

ACADEMIC REGULATIONS
COURSE STRUCTURE AND SYLLABI
FOR
M.TECH.
COMPUTER SCIENCE & ENGINEERING
2011-2012



COLLEGE OF ENGINEERING
(AUTONOMOUS)

GAYATRI VIDYA PARISHAD COLLEGE OF ENGINEERING
(AUTONOMOUS)
ACCREDITED BY NAAC WITH A GRADE WITH A CGPA OF 3.47/4.00
AFFILIATED TO JNTU KAKINADA
MADHURAWADA, VISAKHAPATNAM 530048

Vision

*To evolve into and sustain as a Centre of
Excellence in Technological Education
and Research with a holistic approach.*

Mission

To produce high quality engineering graduates with the requisite theoretical and practical knowledge and social awareness to be able to contribute effectively to the progress of the society through their chosen field of endeavor.

To undertake Research & Development, and extension activities in the fields of Science and Engineering in areas of relevance for immediate application as well as for strengthening or establishing fundamental knowledge.

FOREWORD

It is two years since the G.V.P College of Engineering has become Autonomous with the appreciation and support of erstwhile JNTU and the fast growing new JNTU-K. The college is progressing well with its programmes and procedures drawing more and more accolades from its sister autonomous colleges and higher authorities. The student community, also could adjust well to the new system without any acrimony.

The College is enriched with the experience of running the first batch of Post-graduate programmes under Autonomous stream. It is a moment of pride and achievement that the first Autonomous batch of M.Tech in some branches left the college to the satisfaction of all concerned including firms visited the campus for placements. It is with this experience the curricula of some of the programmes have been reviewed and revamped for better performance and results.

Another larger than canvas picture is foreseen for the programmes wherein the college is getting the funds through TEQIP - II for up-scaling the PG education and research under sub- component 1.2.

The help extended by the first set of Boards of Studies, Academic council and Governing Body in designing the system and its mechanism is overwhelming. The encouragement given by the affiliating JNTU-K has left no task insurmountable.

The new BOS and Academic Council are expected to help in this direction as the term of the present ones expired.

Principal

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

The M.Tech Degree of JNTU-KAKINADA shall be recommended to be conferred on candidates who are admitted to the program and fulfill all the requirements for the award of the Degree.

1.0 ELGIBILITY FOR ADMISSION:

Admission to the above program shall be made subject to the eligibility, qualifications and specialization as per the guidelines prescribed by the APSCHE and AICTE from time to time.

2.0 AWARD OF M.TECH. DEGREE:

- a. 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.
- b. 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.
- c. The duration of each semester will normally be 20 weeks with 5 days a week. A working day shall have 7 periods each of 50minutes.

3.0 COURSES OF STUDY:

M.TECH. COURSES	INTAKE
Chemical Engineering	18
Computer Science and Engineering	18
CAD/CAM	18
Infrastructural Engineering and Management	18
Structural Engineering	18
Power System Control and Automation	18
Embedded Systems & VLSI Design	18
Communications & Signal Processing	18
Software Engineering	18

4.0 ATTENDANCE:

The attendance shall be considered subject wise.

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

- c. Shortage of attendance below 65% shall in no case be condoned.
- d . A student who gets less than 65% attendance in a maximum of two subjects in any semester shall not be permitted to take the end- semester examination in which he/she falls short. His/her registration for those subjects will be treated as cancelled. The student should re-register and repeat those subjects as and when offered next.
- e . If a student gets less than 65% attendance in more than two subjects in any semester he/she shall be detained and has to repeat the entire semester.
- f. 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 100 marks for each theory subject and 100 marks for each practical, on the basis of Internal Evaluation and End Semester Examination.

- a. 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.
- b. 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. One part of the internal evaluation shall be made based on the average of the marks secured in the two Mid–Term Examinations of 30 each conducted one in the middle of the Semester and the

other immediately after the completion of instruction. Each mid-term examination shall be conducted for a duration of 120 minutes with 4 questions without any choice. The remaining 10 marks are awarded through an average of continuous evaluation of assignments / seminars / any other method, as notified by the teacher at the beginning of the semester.

- c. For Practical subjects, 50 marks shall be awarded based on the performance in the End Semester Examinations, 50 marks shall be awarded based on the day-to-day performance as Internal marks. A candidate has to secure a minimum of 50% in the external examination and has to secure a minimum of 50% on the aggregate to be declared successful.
- d. There shall be a seminar presentation during III semester. For seminar, a student under the supervision of a faculty member, shall collect the literature on a topic and critically review the literature and submit it to the Department in a report form and shall make an oral presentation before the Departmental Committee. The Departmental Committee consists of the Head of the Department, supervisor and two other senior faculty members of the department. For Seminar there will be only internal evaluation of 50 marks. A candidate has to secure a minimum of 50% to be declared successful.
- e. For Seminar in I, II Semesters in case of the course structure of having 5 Theory + 2 Labs. + 1 Seminar, a student has to deliver a seminar talk in each of the subjects in that semester which shall be evaluated for 10 marks each and average marks allotted shall be considered. A letter grade from A to C corresponding to the marks allotted may be awarded for the

two credits so as to keep the existing structure and evaluation undisturbed.

A – Excellent	(average marks ≥ 8)
B – Good	($6 \leq$ average marks < 8)
C – Satisfactory	($5 \leq$ average marks < 6)

If a satisfactory grade is not secured, one has to repeat in the following semester.

- f. In case the candidate does not secure the minimum academic requirement in any subject (as specified in 5.0 a, c) 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 subject. In such a case the 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). In the event of re-registration, the internal marks and end examination marks obtained in the previous attempt are nullified.

- g. In case the candidates secure 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 for the subject(s) when next offered.
- h. Laboratory examination for M.Tech subjects must be conducted with two Examiners, one of them being Laboratory Class Teacher and second examiner shall be other than Laboratory Teacher.

6.0 EVALUATION OF PROJECT / DISSERTATION WORK:

Every candidate shall be required to submit the thesis or dissertation after taking up a topic approved by the Departmental Research Committee (DRC).

- a. A Departmental Research Committee (DRC) shall be constituted with the Head of the Department as the chairman and two senior faculty as members to oversee the proceedings of the project work from allotment to submission.
- b. A Central Research Committee (CRC) shall be constituted with a Senior Professor as chair person, Heads of all the Departments which are offering the M.Tech programs and two other senior faculty members.
- c. 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.)
- d. After satisfying 6.0 c, a candidate has to submit, in consultation with his project supervisor, the title, objective and plan of action of his project work to the DRC for its approval. Only after obtaining the approval of DRC the student can initiate the Project work
- e. If a candidate wishes to change his supervisor or topic of the project he can do so with approval of DRC. However, the Departmental Project Review Committee shall examine whether the change of topic/supervisor leads to a major change in his initial plans of project proposal. If so, his date of registration for the Project work shall start from the date of change of Supervisor or topic as the case may be whichever earlier.

- f. A candidate shall submit and present the status report in two stages at least with a gap of 3 months between them after satisfying 6.0 d.
- g. 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 shall be permitted to submit his dissertation only after successful completion of all theory and practical subject with the approval of CRC but not earlier than 40 weeks from the date of registration of the project work. For the approval by CRC the candidate shall submit the draft copy of the thesis to the Principal through the concerned Head of the Department and shall make an oral presentation before the CRC.
- h. Three copies of the dissertation certified by the supervisor shall be submitted to the College after approval by the CRC.
- i. The dissertation shall be adjudicated by one examiner selected by the Principal. For this HOD shall submit in consultation with the supervisor a panel of 5 examiners, who are experienced in that field.
- j. If the report of the examiner is not favorable, the candidate shall revise and resubmit the dissertation, in a time frame as prescribed by the CRC. If the report of the examiner is unfavorable again, the dissertation shall be summarily rejected then the candidate shall change the topic of the Project and option shall be given to change the supervisor also.
- k. 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 dissertation.

The Board shall jointly report candidate's work as:

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

7.0 AWARD OF DEGREE AND CLASS :

A candidate shall be eligible for the respective degree if he satisfies the minimum academic requirements in every subject and secures satisfactory or higher grade report on his dissertation and viva-voce.

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 three classes.

