

**ACADEMIC REGULATIONS
WITH
COURSE STRUCTURE
& DETAILED SYLLABUS**

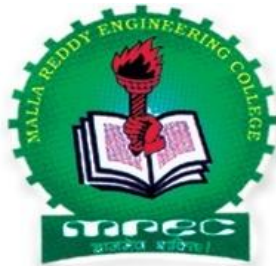
for

M.Tech (Computer Science & Engg.)

with

COMPUTER SCIENCE AND ENGINEERING SPECIALISATION

(Applicable for batches admitted from Academic Year 2011-12)



**MALLA REDDY ENGINEERING COLLEGE
(AUTONOMOUS)**

(Approved by AICTE & Affiliated to JNTUH)

Maisammaguda, Dhulapally (Po) Via (Hakimpet), Secunderabad- 500 014.

MALLA REDDY ENGINEERING COLLEGE

(Autonomous)
Maisammaguda, Dhulapally, Post Via (Hakimpet), Secunderabad- 500 014.

Academic Regulations 2011 for M.Tech. (Regular)

(Effective for the students admitted into first year from the academic year 2011-2012 onwards)

The M.Tech Degree of Malla Reddy Engineering College, Hyderabad shall be conferred on candidates by the Jawaharlal Nehru Technological University Hyderabad, Hyderabad who are admitted to the program and fulfill all the requirements for the award of the Degree.

1.0 ELIGIBILITY FOR ADMISSIONS:

Admission to the above program shall be made subject to the eligibility, qualifications and specialization prescribed by the university/college from time to time.

Admissions shall be made on the basis of merit rank obtained by the qualifying candidate at an Entrance Test conducted by the university/college or on the basis of any other order of merit approved by the university/college (say PGECET), subject to reservations prescribed by the university/college 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 and completes it successfully for not less than two academic years and not more than four academic years.

2.2 A student, who fails to fulfill all the academic requirements for the award of the degree within four academic years from the year of his admission, shall forfeit his seat in M.Tech course.

2.3 The minimum instruction for each semester 90 clear instruction days.

3.0 A. COURSE OF STUDY:

The following specializations are offered at present for the M.Tech course of study.

1. Computer Science and Engineering
2. Control Systems
3. Control Engineering
4. Structural Engineering
5. Thermal Engineering
6. Digital Systems & Computer Electronics

and any other course as approved by the authorities of the university/college from time to time.

3.0 B. Departments offering M. Tech Programs with Specializations mentioned below:

Civil Engineering Department	Structural Engineering
Computer Science and Engineering Department	Computer Science and Engineering
Electrical Electronics Engineering Department	Control Systems
-do-	Control Engineering
Electronics and Communication Engineering Department	Digital Systems and Computer Electronics
Mechanical Engineering Department	Thermal Engineering

4.0 ATTENDANCE:

The programs are offered on a unit basis with each subject being considered unit.

4.1 A candidate shall be deemed to have eligibility to write end semester examinations in a subject if he has put in at least 75% of attendance in that subject.

4.2 Shortage of attendance up to 10% in any subject (i.e. 65% and above and below 75%) may be condoned by the College Academic Committee on genuine and valid reasons on representation by the candidate with supporting evidence.

4.3 A candidate shall get minimum required attendance at least in three (3) theory subjects in the present semester to get promoted to the next semester. In order to qualify for the award of the M.Tech. Degree, the candidate shall complete all the academic requirements of the subjects, as per the course structure.

4.4 Shortage of attendance below 65% shall **in no case be condoned**.

4.5 A stipulated fee shall be payable towards condonation of shortage of attendance.

5.0 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, on the basis of Internal Evaluation and End Semester Examination.

5.1 For the theory subjects 60 marks shall be awarded based on the performance in the End Semester Examination, 40 marks shall be awarded based on the Internal Evaluation. The internal evaluation shall be made based on the better 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 for a total of 30 marks. Each mid term examination shall be conducted for a duration of 120 minutes with 4 questions to be answered out of 6 questions. In addition, there shall be two assignments evaluated for 10 marks each and

average of the two taken as the final assignment mark. The sum of the best of the two mid examinations and the assignment marks obtained shall be the final marks for internal evaluation.

5.2 For practical subjects, 60 marks shall be awarded based on the performance in the End Semester Examinations, 40 marks shall be awarded based on the day-to-day performance as internal Marks.

5.3 There shall be two seminar presentations during I year I semester and II Semesters. 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 Departmental Committee. The Departmental Committee consists 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% to be declared successful.

5.4 There shall be a Comprehensive Viva-Voce in II year I Semester. The Comprehensive Viva-Voce will be conducted by a Committee consisting of Head of the Department and two Senior Faculty members in that area of specialisation. The Comprehensive Viva-Voce is aimed to assess the students' understanding in various subjects he/she studies during the M.Tech course of study. The Comprehensive Viva-Voce is valued for 100 marks by the Committee. There are no internal marks for the Comprehensive viva-Voce

5.5 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 Examination and a minimum aggregate of 50% of the total marks in the End Semester Examination and Internal Evaluation taken together.

5.6 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 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 he has failed in the end examination. In such case candidate must re-register for the subject(s) and secure required minimum attendance. Attendance in the re-registered subject(s) has to be calculated separately to become eligible to write the end examination in the re-registered subject(s). The attendance of re-registered subject(s) shall be calculated separately to decide upon the eligibility for writing the end examination in those subject(s). In the event of taking another chance, the internal marks and end examination marks obtained in the previous attempt are nullified.

5.7 In case the candidate secures less than the required attendance in any subject(s), he shall not be permitted to appear for the End Examination in that subject(s). He shall re-register the subject when next offered.

5.8 Laboratory examination for M.Tech courses must be conducted with two Examiners, one of them being Laboratory Class Teacher and second examiner shall be other Laboratory Teacher or any other member from inside/outside of the college.

6.0 EVALUATION OF PROJECT/ DISSERTATION WORK:

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

6.1 A Project Review Committee (PRC) shall be constituted with Principal as chair person

Heads of all the Departments which are offering the M.Tech programs and two other senior faculty members.

6.2 Registration of Project Work: A candidate is permitted to register for the project work after satisfying the attendance requirement of all the subjects (theory and practical subjects).

6.3 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 to the Departmental Committee for its approval. Only after obtaining the approval of Departmental Committee the student can initiate the Project work.

6.4 If a candidate wishes to change his supervisor or topic of the project he can do so with approval of Departmental Committee. However, the Departmental Committee shall examine whether the change of topic/supervisor leads to a major change of his initial plans of project proposal. If so, his date of registration for the project work starts from the date of change of Supervisor or topic as the case may be.

6.5 Candidate shall submit status report (in a bound-form) in two stages at least with a gap of 3 months between them.

6.6 The work on the project shall be initiated in the beginning of the second year and the duration of the project is for 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. For the approval of PRC the candidate shall submit the draft copy of thesis to the Principal (through Head of the Department) and shall make an oral presentation/demonstration before the PRC.

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

6.8 The thesis shall be adjudicated by one examiner selected by the College. For this, Head of the Department shall submit a panel of 5 examiners to the Principal of the College, who are eminent in that field with the help of the concerned guide and Head of the department.

6.9 If the report of the examiner is not favorable, the candidate shall revise and resubmit the Thesis, in the time frame as described by PRC. If the report of the examiner is unfavorable again, the thesis shall be summarily rejected.

6.10 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 candidates work as:

- A. Excellent
- B. Good
- C. Satisfactory
- D. Unsatisfactory

Head of the Department shall coordinate and make arrangements for the conduct of viva-voce examination.

If the report of the viva-voce is unsatisfactory, the candidate will retake the viva-voce

examination after three months. If he fails to get a satisfactory report at the second viva-voce examination, he will not be eligible for the award of the degree.

7.0 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
First Class	Below 70% but not less than 60%
Second Class	Below 60% but not less than 50%
Pass Class	Below 50% but not less than 40%

(The marks in internal evaluation and end examination shall be shown separately in the marks memorandum)

8.0 WITH-HOLDING OF RESULTS:

If the candidate has not paid any dues to the university or if any case of in-discipline is pending against him, the result of the candidate will be withheld and he will not be allowed into the next higher semester. The issue of the degree is liable to be withheld in such cases.

9.0 TRANSITORY REGULATIONS:

Candidate who have discontinued or have been detained for want of attendance or who have failed after having undergone the course are eligible for admission to the same or equivalent subjects as and when subjects are offered, subject to rule 5.5 and 2.0 of these regulations.

10.0 GENERAL:

10.1 The academic regulations should be read as a whole for purpose of any interpretation.

10.2 In case of any doubt or ambiguity in the interpretation of the above rules, the decision of the Principal is final.

10.3 The College may change or amend the academic regulations and syllabus at any time and the changes and amendments made shall be applicable to all the students with effect from the date notified by the University/College.

10.4 Wherever the word he, him or his occur, it will also include she, her and hers.

10.5 There shall be no transfers within the constituent colleges of Jawaharlal Nehru Technological University.

MALLA REDDY ENGINEERING COLLEGE

(*Autonomous*)

**M.TECH (COMPUTER SCIENCE & ENGINEERING)
COURSE STRUCTURE**

I YEAR I SEMESTER

Code	Group	Subject	L	T	P	Internal Val	End Exam	Total
CSE511101		Advanced Problem Solving	3	1	0	40	60	100
CSE511102		Computer System Design	3	1	0	40	60	100
CSE511103		Embedded Systems and RTOS	3	1	0	40	60	100
CSE511104		Java and Web Technologies	3	1	0	40	60	100
CSE511105 CSE511106 CSE511107	Elective -I	1. Object Oriented Modeling 2. Software Quality Assurance and Testing 3. Software Architecture and Design Patterns	3	1	0	40	60	100
CSE511108 CSE511109 CSE511110	Elective -II	1. Software Design and Engineering 2. Advanced Compiler Design 3. Image Processing and Pattern Recognition	3	1	0	40	60	100
CSE511111	Lab	Java and Web Technologies Lab	0		3	40	60	100
CSE511112		Seminar	-	3	-	50	-	50
		Total Marks (6 Theory + 1 Lab + Seminar)						750

I YEAR II SEMESTER

Code	Group	Subject	L	T	P	Internal Val.	End Exam	Total
CSE521101		Distributed Computing	3	1	0	40	60	100
CSE521102		Distributed Databases	3	1	0	40	60	100
CSE521103		Advanced Computer Architecture	3	1	0	40	60	100
CSE521104		Advanced Computer Networks	3	1	0	40	60	100
CSE521105 CSE521106 CSE521107	Elective – III	Web Services Open Source Development Semantic Web and Social Networks	3	1	0	40	60	100
CSE521108 CSE521109 CSE521110	Elective - IV	Wireless Networks and Mobile Computing Information Security Storage Area Networks	3	1	0	40	60	100
CSE521111	Lab	Databases and Compiler Lab	0		3	40	60	100
CSE521112		Seminar	-	3	-	50	-	50
		Total Marks (6 Theory + 1 Lab + Seminar)						750

II YEAR I SEMESTER

Code	Group	Subject	L	T	P	Internal Val.	End Exam	Total
CSE611101		Comprehensive Viva	-	-	-	-	100	100
CSE611102		Project Seminar	0		6	-	-	-
CSE611102		Project Work	-	-	-	-	-	-
		Total Marks						100

II YEAR II SEMESTER

Code	Group	Subject	L	T	P	Internal Val.	End Exam	Total
CSE611102		Project Work and Seminar (contd)	-	-	-	Grade		
		Grade (A/B/C/D)				A. Excellent B. Good C. Satisfactory D. Unsatisfactory		

Note: All End Examinations (Theory and Practical) are of three hours duration.

