S. Y. B. Tech. Computer Engineering Semester – II										
Course Code	Course Title	Teaching Scheme Hours /Week			Examination Scheme				Marks	Credit
		Lecture	Tutorial	Practical	In Semester	End Semester	Oral	Practical		
BSCE 2201	Engineering Mathematics III	3	1	0	50	50	0	0	100	4
CE 2201	Data Structures and Algorithms II	3	1	0	50	50	0	0	100	4
CE 2202	Fundamentals of Computer Networks	3	1	0	50	50	0	0	100	4
CE 2203	Operating Systems	3	0	0	50	50	0	0	100	3
CE 2204	Microprocessor Architectures	3	1	0	50	50	0	0	100	4
CE 2205	Data Structures and Algorithms II Laboratory	0	0	4	25	0	0	25	50	2
CE 2206	Microprocessor Architectures Laboratory	0	0	2	25	0	0	0	25	1
CE 2207	Operating Systems Laboratory	0	0	4	25	0	25	0	50	2
	Total	15	4	10	325	250	25	25	625	24
	Grand Total	29			625				625	24

MKSSS's Cummins College of Engineering For Women
★
Karve Nagar
Pune-411052.

DEAN ACADEMICS
MKSSS's Cummins College
of Engineering for Women
Karvenagar, Pune-411052

Principal
MKSSS's Cummins College of Engg.
For Women, Karvenagar, Pune-52

APPROVED BY
Governing Body Members
MKSSS's Cummins College
of Engineering for Women
Karvenagar, Pune-411052

BSCE- 2201 ENGINEERING MATHEMATICS III

Teaching Scheme
Lecture : 3 Hrs./week
Tutorials: 1Hr/week

Examination Scheme
In semester : 50 marks
End semester : 50 marks
Credits : 04

Prerequisite:

1. First order linear ordinary differential equations.
2. Basics of Vector Algebra
3. Integration – basic properties, standard results, Beta & Gamma Functions.
4. Partial Fractions.
5. Permutation & Combination. Basics of probability.

Course Objectives: Mathematics is a necessary path to scientific knowledge which opens new perspective of mental activity. Our aim is to provide sound knowledge of engineering mathematics to make the students think mathematically and strengthen their thinking power to analyze and solve engineering problems in their respective areas.

Course Outcome: Students will be able to

CO1. Solve Higher order Linear differential Equations, Simultaneous Differential Equations.

CO2. Calculate Divergence, Curl, Directional derivative, Solenoidal, Irrotational, Scalar potential, vector identities, Line integral.

CO3. Find Fourier Transform, Inverse Fourier Transform.

CO4. Find Z-transforms, Inverse Z – Transform**,** difference equation.

CO5. Calculate Moments, Mean, Variance, Covariance, Correlation, Probability Distributions, Compute Skewness, Kurtosis, Linear Regression.

Unit 1: Higher Order Linear Differential equation and application (07)

Higher order Linear differential Equation with constant coefficients, Cauchy’s and Legendre’s Differential Equations, Simultaneous Differential Equations, Modelling of electrical circuits.

Unit 2: Vector Calculus (07)

Physical interpretation of vector differentiation, vector differential operator, Gradient, Divergence, Curl, Directional derivative, Solenoidal, Irrotational and Conservative fields, Scalar potential, vector identities, Line integral.

Unit 3: Fourier Transform (07)

Complex exponential form of Fourier series, Fourier integral theorem, sine and cosine integrals, Fourier transform, Fourier Sine and Cosine transform , Inverse Fourier Transform, Discrete Fourier Transform.

Unit 4: Z – Transform (06)

Definition, standard properties, Z- Transform of standard sequences, Inverse Z – Transform using standard results, Inversion integral method, solution of difference equation to solve Computer Engineering Problems.

Unit 5: Probability (08)

Theorem of total probability, Theorem of Compound Probability, Baye's theorem, Moments, Mean, dispersion,Variance, Covariance ,Correlation, Random variables, Distributions – Binomial , Poisson, Normal .

Unit 6: Data Analytic (07)

Types of data: Concepts of population and sample, quantitative & qualitative data, cross-sectional and time-series data, discrete and continuous data, Skewness, Kurtosis, Linear Regression.

