such as Google Apps, Microsoft Azure and Amazon Web Services and other businesses cloud applications.

SYLLABUS:

UNIT I

Introduction to Cloud: Cloud Computing at a Glance, the Vision of Cloud Computing, Defining a Cloud, A Closer Look, Cloud Computing Reference Model. Characteristics and Benefits, Challenges Ahead, Historical Developments.

Virtualization: Introduction, Characteristics of Virtualized Environment, Taxonomy of Virtualization Techniques, Virtualization and Cloud computing,
Pros and Cons of Virtualization, Technology Examples- VMware and Microsoft Hyper-V.

Before the Move into the Cloud: Know Your Software Licenses, the Shift to a Cloud Cost Model, Service Levels for Cloud Applications.

UNIT II

Cloud Computing Architecture : Introduction, Cloud Reference Model, Architecture, Infrastructure / Hardware as a Service, Platform as a Service, Software as a Service, Types of Clouds, Public Clouds, Private Clouds, Hybrid Clouds, Community Clouds, Economics of the Cloud, Open Challenges, Cloud Interoperability and Standards, Scalability and Fault Tolerance.

Ready for the Cloud: Web Application Design, Machine Image Design, Privacy Design, Database Management, Data Security, Network Security, Host Security, Compromise Response.

UNIT III

Defining the Clouds for Enterprise: Storage as a service, Database as a service, Process as a service, Information as a service and Integration as a service and Testing as a service.

Scaling a cloud infrastructure - Capacity Planning, Cloud Scale.

Disaster Recovery: Disaster Recovery Planning, Disasters in the Cloud, Disaster Management.

UNIT IV

Aneka: Cloud Application Platform Framework Overview, Anatomy of the Aneka Container, From the Ground Up: Platform Abstraction Layer, Fabric Services, Foundation Services, Application Services, Building Aneka Clouds, Infrastructure Organization, Logical Organization, Private Cloud Deployment Mode, Public Cloud Deployment Mode, Hybrid Cloud Deployment Mode, Cloud Programming and Management ,Aneka SDK, Management Tools.

UNIT V

Cloud Applications: Scientific Applications – Health care, Geo science and Biology. Business and Consumer Applications- CRM and ERP, Social Networking, Media Applications and Multiplayer Online Gaming.

Cloud Platforms in Industry: Amazon Web Services- Compute Services, Storage Services, Communication Services and Additional Services. Google App Engine-Architecture and Core Concepts, Application Life-Cycle, cost model. Microsoft Azure- Azure Core Concepts, SQL Azure.

Text Books

1. "Mastering Cloud Computing" by Rajkumar Buyya, Christian Vecchiola, S.Thamarai Selvi from TMH 2013.
2. George Reese "Cloud Application Architectures", First Edition, O'Reilly Media 2009.

References

1. "Cloud Computing and SOA Convergence in Your Enterprise" *A Step-by-Step Guide* by David S. Linthicum from Pearson 2010.
2. "Cloud Computing" 2nd Edition by Dr. Kumar Saurabh from Wiley India 2012.
3. "Cloud Computing" – web based Applications that change the way you work and collaborate Online – Micheal Miller.Pearson Education.

SWARNANDHRA

COLLEGE OF ENGINEERING AND TECHNOLOGY
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

ADVANCED COMPUTER ARCHITECTURE

M. TECH(R16) II SEMESTER (ELECTIVE-III)

COURSE OBJECTIVES:

1. Give students a broad and deep knowledge of contemporary computer architecture issues and techniques.
2. Give students knowledge of advanced hardware-based techniques for exploiting instruction level parallelism.
3. Give students knowledge of various architectures and techniques used for building high performance scalable multithreaded and multiprocessor systems.
4. Give students ability to apply the learned knowledge to conduct computer architecture research using performance simulators.

COURSE OUTCOMES:
1) Identify the factors affecting performance in superscalar processors and the key components, options and tradeoffs that a designer has to consider when designing such processors.
2) Identify various simulation techniques used to study superscalar processor performance.
3)Compare a trace cache to conventional instruction cache and explain advantages and disadvantages of each approach.