L – Theory.
T – Tutorial.
P – Practical.
D – Drawing.
C – Credits

2011-2012

Code: CSE511101

MALLA REDDY ENGINEERING COLLEGE
(*Autonomous*)

I SEMESTER

L	T/P/D
3	0

ADVANCED PROBLEM SOLVING

Unit I

OOP Using Java - Class and Objects, Variables, Operators, Expressions, Methods, Decision statements, Loops, Arrays, OOP concepts- Encapsulation, Inheritance, Polymorphism, Abstraction, Modularity, Exception handling, Input and Output, Java and Pointers, Interfaces, Packages, Abstract classes, Casting in Inheritance hierarchy, Casting with Interfaces, Vectors in java.util, Data Structures and OOP, Writing a java program-Design, coding, testing and debugging. Basic concepts(Review)- Abstract Data Types, Data structures, Algorithms- Characteristics of Algorithms, Performance analysis- Time complexity and Space complexity, Asymptotic Analysis-Big O, Omega and Theta notations.

Unit II

Linear data structures- The List ADT, Array and Linked Implementations, Singly Linked Lists-Operations-Insertion,Deletion,Traversals,DoublyLinkedLists-Operations-Insertion,Deletion,SkipLists-implementation,StackADT,definitions,operations, Array and Linked implementations, applications-infix to postfix conversion, recursion implementation, tail recursion, nontail recursion, indirect recursion, Queue ADT, definitions and operations ,Array and Linked Implementations, Priority Queue ADT, Deque ADT, Implementation using doubly linked lists, Stacks and Queues in java.util.

Unit III

Non Linear data structures-Trees-Basic Terminology, Binary tree ADT, array and linked representations, iterative traversals, threaded binary trees, Applications-Disjoint-Sets, Union and Find algorithms, Huffman coding, General tree to binary tree conversion, Realizing a Priority Queue using Heap.

Search Trees- Binary Search Tree ADT, Implementation, Operations- Searching, Insertion and Deletion, Balanced Search trees-AVL Trees, Operations – Insertion and Searching, B-Trees, B-Tree of order m, Operations- Insertion, Deletion and Searching, Introduction to Red-Black Trees, Splay Trees, B*-Trees, B+-Trees(Elementary treatment), Comparison of Search Trees, Trees in java.util.

Unit IV

Searching- Linear Search, Binary Search, Hashing-Hash functions, Collision-Handling schemes, Hashing in java.util, Dictionary ADT, Linear list representation, Skip list representation, Hash table representation, Comparison of Searching methods.

Sorting- Bubble Sort, Insertion Sort, Shell sort, Heap Sort, Radix Sort, Quick sort, Merge sort, Comparison of Sorting methods, Sorting in java.util.

Unit V

Graphs-Basic Terminology, Graph Representations- Adjacency matrix, Adjacency lists, Adjacency multilists, Graph traversals- DFS and BFS, Spanning trees-Minimum cost

spanning trees, Kruskal's Algorithm for Minimum cost Spanning trees, Shortest paths- Single Source Shortest Path Problem, All Pairs Shortest Path Problem.

Text Processing - Pattern matching algorithms- The Knuth-Morris-Pratt algorithm, The Boyer-Moore algorithm, Tries- Standard Tries, Compressed Tries, Suffix tries.

TEXT BOOKS :

1. Data structures and Algorithms in Java,Adam Drozdek,Cengage Learning.
2. Data structures and Algorithms in Java,Michael T.Goodrich and R.Tomassia , Wiley India edition.
3. Data structures , Algorithms and Applications in Java,S.Sahani, Universities Press.

REFERENCE BOOKS :

1. Data structures and algorithms in Java,Robert Lafore,Pearson Education.
2. Data structures with Java,W.H.Ford and W.R.Topp, Pearson Education.
3. Classic Data structures in Java,T.Budd,Pearson Education.
4. Data Structures using Java,D.S. Malik and P.S.Nair, Cengage Learning,
5. An Introduction to Data structures and Algorithms, J.A.Storer, Springer.
6. Data structures and Java Collections Frame Work, W.J.Collins, Mc Graw Hill.
7. Data structures with Java,J.R.Hubbard and A.Huray,PHI.
8. Data Structures using Java, Y.Langsam, M.Augenstein, A.Tanenbaum, Pearson Education.
9. Data structures with Java,J.R.Hubbard,Schaum's Outlines, TMH.

2011-2012

Code: CSE511102

MALLA REDDY ENGINEERING COLLEGE
(Autonomous)

I SEMESTER

L T/P/D
3 0

COMPUTER SYSTEM DESIGN

UNIT I

Computer structure – hardware, software, system software, Von-neumann architecture – case study. IA -32 Pentium: registers and addressing, instructions, assembly language, program flow control, logic and shift/rotate instructions, multiply, divide MMX,SIMD instructions, I/O operations, subroutines.

Input/Output organization, interrupts, DMA, Buses, Interface circuits, I/O interfaces, device drivers in windows, interrupt handlers

UNIT II

Processing Unit: Execution of a complete instruction, multiple bus organization, hardwired control, micro programmed control.

Pipelining: data hazards, instruction hazards, influence on instruction sets, data path & control consideration, RISC architecture introduction.

UNIT – III

Memory: types and hierarchy, model level organization, cache memory, performance considerations, mapping, virtual memory, swapping, paging, segmentation, replacement policies.

Processes and Threads: processes, threads, inter process communication, classical IPC problems, Deadlocks.

UNIT –IV

File system: Files, directories, Implementation, Unix file system

Security: Threats, intruders, accident data loss, basics of cryptography, user authentication.

UNIT –V

Matrix Theory: Sparse Matrix, LVD Composition, algorithm, Spectral Factorization

TEXT BOOKS:

1. Computer Organization – Carl Hamacher, Zvonks Vranesic, SafeaZaky, Vth Edition, McGraw Hill.
2. Modern Operating Systems, Andrew S Tanenbaum 2nd edition Pearson/PHI
3. Higher Engineering Mathematics by B.S Grewel, Wiley Publications

REFERENCE BOOKS:

1. Computer Organization and Architecture – William Stallings Sixth Edition, pearson/PHI
2. Morris Mano -Computer System Architecture –3rd Edition-Pearson Education.
3. Operating System Principles- Abraham Silberchatz, Peter B. Galvin, Greg Gagne 7th Edition, John Wiley
4. Operating Systems – Internals and Design Principles Stallings, Fifth Edition–2005, Pearson Education/PHI

2011-2012

Code: CSE511103

MALLA REDDY ENGINEERING COLLEGE
(Autonomous)

I SEMESTER

L T/P/D
3 0

EMBEDDED SYSTEMS AND RTOS

UNIT I

Introduction to Embedded Systems: Embedded Systems, Processor Embedded into a System, Embedded Hardware Units and Devices in a System, Embedded Software, Complex System Design, Design Process in Embedded System, Formalization of System Design, Classification of Embedded Systems

UNIT II

8051 and Advanced Processor Architecture: 8051 Architecture, 8051 Micro controller Hardware, Input/Output Ports and Circuits, External Memory, Counter and Timers, Serial data Input/Output, Interrupts, Introduction to Advanced Architectures, Real World Interfacing, Processor and Memory organization - **Devices and Communication Buses for Devices Network:** Serial and parallel Devices & ports, Wireless Devices, Timer and Counting Devices, Watchdog Timer, Real Time Clock, Networked Embedded Systems, Internet Enabled Systems, Wireless and Mobile System protocols

UNIT III

Embedded Programming Concepts: Software programming in Assembly language and High Level Language, Data types, Structures, Modifiers, Loops and Pointers, Macros and Functions, object oriented Programming, Embedded Programming in C++ & JAVA

UNIT IV

Real – Time Operating Systems: OS Services, Process and Memory Management, Real – Time Operating Systems, Basic Design Using an RTOS, Task Scheduling Models, Interrupt Latency, Response of Task as Performance Metrics - **RTOS Programming:** Basic functions and Types of RTOS, RTOS VxWorks, Windows CE

UNIT V

Embedded Software Development Process and Tools: Introduction to Embedded Software Development Process and Tools, Host and Target Machines, Linking and Locating Software, Getting Embedded Software into the Target System, Issues in Hardware-Software Design and Co-Design - **Testing, Simulation and Debugging Techniques and Tools:** Testing on Host Machine, Simulators, Laboratory Tools

TEXT BOOKS:

1. Embedded Systems, Raj Kamal, Second Edition TMH.

REFERENCE BOOKS :

1. Embedded/Real-Time Systems, Dr.K.V.K.K.Prasad, dreamTech press
2. The 8051 Microcontroller and Embedded Systems, Muhammad Ali Mazidi, Pearson.

3. The 8051 Microcontroller, Third Edition, Kenneth J. Ayala, Thomson.
4. An Embedded Software Primer, David E. Simon, Pearson Education.
5. Micro Controllers, Ajay V Deshmukhi, TMH.
6. Microcontrollers, Raj kamal, Pearson Education.
7. Introduction to Embedded Systems, Shibu K.V, TMH.

2011-2012

Code: CSE511104

MALLA REDDY ENGINEERING COLLEGE
(Autonomous)

I SEMESTER

L T/P/D
3 0

JAVA AND WEB TECHNOLOGIES

UNIT I

HTML Common tags- List, Tables, images, forms, Frames; Cascading Style sheets;

Introduction to Java Scripts, Objects in Java Script, Dynamic HTML with Java Script, CSS

UNIT II

XML: Document type definition, XML Schemas, Document Object model, Presenting XML, Using XML Processors: DOM and SAX. Review of Applets, Class, Event Handling, AWT Programming.

Introduction to Swing: JApplet, Handling Swing Controls like Icons – Labels – Buttons – Text Boxes – Combo – Boxes – Tabbed Pains – Scroll Pains – Trees – Tables Differences between AWT Controls & Swing Controls Developing a Home page using Applet & Swing.

UNIT III

Java Beans: Introduction to Java Beans, Advantages of Java Beans, JDK Introspection, Using Bound properties, Bean Info Interface, Constrained properties Persistence, Customizes, Java Beans API. Web servers: Tomcat Server installation & Testing.

