

GITAM UNIVERSITY

(Declared as Deemed to be University U/S 3 of UGC Act, 1956)



**REGULATIONS & SYLLABUS
OF
M.Tech. (Computer Science & Technology)
(w.e.f 2008 -09 admitted batch)**

Gandhi Nagar Campus, Rushikonda
VISAKHAPATNAM – 530 045
Website: www.gitam.edu

REGULATIONS

(W.e.f. 2008-09 admitted batch)

1.0 ADMISSIONS

- 1.1 Admissions into M.Tech. (Computer Science & Technology) programme of GITAM University are governed by GITAM University admission regulations.

2.0 ELIGIBILITY CRITERIA

- 2.1 A pass in B E / B Tech / AMIE or equivalent in any branch of specialization or Masters Degree in Physics, statistics, Mathematics or Applied Mathematics, Applied Statistics, Applied Physics, Geo Physics or M.Sc. in Computer Science or Information Systems or MCA.
- 2.2 Admissions into M.Tech will be based on the following:
- (i) Score obtained in GAT (PG), if conducted.
 - (ii) Performance in Qualifying Examination / Interview.

The actual weightage to be given to the above items will be decided by the authorities before the commencement of the academic year. Candidates with valid GATE score shall be exempted from appearing for GAT (PG).

3.0 STRUCTURE OF THE M.TECH. PROGRAMME

- 3.1 The Programme of instruction consists of :
- (i) A core programme imparting to the student specialization of engineering branch concerned.
 - (ii) An elective programme enabling the students to take up a group of departmental courses of interest to him/her.
 - (iii) Carry out a technical project approved by the Department and submit a report
- 3.2 Each academic year consists of two semesters. Every branch of the M.Tech programme has a curriculum and course content (syllabi) for the subjects recommended by the Board of Studies concerned and approved by Academic Council.
- 3.3 Project Dissertation has to be submitted by each student individually.

4.0 CREDIT BASED SYSTEM

- 4.1 The course content of individual subjects - theory as well as practicals – is expressed in terms of a specified number of credits. The number of credits assigned to a subject depends on the number of contact hours (lectures & tutorials) per week.
- 4.2 In general, credits are assigned to the subjects based on the following contact hours per week per semester.
- One credit for each Lecture hour.
 - One credit for two hours of Practicals.
 - Two credits for three (or more) hours of Practicals.

4.3 The curriculum of M.Tech programme is designed to have a total of 70 -85 credits for the award of M.Tech degree. A student is deemed to have successfully completed a particular semester's programme of study when he / she earns all the credits of that semester i.e., he / she has no 'F' grade in any subject of that semester.

5.0 MEDIUM OF INSTRUCTION

The medium of instruction (including examinations and project reports) shall be English.

6.0 REGISTRATION

Every student has to register himself/herself for each semester individually at the time specified by the College / University.

7.0 CONTINUOUS ASSESSMENT AND EXAMINATIONS

7.1 The assessment of the student's performance in each course will be based on continuous internal evaluation and semester-end examination. The marks for each of the component of assessment are fixed as shown in the Table 2.:

Table 2: Assessment Procedure

S.No.	Component of assessment	Marks allotted	Type of Assessment	Scheme of Examination
1	Theory	40	Continuous evaluation	(i) Two mid semester examinations shall be conducted for 10 marks each. (ii) Two quizzes shall be conducted for 5 marks each. (iii) 5 marks are allotted for assignments. (iv) 5 marks are allotted for attendance
		60	Semester-end examination	The semester-end examination in theory subjects will be for a maximum of 60 marks.
	Total	100		
2	Practicals	100	Continuous evaluation	(i) 40 marks are allotted for record work and regular performance of the student in the lab. (ii) One examination for a maximum of 20 marks shall be conducted by the teacher handling the lab course at the middle of the semester (iii) One examination for a maximum of 40 marks shall be conducted at the end of the semester (as scheduled by the Head of the Department concerned).

3	Project work	100	Project evaluation	(i) 50 marks are allotted for continuous evaluation of the project work throughout the semester by the guide. (ii) 50 marks are allotted for the presentation of the project work & viva-voce at the end of the semester.*
4	Comprehensive Viva	100	Viva-voce	100 marks are allotted for comprehensive viva to be conducted at the end of programme.*

* Head of the Department concerned shall appoint two examiners for conduct of the examination.

8.0 REAPPEARANCE

- 8.1 A Student who has secured 'F' Grade in any theory course / Practicals of any semester shall have to reappear for the semester end examination of that course / Practicals along with his / her juniors.
- 8.2 A student who has secured 'F' Grade in Project work shall have to improve his report and reappear for viva – voce Examination of project work at the time of special examination to be conducted in the summer vacation after the last academic year.

9.0 SPECIAL EXAMINATION

- 9.1 A student who has completed the stipulated period of study for the degree programme concerned and still having failure grade ('F') in not more than 5 courses (Theory / Practicals), may be permitted to appear for the special examination, which shall be conducted in the summer vacation at the end of the last academic year.
- 9.2 A student having 'F' Grade in more than 5 courses (Theory/practicals) shall not be permitted to appear for the special examination.

10.0 ATTENDANCE REQUIREMENTS

- 10.1 A student whose attendance is less than 75% in all the courses put together in any semester will not be permitted to attend the end - semester examination and he/she will not be allowed to register for subsequent semester of study. He /She has to repeat the semester along with his / her juniors.
- 10.2 However, the Vice Chancellor on the recommendation of the Principal / Director of the University college / Institute may condone the shortage of attendance to the students whose attendance is between 66% and 74% on genuine medical grounds and on payment of prescribed fee.

11.0 GRADING SYSTEM

- 11.1 Based on the student performance during a given semester, a final letter grade will be awarded at the end of the semester in each course. The letter grades and the corresponding grade points are as given in Table 3.

Table 3: Grades & Grade Points

Grade	Grade points	Absolute Marks
O	10	90 and above
A+	9	80 – 89
A	8	70 – 79
B+	7	60 – 69
B	6	50 – 59
C	5	40 – 49
F	Failed, 0	Less than 40

- 11.2 A student who earns a minimum of 5 grade points (C grade) in a course is declared to have successfully completed the course, and is deemed to have earned the credits assigned to that course. However, a minimum of 24 marks is to be secured at the semester end examination of theory courses in order to pass in the theory course

12.0 GRADE POINT AVERAGE

- 12.1 A Grade Point Average (GPA) for the semester will be calculated according to the formula:

$$\text{GPA} = \frac{\sum [C \times G]}{\sum C}$$

Where

C = number of credits for the course,
G = grade points obtained by the student in the course.

- 12.2 Semester Grade Point Average (SGPA) is awarded to those candidates who pass in all the subjects of the semester.
- 12.3 To arrive at Cumulative Grade Point Average (CGPA), a similar formula is used considering the student's performance in all the courses taken in all the semesters completed up to the particular point of time.
- 12.4 The requirement of CGPA for a student to be declared to have passed on successful completion of the M.Tech programme and for the declaration of the class is as shown in Table 4.

Table 4: CGPA required for award of Degree

Distinction	≥ 8.0*
First Class	≥ 7.0
Second Class	≥ 6.0
Pass	≥ 5.0

* In addition to the required CGPA of 8.0, the student must have necessarily passed all the courses of every semester in first attempt.