UNIT -I: Fundamentals of Computer Design: Fundamentals of Computer design, Changing faces of computing and task of computer designer, Technology trends, Cost price and their trends, Measuring and reporting performance, Quantitative principles of computer design, Amdahl's law. Instruction set principles and examples- Introduction, Classifying instruction set- Memory addressing- type and size of operands, Operations in the instruction set.

UNIT –II: Pipelines: Introduction, Basic RISC instruction set, Simple implementation of RISC instruction set, Classic five stage pipe lined RISC processor, Basic performance issues in pipelining, Pipeline hazards, and Reducing pipeline branch penalties.
Memory Hierarchy Design: Introduction, Review of ABC of cache, Cache performance, Reducing cache miss penalty, Virtual memory.

UNIT -III: Instruction Level Parallelism the Hardware Approach: Instruction-Level parallelism, Dynamic scheduling, Dynamic scheduling using Tomasulo's approach, Branch prediction, high performance instruction delivery- hardware based speculation.

UNIT –IV: Multi Processors and Thread Level Parallelism: Multi Processors and Thread level Parallelism- Introduction, Characteristics of application domain, Systematic shared memory architecture, Distributed shared – memory architecture, Synchronization.

UNIT – V: Inter Connection and Networks: Introduction, Interconnection network media, Practical issues in interconnecting networks, Examples of inter connection, Cluster, Designing of clusters.

TEXT BOOKS
1) John L. Hennessy, David A. Patterson – Computer Architecture: A Quantitative Approach, 3rd Edition, An Imprint of Elsevier.
REFERENCES
1) John P. Shen and Miikko H. Lipasti – Modern Processor Design : Fundamentals of Super Scalar Processors
2) Computer Architecture and Parallel Processing – Kai Hwang, Faye A.Brigs., MC Graw Hill.
3) Advanced Computer Architecture – A Design Space Approach – Dezso Sima, Terence Fountain, Peter Kacsuk , Pearson Ed.

DATA WAREHOUSING AND DATA MINING
M.TECH (R16) II SEMESTER (ELECTIVE-III)

COURSE OBJECTIVES:

1. To introduce the basic concepts of Data Warehouse and Data Mining techniques.
2. Examine the types of the data to be mined and apply pre-processing methods on raw data.
3. Discover interesting patterns, analyses supervised and unsupervised models and estimate the accuracy of the algorithms.

COURSE OUTCOMES:

Students should be able to
1. Process raw data to make it suitable for various data mining algorithms.
2. Discover and measure interesting patterns from different kinds of databases.
3. Apply the techniques of clustering, classification, association finding, feature selection and visualization to real world data.

SYLLABUS:

UNIT I: Introduction:
Fundamentals of data mining, Data Mining Functionalities, Classification of Data Mining systems, Major issues in Data Mining.
Data Pre-processing: Needs Pre-processing the Data, Data Cleaning, Data Integration and Transformation, Data Reduction, Discretization and Concept Hierarchy Generation.

UNIT II: Data Warehouse and OLAP:
Data Warehouse and OLAP Technology for Data Mining Data Warehouse, Multidimensional Data Model, Data Warehouse Architecture, Data Warehouse Implementation, Further Development of Data Cube Technology, From Data Warehousing to Data Mining.

UNIT III: Mining Association Rules in Large Databases:
Association Rule Mining, Mining Single-Dimensional Boolean Association Rules from Transactional Databases, Mining Multilevel Association Rules from Transaction Databases, Mining Multidimensional Association Rules from Relational Databases and Data Warehouses, From Association Mining to Correlation Analysis.

UNIT IV: Advanced Classification and Prediction:
Issues Regarding Classification and Prediction, Classification by Decision Tree Induction, Bayesian Classification, Classification by Back propagation, Classification Based on Concepts from Association Rule Mining, support vector machines, Prediction, Classifier Accuracy.

UNIT V: Advanced Cluster Analysis:
Types of Data in Cluster Analysis, A Categorization of Major Clustering Methods, Partitioning Methods, Density-Based Methods, Grid-Based Methods, Model-Based Clustering Methods, Outlier Analysis, Clustering Graph and Network Data.

Web and Text Mining
Introduction, web mining, web content mining, web structure mining, we usage mining, Text mining – unstructured text, episode rule discovery for texts, hierarchy of categories, text clustering.