Introduction to Servlets: Lifecycle of a Servlet, JSDK The Servlet API, The javax.servelet Package, Reading Servlet parameters, Reading Initialization parameters.

UNIT IV

More on Servlets: The javax.servelet HTTP package, Handling Http Request & Responses, Using Cookies-Session Tracking, Security Issues.

Introduction to JSP: The Problem with Servlet. The Anatomy of a JSP Page, JSP Processing. JSP Application Design with MVC architecture. AJAX.

UNIT V

JSP Application Development: Generating Dynamic Content, Using Scripting Elements. Implicit JSP Objects, Conditional Processing – Displaying Values Using an Expression to Set an Attribute, Declaring Variables and Methods Error Handling and Debugging Sharing Data Between JSP pages, Requests, and Users Passing Control and Date between Pages – Sharing Session and Application Data – Memory Usage Considerations

Database Access Database Programming using JDBC Studying Javax.sql.* package Accessing a Database from a JSP Page Application – Specific Database Actions Deploying JAVA Beans in a JSP Page

TEXT BOOKS:

1. Web Programming, building internet applications, Chris Bates 2nd edition, WILEY Dreamtech (UNIT 1,2)

2. The complete Reference Java 2 Fifth Edition ,Patrick Naughton and Herbert Schildt., TMH (Chapters: 25) (UNIT 2,3)
3. Java Server Pages –Hans Bergsten, SPD O’Reilly (UNITs 3,4,5)

REFERENCE BOOKS:

1. Programming world wide web-Sebesta,Pearson
2. Core SERVLETS ANDJAVASERVER PAGES VOLUME 1: CORE TECHNOLOGIES , Marty Hall and Larry Brown Pearson
3. Internet and World Wide Web – How to program , Dietel and Nieto PHI/Pearson.
4. Jakarta Struts Cookbook , Bill Siggelkow, S P D O’Reilly for chap 8.
5. Murach’s beginning JAVA JDK 5, Murach, SPD
6. An Introduction to web Design and Programming –Wang-Thomson
7. Professional Java Server Programming,S.Allamaraju and othersApress(dreamtech).
8. Java Server Programming ,Ivan Bayross and others,The X Team,SPD
9. Web Warrior Guide to Web Programmimg-Bai/Ekedaw-Thomas
10. Beginning Web Programming-Jon Duckett WROX.
11. Java Server Pages, Pekowsky, Pearson.
12. Java Script,D.Flanagan,O’Reilly,SPD.

2011-2012

Code: CSE511105

MALLA REDDY ENGINEERING COLLEGE
(Autonomous)

I SEMESTER

L	T/P/D
3	0

OBJECT ORIENTED MODELING

ELECTIVE – I

UNIT I

Introduction to UML: The meaning of Object Orientation, object identity, Encapsulation, information hiding, polymorphism, generosity, importance of modeling, principles of modeling, object oriented modeling, conceptual model of the UML, Architecture.

Basic Structural Modeling: Classes, Relationships, common Mechanisms, and diagrams.

Class & Object Diagrams: Terms, concepts, modeling techniques for Class & Object Diagrams.

Collaboration Diagrams: Terms, Concepts, depicting a message, polymorphism in collaboration diagrams, iterated messages, use of self in messages.

Sequence Diagrams: Terms, concepts, depicting asynchronous messages with/without priority, callback mechanism, broadcast messages.

UNIT II

Basic Behavioral Modeling: Use cases, Use case Diagrams, Activity Diagrams.

Advanced Behavioral Modeling: Events and signals, state machines, processes and Threads, time and space, state chart diagrams.

Architectural Modeling: Component, Deployment, Component diagrams and Deployment diagrams.

UNIT III

The Unified process: use case driven, architecture centric, iterative, and incremental

The Four Ps: people, project, product, and process.

Use case driven process: why use case, capturing use cases, analysis, design, and implementation to realize the use cases, testing the use cases.

Architecture-centric process: architecture in brief, why we need architecture, use cases and architecture, the steps to architecture, an architecture description.

UNIT IV

Iterative incremental process: iterative incremental in brief, why iterative incremental development? The iterative approach is risk driven, the generic iteration.

The Generic Iteration workflow: phases are the first division workflow, planning proceeds doing, risks affect project planning, use case prioritization, resource needed, assess the iteration and phases

Inception phase: early in the inception phase, the archetypal inception iteration workflow, execute the core workflows, requirements to test.

UNIT V

Elaboration Phase: elaboration phase in brief, early in the elaboration phase, the architectural elaboration iteration workflow, execute the core workflows-Requirements to test.

Construction phase: early in the construction phase, the archetypal construction iteration workflow, execute the core workflow.

Transition phase: early in the transition phase, activities in transition phase

Case Studies: Automation of a Library, Software Simulator application (2-floor elevator simulator)

TEXT BOOKS:

- 1 The Unified Modeling Language User Guide By Grady Booch, James Rumbaugh, Ivar Jacobson 2nd Edition, Pearson Education.
2. UML 2 Toolkit By Hans-Erik Eriksson, Magnus Penker, Brian Lyons, David Fado WILEY-Dreamtech India Pvt. Ltd.
3. The Unified Software Development Process By Ivar Jacobson, Grady Booch, James Rumbaugh, Pearson Education

REFERENCE BOOKS:

1. Fundamentals of Object Oriented Design in UML By Meilir Page-Jones, Pearson Education
2. Object Oriented Analysis & Design By Atul Kahate, The McGraw-Hill.
3. Practical Object-Oriented Design with UML By Mark Priestley, TATA McGrawHill
4. Object Oriented Analysis & Design By Brett D McLaughlin, Gary Pollice and David West, O'REILY .
5. Object-Oriented Analysis and Design using UML By Simon Bennet, Steve McRobb and Ray Farmer, 2nd Edition, TATA McGrawHill.
6. Object-Oriented Analysis and Design with the Unified Process By John W. Satzinger, Robert B Jackson and Stephen D Burd, THOMSON Course Technology.
7. UML and C++,R.C.Lee, and W.M.Tepfenhart,PHI.

2011-2012

Code: CSE511106

MALLA REDDY ENGINEERING COLLEGE
(*Autonomous*)

I SEMESTER

L	T/P/D
3	0

SOFTWARE QUALITY ASSURANCE AND TESTING

ELECTIVE – I

UNIT I

Software Quality Assurance Framework and Standards SQA Framework: What is Quality? Software Quality Assurance, Components of Software Quality Assurance – **Software Quality Assurance Plan:** Steps to develop and implement a Software Quality Assurance Plan – **Quality Standards:** ISO 9000 and Companion ISO Standards, CMM, CMMI, PCMM, Malcom Balridge, 3 Sigma, 6 Sigma

UNIT II

Software Quality Assurance Metrics and Measurement Software Quality Metrics: Product Quality metrics, In-Process Quality Metrics, Metrics for Software Maintenance, Examples of Metric Programs – **Software Quality metrics methodology:** Establish quality requirements, Identify Software quality metrics, Implement the software quality metrics, analyze software metrics results, validate the software quality metrics – **Software quality indicators – Fundamentals in Measurement theory**

UNIT III

Software Testing Strategy and Environment: Establishing testing policy, structured approach to testing, test factors, Economics of System Development Life Cycle (SDLC) Testing

Software Testing Methodology : Defects hard to find, verification and validation, functional and structural testing, workbench concept, eight considerations in developing testing methodologies, testing tactics checklist

UNIT IV

Software Testing Techniques : Black-Box, Boundary value, Bottom-up, Branch coverage, Cause-Effect graphing, CRUD, Database, Exception, Gray-Box, Histograms, Inspections, JADs, Pareto Analysis, Prototyping, Random Testing, Risk-based Testing, Regression Testing, Structured Walkthroughs, Thread Testing, Performance Testing, White-Box Testing
Software Testing Tools : taxonomy of Testing tools, Methodology to evaluate automated testing tools, Load Runner, Win runner and Rational Testing Tools, Silk test, Java Testing Tools, JMetra, JUNIT and Cactus.

UNIT V

Testing Process:Eleven Step Testing Process: Assess Project Management Development Estimate and Status, Develop Test Plan, Requirements Phase Testing, Design Phase Testing,

Program Phase Testing, Execute Test and Record Results, Acceptance Test, Report test results, testing software installation, Test software changes, Evaluate Test Effectiveness.

Testing Specialized Systems and Applications: Testing Client/Server – Web applications, Testing off the Shelf Components, Testing Security, Testing a Data Warehouse

TEXT BOOKS:

1. Effective Methods for Software Testing, 2nd Edition, William E. Perry , Second Edition, Wiley India, 2006.
2. Software Quality, Mordechai Ben-Menachem/Garry S. Marliss, Thomson Learning publication, 1997.

REFERENCE BOOKS:

1. Testing and Quality Assurance for Component-based Software, by Gao, Tsao and Wu, Artech House Publishers
2. Software Testing Techniques, by Boris Beizer, Second Edition, Dreamtech Press
3. Managing the Testing Process, by Rex Black, Wiley
4. Handbook of Software Quality Assurance, by G. Gordon Schulmeyer, James I. McManus, Second Edition, International Thomson Computer Press
5. Software Testing and continuous Quality Improvement, by William E. Lewis, Gunasekaran Veerapillai, Second Edition, Auerbach Publications
6. Metrics and Models for Software Quality Engineering, by Stephen H. Kan, by Pearson Education Publication
7. Software Testing Tools, K.V.K.K. Prasad, Dream tech press, 2008.
8. Practical Software Testing, Ilene Burnstein, Springer, 2003.
9. Software Testing, Srinivasan Desikan & Gopalaswamy Ramesh, Pearson Education, 2006.
10. Software testing techniques, Scott Loveland & Geoffrey Miller, Shroff Publishers, 2005.
11. Software Quality, Martin Wieczorek & Dirk Meyerhoff, Springer, 2001.

2011-2012

Code: CSE511107

MALLA REDDY ENGINEERING COLLEGE
(*Autonomous*)

I SEMESTER

L	T/P/D
3	0

SOFTWARE ARCHITECTURE AND DESIGN PATTERNS
ELECTIVE-I

UNIT I

Envisioning Architecture: The Architecture Business Cycle, What is Software Architecture, Architectural patterns, reference models, reference architectures, architectural structures and views.

Creating Architecture: Quality Attributes, Achieving qualities, Architectural styles and patterns, designing the Architecture, Documenting software architectures, Reconstructing Software Architecture.

UNIT II

Analyzing Architectures: Architecture Evaluation, Architecture design decision making, ATAM, CBAM.

UNIT III

Moving from one system to many: Software Product Lines, Building systems from off the shelf components, Software architecture in future.

UNIT IV

Patterns: Pattern Description, Organizing catalogs, role in solving design problems , Selection and usage.