Text Books:

1. B. S. Grewal, **'Higher Engineering Mathematics** ', *Khanna Publications.*
2. B. V. Ramana, **'Higher Engineering Mathematics** ', *Tata McGraw Hill Publications* (2007)
3. C.R.Wylie, L.C. Barrette, '**Advanced Engineering Mathematics**', *McGraw Hill Publications, New Delhi.(*6th edition)(2003)

References:

1. Peter V. O'neil,**'Advanced Engineering Mathematics'** ,*Thomson Brooks / Cole, Singapore (*5th edition) (2007).
2. Erwin Kreyszig ,**'Advanced Engineering Mathematics'** *Wiley Eastern Ltd.* (8th Student Edition), (2004).
3. S.P.Gupta ' **Statistical Methods'**, S.Chand & sons

CE 2201 DATA STRUCTURES AND ALGORITHMS II

Teaching Scheme	**Examination Scheme**
Lectures : 3 Hrs/Week	In Semester : 50 Marks
Tutorials : 1 Hr/Week	End Semester : 50 Marks
	Credits : 4

Prerequisite:
CE 2102 - Data Structures and Algorithms I

Course Objectives:
To facilitate the learners:

1. To learn and understand representation, implementation and applications of trees, search trees, graphs, multiway trees data structures.
2. To choose and apply data structures for developing solutions for solving problems in various domains.
3. To analyze algorithms using time complexity analysis.
4. To understand and apply the concepts of hashing and file handling.

Course Outcomes:
By taking this course, the learner will be able to

1. Apply appropriate non linear data structure to construct efficient algorithms to approach the problems.
2. Distinguish between various non linear data structure based on their representations and applications.
3. Apply the concept of Hashing techniques for solving a problem.
4. Make use of File handeling and Java collection Frameworks for solving a problem.

Unit 1: Trees **(08)**
Introduction to Non Linear Data Structure, Binary Trees,Types of Binary Trees,Properties of Binary Trees, Binary Tree as Abstract Data Type, Representation using Sequential and Linked Organization, Binary Tree creation , Recursive and Non Recursive Tree Traversals, Threaded Binary Tree and operations, Traversals of Inorder Threaded Binary Tree, Applications of Binary Trees.

Unit 2: Search trees **(08)**
Representation of Symbol Tables- Static Tree Table and Dynamic Tree Table, Binary Search Tree and its operations, Binary Search Trees as Abstract Data Type, Height Balanced Tree : AVL Tree and operations ,Red Black Tree.

Unit 3: Graphs **(07)**
Basic Terminologies, Storage Representation, Graph Traversals, Graph as Abstract Data Type, Spanning Trees, Minimum Spanning Trees, Kruskal’s Algorithm, Prim’s Algorithm, Dijkstra's Single Source Shortest Path Algorithm,Topological Sorting, Case Study : Data structures used in Google map.

Unit 4: Multiway trees and Heap **(06)**
Multiway search tree , B Tree and operations, B+ Tree,Applications of Btrees, Heap basic concepts, Realization of Heap, Heap as an Abstract Data Type, Heap implementation, Heap Sort, Heap as a Priority Queue.

Unit 5: Hashing **(07)**
General idea of Hashing, Hash Table, Hash function, Rehashing, Issues in Hashing, Collision Resolution Strategies: Linear Probing, Quadratic Probing, Double Hashing, Open addressing and Chaining.

Unit 6: File Organization and Java Collection Framework (06)

File Organization, Sequential File, Direct Access File and its Primitive operations, Java Collection Framework : Arraylist , Treeset, Hashset and Hashmap Class.

Text Books:

1. Sartaj Sahani, **“Data Structures, Algorithms and Applications in JAVA”,** *Universities Press* (2nd edition), (2007).
2. Robert Lafore , **“Data Structures Algorithms in JAVA”**, *Techmedia*,(1st edition), (2006).
3. Ivor Horton, **“Beginning Java”**, *Wiley India Edition*,(Java 7 edition),(2012).
4. E. Horowitz, S. Sahni, D. Mehta, **“Fundamentals of Data Structures in C++”**, *Galgotia Publications* ,(2nd edition), (2008).