TEXT BOOKS:

1. Data Mining, Concepts and Techniques, JIAWEI HAN & MICHELINE KAMBER Harcourt India.

REFERENCE BOOKS:

1. Data Mining Introductory and advanced topics, MARGARET H DUNHAM, PEARSON EDUCATION
2. Data Mining Techniques, ARUN K PUJARI, University Press.
3. Data Warehousing in the Real World, SAM ANAHORY & DENNIS MURRAY. Pearson Edn Asia.

MACHINE LEARNING

M. Tech (R16) II SEMESTER (ELECTIVE-IV)

COURSE OBJECTIVES:

The main objective of this course is for the students to achieve basic knowledge of artificial intelligence, a deepened technical understanding of machine learning research and theories, as well as practical experience of the use and design of machine learning and data mining algorithms for applications and experiments. The course has a strong focus towards applied IT. The student not only learns how to critically review and compare different algorithms and methods, but how to plan, design, and implement learning components and applications and how to conduct machine learning experiments.

COURSE OUTCOMES:

The student will be

1. Able evaluate and compare the performance or, other qualities, of algorithms for typical learning problems.
2. The student will be able to design a supervised or unsupervised learning system.

Syllabus:

UNIT I: Introduction:

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

UNIT II: Linear Regression & Logistic Regression:

Predicting numeric values: regression - Finding the best fit lines with linear regression, locally weighted linear regression, Shrinking Coefficients, The bias / Variance tradeoff.

Logistic Regression: Classification with logistic regression and the sigmoid function, using optimization to find the best regression coefficients.

UNIT III: Artificial Neural Networks:

Introduction, Neural network representation, Appropriate problems for neural network learning, Perceptions, Multilayer networks and the back propagation algorithm, Remarks on the back propagation algorithm, An illustrative example face recognition, Advanced topics in artificial neural networks

UNIT IV: Evaluation Hypotheses: Motivation, Estimation hypothesis accuracy, Basics of sampling theory, a general approach for deriving confidence intervals, Difference in error of two hypotheses, Comparing learning algorithms.

UNIT V: Support vector machines & Dimensionality Reduction techniques:

Separating data with the maximum margin, finding the maximum margin, efficient optimization with SMO algorithm, speeding up optimization with full Platt SMO, Using Kernels for more Complex data. Dimensionality Reduction techniques: Principal Component analysis, Example.

TEXT BOOKS:

1. Machine Learning, Tom M. Mitchell, MGH
2. Machine Learning in Action, Peter Harington, 2012, Cengage.

REFERENCE BOOKS:

1. Introduction to Machine Learning, Ethem Alpaydin, PHI, 2004

SWARNANDHRA

COLLEGE OF ENGINEERING AND TECHNOLOGY
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

MOBILE APPLICATION DEVELOPMENT

M. TECH (R16) II SEMESTER (ELECTIVE-IV)

COURSE OBJECTIVES:

1. To make the student understand the concept of mobile computing paradigm, its novel applications and limitations.
2. To understand the typical mobile networking infrastructure through a popular GSM protocol
3. To understand the issues and solutions of various layers of mobile networks, namely MAC layer, Network Layer & Transport Layer
4. To understand the mobile application development language concepts.

COURSE OUTCOMES:

1. Able to think and develop new mobile application.
2. Able to take any new technical issue related to this new paradigm and come up with a solution(s).
3. Able to develop new ad hoc network applications and/or algorithms/protocols.
4. Able to understand & develop any existing or new protocol related to mobile environment

SYLLABUS:

UNIT I:

Mobile Communications: Mobile Communication, Guided and Unguided media, Mobile Computing, Architecture, Limitations and Novel Applications, Mobile Devices, Mobile System Networks.

UNIT II: Mobile Devices and Systems: Cellular Networks and Frequency Reuse, Mobile Smart Phones, Smart Mobiles and Systems, Handheld Pocket Computers, Handheld Devices, Smart Systems.

UNIT III: Introduction about Android: Features of Android, Applications, Android architecture, Android emulator, Installation procedure.