Creational and Structural patterns: Abstract factory, builder, factory method, prototype, singleton, adapter, bridge, composite, façade, flyweight, Proxy.

UNIT V

Behavioral patterns: Chain of responsibility, command, Interpreter, iterator, mediator, memento, observer, state, strategy, template method, visitor.

Case Studies :A-7E – A case study in utilizing architectural structures, The World Wide Web - a case study in interoperability, Air Traffic Control – a case study in designing for high availability, Celsius Tech – a case study in product line development

TEXT BOOKS:

1. Software Architecture in Practice, second edition, Len Bass, Paul Clements & Rick Kazman, Pearson Education, 2003.
2. Design Patterns, Erich Gamma, Pearson Education, 1995.

REFERENCE BOOKS:

1. Beyond Software architecture, Luke Hohmann, Addison wesley, 2003.
2. Software architecture, David M. Dikel, David Kane and James R. Wilson, Prentice Hall PTR, 2001
3. Pattern Oriented Software Architecture, F.Buschmann & others, John Wiley & Sons.

4. Head First Design patterns, Eric Freeman & Elisabeth Freeman, O'REILLY, 2007.
5. Design Patterns in Java, Steven John Metsker & William C. Wake, Pearson education, 2006
6. J2EE Patterns, Deepak Alur, John Crupi & Dan Malks, Pearson education, 2003.
7. Design Patterns in C#, Steven John metsker, Pearson education, 2004.

2011-2012

Code: CSE511108

MALLA REDDY ENGINEERING COLLEGE
(*Autonomous*)

I SEMESTER

L	T/P/D
3	0

SOFTWARE DESIGN AND ENGINEERING

ELECTIVE –II

UNIT I

Introduction to Software Engineering: The evolving role of software, Changing Nature of Software, legacy software, Software myths. **A Generic view of process:** Software engineering- A layered technology, a process framework, The Capability Maturity Model Integration (CMMI), Process patterns, process assessment, personal and team process models. **Process models:** The waterfall model, Incremental process models, Evolutionary process models, Specialized process models, The Unified process.

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

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

UNIT II

Role of Software Design

The nature of the design process, transferring design knowledge, constraints upon the design process and product, recording design decisions, designing with others, context for design, economic factors, assessing design qualities, quality attributes of the design product, assessing the design process.

Transferring Design Knowledge-Representing abstract ideas, design viewpoints, the architecture concept, design methods, design patterns, Design representations, rationale for design methods.

Design Processes and Strategies: The role of strategy in design methods, describing the design process – The D – Matrix, design by top-down decomposition, design by composition, organizational influences upon design.

UNIT III

Designing with objects and components: Designing with objects: design practices for object-oriented paradigm, Object- oriented frame works, Hierarchical object oriented design process and heuristics, the fusion method, the unified process.

Component – based design: The component concept, designing with components, designing components, COTS. **Performing User interface design-**The Golden rules, Interface analysis and design models, user and task analysis, analysis of display content and work environment, applying interface design steps, user interface design issues, design evaluation.

UNIT IV

Project Management and Metrics: Project Management: The management spectrum: people, product, process and project, W5HH principle, critical practices.

Metrics for Process and Projects: Process metrics, project metrics, size-oriented metrics, function-oriented metrics, Object-oriented and use-case metrics, metrics for software quality, integrating metrics with in the software process.

UNIT V

Project Scheduling and Risk Management: Project Scheduling: Basic concepts, project scheduling, defining a task set and task network, timeline charts, tracking the schedule, tracking the progress for an OO project, Earned value analysis.

Risk Management: Reactive Vs. Proactive risk strategies, software risks, risk identification, risk projection, risk refinement, risk mitigation and monitoring, the RMMM plan.

TEXT BOOKS :

1. Software design, David Budgen, second edition, Pearson education, 2003.
2. Software Engineering: A practitioner's Approach, Roger S Pressman, seventh edition. McGraw Hill International Edition, 2009.

REFERENCE BOOKS :

1. Applying domain – driven design and patterns, Jimmy Nilsson, Pearson education,2006
2. Software Engineering, Ian Sommerville, seventh edition, Pearson education,2004.
3. Software Project Management, Bob Hughes & Mike Cotterell, fourth edition, Tata McGraw Hill,2006
4. The art of Project management, Scott Berkun, O'Reilly, 2005.
5. Software Engineering Project Management, Richard H. Thayer & Edward Yourdon, second edition, Wiley India, 2004.
6. Software Engineering Foundations, Yingxu Wang, Auerbach Publications, 2008
7. Applied Software Project Management, Andrew Stellman & Jennifer Greene, O'Reilly, 2006.
8. Software Design, Eric Braude, John Wiley&Sons.

2011-2012

Code: CSE511109

MALLA REDDY ENGINEERING COLLEGE
(Autonomous)

I SEMESTER

L	T/P/D
3	0

ADVANCED COMPILER DESIGN

ELECTIVE-II

UNIT I: Overview of Compilation: Phases of Compilation – Lexical Analysis, Regular Grammar and regular expression for common programming language features, pass and Phases of translation, interpretation, bootstrapping, data structures in compilation – LEX lexical analyzer generator.

UNIT II: Parsing: Context free grammars, Top down parsing – Backtracking, LL (1), recursive descent parsing, Predictive parsing, Preprocessing steps required for predictive parsing.

Bottom up parsing: - Shift Reduce parsing, LR and LALR parsing, Error recovery in parsing, handling ambiguous grammar, YACC – automatic parser generator.

UNIT III: Semantic analysis: Intermediate forms of source Programs – abstract syntax tree, Attributed grammars, Syntax directed translation, Conversion of popular Programming languages language Constructs into Intermediate code forms, Type checker.

Symbol Tables: Symbol table format, organization for block structured languages, hashing, tree structures representation of scope information. Block structures and non block structure storage allocation: static, Runtime stack and heap storage allocation, storage allocation for arrays, strings and records.

UNIT IV: Code Generation- Processing the intermediate Code- Interpretation, Code generation, Simple code generation, code generation for basic blocks, BURS Code generation and dynamic programming, Register allocation by graph coloring, Evaluation of code generation techniques Preprocessing the intermediate code, post processing the target code, machine code generation.

Code optimization: Consideration for Optimization, Machine dependent and machine independent code optimization, Scope of Optimization, local optimization, loop optimization, frequency reduction, folding, DAG representation.

UNIT V: Data flow analysis: Dataflow Analysis, Intermediate representation for flow analysis, Various dataflow analyses, Transformations using dataflow analysis Speeding up dataflow analysis, Alias analysis.

Loop Optimizations – Dominators, Loop-invariant computations, Induction variables, Array bounds checks, Loop unrolling

TEXT BOOKS :

1. Principles of compiler design -A.V. Aho . J.D.Ullman; Pearson Education

2. Modern Compiler Design- Dick Grune, Henry E. Bal, Criel T. H. Jacobs, Wiley dreamtech.

REFERENCE BOOKS:

1. Advanced Compiler Design Implementation, S.S.Muchnick, Elsevier.
2. Compilers principles, techniques and tools A.V.Aho, Ravi Sethi& J.D. Ullman; Pearson ed.,
3. lex &yacc – John R. Levine, Tony Mason, Doug Brown, O'reilly
4. Modern Compiler Implementation in C- Andrew N. Appel, Cambridge University Press.
5. Engineering a Compiler-Cooper & Linda, Elsevier.
6. Compiler Construction, Loudon, Thomson..

2011-2012

Code: CSE511110

MALLA REDDY ENGINEERING COLLEGE
(Autonomous)

I SEMESTER

L	T/P/D
3	0

IMAGE PROCESSING AND PATTERN RECOGNITION

ELECTIVE –II

UNIT – I

Fundamental steps of image processing, components of an image processing system. The image model and image acquisition, sampling and quantization, relationship between pixels, distance functions, scanner.

UNIT – II

Statistical and spatial operations, Intensity functions transformations, histogram processing, smoothing & sharpening – spatial filters Frequency domain filters, homomorphic filtering, image filtering & restoration. Inverse and weiner filtering, FIR weiner filter, Filtering using image transforms, smoothing splines and interpolation.

Morphological and other area operations, basic morphological operations, opening and closing operations, dilation erosion, Hit or Miss transform, morphological algorithms, extension to grey scale images.

UNIT- III

Segmentation and Edge detection region operations, basic edge detection, second order detection, crack edge detection, gradient operators, compass and laplace operators, edge linking and boundary detection, thresholding, regionbased segmentation, segmentation by morphological watersheds.

Image compression: Types and requirements, statistical compression, spatial compression, contour coding, quantizing compression, image data compression-predictive technique, pixel coding, transfer coding theory, lossy and lossless predictive type coding, Digital Image Water marking.

UNIT –IV

Representation and Description

Chain codes, Ploygonal approximation, Signature Boundary Segments, Skeltons, Boundary Descriptors, Regional Descriptors, Relational Descriptors, Principal components for Description, Relational Descriptors

UNIT V

Pattern Recognition Fundamentals: Basic Concepts of pattern recognition, Fundamental problems in pattern recognition system, design concepts and methodologies, example of automatic pattern recognition systems, a simple automatic pattern recognition model

Pattern classification: Pattern classification by distance function: Measures of similarity, Clustering criteria, K-means algorithm, Pattern classification by likelihood function: Pattern classification as a Statistical decision problem, Bayes classifier for normal patterns.

TEXT BOOKS:

1. Digital Image Processing Third edition, Pearson Education, Rafael C. Gonzalez, Richard E. Woods
2. Pattern recognition Principles: Julius T. Tou, and Rafael C. Gonzalez, Addison-Wesley Publishing Company

REFERENCE BOOKS:

1. Image Processing, Analysis and Machine Vision, Second Edition, Milan Sonka, Vaclav Hlavac and Roger Boyle. Thomson learning.
2. Digital Image Processing – Williamk. Prattl –John wiley edition
3. Fundamentals of digital image processing – by A.K. Jain. PH
4. Pattern classification, Richard Duda, Hart and David strok John Weily publishers.
5. Digital Image Processing, S.Jayaraman, S.Esakkirajan, T.Veerakumar, TMH.
6. Pattern Recognition, R.Shinghal, Oxford University Press.

2011-2012

Code: CSE511111

MALLA REDDY ENGINEERING COLLEGE
(Autonomous)

I SEMESTER

L T/P/D
0 3

JAVA AND WEB PROGRAMMING LAB

List of Sample Problems/Experiments

** 1. Develop static pages (using Only HTML) of an online Book store. The pages should resemble:

www.amazon.com The website should consist the following pages.

Home page, Registration and user Login
User Profile Page, Books catalog
Shopping Cart, Payment By credit card
Order Conformation

**2. Validate the Registration, user login, user profile and payment by credit card pages using JavaScript.

**3. Create and save an XML document at the server, which contains 10 users information. Write a program, which takes User Id as an input and returns the user details by taking the user information from the XML document.