% of Marks secured	Class Awarded
70% and above	First Class with Distinction
60% and above but less than 70%	First Class
50% and above but less than 60%	Second Class

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

The grade of the dissertation shall also be mentioned in the marks memorandum.

8.0 WITHHOLDING OF RESULTS:

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

9.0 TRANSITORY REGULATIONS:

A candidate who has discontinued or has been detained for want of attendance or who has failed after having studied the subject is eligible for admission to the same or equivalent subject(s) as and when subject(s) is/are offered, subject to 6.0 e and 2.0

10.0 GENERAL

1. The academic regulations should be read as a whole for purpose of any interpretation.
2. In case of any doubt or ambiguity in the interpretation of the above rules, the decision of the Chairman Academic Council is final.
3. The College may change or amend the academic regulations and syllabus at any time and the changes amendments made shall be applicable to all the students with effect from the date notified by the College.
4. Wherever the word he, him or his occur, it will also include she, hers.

COURSE STRUCTURE

I SEMESTER

COURSE CODE	THEORY/LAB	L	P	C
11CS2101	Advanced Data Structures and Algorithms	4	-	4
11CS2102	Computer Organization and Architecture	4	-	4
11CS2103	Advanced Operating Systems	4	-	4
11CS2104	Advanced Database Management Systems	4	-	4
	ELECTIVE – 1			
11CS2105	Middleware Technologies	4	-	4
11CS2106	Software Project Management	4	-	4
11CS2107	Software Threading: Programming on Multi-Core Processors	4	-	4
11CS2108	<i>Advanced Data Structures Lab (Through C++)</i>	-	4	2
11CS2109	<i>Operating Systems & DBMS Lab</i>	-	4	2
11CS21S1	<i>Seminar</i>	-	-	2
	TOTAL	20	8	26

II SEMESTER

COURSE CODE	THEORY/LAB	L	P	C
11CS2110	Automata & Compiler Design	4	-	4
11CS2111	Advanced Software Engineering	4	-	4
11CS2112	Data Warehousing and Data Mining	4	-	4
11CS2113	Network Security and Cryptography	4	-	4
	ELECTIVE -2			
11CS2114	Embedded Systems	4	-	4
11CS2115	Storage Area Networks and Management	4	-	4
11CS2116	Programming on Parallel & Distributed Computing	4	-	4
11CS2117	<i>Network security and cryptography lab</i>	-	4	2
11CS2118	<i>Data Warehousing and Data Mining Lab</i>	-	4	2
11CS21S2	<i>Seminar</i>	-	-	2
	TOTAL	20	8	26

III SEMESTER

COURSE CODE	THEORY/LAB	L	P	C
COMMENCEMENT OF PROJECT WORK				
11CS21S3	SEMINAR	-	-	2

IV SEMESTER

COURSE CODE	THEORY/LAB	L	P	C
11CS2119	PROJECT WORK DISSERTATION / THESIS EXCELLENT/ GOOD/SATISFACTORY/NOT- SATISFACTORY	-	-	56

ADVANCED DATA STRUCTURES AND ALGORITHMS**Course code:11CS2101****L P C**
4 0 4

UNIT – I: Lists, Stacks, Queues and Trees Lists, Stacks and Queues: Abstract Data Types (ADTs), The List ADT, Vector and list in the STL, Implementation of vector, Implementation of list, The Stack ADT, The Queue ADT.

Trees: The Search Tree ADT – Binary Search Trees, AVL Trees, Splay Trees, B-Trees

UNIT - II : Hashing and Priority Queues Hashing: General idea, Hash Function, Separate Chaining, Hash Tables Without Linked Lists, Rehashing, Extendible Hashing Priority Queues: Implementations, Binary Heap, Applications of Priority Queues, d -Heaps, Leftist Heaps, Skew Heaps, Binomial Queues.

UNIT - III : Sorting: A Lower Bound for Simple sorting Algorithms, Shellsort, Heapsort, Mergesort, Quicksort, Indirect Sorting, A General Lower Bound for sorting, Bucket Sort, External Sorting.

UNIT - IV : The Disjoint Set Class Equivalence Relations, The Dynamic Equivalence Problem, Basic Data Structure, Smart Union Algorithms, Path Compression, Worst Case of Union-by-Rank and Path Compression, An Application.

UNIT - V : Graph Algorithms Definitions, Topological sort, Shortest-Path Algorithms, Network Flow Problems, Minimum Spanning Tree, Applications of Depth-First Search, introduction to NP-Completeness.

UNIT - VI : Algorithm Design Techniques: Greedy Algorithms, Divide and Conquer, Dynamic Programming, Randomized Algorithms, Backtracking Algorithms.

UNIT – VII: Amortized Analysis: An Unrelated Puzzle, Binomial Queues, Skew Heaps, Fibonacci Heaps, Splay Trees.

UNIT – VIII : Advanced Data Structures and Implementation :Top-Down Splay Trees, Red-Black Trees, Deterministic Skip lists, AA-Trees, Treaps, k -d Trees, Pairing Heaps .

Text Books:

1. Mark Allen Weiss : Data Structures and Algorithm Analysis in C++, 3rd Edition, Pearson, 2007.

Reference Books:

1. Sartaj Sahni : Data Structures Algorithms and Applications in C++, 2nd Edition, Universities Press, 2007.
2. Ellis Horowitz, Sartaj Sahni, S. Rajasekharan : Fundamentals of Computer Algorithms, 2nd Edition, Universities Press, 2008.
3. A. V. Aho, J. D. Ullman : Data Structures and Algorithms , 1st Edition, Pearson Publication, 2003.
4. Adam Drozdek : Data Structures and Algorithms in C++, 3rd Edition, Cengage, 2006.
5. Horowitz Sahni, Mehta : Fundamentals of Data Structures in C++, 2nd Edition, University Press 2007.

COMPUTER ORGANIZATION AND ARCHITECTURE**Course Code:11CS2102****L P C**
4 0 4

Aim: To give an exposure on different concepts of Computer Architecture.

Ojective: The students will be able to learn Computer structure, parallel processing, pipelining, Multicomputers and multiprocessors. Also new concept of Multicore computer is introduced in this syllabus. Students will learn current present processors' architecture like Pentium, ARM.

UNIT I: Overview of computer organization: Introduction, Programmer's view, Architect's view, Implementer's view, Te processor, Memory, Input/output. (Text Book 2)

UNIT II : Instruction Sets: Machine Instruction Characteristics, Types of Operands, Intel X86 and ARM Data Types, Types of Operations, Intel X86 and ARM Operation Types, Addressing, x86 and ARM Addressing Modes, Instruction Formats, x856 and ARM Instruction Formats, Assembly Language. (Text Book1)

UNIT III: Processor Structure and Function: Processor Organization, Register Organization, The Instruction Cycle, The x86 Processor Family, The ARM Processor. (Text Book1)
RISC Processors: Introduction, Evolution of CISC Processors, RISC Design Principles. (Text Book2)

UNIT IV: Cache Memory: Introduction, How Cache Memory Works, Why Cache Memory Works, Cache Design Basics, Mapping Function, Replacement Policies, Write Policies, Space Overhead, Mapping Examples, Types of Cache Misses, Types of Caches.

Input/Output Organization: Accessing I/O Devices, I/O data Transfer, External Interface, Universal Serial Bus. (Text Book2)

UNIT V: Pipelining and Vector Processing: Basic Concepts, Handling Resource Conflicts, Data Hazards, Handling Branches, Performance Enhancements, Pentium Implementation, Vector Processors, Performance. (Text Book2)

UNIT VI: Instruction-Level Parallelism and Superscalar Processors: Overview, Design Issues, Pentium4, ARM Cortex-A8. (Text Book1)

UNIT VII: Parallel Processing: The Use of Multiple Processors, Symmetric MultiProcessors, Cache Coherence and the MESI protocol, Multithreading and Chip MultiProcessors, Clusters, NonUniform Memory Access Computers, Vector Computation. (Text Book1)

UNIT VII: Multicore Computers: Hardware Performance Issues, Software Performance Issues, Multicore Organization, Intel x86 Multicore Organization, ARM11 MPCore. (Text Book1)

Text Books:

1. William Stallings: Computer Organization and Architecture Designing for Performance, 8th Edition, Pearson Education, 2010.
2. Sivarama P.Dandamudi: Fundamentals of Computer Organization and Design, Springer International Edition, 2009.