GSM: GSM Services and System Architecture, Radio Interface of GSM, Protocols of GSM, Handover, Security

UNIT IV: MOBILE DEVICES: APPLICATION SERVERS AND MANAGEMENT

Mobile Agent, Application Framework, Application Server, Gateways, Service Discovery, Device management, Security**.**

UNIT V: MOBILE APPLICATION LANGUAGES: Mobile Application Development, XML, JAVA, J2ME, Java Card

Mobile Application Development Platforms: operating system, Windows mobile and CE, Android, Symbian.

TEXT BOOKS:

1. Mobile Computing, Raj Kamal, 2nd Edition, Oxford University Press, 2012.

REFERENCE BOOKS:

1. Mobile Communications, Jochen Schiller, Pearson Education, Second Edition, 2008.
2. Wireless Communications and Networks, 2nd Edition, William Stallings, Person Education,2007.
3. Handbook of Wireless Networks and Mobile Computing, Ivan Stojmenovic, Wiley, 2007.
4. Wireless and Mobile Networks: Concepts and Protocols, Dr. Sunilkumar, et al, Wiley India

SWARNANDHRA

COLLEGE OF ENGINEERING AND TECHNOLOGY
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BIG DATA ANALYTICS

M. TECH (R16) II SEMESTER (ELECTIVE-IV)

COURSE OBJECTIVES:

1. Understand and apply the Big Data Flow to actual projects
2. An able to describe and apply the Data Analytics lifecycle to Big Data projects and lead other team members in the process
3. Identify and successfully apply appropriate techniques and tools to solve actual Big Data problems

COURSE OUTCOMES: At the end of the module

1. Students will possess the skills necessary for utilizing tools (including deploying them on Hadoop/MapReduce) to handle a variety of big data analytics.
2. And to be able to apply the analytics techniques on a variety of applications.

UNIT-I

Data structures in Java: Linked List, Stacks, Queues, Sets, Maps; Generics: Generic classes and Type parameters, Implementing Generic Types, Generic Methods, Wrapper Classes, Concept of Serialization

UNIT-II

Working with Big Data: Google File System, Hadoop Distributed File System (HDFS) – Building blocks of Hadoop (Namenode, Datanode, Secondary Namenode, Job Tracker, Task Tracker), Introducing and Configuring Hadoop cluster (Local, Pseudo-distributed mode, Fully Distributed mode), Configuring XML files.

UNIT-III

Writing MapReduce Programs: A Weather Dataset, Understanding Hadoop API for MapReduce Framework (Old and New), Basic programs of Hadoop MapReduce: Driver code, Mapper code, Reducer code, Record Reader, Combiner, Partitioner

UNIT-IV

Hadoop I/O: The Writable Interface, Writable Comparable and comparators, Writable Classes: Writable wrappers for Java primitives, Text, Bytes Writable, Null Writable, Object Writable and Generic Writable, Writable collections, Implementing a Custom Writable: Implementing a Raw Comparator for speed, Custom comparators

UNIT-V

Pig: Hadoop Programming Made Easier

Admiring the Pig Architecture, Going with the Pig Latin Application Flow, Working through the ABCs of Pig Latin, Evaluating Local and Distributed Modes of Running Pig Scripts, Checking out the Pig Script Interfaces, Scripting with Pig Latin

Applying Structure to Hadoop Data with Hive:

Saying Hello to Hive, Seeing How the Hive is Put Together, Getting Started with Apache Hive, Examining the Hive Clients, Working with Hive Data Types, Creating and Managing Databases and Tables, Seeing How the Hive Data Manipulation Language Works, Querying and Analyzing Data

TEXT BOOKS:

1. Big Java 4th Edition, Cay Horstmann, Wiley John Wiley & Sons, INC
2. Hadoop: The Definitive Guide by Tom White, 3rd Edition, O'reilly

3. Hadoop in Action by Chuck Lam, MANNING Publ.
4.Hadoop for Dummies by Dirk deRoos, Paul C.Zikopoulos, Roman B.Melnyk,Bruce Brown, Rafael Coss

REFERENCE BOOKS:

1. Hadoop in Practice by Alex Holmes, MANNING Publ.
2. Hadoop MapReduce Cookbook, Srinath Perera, Thilina Gunarathne

SOFTWARE LINKS:

1.Hadoop:http://hadoop.apache.org/
2.Hive: https://cwiki.apache.org/confluence/display/Hive/Home
3.Piglatin: http://pig.apache.org/docs/r0.7.0/tutorial.html