**4. Bean Assignments

- a. Create a Java Bean which gives the exchange value of INR (Indian Rupees) into equivalent American/Canadian/Australian Dollar value.
- b. Create a simple Bean with a label - which is the count of number of clicks. Than create a BeanInfo class such that only the "count" property is visible in the Property Window.
- c. Create two Beans-a) Key Pad .b) Display Pad .After that integrate the two Beans to make it work as a Calculator.
- d. Create two Beans Traffic Light(Implemented as a Label with only three background colours Red,Green,Yellow) and Automobile(Implemented as a TextBox which states its state/movement). The state of the Automobile should depend on the following Light Transition Table.

Light Transition Automobile State

Red -> Yellow	Ready
Yellow -> Green	Move
Green -> Red	Stopped

**5. Install TOMCAT web server. Convert the static web pages of assignments 2 into dynamic web pages using Servlets and cookies. Hint: Users information (user id, password, credit card number) would be stored in web.xml Each user should have a separate Shopping Cart.

****6.** Redo the previous task using JSP by converting the static web pages of assignments 2 into dynamic web pages. Create a database with user information and books information. The books catalogue should be dynamically loaded from the database. Follow the MVC architecture while doing the website.

***7.** Implement the "Hello World!" program using JSP Struts Framework.

NOTE: * - Simple Problems.
 ** - Moderate Problems.
 *** - Complex Problems.

TEXT BOOKS:

1. Java Server Programming for Professionals, 2nd Edition, Bayross and others, O'reilly,SPD, 2007.
2. JOBC, Servlets, and JSP ,Black Book, K. Santosh Kumar, dreamtech.
3. Core Web Programming, 2nd Edition, Volume I, M.Hall and L.Brown, PHPTR.
4. Core Web Programming, 2nd Edition, Volume 2, M.Hall and L.Brown, PHPTR.
5. Core Java, Volume 1, Horstman and Cornell, 8th Edition, Pearson Education, 2008.
6. Core Java, Volume 2, Horstman and Cornell, 8th Edition, Pearson Education, 2008.
7. Java Programming: Advanced Topics, 3rd Edition, J. Wiggles worth and PMcMillan, Thomson, 2007.

2011-2012

Code: CSE521101

MALLA REDDY ENGINEERING COLLEGE
(*Autonomous*)

II SEMESTER

L	T/P/D
3	0

DISTRIBUTED COMPUTING

UNIT I

Introduction

The different forms of computing - Monolithic, Distributed, Parallel and cooperative computing, the meaning of . Distributed computing, Examples of Distributed systems, the strengths and weaknesses of Distributed computing &, operating system concepts relevant to distributed computing, the architecture of distributed applications.

UNIT II

Distributed Computing paradigms:

Paradigm for distributed Applications-Message passing paradigm the client paradigm(Java socket API), The peer-to-peer Paradigm, **Message system (or MOM) Paradigm** - the point-to-point message model and the publish/subscribe message model, RPC model, **The Distributed Objects Paradigms** - RMI, ORB, the object space Paradigm, The Mobile Agent Paradigm, the Network Services Paradigm, The collaborative application (Group ware Paradigm) ,choosing a Paradigm for an application.

UNIT III

Distributed Objects Paradigm (RMI)

Message passing versus Distributed Objects, An Archetypal Distributed Object Architecture, Distributed Object Systems, RPC, RMI, The Java RMI Architecture, Java RMI API, A sample RMI Application, steps for building an RMI application, testing and debugging, comparison of RMI and socket API

Distributed Object Paradigm (CORBA)

The basic Architecture, The CORBA object interface, Inter-ORB protocols, object servers and object clients, CORBA object references, CORBA Naming Service and the Interoperable Naming Service, CORBA object client, object Adapters, Java IDL, An example CORBA application.

UNIT IV

Distributed Document-based Systems

WWW, Lotus Notes, comparison of WWW and Lotus Notes, Distributed Coordination-based systems Introduction to coordination models, Till, JINI, comparison of Till and JINI

Software Agents, Agent Technology, Mobile Agents.

Distributed Multimedia Systems – characteristics of multimedia data, QOS of service management, Resource Management, Stream Adaptation

UNIT V

Grid Computing

Definition of grid, grid types - computational grid, data grid, grid benefits and applications, drawbacks of grid computing, grid components, grid architecture and its relation to various

Distributed Technologies.

Cluster Computing

Parallel computing overview, cluster computing - Introduction, Cluster Architecture, parallel programming models and Paradigms, Applications of Clusters.

TEXT BOOKS:

1. Distributed Computing, Principles and Applications, M.L.Liu, Pearson Education.
2. Distributed Systems, Principles and Paradigms, A.S.Tanenbaum and M.V.Steen, Pearson Education.
3. Client/Server Programming with Java and CORBA, second edition, R.Orfali & Dan.Harkey, John Wiley & Sons
4. Grid Computing, J.Joseph & C.Fellenstein, Pearson education;
5. High Performance Cluster Computing, Rajkumar Buyya, Pearson education.

REFERENCE BOOKS:

1. A Networking Approach to Grid Computing, D.Minoli, Wiley & sons.
2. Grid Computing: A Practical Guide to Technology and Applications, A.A.bbbs, Firewall Media.'
3. Java Network Programming, E.R.Harold, 2nd edition, O'Reilly, SPD,
4. Distributed Systems, Concepts and Design, 3rd edition, GCoulouris, J .Dollimore and Tim Kindbirg, Peall Education. '
5. Java Programming with CORBA, 3rd edition, Brose, Vogel, Duddy, Wiley Dreamtech

2011-2012

Code: CSE521102

MALLA REDDY ENGINEERING COLLEGE
(Autonomous)

II SEMESTER

L T/P/D
3 0

DISTRIBUTED DATABASES

UNIT I

Features of Distributed versus Centralized Databases, Principles of Distributed Databases, Levels Of Distribution Transparency, Reference Architecture for Distributed Databases, Types of Data Fragmentation, Integrity Constraints in Distributed Databases, Distributed Database Design

UNIT II

Translation of Global Queries to Fragment Queries, Equivalence transformations for Queries. Transforming Global Queries into Fragment Queries, Distributed Grouping and Aggregate Function Evaluation, Parametric Queries.
Optimization of Access Strategies, A Framework for Query Optimization, Join Queries, General Queries

UNIT III

The Management of Distributed Transactions, A Framework for Transaction Management, Supporting Atomicity of Distributed Transactions, Concurrency Control for Distributed Transactions, Architectural Aspects of Distributed Transactions.
Concurrency Control, foundation of Distributed Concurrency Control, Distributed Deadlocks, Concurrency Control based on Timestamps, Optimistic Methods for Distributed Concurrency Control.

UNIT IV

Reliability, Basic Concepts, Nonblocking Commitment Protocols, Reliability and concurrency Control. Determining a consistent View of the Network, Detection and Resolution of Inconsistency, Checkpoints and Cold Restart, Distributed Database Administration, Catalog Management in Distributed Databases, Authorization and Protection

UNIT V

Architectural Issues, Alternative Client/Server Architectures, Cache Consistency, Object Management, Object Identifier Management, Pointer Sizzling, Object Migration, Distributed Object Storage, Object Query Processing, Object Query Processor Architectures, Query Processing Issues, Query Execution, Transaction Management, Transaction Management in Object DBMSs, Transactions as Objects
Database Integration, Scheme Translation, Scheme Integration, Query Processing Query Processing Layers in Distributed Multi-DBMSs, Query Optimization Issues Transaction Management Transaction and Computation Model, Multidatabase Concurrency Control, Multidatabase Recovery, Object Orientation and Interoperability, Object Management Architecture CORBA and Database interoperability, Distributed Component Object Model. COM/OLE and Database Interoperability, PUSH-Based Technologies

TEXT BOOKS:

1. Distributed Databases Principles & Systems, Stefano Ceri, Giuseppe Pelagatti, TMH.
2. Principles of Dist. Database Systems, M. Tamer Ozsu, Patrick Valduriez, Pearson Edn..

2011-2012

Code: CSE521103

MALLA REDDY ENGINEERING COLLEGE
(Autonomous)

II SEMESTER

L	T/P/D
3	0

ADVANCED COMPUTER ARCHITECTURE

UNIT I

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 compute 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 line for RISC processor, Basic performance issues in pipelining , Pipeline hazards, Reducing pipe line 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.

ILP software approach- Basic compiler level techniques, static branch prediction, VLIW approach, Exploiting ILP, Parallelism at compile time, Cross cutting issues -Hardware verses Software.

UNIT IV

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 interconnecting networks, Examples of inter connection, Cluster, Designing of clusters.

FSIntel Architecture: inlet IA- 64 ILP in embedded and mobile markets Fallacies and pit falls

TEXT BOOKS:

1. John L Hennessy, David A. Patterson, Computer Architecture: A Quantitative Approach, 3rd Edition, An Imprint of Elsevier.

REFERENCE BOOKS:

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 cd

2011-2012

Code: CSE521104

MALLA REDDY ENGINEERING COLLEGE
(*Autonomous*)

II SEMESTER

L	T/P/D
3	0

ADVANCED COMPUTER NETWORKS

UNIT I Review

Computer Networks and the Internet: What is the Internet, The Network edge, The Network core, Access Networks and Physical media, ISPs and Internet Backbones, Delay and Loss in Packet-Switched Networks, History of Computer Networking and the Internet

Foundation of Networking Protocols: 5-layer TCP/IP Model, 7-Layer OSI Model. Internet Protocols and Addressing, Equal-Sized Packets Model: ATM

Networking Devices: Multiplexers, Modems and Internet Access Devices, Switching and Routing Devices, Router Structure.