Reference Books:

1. Moris Mono : Computer System Architecture, 3rd Edition, Pearson / PHI, 1993.
2. Hamacher, Vranesic, Zaky : Computer Organization, 5th Edition, TMH, 2002.

3. John D. Carpinelli : Computer systems organization and architecture, 1st Edition, Pearson, 2001.
4. Pal Chowdary : Computer organization and Design, 2nd Edition, PHI, 2004.
5. Naresh Jotwani : Computer system organization, 1st Edition, TMH, 2009.

ADVANCED OPERATING SYSTEMS**Course Code:11CS2103**

L	P	C
4	0	4

UNIT – I: Operating System Introduction, Structures-Simple Batch, Multi Programmed, time-shared, Personal Computer, Parallel, Distributed Systems, Real-Time Systems, System components, Operating-System Services, System Calls, Virtual Machines, System Design and Implementation, Microsoft Windows Overview, Modern UNIX Systems Overview, Linux Overview

UNIT – II: Process and CPU Scheduling - Process concepts and scheduling, Operation on processes, Cooperating Processes, Threads, and Inter process Communication Scheduling Criteria, Scheduling Algorithm, Multiple -Processor Scheduling, Real-Time Scheduling, Windows XP Thread and SMP Management , Linux Process and thread Management , Linux Scheduling , Windows XP Scheduling.

UNIT – III: Memory Management and Virtual Memory, Logical versus Physical Address Space, Swapping, Contiguous Allocation, Paging, Segmentation, Segmentation with Paging. Demand Paging, Performance of Demand Paging, Page Replacement, Page Replacement Algorithm, Allocation of Frames, Thrashing, Linux Memory Management, Windows XP Memory Management.

UNIT – IV: File System Interface and Implementation -Access methods, Directory Structure, Protection, File System Structure, Allocation methods, Free space Management, Directory Management, Directory Implementation, Efficiency and Performance. Deadlocks – System Model, Dead locks Characterization, Methods for Handling Dead locks, Dead lock, Prevention, Deadlock Avoidance, Deadlock Detection, and Recovery from Deadlock.

UNIT – V: Process Management and Synchronization - The Critical Section Problem, Synchronization Hardware, Semaphores, and Classical Problems of Synchronization, Critical Regions, Monitors, UNIX Concurrency Mechanism, Linux Kernel Concurrency Mechanism, Windows XP Concurrency Mechanism.

UNIT – VI : Introduction to Distributed systems: Goals of distributed system, hardware and software concepts, design issues. Communication in Distributed systems: remote procedure call and group communication.

UNIT – VII : Synchronization in Distributed systems: Clock synchronization, Mutual exclusion, Election algorithms, the Bully algorithm, a ring algorithm, atomic transactions, Deadlocks: deadlock in distributed systems, Distributed deadlock prevention, and distributed dead lock detection.

UNIT – VIII : Shared Memory, page based Distributed shared memory, shared variable distributed shared memory.

Text Books:

- 1 Abraham Silberchatz, Peter B. Galvin, Greg Gagne : Operating System Principles, 7th Edition, John Wiley, 2006. (I, II, III, IV, V).
- 2 Andrew. S.Tanenbaum : Distributed Operating Systems, 1st Edition, PHI, 1995. (VI, VII, VIII).

Reference Books:

1. Stallings : Operating Systems – Internals and Design Principles, 6th Edition, Pearson Education/PHI, 2009.
12. Charles Crowley : Operating System - A Design Oriented Approach, 1st Edition, TMH, 1998.
3. Andrew S Tanenbaum : Modern Operating Systems, , 3rd Edition, Pearson/PHI,2008.

4. Dhamdhere : Operating Systems – A concept based approach, 2nd Edition, TMH, 2006.
5. Daniel P Bovet and Marco Cesati : Understanding the Linux Kernel, 3rd Edition, O'Reilly, 2005.
6. Distributed Operating Systems – Concepts and Design – Pradeep K. Sinha, IEEE 1997.

ADVANCED DATABASE MANAGEMENT SYSTEMS**Course Code:10CS2104****L P C**
4 0 4

UNIT-I : Introduction of Object Database Systems, Structured Data types, operations on structured data, Encapsulation and ADTS, Inheritance Objects, IDs, and reference types, implementation of various **complex** types in SQL (chapter 23 from text book 1) .

UNIT-II: Database design for ORDBMS, ORDBMS implementation and challenges, OODBMS, comparison of RDBMS, OODBMS and ORDBMS. (chapter 23 from text book 1) .

UNIT-III: Introduction to Parallel databases, architectures for parallel databases, Parallel Query Evaluation – data partitioning and parallelising sequential operator evaluation code, Parallelising individual operations, and parallel Query optimization (chapter 22 from text book 1) .

UNIT-IV: Introduction to distributed databases; features of distributed databases vs centralized databases, Why distributed databases, DDBMS, levels of transparency- reference architecture for DDB, types of data fragmentation, distribution transparency for read-only and update applications, distributed database access primitives, Integrity Constraints in Distributed databases (chapter 1.1,1.2, Chapter 3 from Text book 2) .

UNIT-V: Distributed database design: framework for distributed database design, the design of database fragmentation, allocation of fragments; (chapter 4 from text book 2) .

UNIT-VI: Distributed Query processing: Equivalence of transformations for queries, transforming global queries into fragment

queries, distributed grouping and aggregation functions, parametric queries, A framework for query optimization, join queries and general queries. non-join queries in a distributed DBMS, joins in a distributed DBMS, cost based query optimization. (Chapter 5 and 6 from text book 2) .

UNIT-VII: Management of distributed transactions - framework transaction management,. atomicity of distributed transactions, concurrency control for Distributed transactions, architectural aspects of distributed transactions , foundation of distributed concurrency control, distributed deadlocks, CC based on time stamps, optimistic methods for concurrency control. (Chapters 7 and 8 from text book 2).

UNIT-VIII: Information retrieval and XML data: DBMS Vs IR systems, Introduction to Information retrieval, Indexing for text search, web search engine, managing text in a DBMS, a data model for XML, Querying XML data, efficient evaluation of XML queries. (Chapter 27 from text book 1).

Text Books:

1. Raghurama krishnan and Johannes Gehrke: Database Management Systems, 3rd Edition, TMH, 2006.
2. S Ceri and G Pelagatti: Distributed databases principles and systems, 1st Edition, TMH, 2008.

Reference Books:

1. Silberschatz, Korth: Database System Concepts, 6th Edition, TMH, 2010.
2. Elmasri R, Navathe S B, Somayajulu D V L N, and Gupta S K: Fundamentals of Database Systems, 5th Edition, Pearson Education, 2009.
3. C. J. Date: Introduction to Database Systems, 8th Edition, Pearson Education, 2009.

MIDDLEWARE TECHNOLOGIES**Elective-1****Course code:11CS2105**

L	P	C
4	0	4

UNIT – I : Introduction to client server computing: Evolution of corporate computing models from centralized to distributed computing, client server models. Benefits of client server computing, pitfalls of client server programming.

UNIT – II : CORBA with Java: Distributed programming with Java RMI; Overview of CORBA, CORBA IDL, Client/server programming with CORBA & Java.

UNIT - III : XML Technology XML – Name Spaces – Structuring With Schemas and DTD – Presentation Techniques – Transformation – XML Infrastructure.

UNIT - IV : SOAP: Overview of SOAP – HTTP – XML-RPC – SOAP: Protocol – Message Structure –Intermediaries – Actors – Design Patterns And Faults – SOAP With Attachments.

UNIT- V: Web Services Overview – Architecture – Key Technologies - UDDI – WSDL – eb XML – SOAP and Web Services In E-Com – Overview Of .NET And J2EE.

UNIT - VI : Agent and User Experience: Interacting with Agents - Agent From Direct Manipulation to Delegation - Interface Agent Metaphor with Character - Designing Agents - Direct Manipulation versus Agent Path to Predictable.

UNIT - VII : Agent Communication and Collaboration: Overview of Agent Oriented Programming - Agent Communication Language - Agent Based Framework of Interoperability - Agents for Information Gathering - Open Agent Architecture - Communicative Action for Artificial Agent.