13.0 **ELIGIBILITY FOR AWARD OF THE M.TECH DEGREE**

13.1 **Duration of the programme:**

A student is ordinarily expected to complete the M Tech. programme in four semesters of two years. However a student may complete the programme in not more than four years including study period.

13.2 However the above regulation may be relaxed by the Vice Chancellor in individual cases for cogent and sufficient reasons.

13.3 Project dissertation shall be submitted on or before the last day of the course. However, it can be extended up to a period of 6 months maximum, with the written permission of the Head of the Department concerned.

13.4 A student shall be eligible for award of the M.Tech degree if he / she fulfils all the following conditions.

- a) Registered and successfully completed all the courses and projects.
- b) Successfully acquired the minimum required credits as specified in the curriculum corresponding to the branch of his/her study within the stipulated time.
- c) Has no dues to the Institute, hostels, Libraries, NCC / NSS etc, and
- d) No disciplinary action is pending against him / her.

13.5 The degree shall be awarded after approval by the Academic Council.

RULES

1. With regard to the conduct of the end-semester examination in any of the practical courses of the programme, the Head of the Department concerned shall appoint one examiner from the department not connected with the conduct of regular laboratory work, in addition to the teacher who handled the laboratory work during the semester.
2. In respect of all theory examinations, the paper setting shall be done by an external paper setter having a minimum of three years of teaching experience. The panel of paper setters for each course is to be prepared by the Board of Studies of the department concerned and approved by the Academic Council. The paper setters are to be appointed by the Vice Chancellor on the basis of recommendation of Director of Evaluation / Controller of Examinations.
3. The theory papers of end-semester examination will be evaluated by two examiners. The examiners may be internal or external. The average of the two evaluations shall be considered for the award of grade in that course.
4. If the difference of marks awarded by the two examiners of theory course exceeds 12 marks, the paper will have to be referred to third examiner for evaluation. The average of the two nearest evaluations of the three shall be considered for the award of the grade in that course.
5. Panel of examiners of evaluation for each course is to be prepared by the Board of Studies of the department concerned and approved by the Academic Council.
6. The examiner for evaluation should possess post graduate qualification and a minimum of three years teaching experience.
7. The appointment of examiners for evaluation of theory papers will be done by the Vice Chancellor on the basis of recommendation of Director of Evaluation / Controller of Examinations from a panel of examiners approved by the Academic Council.
8. Project work shall be evaluated by two examiners at the semester end examination. One examiner shall be internal and the other be external. The Vice Chancellor can permit appointment of second examiner to be internal when an external examiner is not available.
9. The attendance marks (maximum 5) shall be allotted as follows :

Percentage of Attendance	Marks
76% to 80%	1
81% to 85%	2
86% to 90%	3
91% to 95%	4
96% to 100%	5

SYLLABUS

M.Tech. (CST)

Programme Code: EPRCS200701

FIRST SEMESTER

Course Code	Name of the Course	Instruction hours				Maximum marks			Credits
		per week				C	S	Total	
		L	T	P	Total				
EPRCS101	Theory of Computation	4	-	-	4	40	60	100	4
EPRCS102	Software Engineering	3	-	-	3	40	60	100	3
EPRCS103	Data Structures and Algorithms	4	-	-	4	40	60	100	4
EPRCS104	Computer Organization	3	-	-	3	40	60	100	3
EPRCS105	Database Management systems	4	-	-	4	40	60	100	4
EPRCS111	Data structures lab	-	-	3	3	100	--	100	2
EPRCS112	Computer Organization Lab	-	-	3	3	100	--	100	2
Total									22

SECOND SEMESTER

Course Code	Name of the Course	Instruction hours				Maximum marks			Credits
		per week				C	S	Total	
		L	T	P	Total				
EPRCS201	Language processors	4	-	-	4	40	60	100	4
EPRCS202	Operating systems	3	-	-	3	40	60	100	3
EPRCS203	Embedded systems	4	-	-	4	40	60	100	4
EPRCS204	Computer Networks	4	-	-	4	40	60	100	4
EPRCS205	Object oriented Analysis and Design	4	-	-	4	40	60	100	4
EPRCS211	Networks lab	-	-	3	3	100	--	100	2
EPRCS212	DBMS lab	-	-	3	3	100	--	100	2
EPRCS213	Object Oriented Analysis and Design lab	-	-	3	3	100	--	100	2
Total									25

L – Lectures T – Tutorials D – Drawing P – Practical C- Continuous Evaluation S- Semester End Examination

M. Tech. (C S T) THIRD SEMESTER

Course Code	Name of the Course	Instruction hours				Maximum marks			Credits
		per week				marks			
		L	T	P	Total	C	S	Total	
EPRCS301	Internet Technologies	4	-	-	4	40	60	100	4
EPRCS302	Cryptography and Network Security	4	-	-	4	40	60	100	4
Elective-I		3	-	-	3	40	60	100	3
EPRCS321	a)Data warehousing &Data mining								
EPRCS322	b)Image processing								
EPRCS323	c)Bioinformatics								
EPRCS324	d)Advanced Databases								
EPRCS325	e)Advanced Algorithms								
Elective-II		3	-	-	3	40	60	100	3
EPRCS331	a)Geo Information Systems								
EPRCS332	b)Network Management								
EPRCS333	c)Advanced Computer Architecture								
EPRCS334	d) Artificial Intelligence								
EPRCS311	Embedded systems Lab	-	-	3	3	100	--	100	2
EPRCS312	Web programming lab	-	-	3	3	100	--	100	2
EPRCS313	Project Phase-I	-	-	3	3	100	--	100	3
Total									21

FOURTH SEMESTER

COURSE STRUCTURE AND SCHEME OF EXAMINATION

Course Code	Name of the Course	Instruction hours				Maximum marks			Credits
		per week				marks			
		L	T	P	Total	C	S	Total	
EPRCS411	Project Phase-II	-	-	24	24	50	50	100	12
EPRCS412	Comprehensive Viva	-	-	-	-	-	100	100	2
Total									14

Total Credits: 82

M.Tech. (CST) FIRST Semester THEORY OF COMPUTATION

Course Code :EPRCS101
Credits :4

Category : CE
No. of hours : 4 per week

UNIT – I:

Preliminaries: Strings, alphabets, and languages Graphs and trees
Inductive proofs, Set notation, Relations, Synopsis of the book.

Finite Automata and Regular Expressions: Finite state systems, Basic definitions, Nondeterministic finite automata, Finite automata with ϵ -moves, Regular expressions, Two-way finite automata Finite automata with output, Applications of finite automata

UNIT – II

Properties of Regular Sets: The pumping lemma for regular sets, Closure Properties of regular sets, Decision algorithms for regular sets, The Myhill-Nerode theorem and minimization of finite automata.

Context-Free Grammars: Motivation and introduction, Context-free grammars, Derivation Trees, Simplification of context-free grammars, Chomsky normal form, Greibach normal form, The existence of inherently ambiguous context-free languages.

UNIT – III

Pushdown Automata: Informal description, Definitions, Pushdown automata and context-free languages.

Properties of Context-Free Languages: The pumping lemma for CFL's, Closure properties of CFL's, Decision algorithms for CFL's.

UNIT-IV

Turing Machines: Introduction, The Turing machine model, Computable languages and functions, Techniques for Turing machine construction, Modifications of Turing machines, Church's hypothesis, Turing machines as enumerators, Restricted Turing machines equivalent to the basic model.