UNIT II

The Link Layer and Local Area Networks: Link Layer: Introduction and Services, Error-Detection and Error-Correction techniques, Multiple Access Protocols, Link Layer Addressing. Ethernet, Interconnections: Hubs and Switches, PPP: The Point-to-Point Protocol. Link Virtualization

Routing and Internet working: Network-, Layer Routing, Least-Cost-Path algorithms, Non-Least-Cost-Path algorithms, Inter domain Routing Protocols, Inter domain Routing Protocols. Congestion Control at Network Layer

UNIT III

Logical Addressing: IPv4 Addresses, IPv6 Addresses - Internet Protocol: Internetworking, IPv4, IPv6, Transition from IPv4 to IPv6 -- Multicasting Techniques and Protocols: Basic Definitions and Techniques, Inter domain Multicast Protocols, Inter domain Multicast Protocols, Node-Level Multicast algorithms - Transport and End-to-End Protocols: Transport Layer, Transmission Control Protocol (TCP). User Datagram Protocol (UDP), Mobile Transport Protocols. TCP Congestion Control – Application Layer: Principles of Network Applications, The Web and HTTP, File Transfer: FTP, Electronic Mail in the Internet. Domain Name System (DNS), P2P File Sharing, Socket Programming with TCP and UDP, Building a Simple Web Server

UNIT IV

Wireless Networks and Mobile IP: Infrastructure of Wireless Networks, Wireless LAN Technologies, IEEE 802.11 Wireless Standard, Cellular Networks, Mobile IP, Wireless Mesh Networks (WMNs) - Optical Networks and WDM Systems: Overview of Optical Networks, Basic Optical Networking Devices, Large-Scale Optical Switches. Optical Routers, Wavelength Allocation in Networks. Case Study: All-Optical Switch

UNIT V

VPNs, Tunneling and Overlay Networks: Virtual Private Networks (VPNs), Multiprotocol Label Switching (MPLS). Overlay Networks – **VoIP and Multimedia Networking:** Overview of IP Telephony, VoIP Signaling Protocols, Real-Time Media Transport Protocols,

Distributed Multimedia Networking, Stream Control Transmission Protocol - Mobile Ad-Hoc Networks: Overview of Wireless Ad-Hoc Networks, Routing in Ad-Hoc Networks, Routing Protocols for Ad-Hoc Networks - Wireless Sensor Networks: Sensor Networks and Protocol Structures, Communication Energy Model, Clustering Protocols, Routing Protocols.

TEXT BOOKS:

1. Computer Networking: A Top-Down Approach Featuring the Internet, *James F Kurose, Keith W.Ross*. Third Edition, Pearson Education. 2007
2. Computer and Communication Networks, *Noda F Mir*. Pearson Education, 2007

REFERENCE BOOKS:

1. Data Communications and Networking, *Behrouz,A. Forouzan*, Fourth Edition, Tata McGraw Hill, 2007
 2. Guide to Networking Essentials, *Greg Tomsho ,Ed Tittel, David Johnson*, Fifth Edition, Thomson.
 3. An Engineering Approach to Computer Networking, *S.Keshav*, Pearson Education.
 4. Campus Network Design Fundamentals, *Diane Teare, Catherine Paquet*, Pearson Education (CISCO Press)
 5. Computer Networks, *Andrew S. Tanenbaum*, Fourth Edition, Prentice Hall.
 6. The Internet and Its Protocols, *A.Farrel*, Elsevier
- .

2011-2012

Code: CSE521105

MALLA REDDY ENGINEERING COLLEGE
(*Autonomous*)

II SEMESTER

L	T/P/D
3	0

WEB SERVICES

ELECTIVE III

UNIT I

Evolution and Emergence of Web Services - Evolution of distributed computing, Core distributed computing technologies.. :client/server, CORBA, JAVA RMI, Micro Soft DCOM, MOM, Challenges in Distributed Computing, role of J2EE and XML in distributed computing, emergence of Web Services and Service Oriented Architecture (SOA).

UNIT II

Introduction to Web Services - The definition of web services, basic operational model of web services, tools and technologies enabling web services, benefits and challenges of using web services - Web Services Architecture - Web services Architecture and its characteristics, core building blocks of web services, standards and technologies available for implementing web services, web services communication, basic steps of implementing web services, developing web services enabled applications.

UNIT III

Core fundamentals of SOAP- SOAP Message Structure, SOAP encoding, SOAP message exchange models, SOAP communication and messaging, SOAP security - **Developing Web Services using SOAP** - Building SOAP Web Services, developing SOAP Web Services using Java, limitations of SOAP.

UNIT IV

Describing Web Services - WSDL - WSDL in the world of Web Services, Web Services life cycle, anatomy of WSDL definition document, WSDL bindings, WSDL Tools, limitations of WSDL - **Discovering Web Services** -- Service discovery, role of service discovery in a SOA, service discovery mechanisms, UDDI - ODDI Registries, uses of UDDI Registry, Programming with UDDI, UDDI data structures, support for categorization in UDDI Registries, Publishing API, Publishing information to a UDDI Registry, searching information in a UDDI Registry, deleting information in a UDDI Registry, limitations of UDDI.

UNIT V

Web Services Interoperability - Means of ensuring Interoperability, Overview of .NET and J2EE. Web Services Security - XML security frame work, XML encryption, XML digital signature, XKMS structure, guidelines for signing XML documents.

TEXT BOOKS:

1. Developing Java Web Services, R. Nagappan, R. Skoczylas, R.P. Sriganesh, Wiley India, 2008.
2. Developing Enterprise Web Services, S. Chatterjee, J. Webber, Pearson Education, 2008.
3. XML, Web Services, and the Data Revolution, FP.Coyle, Pearson Education.

REFERENCE BOOKS:

1. Building Web Services with Java, 2nd Edition, S. Graham and others, Pearson End., 2008.
2. Java Web Services, D.A. Chappell & T. Jewell, O'Reilly, SPD.
3. McGovern, et al.: "Java Web Services Architecture", Morgan Kaufmann Publishers, 2005.
4. NEE Web Services, Richard Monson-Haefel, Pearson Education.
5. Web Services, G. Alonso, F Casati and others, Springer, 2005.

2011-2012

Code: CSE521106

MALLA REDDY ENGINEERING COLLEGE
(*Autonomous*)

II SEMESTER

L	T/P/D	C
3	0	3

OPEN SOURCE DEVELOPMENT

ELECTIVE III

UNIT I

Introduction to Linux: Who's Using Linux?, System features, About Linux Copy right, Open Source and the Philosophy of Linux, Sources of Linux Information, The K desktop Environment, KDE Applications, The GNOME Desktop Environment, GNOME Applications, Virtual Consoles, Popular Commands, Shells, Useful Keys and How to get them to work, Filename Expansion, Remote logins and Command Executions, Manual Pages, Startup Files, Important Directories, Advanced Shells and Shell Scripting.

UNIT II

Introduction to open source development : A brief history about open source development, The evolution of the open source movement, FLOSS - Free, libre, open source software, Advantages and disadvantages of open source, Pros, Cons, Open source trends and perspectives

Open source business models, dual licensing, Split open source software / commercial products, Product specialists, Platform providers, Business model relationship to license, Open source business model and proprietary software

UNIT III

Licensing: Intellectual property, copyright and licensing, Open source licensing ,History of open source licensing, Commonly used open source licenses, Choosing the right license,Community driven development ,Getting started with open source development, Developers' group: Software design and , Builders' group: Software building , Testers' group: Software Testing, Release management group: Packaging, Release management group: Releasing , Installation and issue tracking, Installation, Issue tracking

UNIT IV

Participating in open source development, Open source communities, Effective communication, Communication etiquette and guidelines, starting your own open source project, providing the ecosystem for your open source, Accepting contributions.

UNIT V

Starting your own open source project, Providing the ecosystem for your open source project, Accepting contributions, Case Study: Contributing to an open source project, Ruby on Rails and the DB2 module , The ruby forge , Submitting a bug.

TEXT BOOKS:

1. **Running Linux, 5th Edition** A Distribution-Neutral Guide for Servers and Desktops
By Matthias Kalle Dalheimer, Matt Welsh Publisher: O'Reilly Media Released:
December 2005
2. **Getting Started with Open source development**, Ist Edition By Rachna Kapur, Mario
Briggs, Tapas Saha, Ulisses Costa, Pedro Carvalho, Raul F. Chong, Peter Kohlmann.
Publisher : IBM Corporation (July 2010)

REFERENCES :

1. Linux in a Nutshell, 6th Edition, By Ellen Siever, Stephen Figgins, Robert Love, Arnold
Robbins Publisher: O'Reilly Media Released: September 2009
2. Understanding the Linux Kernel, 3rd Edition, By Daniel P. Bovet, Marco Cesati
Publisher: O'Reilly Media Released: November 2005
3. Linux Device Drivers, 3rd Edition, By Jonathan Corbet, Alessandro Rubini, Greg Kroah-
Hartman Publisher: O'Reilly Media Released: February 2005

MALLA REDDY ENGINEERING COLLEGE
(Autonomous)

II SEMESTER

L	T/P/D
3	0

SEMANTIC WEB AND SOCIAL NETWORKS

ELECTIVE- III

UNIT -I: Web Intelligence

Thinking and Intelligent Web Applications, The Information Age ,The World Wide Web, Limitations of To days Web, The Next Generation Web, Machine Intelligence, Artificial Intelligence, Ontology, Inference Engines, Software Agents, Berners-Lee www, Semantic Road Map, Logic on the semantic Web.

UNIT -II: Knowledge Representation for the Semantic Web

Ontologies and their role in the semantic web, Ontologies Languages for the Semantic Web - Resource Description Framework(RDF) / RDF Schema, Ontology Web Language(OWL),UML,XML/XML Schema.

UNIT-III: Ontology Engineering

Ontology Engineering, constructing Ontology, Ontology Development Tools, Ontology Methods, Ontology Sharing and Merging, Ontology Libraries and Ontology mapping,Logic,Rule and Inference Engines.

UNIT-IV: Semantic Web Applications, Services and Technology

Semantic Web applications and services, Semantic Search,e-learning,Semantic Bioinformatics,Knowledge Base .XML Based Web Services, Creating an OWL-S Ontology for Web Services, Semantic Search Technology, Web Search Agents and Semantic Methods,

UNIT- V: Social Network Analysis and semantic web

What is social Networks analysis ?, development of the social networks analysis, Electronic Sources for Network Analysis - Electronic Discussion networks, Blogs and Online Communities, Web Based Networks, Building Semantic Web Applications with social network features.

TEXT BOOKS:

1. Thinking on the Web - Berners Lee, Gödel and Turing, Wiley interscience, 2008.
2. Social Networks and the Semantic Web, Peter Mika, Springer, 2007.

REFERENCE BOOKS:

1. Semantic Web Technologies, Trends and Research in Ontology Based Systems, J.Davies, R.Studer, P.Warren, *Johri*. Wiley & Sons.
2. Semantic Web and Semantic Web Services -Liyang Lu Chapman and Hall/CRC Publishers,(Taylor & Francis Group)
3. Information Sharing on the semantic Web - Heiner Stuckenschmidt; Frank Van Harmelen, Springer Publications.
4. Programming the Semantic Web, T.Segaran, C.Evans, J.Taylor, O'Reilly, SPD.

2011-2012

Code: CSE521108

MALLA REDDY ENGINEERING COLLEGE
(Autonomous)

II SEMESTER

L T/P/D
3 0

WIRELESS NETWORKS AND MOBILE COMPUTING
ELECTIVE – IV

UNIT I : Introduction to Mobile And Wireless Landscape

Definition of Mobile and Wireless, Components of Wireless Environment, Challenges
Overview of Wireless Networks, Categories of Wireless Networks
Wireless LAN: Infrared Vs radio transmission, Infrastructure and Ad-hoc Network, IEEE
802.11, HIPERLAN, Bluetooth

Global System for Mobile Communications (GSM)

GSM Architecture, GSM Entities, Call Routing in GSM, PLMN Interfaces, GSM Address
and -Identifiers, Network Aspects in GSM, GSM Frequency Allocation, Authentication and
Security

UNIT II: Mobile Network Layer

Mobile IP (Goals, assumptions, entities and terminology, IP packet delivery, agent
advertisement and discovery, registration, tunneling and encapsulation, optimizations),
Dynamic Host Configuration Protocol (DHCP), Mobile Ad-hoc networks: Routing,
destination Sequence Distance Vector, Dynamic Source Routing.