UNIT - VIII : Agent Architecture Agents for Information Gathering - Open Agent Architecture – Communicative Action for Artificial Agent.

Text Books:

1. Frank. P. Coyle : XML, Web Services and The Data Revolution, 1st Edition, Pearson Education, 2002.
2. Jeffrey M. Bradshaw : Software Agents, 1st Edition, PHI, 2010.

Reference Books:

1. M.L. Liu : Distributed Computing, Principles and applications, 1st Edition, Pearson Education, 2008.
2. Ramesh Nagappan, Robert Skoczylas and Rima Patel Sriganesh: Developing Java Web Services, 1st Edition, Willey Publishing, 2004.

**SOFTWARE PROJECT MANAGEMENT
(Elective-1)****Course code:11CS2106****L P C
4 0 4**

UNIT – I :Conventional Software Management: The waterfall model, conventional software Management performance. Evolution of Software Economics : Software Economics, pragmatic software cost estimation.

UNIT – II : Improving Software Economics : Reducing Software product size, improving software processes, improving team effectiveness, improving automation, Achieving required quality, peer inspections.

UNIT – III : The old way and the new : The principles of conventional software engineering, principles of modern software management, transitioning to an iterative process. Life cycle phases : Engineering and production stages, inception, Elaboration, construction, transition phases.

UNIT – IV : Artifacts of the process : The artifact sets, Management artifacts, Engineering artifacts, programmatic artifacts. Model based software architectures : A Management perspective and technical perspective.

UNIT – V : Flows of the process : Software process workflows, Inter trans workflows.Checkpoints of the Process : Major Mile Stones, Minor Milestones, Periodic status assessments. Interactive Process Planning : Work breakdown structures, planning guidelines, cost and schedule estimating, Interaction planning process, Pragmatic planning.

UNIT – VI :Project Organizations and Responsibilities : Line-of-Business Organizations, Project Organizations, evolution of Organizations. Process Automation : Automation Building Blocks, The Project Environment.

UNIT – VII : Project Control and Process instrumentation : The server care Metrics, Management indicators, quality indicators, life cycle expectations, pragmatic Software Metrics, Metrics automation. Tailoring the Process : Process dicriminants, Example.

UNIT – VIII : Future Software Project Management : Modern Project Profiles Next generation Software economics, modern Process transitions. Case Study : The Command Center Processing and Display System-Replacement(CCPDS-R).

Text Book :

1. Walker Rayce : Software Project Management A Unified Framework, 1st Edition, Pearson Education, 2005.

Reference Books:

1. Richard H.Thayer : Software Engineering Project Management, 2nd Edition, IEEE Computer Society, 1997.
2. Shere K.D. : Software Engineering and Management, 1st Edition, Prentice Hall, 1988.

SOFTWARE THREADING: PROGRAMMING ON MULTI-CORE PROCESSORS

(Elective-1)

Course code:11CS2107

L	P	C
4	0	4

UNIT-I: Overview of Multi- core Processors Platforms

Introduction to Multi-core Architecture - An overview of Parallel Computing Platforms (*SIMD & MIMD systems, an Overview of Single-Core, Multi-Processor, Multi-Core Architectures*); Concurrency on Software; Fundamental concepts of Parallel Processing; Programming Paradigms, Understanding (*Performance and Scalability analysis; Amdahl's Law, and Gustafson Law*); An Overview of Hyper-threading technology- (*Multi-threading issues*) on Multi-Core Processor Systems.

UNIT-II: Multi- core Processors – Overview of Software Threading Multi-Core - An Overview of Threading: Multi-Core - An overview of Caches, Virtual Memory; NUMA Programming; System View of Threading (Threading inside, the OS, Threading inside the Hardware, Threading above the Operating System); Programming Models and Threading; Virtualization (*Virtual Machines, System Virtualization, VMs-Threads & Scheduling*); Prefetching (*Hardware & Software*).

UNIT III : Multi- core Processors Overview of Threading API

Thread Basics; Why Threads?; Threading APIs and Parallel Programming Constructs (*Synchronization, Critical Sections, Deadlocks, Synchronization Primitives, Message Passing features, Key features of Threaded APIs*); An overview of Threading APIs for Microsoft Windows; An overview of Threading APIs for Microsoft .NET framework.

UNIT IV :Shared Memory-Linux :POSIX threads & Intel TBB: Programming Shared Address Space Platforms; An Overview of POSIX Threads; Key features of POSIX Threads (*Creating threads, Managing threads, Thread synchronization, Signaling*); Multi-thread Optimization; Performance Issues based on POSIX threads for Scientific Computations; Threading Building blocks; An Overview of Memory Allocators, An Overview of Intel Threading building blocks (*Intel TBB*); Intel TBB Containers; Intel TBB-Task Scheduling; Intel TBB Threads-Mutual Exclusion; Performance Issues for Scientific Computations - POSIX threads - Intel TBB; An Overview of Compiler Optimization techniques on Multi-Core Processors.

UNIT-V : (Shared Memory : Open MP): Programming Shared Address Space Platforms; An Overview of OpenMP – A standard for Directive Parallel Programming; The OpenMP programming Model (*Concurrent Tasks, Synchronization Constructs, Data Handling*); Open libraries; OpenMP-Environment Variables; Explicit threads versus OpenMP based Programming on Multi-Core processors; Performance Issues for Scientific Computations - OpenMP; An Overview of Compiler Optimization techniques on Multi-Core Processors.

UNIT-VI: Message Passing (MPI- 1.0) on multi core Processors: Programming Using Message-Passing Paradigm – Principles of Message Passing Programming; An overview of the Message Passing Building blocks (*Sending and Receiving Operations; Blocking & Non-Blocking Message Passing Operations*); an Overview of Message Passing Interface (MPI; - MPI 1.0); Point-Point Message Passing Communication Library calls; Collective communication and computation library calls; MPI Groups & Communicators; Topologies; Programming and Performance issues for Numerical Computations.

UNIT-VII: Multi- core Processors - Tuning & Performance

Tuning and Performance on Multi-Core Processors (Compilers-General Tuning and Optimization; Compiler Optimization Switches, Using Mathematical Libraries); Application and System Benchmarks; Common Parallel Programming Problems - (OpenMP, Intel TBB, MPI) on Multi-Core Processors; Tuning for I/O- Multi-threading on Multi-Core processors.

UNIT-VIII: Multi- core Processors software tools: An Overview of Software tools on Multi Core Processors -Thread Checker, Thread Analyzer, Thread Profiler, Thread Performance Analyzer, Open Source Tools; Memory Performance Tools; An overview of Multi-threaded debugging techniques; Open Source tools on Multi-Core Processors.

Text books

1. Shameem Akhter, Jason Roberts: Multi-Core Programming – Increasing Performance through Software Multi-threading, Intel PRESS, Intel Corporation, 2006.
2. Grama Ananth, Anshul Gupta, George Karypis and Vipin Kumar: Introduction to Parallel computing, Boston, MA : Addison-Wesley, 2003.
3. Michael J. Quinn : Parallel Programming in C with **MPI** and **OpenMP** McGraw-Hill International Editions, Computer Science Series, McGraw-Hill, Inc., 2004.

Reference Books:

- 1.Pacheco S. Peter: Parallel Programming with **MPI**, University of Sanfrancisco,Morgan Kaufman Publishers,Inc., Sanfrancisco, California, 1992.
- 2.Chandra, Rohit Murgan, Leonardo Dagum, Dave Kohr, Dror Maydan, Jeff McDonald, and Ramesh Menon: Parallel

- Programming in **OpenMP** San Francisco Moraan Kaufmann, 2001.
- 3 James Reinders: Intel Threading Building Blocks, O'REILLY series, 2007.
 - 4 Andrews, Grogory R: Foundations of Multithreaded, Parallel, and Distributed Progmaming, Boston, MA: Addison-Wesley, 2000.
 - 5 Butenhof, David R: Programming with **POSIX Threads**, Boston, MA : Addison Wesley Professional: 1997.
 - 6 Marc Snir, Steve Otto, Steyen Huss-Lederman, David Walker and Jack Dongarra: MPI-The Complete Reference: Volume 1, 2nd Edition, The **MPI Core**, 1998.
 - 7 William Gropp, Steven Huss-Lederman, Andrew Lumsdaine, Ewing Lusk, Bill Nitzberg, William Saphir and Marc Snir: MPI-The Complete Reference: Volume 2, 2nd Edition, The **MPI-2 Extensions**, 1998.