UNIT-V

Undecidability: problems, Properties of recursive and recursively enumerable languages, Universal Turing machines and an undecidable problem, Rice's theorem and some more undecidable problems, Undecidability of Post's correspondence problem, Valid and invalid computations of TM's a tool for proving CFL problems undecidable, Greibach's theorem, Introduction to recursive function theory, Oracle computations.

The Chomsky Hierarchy: Regular grammars, Unrestricted grammars, Context-sensitive languages, Relations between classes of languages, Polynomial time and space, Some NP-complete problems, The classes co-N P.

Text Books:

- 1.Hopcroft. J.E and J.D. Ullman. Introduction to Automata Theory, Languages, and Computation, Addison-Wesley, Reading, Mass. 1979.
- 2.Harry R Lewis, Christos H. Papadimitriou : Elements of The Theory of Computation", Prentice-Hall of India Private Limited, New Delhi

M.Tech. (CST) FIRST Semester
SOFTWARE ENGINEERING

Course Code : EPRCS102
Credits : 3

Category : CE
No. of hours : 3 per week

UNIT-I:

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

UNIT-II:

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

UNIT-III:

Design Engineering: Design process and Design quality, Design concepts, the design model. **Creating an architectural design:** software architecture, Data design, Architectural styles and patterns, Architectural Design. **Object-Oriented Design:** Objects and object classes, An Object-Oriented design process, Design evolution. **Performing User interface design:** Golden rules, User interface analysis and design, interface analysis, interface design steps, Design evaluation.

UNIT-IV:

Testing Strategies: A strategic approach to software testing, test strategies for conventional software, Black-Box and White-Box testing, Validation testing, System testing, the art of Debugging. **Product metrics:** Software Quality, Metrics for Analysis Model, Metrics for Design Model, Metrics for source code, Metrics for testing, Metrics for maintenance.

UNIT-V:

Plans for testing: Snooping for information, Coping with complexity through teaming, Testing plan focus areas, Testing for recoverability, Planning for troubles. **Preparing for the tests:** Software Reuse, Developing good test programs, Data corruption, Tools, Test Execution, Testing with a virtual computer, Simulation and Prototypes, Managing the Test, Customer's role in testing

TEXT BOOKS:

1. Software Engineering, A practitioner's Approach- Roger S. Pressman, 6th edition. McGraw Hill International Edition.
2. Software Engineering- Sommerville , 7th edition, Pearson education.
3. Software Testing Techniques – Loveland, Miller, Prewitt, Shannon, Shroff Publishers & Distribution Pvt Ltd.,

REFERENCE BOOKS:

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

DATA STRUCTURES AND ALGORITHMS

Course Code : EPRCS 103
Credits : 4

Category : CE
No. of hours : 4 per week

UNIT I

Introduction

Object oriented programming concepts, Overview of C++ classes, pointers, parameters passing. Data Representation: Linear lists, formula based representation, indirect addresses, simulating pointers, Arrays, matrices, Basics of time complexity

UNIT II

Linked Lists, Stacks And Queues

Single Linked Lists, Double Linked Lists, Circular Lists Operations and Applications, Array and linked representation of stacks Queues: Definitions and operations, Array and linked representation of queues. Circular Queues, Dequeues, Priority Queue, Applications.

UNIT III

Trees

Definitions And properties, Representation of Binary Trees, Operations. Binary tree traversal. AVL Trees and Operations on AVL Trees, B+Trees, Operations on B + trees and applications.

UNIT IV

Searching And Sorting

Merge Sort, Quick Sort, Selection Sort, Heap Sort. Complexity analysis. Sequential Search, Binary Search. Various types of hashing.

UNIT V

Graphs

Definitions and Representation of Graphs, Graph Search Methods, Applications, Spanning Tree, Minimum Spanning Tree, Prim's Algorithm, Kruskal's Algorithm.

Text Books:

1. Data Structures, Algorithms and Applications in C++, S Sahani, Tata McGraw Hill.
2. Data Structures using C and C++ , Yedidyah Langsam, MosheJ Augenstein, aaron M Tenenbaum , Prentice Hall India.
3. Data Structures & Algorithm Analysis in C++, Mark Allen Weiss. Second Edition, Pearson Edition. Asia.

Reference Books:

1. Data Structures & Algorithm in C++, Adam Drozdek. Vikas publication House.
2. Data Structure, Algorithm and OOP, Gregory L. Heileman (Tata McGraw Hill Edition)

DATABASE MANAGEMENT SYSTEM

Course Code : EPRCS105 Category : CE
Credits : 4 No. of hours : 4 per week

UNIT-I

Introduction:

Data, storing data in dbms, relational model, levels of abstraction, data independence, queries in dbms, transaction management, structure of dbms and people who work with dbms

UNIT-II

ER Model, Relational Model

ER model, entities, attributes & entity sets; relationship & relationship sets, ER diagrams, key and participation constraints, weak entities, class hierarchies, aggregation, conceptual design with ER model, case study: requirement analysis and conceptual design,

UNIT-III

Relational Algebra And SQL:

Relational algebra, relational calculus, Basic SQL query, nested queries, aggregate operators, null values, integrity constraints, Triggers

UNIT-IV

Database Design:

Schema refinement, Functional dependencies, closure of set of FDs and attribute, Third & Boyce-code normal forms, properties of decomposition, Decomposition into 3NF and BCNF, Schema refinement in database design

UNIT-V

Transaction Processing:

ACID properties, Transaction & schedule, Concurrent execution of transaction, Lock-based concurrency control, crash recovery, Serializability and recoverability, Lock management, lock conversions, dealing with dead locks, Optimistic concurrency control, timestamp-based concurrency, multi version concurrency control, ARIES, logs, the write-ahead log protocol, check pointing, recovering from a system crash, media recovery

Text Book:

Database Management Systems; Raghu Ramakrishnan, Johannes Gehrke 4th Edition, McGraw Hill

Reference Book:

Database System Concepts; A. Silberschatz, H. Korth 5th Edition, McGraw Hill

**M.Tech. (CST) FIRST Semester
DATA STRUCTURES LAB**

Course Code : EPRCS111
Credits : 2

Category : CE
No. of hours : 3 per week

Develop algorithms and write programs in C++ to implement the following:

1. Operations on stacks.
2. Operations on queues
3. In order, preorder, post order traversal of a binary tree using recursion
4. In order, preorder, post order traversal of a binary tree using non recursion
5. Addition and multiplication of two sparse matrices
6. Converting a given infix expression to postfix form
7. Evaluating a given postfix expression
8. Operations of a dequeue
9. Hashing Algorithms
10. Sorting Algorithms: Quick Sort, Merge Sort, Insertion Sort, Heap Sort
11. Depth First Search of a graph
12. Breadth First Search of a Graph
13. Finding the shortest path from a given node to all other nodes a digraph
 - a) Directed Graph
 - b) Undirected Graph
14. Polynomial Addition, multiplication.