Mobile Transport Layer

Traditional TCP, Indirect TCP, Snooping TCP, Mobile TCP, Fast retransmit/fast recovery,
Transmission /time freezing, Selective retransmission, Transaction oriented TCP.

UNIT III: Broadcast Systems

Overview, Cyclical repetition of data, Digital audio broadcasting: Multimedia object transfer
protocol, Digital video broadcasting: DVB data broadcasting, DVB for high-speed internet
access, Convergence of broadcasting and mobile communications.

UNIT IV : Protocols and Tools:

Wireless Application Protocol- WAP. (Introduction, protocol architecture, and treatment of
protocols of all layers), Bluetooth (User scenarios, physical layer, MAC layer, networking,
security, link management) and J2ME.

Wireless Language and Content - Generation Technologies

Wireless Content Types, Markup Languages: HDML, WML, HTML, CHTML, XHTML,
Voice XML.

Content- Generation Technologies: CGI with Perl, Java Servlet, Java Server Pages, Active
Server Pages, XML with XSL Style sheets, XML Document, XSL Style sheet

UNIT V: Mobile and Wireless Security

Creating a Secure Environment, Security Threats, Security Technologies, Other Security
Measures, WAP Security Smart Client Security

TEXT BOOKS:

1. Jochen Schiller, "Mobile Communications", Pearson Education, Second Edition, 2008.
2. Martyn Mallick, "Mobile and Wireless Design Essentials", Wiley, 2008.
3. Asoke K Talukder, et ai, "Mobile Computing", Tata McGraw Hill, 2008.

REFERENCE BOOKS:

1. Mobile Computing, Raj Kamal,Oxford University Press.
2. William Stallings,"Wireless Communications & Networks", Person, Second Edition, 2007.
3. Frank Adelstein et ai, "Fundamentals of Mobile and Pervasive Computing",TMH,2005.
4. Jim Geier, "Wireless Networks first-step", Pearson, 2005.
5. Sumit Kasera et ai, "2.5G Mobile Networks: GPRS and EDGE", TMH, 2008.
6. Matthew S.Gast, "802.11 Wireless Networks", O'Reilly, Second Edition, 2006.
7. Ivim Stojmenovic , "Handbook of Wireless Networks and Mobile Computing", Wiley, 2007.

2011-2012

Code: CSE521109

MALLA REDDY ENGINEERING COLLEGE
(Autonomous)

II SEMESTER

L	T/P/D
3	0

INFORMATION SECURITY

ELECTIVE-IV

UNIT I

Security Goals, 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 & Algorithms (DES, AES, RC4), Block Cipher Modes of Operation, Location of Encryption Devices, Key Distribution, Public key cryptography principles, public key cryptography algorithms (RSA, RABIN, ELGAMAL, Diffie-Hellman, ECC), Key Distribution

UNIT III

Approaches of Message Authentication, Secure Hash Functions (SHA-512, WHIRLPOOL) and HMAC Digital Signatures: Comparison, Process- Need for Keys, Signing the Digest, Services, Attacks on Digital Signatures, Kerberos, X.509 Directory Authentication Service

UNIT IV

Email Security: Pretty Good Privacy (PGP) and SIMIME.
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, Virus Countermeasures Firewall Design principles, Trusted Systems, Intrusion Detection Systems

TEXT BOOKS:

1. Networks~ Security Essentials (Applications and Standards) by William Stallings Pearson Education, 2008.
2. Cryptography & Network Security by Behrouz A. Forouzan, TMH 2007.

REFERENCE BOOKS:

1. Information Security by Mark Stamp, Wiley - India, 2006.
2. Information Systems Security, Godbole, Wiley Student Edition.
3. Cryptography and Network Security by William Stallings, Fourth Edition, Pearson Education 2007.
4. Fundamentals of Computer Security, Springer.
5. Network Security: The complete reference, Robert Bragg, Mark Rhodes, TMH
6. Computer Security Basics by Rick Lehtinen, Deborah Russell & G.T.Gangemi Sr., SPD O'REILLY 2006.
7. Modern Cryptography by Wenbo Mao, Pearson Education 2007.
8. Principles of Information Security, Whitman, Thomson.

2011-2012

Code: CSE521110

MALLA REDDY ENGINEERING COLLEGE
(Autonomous)

II SEMESTER

L	T/P/D
3	0

STORAGE AREA NETWORKS

ELECTIVE-IV

UNIT I: Introduction to Storage Technology

Review data creation and the amount of data being created and understand the value of data to a business, challenges in data storage and data management, Solutions available for data storage, Core elements of a data center infrastructure, role of each element in supporting business activities.

UNIT II: Storage Systems Architecture

Hardware and software component of the host environment, Key protocols and concepts used by each component, Physical and logical components of a connectivity environment, Major physical components of a disk drive and their function, logical constructs of a physical disk, access characteristics, and performance Implications, Concept of RAID and its components, Different RAID levels and their suitability for different application environments: RAID 0, RAID 1, RAID 3, RAID 4, RAID 5. RAID 0+ 1, RAID 1 +0, RAID 6, Compare and contrast integrated and modular storage systems, High-level architecture and working of an intelligent storage system

UNIT III: Introduction to Networked Storage

Evolution of networked storage. Architecture. components, and topologies of FC-SAN, NAS, and IP-SAN, Benefits of the different networked storage options, Understand the need for long-term archiving solutions and describe how CAS fulfills the need, Understand the appropriateness of the different networked storage options for different application environments

UNIT IV: Information Availability & Monitoring & Managing Datacenter

List reasons for planned/unplanned outages and the impact of downtime, Impact of downtime, Differentiate between business continuity (BC) and disaster recovery (DR), RTO and RPO, Identify single points of failure in a storage infrastructure and list solutions to mitigate these failures, Architecture of backup /recovery and the different backup/recovery topologies, replication technologies and their role in ensuring information availability and business continuity, Remote replication technologies and their role in providing disaster recovery and business continuity capabilities

Identify key areas to monitor in a data center, Industry standards for data center monitoring and management, Key metrics to monitor for different components in a storage infrastructure, Key management tasks in a data center

UNIT V: Securing Storage and Storage Virtualization

Information security, Critical security attributes for information systems, Storage security

domains, List and analyzes the common threats in each domain, Virtualization technologies, block-level and file-level virtualization technologies and processes

Case Studies:

The technologies described in the course are reinforced with EMC examples of actual solutions.

Realistic case studies enable the participant to design the most appropriate solution for given sets of criteria.

TEXT BOOKS:

1. EMC Corporation, Information Storage and Management, Wiley.
2. Robert Spalding. "Storage Networks: The Complete Reference", Tata McGraw Hill. Osborne, 2003.
3. Marc Farley, "Building Storage Networks", Tata McGraw Hill, Osborne, 2001.
4. Meeta Gupta, Storage Area Network Fundamentals. Pearson Education Limited, 2002.

2011-2012

Code: CSE521111

MALLA REDDY ENGINEERING COLLEGE
(Autonomous)

II SEMESTER

L T/P/D
3 0

DATABASES AND COMPILER LAB

DATABASES

Objective: This lab enables the students to practice the concepts learnt in the subject DBMS by developing a database for an example company named "Roadway Travels" whose description is as follows. The student is expected to practice the designing, developing and querying a database in the context of example database "Roadway travel". Students are expected to use "Mysql" database.

Roadway Travels

"Roadway Travels" is in business since 1997 with several buses connecting different places in India. Its main office is located in Hyderabad.

The company wants to computerize its operations in the following areas:

- ? Reservations
- ? Ticketing
- ? Cancellations

Reservations:

Reservations are directly handled by booking office. Reservations can be made 60 days in advance in either cash or credit. In case the ticket is not available, a wait listed ticket is issued to the customer. This ticket is confirmed against the cancellation.

Cancellation and Modifications:

Cancellations are also directly handed at the booking office. Cancellation charges will be charged. *Wait listed tickets that do not get confirmed are fully refunded.*

Week1: E-R Model

Analyze the problem carefully and come up with the entities in it. Identify what data has to be persisted in the database. This contains the entities, attributes etc.

Identify the primary keys for all the entities. Identify the other keys like candidate keys, partial keys, if any.

Example: **Entities:**

1. BUS
2. Ticket
3. Passenger

PRIMARY KEY ATTRIBUTES:

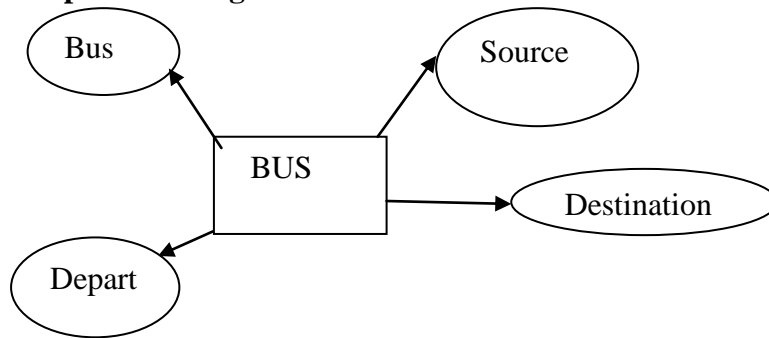
1. Ticket ID (Ticket Entity)
2. Passport ID (Passenger Entity)

Apart from the above mentioned entities you can identify more. The above mentioned are few.

Week2: Concept design with E-R Model

Relate the entities appropriately. Apply cardinalities for each relationship. Identify strong entities and weak entities (if any). Indicate the type of relationships (total/partial). Try to incorporate generalization, aggregation, specialization etc wherever required.

Example: E-R diagram for bus



Week3: Relational Model

Represent all the entities (Strong, Weak) in tabular fashion. Represent relationships in a tabular fashion. There are different ways of representing relationships as tables based on the cardinality. Represent attributes as columns in tables or as tables based on the requirement. Different types of attributes (Composite, Multivalued.. and Derived) have different way of representation.

Example: The passenger tables look as below. This is an example. *You* can add more attributes based on your E-R model.

Passenger

Name	Age	Sex	Address	Passport ID
				!

Week4: Normalization

Database normalization is a technique for designing relational database tables to minimize duplication of information and, in so doing, to safeguard the database against certain types of logical or structural problems, namely data anomalies. For example, when multiple instances of a given piece of information occur in a table, the possibility exists that these instances will not be kept consistent when the data within the table is updated, leading to a loss of data integrity. A table that is sufficiently normalized is less vulnerable to problems of this kind because its structure reflects the basic assumptions for when multiple instances of the same information should be represented by a single instance only.