ADVANCED DATA STRUCTURES LAB**Course code:11CS2108****L P C**
0 4 2

Implement the following using C/C++/Java

- 1) Write a program to perform the following operations on singly linked list.
 - i) Creation ii) Insertion iii) Deletion iv) Traversal.
- 2) Write a program to perform the following operations on doubly linked list.
 - i) Creation ii) Insertion iii) Deletion iv) Traversal in both ways
- 3) Write a program that implements stack (its operations) using
 - i) Arrays ii) linked list
- 4) Write a programs that implements Queue (its operations) using
 - i) Arrays ii) linked list
- 5) Write C program that implements the Quick sort method to sort a given list of integers in ascending order.
- 6) Write C program that implement the Merge sort method to sort a given list of integers in ascending order.
- 7) Write C program that implement the SHELL sort method to sort a given list of integers in ascending order.
- 8) Write a program to perform the following:
 - i) Creating a Binary Tree of integers
 - ii) Traversing the above binary tree in preorder, in order and postorder.

- 9) Write a C program to perform the following:
 - i) Creating a AVL Tree of integers
 - ii) Traversing the above binary tree in preorder, inorder and postorder.
- 10) Write a C program that uses functions to perform the following:
 - i) Creating a SplayTree of integers.
 - ii) Traversing the above binary tree in preorder, in order and postorder.
- 11) Write a C program to perform the following:
 - i) Creating a B-Tree of integers.
 - ii) Traversing the above binary tree in preorder, in order and postorder.
- 12) Write a program that implements Kruskal's algorithm using a disjoint set data structure. The program takes as input a file (data.txt), in which each line either represents a vertex or an edge. For the edge lines, the first integer on that line representing the starting vertex, the second the ending vertex, and the third the weight of the edge. Use this file to construct, line by line, the graph upon which Kruskal's algorithm will be run (do NOT hardcode this graph!).
- 13) Write a program to simulate various graph traversing algorithms.
- 14) Write a program to find the minimal spanning tree of a graph using the Prim's algorithm. The program should be able to read in the weight matrix of a graph and produce the minimal spanning tree. Generate weight matrices (using a random number generator) with a large number of nodes and estimate the time complexity of the algorithm.

- 15) Write a program to find the closest pair of points using a divide and conquer strategy. Use the random number generator to generate a large number of points in a unit square as input to the algorithm. Test the correctness of the algorithm by using a brute force method.
- 16) Use dynamic programming to find the optimal binary search tree for a given set of numbers together with their probabilities. Remember that the numbers may be generated in any order, so, a presorting step is also required.

OPERATING SYSTEMS AND DBMS LAB**Course code:11CS2109****L P C**
0 4 2

Aim: To provide necessary operating system concepts like Disk scheduling, paging, deadlock avoidance and concurrency techniques.

Objective :

To provide an understanding of the design aspects of operating system.

Recommended Systems/Software Requirements:

· Intel based desktop PC with minimum of 166 MHZ or faster processor with atleast 64 MB RAM and 100 MB free disk space
JDK kit

Part – A

1. Simulate the following unix commands:
 - a)mv
 - b)cp
 - c)ls(Use system calls)
2. Simulate the following CPU scheduling algorithms
 - a) Round Robin b) SJF c) FCFS d) Priority
3. Simulate all file allocation strategies
 - a) Sequential b) Indexed c) Linked
4. Simulate Bankers Algorithm for Dead Lock Avoidance
5. Simulate Bankers Algorithm for Dead Lock Prevention
6. Simulate all page replacement algorithms
 - a) FIFO b) LRU c) LFU

Part-B

Aim: To teach the student logical database design and querying the database using SQL & PL/SQL.

Objective: Student will get knowledge of creating and maintaining tables of a database using SQL, handling all types of queries, and writing all kinds of programming scripts in PL/SQL, transaction managements, creation of stored procedures, functions, cursors & triggers.

Recommended Systems/Software Requirements:

- Intel based desktop PC

- Mysql /Oracle latest version Recommended

1) Creation, altering and dropping of tables and inserting rows into a table (use constraints while creating tables) examples using SELECT command.

2) Queries (along with sub Queries) using ANY, ALL, IN, EXISTS, NOTEXISTS, UNION, INTERSET, Constraints.

Example:- Select the roll number and name of the student who secured fourth rank in the class.

3) Queries using Aggregate functions (COUNT, SUM, AVG, MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views.

4) Queries using Conversion functions (to_char, to_number and to_date), string functions

(Concatenation, lpad, rpad, ltrim, rtrim, lower, upper, initcap, length, substr and instr), date

functions (Sysdate, next_day, add_months, last_day, months_between, least, greatest, trunc, round, to_char, to_date)

- 5) i) Creation of simple PL/SQL program which includes declaration section, executable section and exception –Handling section (Ex. Student marks can be selected from the table and printed for those who secured first class and an exception can be raised if no records were found)
- ii) Insert data into student table and use COMMIT, ROLLBACK and SAVEPOINT in PL/SQL block.
- 6) Develop a program that includes the features NESTED IF, CASE and CASE expression. The program can be extended using the NULLIF and COALESCE functions.
- 7) Program development using WHILE LOOPS, numeric FOR LOOPS, nested loops using ERROR Handling, BUILT –IN Exceptions, USE defined Exceptions, RAISE- APPLICATION ERROR.
- 8) Programs development using creation of procedures, passing parameters IN and OUT of PROCEDURES.
- 9) Program development using creation of stored functions, invoke functions in SQL Statements and write complex functions.
- 10) Program development using creation of package specification, package bodies, private objects, package variables and cursors and calling stored packages.
- 11) Develop programs using features parameters in a CURSOR, FOR UPDATE CURSOR, WHERE CURRENT of clause and CURSOR variables.
- 12) Develop Programs using BEFORE and AFTER Triggers, Row and Statement Triggers and INSTEAD OF Triggers.

AUTOMATA & COMPILER DESIGN**Course code:11CS2110****L P C****4 0 4**

UNIT-I : Formal Language and Regular Expressions : Languages, Definition Languages regular expressions, Finite Automata – DFA, NFA. Conversion of regular expression to NFA, NFA to DFA. Applications of Finite Automata to lexical analysis, lex tools.

UNIT-II : Context Free grammars and parsing : Context free grammars, derivation, parse trees, ambiguity LL(K) grammars and LL(1) parsing .

UNIT-III : Bottom up parsing handle pruning LR Grammar Parsing, LALR parsing, parsing ambiguous grammars, YACC programming specification.

UNIT-IV : Semantics : Syntax directed translation, S-attributed and L-attributed grammars, Intermediate code – abstract syntax tree, translation of simple statements and control flow statements.

UNIT-V : Context Sensitive features – Chomsky hierarchy of languages and recognizers. Type checking, type conversions, equivalence of type expressions, overloading of functions and operations.

UNIT-VI : Run time storage : Storage organization, storage allocation strategies, scope access to now local names, parameters, language facilities for dynamics storage allocation.

UNIT-VII : Code optimization : Principal sources of optimization, optimization of basic blocks, peephole optimization, flow graphs, Data flow analysis of flow graphs.

UNIT-VIII : Code generation : Machine dependent code generation, object code forms, generic code generation algorithm, Register allocation and assignment. Using DAG representation of Block.

Text Books:

1. Sipser Michael: Introduction to Theory of computation, 1st Edition, Thomson, 2009.
2. Aho, Ullman, Ravisethi: Compilers Principles, Techniques and Tools, 2nd Edition, Pearson Education, 2009.

Reference Books:

1. Tremblay J P, Sorenson G P: The Theory & Practice of Compiler writing, 1st Edition, BSP publication, 2010.
2. Appel W & Andrew G M: Modern Compiler Implementation in C, 1st Edition, Cambridge University Press, 2003.
3. Loudon: Compiler Construction, Principles & Practice, 1st Edition, Thomson Press, 2006.

ADVANCED SOFTWARE ENGINEERING**Course code: 11CS2111****L P C**
4 0 4

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.