**M.Tech. (CST) FIRST Semester
COMPUTER ORGANIZATION LAB**

Course Code : EPRCS112
Credits : 2

Category : CE
No. of hours : 3 per week

List Of Experiments Hardware

1. Verification of gates
2. Half adder and full adder
3. Half subtractor and full subtractor
4. Decoder and multiplexer
- 5 R-S and J-K flip flops
- 6 Counter(Synchronous or ripple)
7. Shift register

Software

1. Maximum from a set of numbers
2. Addition of two sixteen bit numbers
3. Sorting a set of numbers
4. Conversion of BCD to HEX
5. Conversion of HEX to BCD
6. Flashing and rolling display of a word
7. Display of seconds in the data field

**M.Tech. (CST) SECOND SEMESTER
OPERATING SYSTEMS**

Course Code : EPRCS202
Credits : 3

Category : CE
No. of hours : 3 per week

UNIT-I

Introduction to Operating Systems: Definition, Types of OS: Batch processing system, Multiprogramming Batch System, Time sharing systems, Introduction to Distributed systems, Multi processing systems, Real time system, Definition, Hardware concepts, Software concepts, Design issues. Operating systems structures: Services, system calls, systems programs, system structure, virtual machines.

UNIT-II

Process Management: Process Concepts, Process scheduling, Operation on processes, Co-operating processes, Threads, Inter process communications. CPU Scheduling, Scheduling algorithms, Multiple Processors and Real time scheduling. Process synchronization: Critical section problems, Semaphores, Classical problems of Synchronization.

UNIT-III

Dead Locks: Characterization, Handling, Prevention, Avoidance, Detection & Recovery.

UNIT-IV

Storage Management: Memory Management: Swapping, Paging, Segmentation, Segmentation & Paging. Virtual memory, Demand paging, Page replacement, Frames, Thrashing, Demand Segmentation. File Systems: Concepts, Access methods, File system structure, Protection, Allocation methods.

UNIT-V

I/O management: I/O Hardware, I/O Interfaces, Kernel I/O Subsystems, Disk scheduling, Case study overview: Unix, Linux, And Windows NT OS.

Text Books:

1. Applied Operating Systems concepts with Java Silberschatz, Galvin, Grey Gagne.
2. Distributed Operating Systems by Andrew S Tanenbaum.

Reference Book:

Modern Operating Systems – Andrew S. Tanenbaum, PHI

**M.Tech. (CST) SECOND SEMESTER
COMPUTER NETWORKS**

Course Code: EPRCS204
Credits : 4

Category: CS
No. of hours: 4 per week

Unit-I

Reference Models: OSI, TCP/IP and Differences between OSI and TCP/IP. Examples of Networks: Novel Netware, Internet, Connection Oriented Networks.

MAC Sub layer: Channel Allocation Problems: Static and Dynamic. Multiple Access Protocols: ALOHA, CSMA, Collision-free protocols.

Unit-II

IEEE 802.x Standards Ethernet, Wireless LANS: 802.11 protocol stack, physical layer, MAC sub layer protocol, frame structure and services. Bluetooth, Data link Layer Switching: Bridges from 802.x to 802.y, Local Internetworking, Spanning tree bridges, Remote bridges, Repeaters, Hubs, Bridges, Switches, Routers and Gateways.

Unit-III

Network Layer: Design Issues: Store and forward packet switching, Services provided to the Transport Layer, Implementation of connection less and connection oriented services and comparisons. Routing Algorithms, Congestion Control Algorithms, The Network Layer in the Internet: IP, ICP and IPV6.

Unit-IV

Transport Layer: Transport Services: Services provided to the upper layer, Primitives, Berkeley Sockets and examples. Elements of Transport Protocols: Addressing, Connection Establishment, Release, Flow control and Buffering, Multiplexing and crash Recovery. The Internet Transport Protocols: TCP, UDP.

Unit-V

Application Layer:

The Domain Name System (DNS): The DNS Name Space, Resource, Records, Name server. Electronic Mail: Architecture and services, the user Agent, Message Formats, Message Transfer, Final Delivery. The World Wide Web (WWW): Architecture Overview, Static Web Documents , Dynamic Web Documents, HTTP – The Hyper Text Transfer Protocol, Performance Enhancements, The Wireless Web.

Text Book:

1. Computer Networks – Andrew S Tanenbum, 4th Edition. Pearson Education/PHI.

Reference Books:

1. Data and Computer Communications, 7th Edition by William Stallings.
2. Data Communications and Networking by Behrouz a Forouzan, 4th Edition. Tata McGraw-Hill.

**M.Tech. (CST) SECOND SEMESTER
COMPUTER NETWORKS LAB**

Course Code : EPRCS211 Category : CE
Credits : 2 No. of hours : 3 per week

1. Identifying well known ports on a Remote System:

By trying to listen to the various well known ports by opening client connections. If the exception does not occur then the remote port is active else the remote port is inactive.

2. Writing a Chat application:

i) One-One: By opening socket connection and displaying what is written by one party to the other.

ii) Many-Many (Broad cast): Each client opens a socket connection to the chat server and writes to the socket. Whatever is written by one party can be seen by all other parties.

3. Data retrieval from a Remote Database:

At the remote database a server listens for client connections. This server accepts SQL queries from the client, executes it on the database and sends the response to the client.

4. Mail Client:

i) POP Client : Gives the server name , user name and password retrieve the mails and allow manipulation of mail box using POP commands.

ii) SMTP Client : Gives the server name, send e-mail to the recipient using SMTP commands- (Core Java 2 pg:163.)

5. Simulation of Telnet:

Provide a user interface to contact well-known ports, so that client-server interaction can be seen by the user.

6. Simple file transfer between two systems (without protocols):

By opening socket connection to our server on one system and sending a file from one system to another.

7. TFTP-Client:

To develop a TFTP client for file transfer. (Unix Network programming-Stevens)

8. HTTP-Server:

Develop a HTTP server to implement the following commands.

GET, POST, HEAD, DELETE.

The server must handle multiple clients.

Reference Books:

1. Java Network Programming, Harold Orielly
2. An Introduction to Computer Networking, Kenneth C. Mansfield Jr and James L. Antonakos Pearson Education Asia.

**M.Tech. (CST) SECOND SEMESTER
DATABASE MANAGEMENT SYSTEM LAB**

Course Code : EPRCS212
Credits : 2

Category : CE
Hours : 3 per week

Each student is assigned with a problem. The student is to develop a logical and physical database design for the problem.

A. The logical design performs the following tasks:

1. Map the ER/EER diagrams to a relational schema. Be sure to underline all primary keys, include all necessary foreign keys and indicate referential integrity constraints.
2. Identify the functional dependencies in each relation
3. Normalize to the highest normal form possible

B. Perform physical design based above logical design using Oracle/MSSQL on Windows platform and MySQL/PostgreSQL on Linux platform

C. Perform DML and DDL using all possible SQL commands and with the help any one host languages like C, C++, VB etc (ie embedded SQL)

D. Perform DML and DLL using PL/SQL and PL/pgSQL for the above problems

Reference Books:

1. Oracle PL/SQL Programming Steven Feuerstein O'Reilly Publishers
2. PL/pgSql, search internet for necessary documentation

M.Tech. (CST) SECOND SEMESTER OBJECT ORIENTED ANALYSIS AND DESIGN LAB

Course Code : EPRCS213
Credits : 2

Category : CE
Hours : 3 per week

The student is expected to take up about five mini-projects and model them and produce Use Cases, Analysis Documents - both static & dynamic aspects, Sequence Diagrams and State-Charts, Database Design using Rational Products_A sample collection of ideas is given. Numerous other ideas can be found in the pages from the list of references given below.