Week5: Installation of Mysql and practicing DDL commands

Installation of MySQL. In this week you will learn Creating databases, How to create tables, altering the database, dropping tables and databases If not required. You will also try truncating, renaming commands etc.

Example for creation of a table.

```
CREATE TABLE Passenger (Passport id INTEGER PRIMARY KEY, Name CHAR  
(50) NULL, Age Integer, Sex Char);
```

Note: Detailed creation of tables is given at the end.

Week6: Practicing DML commands

DML commands are used to for managing data within schema objects. Some examples:

? SELECT - retrieve data from the a database

? INSERT - insert data into a table

? UPDATE - updates existing data within a table

? DELETE - deletes all records from a table, the space for the records remain

Inserting values into Bus table:

Insert into Bus values (1234,'hyderabad', 'tirupathi');

Insert into Bus values (2345,'hyderabad', 'Banglore');

Inserting values into Bus table:

Insert into Passenger values (1, 45,'ramesh', 45,'M','abc 123');

Insert into Passenger values (2, 78,'geetha', 36,'F','abc124');

Few more Examples of DML commands:

Select * from Bus; (selects all the attributes and display)

UPDATE BUS SET Bus No = I WI-IERE BUS NO=2;

Week7: Querying

In this week you are going to practice queries (along with sub queries) using ANY, ALL, IN, EXISTS, NOT EXISTS, UNION, INTERSECT, Constraints etc.

Practice the following Queries:

1. Display unique PNR-no of all passengers.
2. Display all the names of male passengers.
3. Display the ticket numbers and names of all the passengers.
4. Display the' source and destination having journey time more than 10 hours.
5. Find the ticket numbers of the passengers whose name start with 'A' and ends with 'H'.
6. Find the names of passengers whose age is between 30 and 45.
7. Display all the passengers names beginning with 'A'
8. Display the sorted list of passengers names
9. Display the Bus numbers that travel on Sunday and Wednesday
10. Display the details of passengers who are traveling either in AC or NON_AC (Using only IN operator)

Week8 and week9: Querying (continued...)

You are going to practice queries using Aggregate functions (COUNT, SUM, AVG, and MAX and MIN), GROUP BY, I-IAVING and Creation and dropping of Views.

? Write a Query to display the Information present in the Passenger and cancellation tables.

Hint: Use UNION Operator.

? Write a Query to display different travelling options available in British Airways.

? Display the number of days in a week on which the 9WO1 bus is available.

? Find number of tickets booked for each PNR_no using GROUP BY CLAUSE. **Hint:** Use GROUP BY on PNR_No.

- ? Find the distinct PNR numbers that are present.
- ? Find the number of tickets booked in each class where the number of seats is greater than 1.
Hint: Use GROUP BY, WHERE and HAVING CLAUSES.
- ? Find the total number of cancelled seats.
- ? Write a Query to count the number of tickets for the buses, which travelled after the date '14/3/2009'. **Hint:** Use HAVING CLAUSES.

Week10: Triggers

In this week you are going to work on Triggers. Creation of insert trigger, delete trigger, update trigger. Practice triggers using the above database.

Eg: **CREATE TRIGGER up check BEFORE UPDATE ON passenger FOR EACH ROW**

BEGIN

IF NEW.Tickent NO > 60 THEN
SET New.Tickent No = Ticket no;

ELSE

SET New.Ticket no = 0;

END IF;

END;

Week11: Procedures

In this session you are going to learn Creation of stored procedure, Execution of procedure and modification of procedure. Practice procedures using the above database.

Eg: **CREATE PROCEDURE my Proc ()**

BEGIN

SELECT COUNT (Tickets) FROM Ticket WHERE age>=40;

End;

Week12: Cursors

In this week you need to do the following: Declare a cursor that defines a result set.

Open the cursor to establish the result set. Fetch the data into local variables as needed from the cursor, one row at a time. Close the cursor when done. .

CREATE PROCEDURE my Proc (in_customer_id INT)

BEGIN

DECLARE v_id {NT};

DECLARE v_name VARCHAR (30);

DECLARE c1 CURSOR FOR SELECT stdId, stdFirstname FROM students WHERE staid=in customer_id;

OPEN c1;

FETCH c1 into v_id, v_name;

Close c1;

END; .

Tables:

BUS

Bus No: Varchar: Pk

Source: Varchar

Destination: Varchar

Passenger

PNR_No : Numeric (9) : PK

Ticke_CNo: Numeric (9)

Name: Yarchar (15)

Age: int (4)

Sex: Char(10): Male/Female

PPNO: Yarchar (15)

Reservation

PNR_No: Numeric (9) : FK

Journey_date: date time (8)

No_of seats : int (8)

Address: Varchar (50)

Contact_No: Numeric (9) ---> Should not be less than 9 and Should not accept any other character other than Integer

Status: Char (2) : Yes / No

Cancellation

PNR_No: Numeric (9) : FK

Journey_date: date time (8)

No_of seats : int (8)

Address: Yarchar (50)

Contact No: Numeric (9) ---> should not be less than 9 and Should not accept any other character other than Integer

Status: Char (2) : Yes / No

Ticket

Ticket_No: Numeric (9): PK

Journey_date: date time (8)

Age: int (4)

Sex: Char (10) : Male / Female

Source: Varchar

Destination: Varchar

Dep_time : Varchar

ii) COMPILER

Consider the following mini Language, a simple procedural high-level language, only operating on integer data, with a syntax looking vaguely like a simple C crossed with Pascal.

The syntax of the language is defined by the following BNF grammar:

<Program> ::= <block>

<block> ::= { <variable definition> <slits> }

{ <slist> }

<variable definition> ::= int <vardeflist> ;

<Vardeflist> ::= <Verdec> |<Verdec> ,<vardeflist>

<Verdec> ::= <identifier> | <identifier> [<constant>]

<slist> ::= <statement> |<statement> ; <slits>

<statement> ::= <assignment> | <if statement> |<while statement>

<block> | <print statement> | <empty>
 <assignment> ::= <identifier> = <expression>
 | <identifier> [<expression>] = <expression>
 <if statement> ::= if <bexpression> then <slist> else <slist> endif
 | if <bexpression> then <slist> endif
 <Whilestatement> ::= while <bexpression> do <slist> enddo
 <printstatement> ::= print (<expression>)
 <expression> ::= <expression> <addingop> <term> | <term> | <addingop> <term>
 <bexpression> ::= <expression> <relop> <expression>
 <relop> ::= < | <= | == | >= | > | !=
 <addingop> ::= + | -
 <term> ::= <term> <multop> <factor> | <factor>
 <multop> ::= * | /
 <factor> ::= <constant> | <identifier> | <identifier> [<expression>]
 | (<expression>)
 <constant> ::= <digit> | <digit> <constant>
 <identifier> ::= <identifier> <letterordigit> | <letter>
 <letterordigit> ::= <letter> | <digit>
 <letter> ::= a|b|c|d|e|f|g|h|i|j|k|l|m|n|o|p|q|r|s|t|u|v|w|x|y|z
 digit ::= 0|1|2|3|4|5|6|7|8|9
 <empty> has the obvious meaning

Comments (zero or more characters enclosed between the standard C/Java-style comment brackets / *...*/) can be inserted. The language has rudimentary support for 1-dimensional arrays. The declaration `int a[3]` declares an array of three elements, referenced as `a[0]`, `a[1]` and `a[2]`. Note also that you should worry about the scoping of names.

A simple program written in this language is:

```

{ int a[3],t 1 ,t2;
t1=2;
a[0]= 1; a[1]=2; a[t1]=3;
t2=-(a[2]+t1 *6)/(a[2]-t1);
if t2>5 then
print(t2);
else {
int t3;
t3=99;
t2=-25;
print( -t 1 +t2*t3); /* this is a comment on 2 lines */
} end if }

```

1. Design a Lexical analyzer for the above language. The lexical analyzer should ignore redundant spaces, tabs and new lines. It should also ignore comments. Although the syntax specification states that identifiers can be arbitrarily long, you may restrict the length to some reasonable value.
2. Implement the lexical analyzer using JLex, flex or lex or other lexical analyzer generating tools.
3. Design Predictive parser for the given language
4. Design LALR bottom up parser for the above language.
5. Convert the BNF rules into Yacc form and write code to generate abstract syntax tree.
6. Write program to generate machine code from the abstract syntax tree generated by the parser. The following instruction set may be considered as target code.

The following is a simple register-based machine, supporting a total of 17 instructions. It has three distinct internal storage areas. The first is the set of 8 registers, used by the individual instructions as detailed below, second is an area used for the storage of variables and the third is an area used for the storage of program instructions. Instructions can be preceded by a label. This consists of an integer in the range 1 to 9999 and the label is followed by a colon to separate it from the rest of the instruction. The numerical label can be used as the argument to a jump instruction, as detailed below. In the description of the individual instructions below, instruction argument types are specified as follows:

R

specifies a register in the form R0, R1, R2, R3, R4, R5, R6 or R7 (or, r0, r1, etc.).

L

specifies a numerical label (in the range 1 to 9999).

V

specifies a "variable location" (a variable number, or a variable location pointed to by a register - see below).

A

specifies a constant value, a variable location, a register or a variable location pointed to by a register (an indirect address). Constant values are specified as an integer value, optionally preceded by a minus sign, preceded by a # symbol. An indirect address is specified by an @ followed by a register.

So, for example, an A-type argument could have the form 4 (variable number 4), #4 (the constant value 4), r4 (register 4) or @r4 (the contents of register 4 identifies the variable location to be accessed).

The instruction set is defined as follows:

LOAD A,R

loads the integer value specified by A into register R.

STORE R,V

stores the value in register R to variable V.

OUT R

Outputs the value in register R.

NEGR

negates the value in register R.

ADD A,R

Adds the value specified by A to register R, leaving the result in register R.

SUB A,R

subtracts the value specified by A from register R, leaving the result in register R.

MUL A,R

multiplies the value specified by A by register R, leaving the result in register R.

DIV A,R

divides register R by the value specified by A, leaving the result in register R.

JMP L

causes an unconditional jump to the instruction with the label L.

JEQ R,L

jumps to the instruction with the label L if the value in register R is zero.

JNE R,L

jumps to the instruction with the label L if the value in register R is not zero.

JGE R,L

jumps to the instruction with the label L if the value in register R is greater than or equal to

zero.

JGT R,L

jumps to the instruction with the label L if the value in register R is greater than zero.

JLE R,L

jumps to the instruction with the label L if the value in register R is less than or equal to zero.

•

JLT R,L

jumps to the instruction with the label L if the value in register R is less than zero.

NOP

is an instruction with no effect. It can be tagged by a label.

STOP

stops execution of the machine. All programs should terminate by executing a STOP instruction.