UNIT – II : Process models: The waterfall model, Incremental process models, Evolutionary process models, The Unified process. Software Requirements: Functional and non-functional requirements, User requirements, System requirements, Interface specification, the software requirements document.

UNIT – III : Critical Systems : A simple safety-critical system, System dependability, Availability and Reliability, Safety, Security Critical Systems Specification: Risk-driven specification, safety specification, security specification, software reliability specification.

UNIT – IV : 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.

UNIT–V :Real-time Software Design System design, Realtime Operating systems, Monitoring and Control systems, Data Acquisition systems Performing User Interface Design: Design Issues, the UI design process, User analysis, User interface prototyping, Interface evaluation.

UNIT-VI : Critical Systems Development: Dependable processes, dependable programming, fault-tolerance, fault-tolerant architectures. Testing Strategies: A strategic approach to software testing, test strategies for Conventional software, test strategies for object-oriented software, validation testing, system testing, art of debugging.

UNIT-VII : Rapid Software Development : Agile methods, Extreme programming, Rapid Application Development, software prototyping Component-Based Software Engineering: Components and component models, the CBSE process, Component composition.

UNI-VIII :Cleanroom Software Engineering: Cleanroom approach, Functional specification,Cleanroom design, Cleanroom testing. Configuration Management: Configuration management, planning, Change management, Version and release management, System building, CASE tools for configuration management.

Text Books:

1. Ian Sommerville : Software Engineering, , 8th Edition, Pearson, 2007.
2. Roger S Pressman : Software Engineering: A Practitioner's Approach, 7th Edition, TMH, 2009.

Reference Books:

1. K.K. Agarwal & Yogesh Singh : Software Engineering, 3rd Edition, New Age International Publishers, 2009.
2. James F.Peters,Witold Pedrycz : Software Engineering, An Engineering approach, 1st Edition, John Wiley, 2000.
3. Shely Cashman Rosenblatt : Systems Analysis and Design, 4th Edition, Thomson Publications, 2005.
4. Waman S Jawadekar : Software Engineering Principles and Practice, 1st Edition, TMH, 2004.

DATA WAREHOUSING AND DATA MINING**Course code:11CS2112****L P C
4 0 4**

UNIT – I :Introduction: Data mining-On what kinds of Data, Data Mining Functionalities, Classification of Data Mining systems, Data Mining Task Primitives, Integration of a Data Mining System with a Database or Data Warehouse System, Major issues in Data Mining.

UNIT – II : Data Preprocessing: Descriptive data summarization, Data Cleaning, Data Integration and Transformation, Data Reduction, Discretization and Concept Hierarchy Generation.

UNIT -III :Data Warehouse and OLAP Technology: Multidimensional Data Model, Data Warehouse Architecture, Data Warehouse Implementation, From Data Warehousing to Data Mining.

UNIT – IV :Data Cube Computation and Data Generalization: Efficient methods for Data Cube Computation, Further Development of Data Cube and OLAP Technology, Attribute-Oriented Induction.

UNIT – V :Mining Frequent Patterns, Association and Correlations: Basic Concepts, Efficient and Scalable Frequent Itemset Mining Methods, Mining Various kinds of Association Rules, From Association Mining to Correlation Analysis, Constraint-Based Association.

UNIT-VI :Classification and Prediction-1: Issues Regarding Classification and Prediction, Classification by Decision Tree Induction, Bayesian Classification, Rule-Based Classification, Classification by backpropagation.

UNIT-VII: Classification and Prediction-2: Support Vector Machines, Association Classification, Other Classification Methods, Prediction, Accuracy and Error Measures, Evaluating the Accuracy of a Classifier or Predictor.

UNIT –VIII :Cluster Analysis: Types of Data in Cluster Analysis, A Categorization of Major Clustering Methods, Partitioning Methods, Hierarchical Methods, Density-Based Methods, Grid-Based Methods, Model-Based Clustering Methods, Outlier Analysis.

Text Books:

1. Jiawei han & Micheline Kamber : Data Mining – Concepts and Techniques, 2nd Edition, Morgan Kaufmann Publishers, 2008.
2. Margaret H Dunham : Data Mining Introductory and Advanced Topics, 6th Edition, Pearson education, 2009.

Reference Books:

1. Arun K Pujari : Data Mining Techniques, 1st Edition, University Press, 2005.
2. Pang-Ning Tan, Michael Steinbach, Vipin Kumar : Introduction To Data Mining, 1st Edition, Pearson Education, 2009.
3. Sam Aanhory & Dennis Murray : Data Warehousing in the Real World, 1st Edition, Pearson Education, 2008.
4. Paulraj Ponnaiah : Data Warehousing Fundamentals, 1st Edition, Wiley student Edition, 2007.
5. Ralph Kimball : The Data Warehouse Life cycle Tool kit, 2nd Edition, Wiley student Edition, 2005.

NETWORK SECURITY AND CRYPTOGRAPHY**Course code:11CS2113****L P C
4 0 4**

UNIT – I :Introduction:Attacks, Services and Mechanisms, Security attacks, Security services, A Model for Internetwork security. Classical Techniques: Conventional Encryption model, Steganography, Conventional Encryption Principles, Conventional encryption algorithms, cipher block modes of operation, location of encryption devices.

UNIT – II :Number theory: Prime and Relatively prime numbers, Modular arithmetic, Fermat's and Euler's theorems, Testing for primality, Euclid's Algorithm, the Chinese remainder theorem, Discrete logarithms, key distribution Approaches of Message Authentication Security of Hash Functions and MACs – MD5 message Digest algorithm , Secure Hash Algorithm – RIPEMD and HMAC

UNIT–III :Public key cryptography principles, public key cryptography algorithms, digital signatures, digital Certificates, Certificate Authority and Key Management - Diffie-Hellman key Exchange, Kerberos, X.509 Directory Authentication Service.

UNIT – IV :Email privacy: Pretty Good Privacy (PGP) and S/MIME.

UNIT–V :IP Security Overview, IP Security Architecture, Authentication Header, Encapsulating Security Payload, Combining Security Associations and Key Management.

UNIT – VI : Web Security Requirements, Secure Socket Layer (SSL) and Transport Layer Security (TLS), Secure Electronic Transaction (SET).

UNIT – VII : Basic concepts of SNMP, SNMPv1 Community facility and SNMPv3. Intruders, Viruses and related threats.

UNIT – VIII : Firewall Design principles, Trusted Systems. Intrusion Detection Systems.

Text Books:

1. William Stallings : Network Security Essentials (Applications and Standards), 3rd Edition, Pearson Education, 2008.
2. Ryan Russell, Dan Kaminsky, Rain Forest Puppy, Joe Grand, David Ahmad, Hal Flynn Ido Dubrawsky, Steve W. Manzuik and Ryan Permech: Hack Proofing your network , 2nd Edition, Wiley Dreamtech, 2002.

Reference Books:

1. Eric Maiwald, Fundamentals of Network Security, 1st Edition, Dreamtech Press, 2003.
2. Charlie Kaufman, Radia Perlman and Mike Speciner : Network Security - Private Communication in a Public World, 2nd Edition, Pearson/PHI, 2002.
3. Stallings : Cryptography and network Security, 3rd Edition, PHI/Pearson, 2006.
4. Whitman : Principles of Information Security,, 3rd Edition, Thomson, 2009.
5. Robert Bragg, Mark Rhodes : Network Security: The complete reference, 1st Edition, TMH, 2004.
6. Buchmann : Introduction to Cryptography, 2nd Edition, Springer, 2004.

EMBEDDED SYSTEMS
(Elective-2)

Course Code:11CS2114

L	P	C
4	0	4

UNIT – I :Introduction to Classic 8051 family Architecture. Address and data bus with multiplexed I/O pins. Registers, Examples with arithmetic and Boolean instruction set. Applications using Timers Counters and I/O programming for external logic sensing and control. Interrupts and its programming. This is an example of Vanneumann Architecture.

UNIT – II :Introduction to Harvard architecture. Advantages of separate address and data busses providing faster and efficient programming. Built in Flash with two wire programming reducing CPU size. Provision of peripherals and flash ROM, EEPROM, and a large special function register work space for application oriented embedded systems.