Mini-Project-I: A Point-of-Sale (POS) System:

A POS system is a computerized application used to record sales and handle payments; it is typically used in a retail store, it includes hardware components such as a computer and bar code scanner, and software to run the system. It interfaces to various service applications, such as a third-party tax calculator and inventory control. These systems must be relatively fault tolerant; that is, even if remote services are temporarily unavailable they must still be of capturing sales and handling at least cash payments. A POS system must support multiple and varied client-side terminals and interfaces such as browser, PDAs, touch-screens.

Mini-Project-II: Online Bookshop Example:

Following the model of amazon.com or bn.com, design and implement an online bookstore.

Mini-Project-III: A Simulated Company:

Simulate a small manufacturing company. The resulting application will enable the user to take out a loan, purchase a machine, and over a series of monthly production runs, follow the performance of their company.

Mini-Project-IV: A Multi-Threaded Airport Simulation:

Simulate the operations in an airport. Your application should support multiple aircrafts using several runways and gates avoiding collisions/conflicts. Landing: an aircraft uses the runway, lands, and then taxis over to the terminal. Take-Off: an aircraft taxis to the runway and then takes off.

Mini-Project-V: An Automated Community Portal:

Business in the 21st Century is above all BUSY. Distractions are everywhere. The current crop of "enterprise intranet portals" are often high noise and low value, despite the large capital expenditures it takes to stand them up. Email takes up 30 - 70% of an employee's time. Chat and Instant Messaging are either in the enterprise or just around the corner. Meanwhile, management is tasked with un for seen and unfunded leadership and change-agent roles as well as leadership development and succession management. What is needed is a simplified, repeatable process that enhances communications within an enterprise, while allowing management and peers to self-select future leaders and easily recognize high performance team members in a dynamic way.

Additionally, the system should function as a general-purpose content management, business intelligence and peer-review application.

Glass code's goal is to build that system. The software is released under a proprietary license, and will have the following features: Remote, unattended moderation of discussions However, it will have powerful discovery and business intelligence features, and be infinitely extendable, owing to a powerful API and adherence to Java platform standards. Encourages peer review and indicates for management potential leaders, strong team players and reinforces enterprise and team goals seamlessly and with zero administration.

Mini-Project-VI: A Content Management System:

The goal is to enable non-technical end users to easily publish, access, and share information over the web, while giving administrators and managers complete control over the presentation, style, security, and permissions.

Features:

- * Robust Permissions System
- * Templates for easy custom site designs
- * Total control over the content
- * Search engine friendly URL's
- * Role based publishing system
- * Versioning control
- * Visitor profiling

Mini-Project-VII: An Auction Application:

Several commerce models exist and are the basis for a number of companies like eBay.com, pricellne.com etc. Design and implement an auction application that provides auctioning services. It should clearly model the various auctioneers, the bidding process, auctioning etc.

Mini-Project-VIII: A Notes and File Management System:

In the course of one's student years and professional career one produces a lot of personal notes and documents. All these documents are usually kept on papers or individual files on the computer. Either way the bulk of the information is often erased corrupted and eventually lost. The goal of this project is to build a distributed software application that addresses this problem. The system will provide an interface to create, organize and manage personal notes through the Internet for multiple users. The system will also allow users to collaborate by assigning permissions for multiple users to view and edit notes.

Mini-Project-IX: A Customizable Program Editor:

A programmer's editor which will be focused on an individual programmer's particular needs and style. The editor will act according to the specific language the current source file is in, and will perform numerous features, such as auto-completion or file summarization, on the file. These features will be able to be turned on or off by the programmer, and the programming style of the user will be used to create as efficient an editing environment as possible.

Mini-Project-X: A Graphics Editor:

Design and implement a Java class collection that supports the construction of graph editing applications, i.e., applications that include the ability to draw structured and unstructured diagrams.

E.g., The goal of the GEF project is to build a graph editing library that can be used to construct many, high-quality graph editing applications. Some of GEF's features are:

A simple, concrete design that makes the framework easy to understand and extend.

Node-Port-Edge graph model that is powerful enough for the vast majority of connected graph applications.

Model-View-Controller design based on the Swing Java UI library makes GEF able to act as a UI to existing data structures, and also minimizing learning time for developers familiar with Swing.

High-quality user interactions for moving, resizing, reshaping, etc. GEF also supports several novel interactions such as the broom alignment tool and selection-action-buttons.

Generic properties sheet based on Java Beans introspection. XML-based file formats based on the PGML standard Text Book(s): "Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and the Unified Process", Craig Larman, Pearson Education Asia, 2002, 2nd Edition

Reference(s):

"Object Oriented Systems Analysis and Design using UML", Simon Sennet, Steve McRobb, and Ray Farmer, McGraw Hill, 2002, 2nd Edition

"Object-Oriented Analysis & Design," Andrew Haigh, Tata McGraw Hill, 2001

Various Net Resources and Projects:

<http://user-mode-linux.sourceforge.net/case-studies.html>

<http://www.onesmartclick.com/programming/case-studies.html>

<http://www.tigris.org/servlets/ProjectList?type=P> rejects

<http://hotscripts.com/>

<http://www.developingwebs.net/>

<http://sourceforge.net/projects/>

<http://governing.com/gpp/gponline.htm>

<http://www.cio.com/research/government/gov.html>

<http://www.whitehouse.gov/omb/inforeg/egovstrategy.pdf>

<http://www.andhrapradesh.com/>

<http://www.ap-It.com/>

<http://www.aponline.gov.in>

CRYPTOGRAPHY AND NETWORK SECURITY

Course Code : EPRCS302

Category: CE

Credits : 4

No. of Hours: 4 per week

UNIT – I

Introduction: Security goals, Attacks, Services and mechanism, techniques. Symmetric key encipherment: Integer arithmetic, modular arithmetic. Traditional symmetric–key ciphers: substitution cipher, transposition cipher, stream and block cipher. Modern symmetric key cipher: modern block cipher and stream cipher.

Unit -II

DES structure , DES analysis, multiple DES, security of DES, AES, transformation, key expansion, cipher, analysis of AES. Encipherment using modern symmetric key ciphers: use of modern block cipher and stream cipher.RC4, key management, key generation. Primes, primality test, Chinese remainder theorem.

Unit -III

Asymmetric key cryptography: Introduction, RSA cryptosystem, RABIN cryptosystem, ELGAMA cryptosystem, ELLIPTIC curve cryptosystem .Groups, ring, fields, $GF(2^n)$ fields. Message integrity, message authentication.

Unit-IV

Cryptographic hash functions: introduction, SHA-512.

Digital signature: Comparison, process, services, digital process scheme: RSA Digital signature scheme, Elgamal Digital signature scheme, Digital signature standards (DSS).

Key-management: symmetric key distribution, Kerberos, symmetric key agreement, Public key distribution.

Unit-V

Security at application layer: E-MAIL, PGP, S/ MIME .Security at transport layer: SSL architecture, handshake protocols, SSL message formats, Transport layer security. Security at network layer: IPsec, authentication header (AH)., ESP, IPv4,IPv6,AH versus ESP, security provided by IPsec, security association, security policy, internet key exchange, ISAKMP.

Text books:

1. Cryptography and Network Security Behrouz A. Forouzan, TMH
2. Cryptography and Network Security Third Edition, William Stallings, Pearson Education

Reference Books:

1. Fundamentals of Network Security by Eric Maiwald (Dreamtech press)
2. Network Security - Private Communication in a Public World by Charlie Kaufman, Radia Perlman and Mike Speciner, Pearson/PHI.
3. Principles of Information Security, Whitman, Thomson.
5. Network Security: The complete reference, Robert Bragg, Mark Rhodes, TMH
6. Introduction to Cryptography, Buchmann, Springer.
7. Cryptography and Network Security Second edition, Atul Kahate, TMH

**M.Tech. (CST) THIRD SEMESTER
DATA WAREHOUSING AND DATA MINING**

Course Code : EPRCS321

Category : CE

Credits : 3

Hours : 3 per week

UNIT-I

Introduction to Data Mining:

Motivation and importance, What is Data Mining, Relational Databases, Data Warehouses, Transactional Databases, Advanced Database Systems and Advanced Database Applications, Data Mining Functionalities, Interestingness of a pattern Classification of Data Mining Systems, Major issues in Data Mining.

UNIT-II

Data Warehouse and OLAP Technology for Data Mining:

What is a Data Warehouse? Multi-Dimensional Data Model, Data Warehouse Architecture, Data Warehouse Implementation, Development of Data Cube Technology, Data Warehousing to Data Mining. **Data Preprocessing:** Why Pre-process the Data? Data Cleaning, Data Integration and Transformation, Data Reduction, Discretization and Concept Hierarchy Generation

UNIT-III

Concept Description: Characterization and Comparison:

What is Concept Description, Data Generalization and summarization-based Characterization, Analytical Characterization: Analysis of Attribute Relevance, Mining Class Comparisons: Discriminating between different Classes, Mining Descriptive Statistical Measures in large Databases.

UNIT-IV

Mining Association rule in large Databases:

Association Rule Mining, Mining Single-Dimensional Boolean Association Rules from Transactional Databases, Mining Multilevel Association Rules from Transaction Databases, Mining Multidimensional Association Rules from Relational Databases and Data Warehouses, From Association Mining to Correlation Analysis, Constraint-Based Association Mining.

UNIT-V

Classification and Prediction:

Concepts and Issues regarding Classification and Prediction, Classification by Decision Tree Induction, Bayesian Classification, Classification by Back propagation, Classification Based on Concepts from Association Rule Mining, Other Classification Methods like k-Nearest Neighbor Classifiers, Case-Based Reasoning, Generic Algorithms, Rough Set Approach, Fuzzy Set Approaches, Prediction, Classifier Accuracy. **Cluster Analysis:** What is Cluster Analysis, Types of Data in Cluster Analysis, A Categorization of Major Clustering Methods, Hierarchical Algorithms, Partition Algorithms.

Text Book:

1. Data Mining Concepts and Techniques Jiawei Han and Micheline Kamber Morgan Kaufman Publications

Reference Books:

1. Data Mining Introductory and Advanced Topics, Margaret H Dunhan, Pearson Education.
2. Data Mining, Ian H. Witten Eibe Frank, Morgan Kaufman Publications.
3. Data Mining by Tan, Steinbach, Vipin Kumar, Pearson Education.

**M.Tech. (CST) THIRD SEMESTER
IMAGE PROCESSING**

Course Code	:	EPRCS322	Category	:	CE
Credits	:	3	Hours	:	3 per week

UNIT-I

Introduction:

What is Digital Image Processing, Fundamental steps in Digital Image Processing, Components of Image Processing System, Simple Image Model, Image Sampling and Quantization, some basic relationships between pixels

UNIT-II

Image Enhancement in Spatial Domain:

Some Basic Gray Level Transformations, Histogram Processing, Enhancement using Arithmetic/Logic operations, Basics of Spatial filtering, Smoothing spatial filters, Sharpening spatial filters

Color Image Processing: Color Fundamentals, Color Models

UNIT-III

Image Restoration:

A model of the Image Degradation/Restoration process, Noise Models, Restoration in the Presence of Noise Only-Spatial Filtering, Introduction to the Fourier Transform and the Frequency Domain, Periodic Noise Reduction by Frequency Domain Filtering, Linear, Position-Invariant Degradations, Estimating the Degradation Functions.

UNIT-IV

Image Compression:

Fundamentals, Image Compression Models, Lossless Compression: Huffman Coding, Run length Encoding, Contour coding A brief discussion on Lossy Compression, A brief discussion on Compression Standards

Image Segmentation:

Detection of Discontinuities, Brief discussion of Thresholding, Region Based Segmentation and The Use of Motion in Segmentation.

UNIT-V

Morphological Image Processing:

Preliminaries, Dilation and Erosion, Opening and closing, The Hit-or-Miss transform, Some Basic Morphological Algorithms

Object Recognition:

Patterns and Pattern classes, Recognition Based on Decision – Theoretic Methods, Structural Methods.

Text Book:

Digital Image Processing, Second Edition By Rafael C. Gonzalez and Richard E. Woods, Addison Wesley

Reference Books:

1. Image Processing, Analysis & Machine Vision by Milan Sonka, Vaclaw Hlavac, Roger Boyle, Vikas Publishing House.
2. Digital Image Processing and Analysis, B. Chanda & D Dutta Majumder, Pearson.

3. Fundamentals of Digital Image Processing, Anil K Jain, Pearson.
4. Digital Image Processing using MATLAB, Goinzalez

M.Tech. (CST) THIRD SEMESTER BIOINFORMATICS

Course Code : EPRCS323
Credits : 3

Category : CE
Hours : 3 per week

UNIT-I

Introduction:

Definitions, Sequencing, Biological sequence/structure, Genome Projects, Pattern recognition an prediction, Folding problem, Sequence Analysis, Homology and Analogy.

Protein Information Resources: Biological databases, Primary sequence databases, Protein Sequence databases, Secondary databases, Protein pattern databases, and Structure classification databases.

UNIT-II

Genome Information Resources:

DNA sequence databases, specialized genomic resources

DNA Sequence analysis:

Importance of DNA analysis, Gene structure and DNA sequences, Features of DNA sequence analysis, EST (Expressed Sequence Tag) searches, Gene hunting, Profile of a cell, EST analysis, Effects of EST data on DNA databases

UNIT-III

Pair wise alignment techniques:

Database searching, Alphabets and complexity, Algorithm and programs, Comparing two sequences, sub-sequences, Identity and similarity, The Dotplot, Local and global similarity, different alignment techniques, Dynamic Programming, Pair wise database searching.

UNIT-IV

Multiple sequence alignment:

Definition and Goal, The consensus, computational complexity, Manual methods, Simultaneous methods, Progressive methods, Databases of Multiple alignments and searching, Secondary database searching, Importance and need of secondary database searches, secondary database structure and building a sequence search protocol

UNIT-V

Analysis packages:

Analysis package structure, commercial databases, commercial software, comprehensive packages, packages specializing in DNA analysis, Intranet Packages, Internet Packages.

Text Books:

1. Introduction to Bioinformatics, by T K Attwood & D J Parry-Smith Addison Wesley Longman
2. Bioinformatics- A Beginner's Guide by Jean-Michel Claveriw, Cerdric Notredame, WILEY dreamlech India Pvt. Ltd.

Reference Book:

1. Introduction to Bioinformatics by M. Lesk Oxford publishers (Indian Edition)

Text Book:

Concepts and Techniques of Geographic Information Systems, by C. P. Lo & Albert K. W. Yeung, Prentice Hall of India Ltd

Reference Books:

1. An Introduction to Geographical Information Systems, by Ian Heywood, Sarah Corneliu & Steve Carver, Pearson Education.
2. Introduction to Geographic Information Systems, by Kang-rsung Chang, Tata McGraw Hill Publishing Company Limited.