Introduction to PIC family Architecture and instruction set. Introduction to the Risk instruction set and its usage with example programs using Integrated development environment MPLAB simulation.

UNIT – III: Peripheral systems in Pic 16f877a processor.

- (a) Digital Input and Output Programming,
- (b) Timers and Counters
- (c) Capture Control and PWM
- (d) Analog to Digital Converters and their Programming
- (e) Simple data acquisition systems and programming.

UNIT – IV: Introduction to Atmega processor family architecture using typical Atmega 8535 processor. The many features in the

peripherals provided. Introduction to its large instruction set. Using IDU Atmel Studio for programming and simulation.

UNIT – V :Peripheral systems in Atmega 8535

- (a) Digital Input and Output Programming
- (b) Timers and Counters wave form generation.
- (c) Capture Control and PWM
- (d) Analog to Digital Converters and their Programming
- (e) Simple data acquisition programming.

UNIT – VI: Serial Communication buses

- (a) USART, with addressable feature
- (b) SPI bus
- (c) 12c two wire bus
- (d) Introduction to USB bus

UNIT – VII :Application design using an embedded system.Interrupts, and interrupt processing. Interrupt vectors, and their application programming. Interrupt processing in PIC and Atmega processor families. Interrupt latency. Processing multiple interrupts.Logical steps to design a program to meet an objective. Examples in robotics, Motor control, display control, data acquisition etc. preferably with illustrative examples.

UNIT – VIII : Processor simulation and debugging using integrated development environment. The use of IDU gives a comprehensive glimpse of all processor activities to enable the programmer to watch events. Such a program makes it easy to find errors in the program logic and correct it.

A brief introduction to In-circuit debugging of an assembly level program.

Text books:

1. Bendapudy Kanta Rao : Embedded Systems, Prentice Hall India, 1st Edition, 2011.
2. Ajay V Deshmukh : Microcontrollers, 4th Edition, TMH, 2010.
3. Kenneth J Ayala : The 8051 Micro Controller, 3rd Edition, Thomson Publishers, 2009.

Reference books:

1. Raj Kamal : Embedded Sytesms, 2nd Edition, TMH, 2008.
2. Raj Kamal : Microcontrollers, 1st Edition, Pearson Education, 2009.
3. Ali Mazidi Mohammed Gillispie, Mazide Janice: The 8051 Microcontroller and Embedded Systems, 2nd Edition, Pearson Education, 2009.

STORAGE AREA NETWORKS AND MANAGEMENT (Elective-2)

Course code:11CS2115

L P C

4 0 4

UNIT – I :Introduction to Storage Technology Information storage, evolution of storage technology and architecture, data center infrastructure, key challenges in Managing information, information lifecycle. Storage system Environments: components of storage system environment, Disk Drive components, Disk Drive Performance, fundamental laws governing disk performance, logical components of the host, application requirements and disk performance.

UNIT – II : Data Protection: RAID: Implementation of RAID, RAID array components, RAID levels, RAID comparison, RAID Impact on disk performance, host spares. Intelligent Storage System: Components of an Intelligent Storage System, Intelligent Storage array, concepts in Practice: EMC CLARIION and Symmetrix.

UNIT – III : Direct – Attached Storage and Introduction to SCSI :Types of DAS, DAS benefits and limitations, disk drive interfaces, introduction to parallel SCSI, SCSI command model. Storage Area Networks: fibre channel, The SAN and Its evolution, components of SAN, FC connectivity, Fibre channel ports, fibre channel architecture, zoning, fiber channel login types, concepts in practice: EMC Connectrix.

UNIT – IV :Network attached storage: general purpose servers vs NAS Devices, benefits of NAS, NAS file I/O, components of NAS, NAS Implementations, NAS file sharing protocols, NAS I/O

operations, factors effecting NAS Performance and availability, concepts in practice: EMC Celerra.IP SAN: iscsi, fcip.

UNIT – V :Content – addressed storage: Fixed content and Archives, types of archives, features and benefits of CAS, CAS Architecture, object storage and retrieval in CAS, CAS Examples, concepts in practice: EMC Centera. Storage Virtualization: Formas of Virtualization, SNIA Storage virtualization taxonomy, storage virtvalization configurations, storage virtualization challenges, types of storage virtualization, concepts in practice: EMC Invista, Rainifinity.

UNIT – VI: Introduction to business continuity: information availability, BC terminology, BC planning life cycle, Failure analysis, business impact analysis, BC technology solutions, concepts in practice: EMC Power path. Backup and recovery: backup purpose, backup considerations, backup granularity, recovery considerations, backup methods, backup process, backup and restore operations , backup topologies, backup in NAS environments, backup technologies, concepts in practice: EMC Networker, EMC Disk Library(EDL).

UNIT – VII: Local replication: Source and targets, uses of local replicas, data consistency, local replication technologies, restore and restart considerations, creating multiple replicas, management interface, concepts in practice EMC Timefinder and Emc snap view. Remote replication: modes of remote replication, remote replication technologies, network infrastructure, concepts in practice: EMC SRDF,EMC SAN Copy.

UNIT – VIII: Securing the infrastructure: storage security framework, storage security domains, security implementations in storage networking. Managing the Storage infrastructure: Monitoring the Storage infrastructure, Storage management activities, Storage

infrastructure management challenges, Developing an ideal solution, concepts in practice: EMC control center.

Text Books:

1. G. Somasundaram, A. Shrivastava, EMC Corporation : Information Storage and Management, 1st Edition, wiley publishing, 2009.
2. Robert Spalding, Storage Networks : The Complete Reference, 1st Edition, TMH, 2003.

Reference Books:

1. Marc Farley : Building Storage Networks, 2nd Edition, Tata McGraw Hill, Osborne , 2001.
2. Meeta Gupta : Storage Area Network Fundamentals, 2nd Edition, Pearson Education Limited, 2002.

PROGRAMMING ON PARALLEL & DISTRIBUTED COMPUTING SYSTEMS (ELECTIVE-2)

Course code:11CS2116

L	P	C
4	0	4

Unit-I: Introduction:

What is distributed computing. Basic network concepts: Data transmission, layered architecture, protocols, connection-oriented versus connectionless Communication; Basic operating system concepts: Programs, processes, concurrency, child processes, threads. Basic software engineering concepts: Procedural programming versus object-oriented programming, Unified Modeling Language (UML) diagrams, software architecture (presentation logic, application/business logic, service logic), The Internet: background and history, architecture, HTTP, HTML, web server, web browser, web services; Network resources and their identification: computers, services, resources, host names, host identifiers, port addresses, the domain name system, Internet addresses, Uniform Resource Locator (URL), Uniform Resource Identifier (URI) Security; Fault Tolerance.

UNIT-II: Interprocess Communication & Distributed Computing Paradigm

Interprocess communication - Basic model; Primitives (operations): connect, send, receive, disconnect; Connection-oriented/connectionless; Data marshalling: data flattening, data representation, serialization; Event synchronization; Event diagram, sequence diagram; An Overview of distributed computing paradigms; Analysis of Different software forms or architectures for distributed computing;

UNIT-III: The Socket API & Client and Server programming:

Socket API: The basic model; Stream-mode (connection-oriented) socket; Datagram socket (connectionless) socket; Java socket API;

Using socket to implement a client; Using socket to implement a server; A simple middleware using socket; Secure sockets and the Java secure socket extension API

Client and Server program Programming Shared Address Space Platforms; An Overview of POSIX This chapter introduces the most basic distributed computing model; The daytime protocol and a sample client-server suite; The echo protocol and a sample client-server suite-Connection-oriented client-server. Connectionless client-server, Iterative server and concurrent server, Stateful server and stateless server.

UNIT-IV: Group Communications & Distributed Objects

Group Communication :- Unicast versus multicast; Basic model of group communications; The Java multicast API; Sample multicast sender program; Sample multicast listener program; Multicast and message ordering; Reliable multicast/broadcast; Java implementation over multicast IP addresses Distributed Objects :- Message passing versus distributed objects: The basic model; Remote procedure call; Remote method invocation: basic architecture; object registry; remote interface, interface implementation, server implementation, client implementation, algorithm for developing client-side and server-side software.