M.Tech. (CST) THIRD SEMESTER NETWORK MANAGEMENT

Course Code : EPRCS332

Category : CE

Credits : 3

Hours : 3 per week

UNIT – I:

Introduction: Network Management Requirements – Network Management System.

Network Monitoring: Network Monitoring Architecture – Performance Monitoring – Fault Monitoring – Accounting Monitoring.

Network Control: Configuration Control – Security Control.

UNIT –II

SNMP Network Management Concepts: Background – Basic Concepts.

SNMP Management Information: Structure of management Information – Practical Issues.

Standard MIB: MIB-II – Ethernet Interface MIB.

Simple Network Management Protocol: Basic Concepts – Protocol Specifications – Transport-level Support – SNMP Group – Practical Issues.

UNIT -III

Remote Network Monitoring: Statistics Collection: Basic Concepts, Groups: statistics, history, host, host TopN, matrix token Ring extensions to RMON.

Remote Network Monitoring: Alarms and Filters: Groups: alarm, filter, packet capture, event.

RMON2: Overview – Protocol Directory Group – Protocol Distribution Group – Address Map Group –

RMON2 Host Groups – MON2 Matrix Groups – User History Collection Groups – Probe Configuration Group – Extensions to RMON1 for RMON2 Devices.

UNIT - IV

SNMPv2: Management Information : Background –Structure of Management Information

SNMPv2: Protocol: Protocol operations – Transport Mappings – Coexistence with SNMPv1.

SNMPv2: MIBs and Conformance: SNMPv2Management Information Base—Conformance Statements –

Evolution of the interfaces group of MIB-II

UNIT – V

SNMPv3: Cryptographic Algorithms in SNMPv3: Conventional Encryption with DES – The MD5 Secure Hash Function – The SHA-1 Secure Hash Function – Message Authentication with HMAC.

SNMPv3: Architecture and Applications: Background – SNMPv3 Overview – SNMPv3 Architecture MIBs for SNMPv3 applications.

Text Book:

SNMP, SNMPv2, SNMPv3, AND RMON1 and 2 by William Stallings 3rd Edn, Pearson Education

Reference Book:

Network Management Principles and Practice by Mani Subramanian, Pearson Education

**M.Tech. (CST) THIRD SEMESTER
ADVANCED COMPUTER ARCHITECTURE**

Course Code : EPRCS333

Category : CE

Credits : 3

Hours : 3 per week

Unit-I

Introduction to Parallel Processing

Trends towards Parallel Processing, Parallelism in Uniprocessors Systems, Parallel Computer Structures, Architectural Classification Schemes, Parallel Processing Applications.

Unit-II

Memory and Input-Output Subsystem

Hierarchical Memory Architecture, Virtual Memory System, Memory Allocation and Management, Cache Memories and Managements, Input-Output Sub-systems.

Unit-III

Principles of Pipelining and Vector Processing

Principles of linear pipelining, Classification of pipeline processors, General pipelines and reservation tables, Interleaved Memory organization, Instruction and arithmetic pipe lines, Principles of designing pipelined processors, Vector Processing requirements.

Vector Processor The Architecture of CRAY-1, Pipeline chaining and vector loops, The Architecture of CYBER-205, Vector Processing in CYBER-205.

Unit-IV

Structures and Algorithms for Array Processors: SIMD array processors, SIMD interconnection networks parallel algorithms for array processors, Associative array processing.

Unit-V

Multiprocessors Architecture and Programming

Functional structures, Interconnection networks, Parallel memory organization, Multiprocessors Operating system, Exploiting concurrency for multiprocessing.

Data Flow Computers : Data driven computing and languages, Data flow computer Architecture.

Text Book:

Computer Architecture and Parallel Processing - Kai Hwang and Faye A. Briggs. Scalable Parallel Computing – K.Hwang, Tata McGra Hill.

**M.Tech. (CST) THIRD SEMESTER
EMBEDDED SYSTEMS LAB**

Course Code	:	EPRCS311	Category	:	CE
Credits	:	2	Hours	:	3 per week

I - Cycle

8051 Assembly Language and C-Programming and DSP programming with 8051 based Embedded system + PC, and DSP- μ P based Embedded System + PC

1. 8051 Assembly Language Programming Exercises using 8051 Trainer and Pentium Class PC or VT 100/220 Terminal
2. 8051 Assembly Language Programming Exercises using 8051 Trainer, ICE-51 Module, and Pentium Class PC
3. 8051 C- programming Exercises with CYGNAL kit C8051F124DK Development Kit or equivalent , Keil C51 C –Compiler, and Pentium Class PC
4. 8051 C- programming Exercises with SPJ Systems Board - SBC 51 Single Board Computer Development System or equivalent, IDE 51 C- Compiler, and Pentium Class PC
5. DSP Programming using ADSP 2181 trainer, IDE Visual DSP ++ 3.0 C-Compiler, and Pentium Class PC
6. DSP Programming using ADSP 2181 trainer, IDE Visual DSP ++ 3.0 C-Compiler, EZICE Module and Pentium Class PC
7. DSP Programming using TMS 320C6x Development System, IDE Code Composer Studio- C Compiler, and Pentium Class PC

II CYCLE

I/O interface and I/O Programming with 8051-based System + Pentium Class PC

- 1 Interfacing Hex Key Board and Hex Display
2. Interfacing Multiplexed Hex Display
3. Interfacing a D.M. Printer
4. Interfacing Traffic Light Control Board
- 5 Interfacing Stepper Motor :
 - a)With different speeds
 - b)Rotating Clockwise and Anti clockwise Directions
 - c)With in a Given angle

2) LOGIN PAGE:


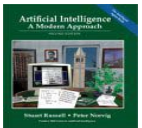



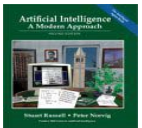



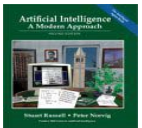


This page looks like below:

Logo	Web Site Name			
Home	Login	Registration	Catalogue	Cart
CSE ECE EEE CIVIL	<p style="text-align: center;">Login : <input type="text"/></p> <p style="text-align: center;">Password: <input type="text"/></p> <p style="text-align: center;"> <input type="button" value="Submi"/> <input type="button" value="Reset"/> </p>			

3) CATALOGUE PAGE:

The catalogue page should contain the details of all the books available in the web site in a table. The details should contain the following:

- 1) Snap shot of Cover Page.
- 2) Author Name.
- 3) Publisher.
- 4) Price.
- 5) Add to cart button.

Logo	Web Site Name																			
Home	Login	Registration	Catalogue	Cart																
CSE ECE EEE CIVIL	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%; text-align: center;">  </td> <td style="width: 35%;"> Book : XML Bible Author : Winston Publication : Wiely </td> <td style="width: 15%; text-align: center;">\$ 40.5</td> <td style="width: 35%; text-align: center;"> <input type="button" value="Add to cart"/> </td> </tr> <tr> <td style="text-align: center;">  </td> <td> Book : AI Author : S.Russel Publication : Princeton hall </td> <td style="text-align: center;">\$ 63</td> <td style="text-align: center;"> <input type="button" value="Add to cart"/> </td> </tr> <tr> <td style="text-align: center;">  </td> <td> Book : Java 2 Author : Watson Publication : BPB publications </td> <td style="text-align: center;">\$ 35.5</td> <td style="text-align: center;"> <input type="button" value="Add to cart"/> </td> </tr> <tr> <td style="text-align: center;">  </td> <td> Book : HTML in 24 hours Author : Sam Peter Publication : Sam publication </td> <td style="text-align: center;">\$ 50</td> <td style="text-align: center;"> <input type="button" value="Add to cart"/> </td> </tr> </table>					Book : XML Bible Author : Winston Publication : Wiely	\$ 40.5	<input type="button" value="Add to cart"/>		Book : AI Author : S.Russel Publication : Princeton hall	\$ 63	<input type="button" value="Add to cart"/>		Book : Java 2 Author : Watson Publication : BPB publications	\$ 35.5	<input type="button" value="Add to cart"/>		Book : HTML in 24 hours Author : Sam Peter Publication : Sam publication	\$ 50	<input type="button" value="Add to cart"/>
	Book : XML Bible Author : Winston Publication : Wiely	\$ 40.5	<input type="button" value="Add to cart"/>																	
	Book : AI Author : S.Russel Publication : Princeton hall	\$ 63	<input type="button" value="Add to cart"/>																	
	Book : Java 2 Author : Watson Publication : BPB publications	\$ 35.5	<input type="button" value="Add to cart"/>																	
	Book : HTML in 24 hours Author : Sam Peter Publication : Sam publication	\$ 50	<input type="button" value="Add to cart"/>																	

Note: Week 2 contains the remaining pages and their description.

Experiment -2:

4) CART PAGE:

The cart page contains the details about the books which are added to the cart.

The cart page should look like this:

Logo	Web Site Name			
Home	Login	Registration	Catalogue	Cart
CSE ECE EEE CIVIL	Book name	Price	Quantity	Amount
	Java 2	\$35.5	2	\$70
	XML bible	\$40.5	1	\$40.5
	Total amount -			\$130.5

5) REGISTRATION PAGE:

Create a "registration form" with the following fields

- 1) Name (Text field)
- 2) Password (password field)
- 3) E-mail id (text field)
- 4) Phone number (text field)
- 5) Sex (radio button)
- 6) Date of birth (3 select boxes)
- 7) Languages known (check boxes – English, Telugu, Hindi, Tamil)
- 8) Address (text area)

Experiment- 3:

VALIDATION:

Write *JavaScript* to validate the following fields of the above registration page.

- 1) Name (Name should contain alphabets and the length should not be less than 6 characters).
- 2) Password (Password should not be less than 6 characters length).
- 3) E-mail id (should not contain any invalid and must follow the standard pattern name@domain.com)
4. Phone number (Phone number should contain 10 digits only).

Note : You can also validate the login page with these parameters.

Experiment -4:

Design a web page using **CSS (Cascading Style Sheets)** which includes the following:

- 1) Use different font, styles: In the style definition you define how each selector should work (font,color etc.).
Then, in the body of your pages, you refer to these selectors to activate

the styles.

- 2) Set a background image for both the page and single elements on the page. You can define the background image for the page like this:

```
BODY {background-image:url(myimage.gif);}
```

- 3) Control the repetition of the image with the background-repeat property.

As background-repeat: repeat
Tiles the image until the entire page is filled, just like an ordinary

background image in plain HTML.

- 3) Define styles for links as

A:link
A:visited
A:active
A:hover

Example:

```
<style type="text/css">
A:link {text-decoration: none}
A:visited {text-decoration: none}
A:active {text-decoration: none}
A:hover {text-decoration: underline; color: red;}
</style>
```

5) Work with layers:

For example:

LAYER 1 ON TOP:

```
<div style="position:relative; font-size:50px; z-index:2;">LAYER
```

```
1</div>
```

```
<div style="position:relative; top:-50; left:5; color:red; font-size:80px; z-
index:1">LAYER 2</div>
```

LAYER 2 ON TOP:

```
<div style="position:relative; font-size:50px; z-index:3;">LAYER
```

```
1</div>
```

```
<div style="position:relative; top:-50; left:5; color:red; font-size:80px; z-
index:4">LAYER 2</div>
```

6) Add a customized cursor:

```
Selector {cursor:value}
```

Experiment -5:

Write an XML file which will display the Book information which includes the following:

- 1) Title of the book
- 2) Author Name
- 3) ISBN number
- 4) Publisher name
- 5) Edition
- 6) Price

Write a Document Type Definition (DTD) to validate the above XML file.

Display the XML file as follows.

The contents should be displayed in a table. The header of the table should be in color GREY. And the Author names column should be displayed in one color and should be capitalized and in bold. Use your own colors for remaining columns.

Use XML schemas XSL and CSS for the above purpose.

Note: Give at least for 4 books. It should be valid syntactically.

Hint: You can use some xml editors like XML-spy

Experiment -6:

VISUAL BEANS:

Create a simple visual bean with a area filled with a color.

The shape of the area depends on the property shape. If it is set to true then the shape of the area is Square and it is Circle, if it is false.

The color of the area should be changed dynamically for every mouse click. The color should also be changed if we change the color in the

“property window “.

Experiment -7:

Install TOMCAT web server and APACHE.

While installation assign port number 4040 to TOMCAT and 8080 to APACHE. Make sure that these ports are available i.e., no other process is using this port.

- 1) Access the above developed static web pages for books web site, using these servers by putting the web pages developed in week-1 and week-2 in the document root.

Access the pages by using the urls : <http://localhost:4040/rama/books.html>

(for tomcat)

<http://localhost:8080/books.html> (for Apache)

Experiment -8:

User Authentication :

Assume four users user1,user2,user3 and user4 having the passwords pwd1,pwd2,pwd3 and pwd4 respectively. Write a servlet for doing the following.

1. Create a Cookie and add these four user id's and passwords to this Cookie.
2. Read the user id and passwords entered in the Login form (week1) and authenticate with the values (user id and passwords) available in the cookies.

If he is a valid user(i.e., user-name and password match) you should welcome him by name(user-name) else you should display “ You are not an authenticated user “.

Use init-parameters to do this. Store the user-names and passwords in the webinf.xml and access them in the servlet by using the getInitParameters() method.

Experiment -9:

Install a database(Mysql or Oracle).

Create a table which should contain at least the following fields: name, password, email-id, phone number(these should hold the data from the registration form).

Practice 'JDBC' connectivity.

Write a java program/servlet/JSP to connect to that database and extract data from the tables and display them. Experiment with various SQL queries.

Insert the details of the users who register with the web site, whenever a new user clicks the submit button in the registration page (week2).

Experiment -10:

Write a JSP which does the following job:

Insert the details of the 3 or 4 users who register with the web site (week9) by using registration form. Authenticate the user when he submits the login form using the user name and password from the database (similar to week8 instead of cookies).

Experiment -11:

Create tables in the database which contain the details of items (books in our case like Book name , Price, Quantity, Amount) of each category. Modify your catalogue page (week 2)in such a way that you should connect to the database and extract data from the tables and display them in the catalogue page using JDBC.

Experiment -12:

Install a database(Mysql or Oracle).

Create a table which should contain at least the following fields: name, password, email-id, phone number(these should hold the data from the registration form).

Write a PHP to connect to that database and extract data from the tables and display them. Experiment with various MySQL queries.

Insert the details of the users who register with the web site, whenever a new user clicks the submit button in the registration page (week2).