UNIT-V: Internet Applications

Basic components and protocols: HTTP, HTML, MIME, web server, browser, web forms; Web document types: static, dynamic, executable, active. CGI: background; interaction and passing of data among browser, web server, and script(s); HTTP Session state information: hidden tags, cookies, session objects. Client-side programming: Applets, JavaScript; Server-side programming: common gateway Interface (CGI), servlets, server pages. Applets; Servlets; session data maintenance; Web services and the Simple Object Access Protocol (SOAP)

Course Contents: Parallel Computing

UNIT-VI: Parallel Processing Platforms

Architecture Models of Parallel Processing Platforms (SIMD & MMD Models; PRAM & Architectural Models; Interconnection Networks for Parallel Computers ; Static & Dynamic Interconnection Networks - Bus-Based; Crossbar Multistage Networks, Network topologies : Completely connected, Star-connect, Linear Array, 2-D /3-D torus, Evaluating of Static & Dynamic Interconnection Networks; Cache Coherence in Multi-Processor Systems; Communication Costs (Message Passing Costs, , Communication costs in shared-Address Space Machines); Communication Operations: (One-to-All Broadcast, All-to-All Broadcast, Reduction, and All-to-All Personalized Communication); Routing Mechanisms for Interconnection networks; Processor Mapping in Parallel Machines; Mapping Techniques for Graphs, Cost –Performance trade-offs and Performance Metrics – Theoretical Studies on Speedup and Performance Analysis

UNIT-VII: Parallel Processing Platforms: Message Programming

Programming Using Message-Passing Paradigm – Principles of Message Passing Programming; An overview of the Message Passing Building blocks (*Sending* and *Receiving* Operations; Blocking & *Non-Blocking* Message Passing Operations); an Overview of Message Passing Interface (MPI; - MPI 1.0 & MPI 2.0); Point-Point Message Passing Communication Library calls; Collective communication and computation library calls; MPI Groups & Communicators; Topologies; Programming and Performance issues for Numerical Computations; Programming Paradigms for Dense Matrix Computations (Shared & Distributed Shared Memory Systems - An Overview of Vector-Vector, Matrix-Vector Matrix-Matrix Multiplication algorithms)

UNIT-VIII: Parallel Processing Platforms: Mixed (Shared & Message passing) Programming

Mixed Programming (MPI, OpenMP, Pthreads, Intel TBB) for Numerical computations, (Dense Matrix Computations); Tuning and Performance on Cluster of Multi-Core Processors (*Compilers-General Tuning and Optimization; Compiler Optimization Switches, Using Mathematical Libraries*); Application and System Benchmarks; Common Parallel Programming Problems - (*OpenMP, Intel TBB, MPI*) on Message Passing Clusters; I/O- Multi-threading & Message Passing on Multi-Core processors;

Text Books:

1. M.L.Liu : Distributed Computing – Principles and Applications, Pearson Education 2005.
2. Grama Ananth, Anshul Gupts, George Karypis and Vipin Kumar: Introduction to Parallel computing, Boston, MA : Addison-Wesley, 2003.

Reference Books: Parallel Computing:

1. Chandra, Rohit Morgan, Leonardo Dagum, Dave Kohr, Dror Maydan, Jeff McDonald, and Ramesh Menon: Parallel Programming in OpenMP San Francisco Moraan Kaufmann, 2001.
2. James Reinders, Intel Threading Building Blocks, O'REILLY series, 2007.
3. Michael J. Quinn: Parallel Programming in C with MPI and OpenMP McGraw-Hill International Editions, Computer Science Series, McGraw-Hill, Inc. Newyork, 2004

4. Andrews, Grogory R: Foundations of Multithreaded, Parallel, and Distributed Progrmaming, Boston, MA : Addison-Wesley, 2000.
5. Butenhof, David R : Programming with POSIX Threads , Boston, MA : Addison Wesley Professional, 1997.
6. Marc Snir, Steve Otto, Steyen Huss-Lederman, David Walker and Jack Dongarra:) MPI-The Complete Reference: Volume 1, 2nd Edition, The MPI Core, 1998.
7. William Gropp, Steven Huss-Lederman, Andrew Lumsdaine, Ewing Lusk, Bill Nitzberg, William Saphir and Marc Snir : MPI-The Complete Reference: Volume 2, The MPI-2 Extensions, 2nd Edition, 1998.
8. Shameem Akhter, Jason Roberts : Multi-Core Programming – Increasing Performance through Software Multi-threading, Intel Corp.,2006
9. Pacheco S. Peter: Parallel Programming with MPI, University of Sanfrancisco, Morgan Kaufman Publishers, Inc., Sanfrancisco, California, 1992.
10. William Gropp, Ewing Lusk, Rajeev Thakur : Using MPI-2, Advanced Features of the Message-Passing Interface, The MIT Press., 1999.
11. Foster : Designing and Building Parallel Programs ; Concepts and tools for Parallel Software Engineering, Addison-Wesley, 1995.
12. J.Dongarra, I.S. Duff, D. Sorensen, and H.V.Vorst : Numerical Linear Algebra for High Performance Computers (Software, Environments, Tools) SIAM, 1999.

13. Jack Dongarra (Editor), Ian Foster (Editor), Geoffrey C. Fox (Editor), William Gropp; Ken Kennedy, Linda Torczon, Andy White; The Sourcebook of Parallel Computing; The Morgan Kaufmann Series in Computer Architecture and Design 2003.
14. Culler, David E., Jaswinder Pal Singh : Parallel Computer Architecture - A Hardware/Software Approach , San Francisco, CA : Morgan Kaufmann; 1999.
15. Kai Hwang, Zhiwei Xu: Scalable Parallel Computing (Technology Architecture Programming), McGraw Hill, 1998.

Reference Books: Distributed Computing:

1. Jim Farle: JAVA – Distributed Computing O'REILLY 1998.
2. Amjad Umar: Distributed Computing – A practical Synthesis, PTR, PHI ,1993.
3. M.L.Liu - Distributed Computing – Principles and Applications, Pearson Education, 2005.
4. George Coulouris, Jean Dollimore, Kindberg Ti: Distributed Systems, Concepts and Design 3/e; 4th Edition, Pearson Education, Addison Wesley 2005.
5. Andrew S Tanenbaum, Maarten Van Steen, Distributed Systems : Principles and Paradigms, Pearson Education, 2002.
6. Andrew S Tanenbaum: Distributed Operating Systems, Prentice Hall, US Ed., 2002.
7. P. K. Sinha: Distributed Operating Systems, PHI, 2005.

NETWORK SECURITY AND CRYPTOGRAPHY LAB**Course code:11CS21 17****L P C
0 4 2**

The following programs should be implemented preferably on platform using C language (for 1-5) and other standard utilities available with UNIX systems (for 6-9) :-

1. Implement the encryption and decryption of 8-bit data using Simplified DES Algorithm (created by Prof. Edward Schaefer) in C
2. Write a program to break the above DES coding
3. Implement Linear Congruential Algorithm to generate 5 pseudo-random numbers in C
4. Implement Rabin-Miller Primality Testing Algorithm in C
5. Implement the Euclid Algorithm to generate the GCD of an array of 10 integers in C
6. a) Implement RSA algorithm for encryption and decryption in C
b) In an RSA System, the public key of a given user is $e=31, n=3599$. Write a program to find private key of the User
7. Configure a mail agent to support Digital Certificates, send a mail and verify the correctness of this system using the configured parameters.
8. Configure SSH (Secure Shell) and send/receive a file on this connection to verify the correctness of this system using the configured parameters.
9. Configure a firewall to block the following for 5 minutes and verify the correctness of this system using the configured parameters:
 - (a) Two neighborhood IP addresses on your LAN
 - (b) All ICMP requests
 - (c) All TCP SYN Packets

DATA WAREHOUSING AND DATA MINING LAB**Course code:11CS2118****L P C**
0 4 2

1. Implementation of multi dimensional data model using oracle warehouse builder/SQL Server.
2. Introduction to Weka : All the features of Weka software will be explored in this assignment.
Implementation of the following programs in C/C++/Java.
3. Implementation of Apriori algorithm
4. Implementation of FP tree algorithm
5. Implementation of Naïve Bayesian classification algorithm
6. Implementation of Backpropagation algorithm
7. Implementation of K-means clustering algorithm
8. Implementation of K-Medoids clustering algorithm