References:

1. Sartaj Sahani, **“Data Structures, Algorithms and Applications in C++”,** *Universities Press* (2nd edition), (2007).
2. R. Gillberg, B. Forouzn, **“Data Structures: A Pseudo code approach with C++”,** *Cenage Learning* (2nd edition) (2007).
3. Y. Langsam, M. Augenstein and A. Tenenbaum, **“Data structures using C and C++”,** *Prentice Hall of India* (2nd edition), (2005) .
4. M. Weiss, **“Data Structures and Algorithm Analysis in C++”,** *Pearson Education* (3rd edition), (2009).
5. A. Aho, J. Hopcroft, J. Ullman, **“Data Structures and Algorithms”**, *Pearson Education* (3rd Impression), (2008).

List of the Tutorial Assignments:

Every student should perform 12 to 14 tutorials which will cover topics of all units mentioned in the syllabus of Data Structures and Algorithms II. Students will perform practice exercise on data representation and corresponding implementation. Tutorial assignments will help students to enhance their ability of problem solving using appropriate data structures.

Following list of tutorials can be considered as guideline for designing tutorials:

1. Practice exercise on creating a binary tree and perform recursive and non recursive traversals of binary tree on given data.
2. Create a binary search tree for the given data and perform its inorder, preorder, postorder traversals.
3. Practice exercise on searching and deleting data values from given binary search tree. Analyze the time complexity of used algorithm.
4. Create a binary search tree for the given data and perform its inorder, preorder, postorder traversals.
5. Practice exercise on different rotations of AVL tree.
6. Construct AVL tree for the given numeric data elements. Perform the appropriate rotations whenever needed.
7. Simulate flight path data using graph data structure to find minimum cost path.
8. Practice assignment on converting a binary tree to threaded binary tree and its traversals.
9. Design a heap data structure for student data and find out minimum/maximum marks obtained in particular subject.
10. Use sequential file to maintain employee information. Write algorithm to add, delete and search employee information from the file.
11. Design a solution for company survey about its products in an area. Choose the appropriate algorithm to complete the survey within short time period and cover all

houses under that area. Give justification for your answer and also analyze your algorithm for time complexity.

12. Given the input data and hash function , show the result using following hashing methods

    a. Linear Probing b. Quadratic Probing c. Double hashing h2(x)=7-(x Mod 7)
13. Use different hashing functions to hash given values.
14. Construct a Btree of order 3 by inserting numbers of given data.

CE2202 FUNDAMENTALS OF COMPUTER NETWORKS

Teaching Scheme	**Examination Scheme**
Lectures : 3 Hrs/Week	In Semester : 50 Marks
Tutorials : 1Hr/Week	End Semester : 50 Marks
	Credits : 4

Course Objectives:
To facilitate the learners

1. To learn and understand fundamental concept of networking.
2. To learn different methods for framing, flow control, error control.
3. To understand OSI model & TCP/IP protocol stack.
4. To learn various functions of physical & data link layer.

Course Outcomes:
By taking this course, the learner will be able to

1. **Build** the knowledge of fundamental concepts of networking to recognize various network standers and protocols.
2. **Build** the knowledge of design requirements of layered network architecture.
3. **Analyze** different error and flow control strategies.
4. **Experiment with** different line coding techniques, modulation techniques and switching techniques to build design requirements of physical layer.

Unit 1: Introduction to Computer Networks (08)

Concept of Data in Networking-Representation, Transmission, Data Flow, types of Connection- Point to Point ,Point to Multi Point , Network Standards, type of Networks- LAN,WAN,MAN, Ad-hoc Network, Networking Topologies: Bus, Mesh, Star, Ring and Hierarchical , The Internet-dial up,DSL service, Internet Standards, Internet administration.

Unit 2: Network Models (06)

Principles of protocol layering,The TCP/IP Protocol suite:Layers, description of each layer , encapsulation and decapsulation, addressing,multiplexing and demultiplexing, OSI Model, OSI verses TCP/IP suite.

Unit 3: Physical Layer (08)

Digital signals,Digital to digital conversion:line Coding techniques(unipolar,polar and bipolar), analog to digital conversion:(PCM, DM),Transmission modes:parallel,serial, introduction to Multiplexing and types: FDM, TDM,Transmission Media- Guided(Twisted pair cable, coaxial cable, Fiber Optic),Unguided media:propagation methods, types of waves (radio waves,microwaves and infrared waves) .Introduction to Switching-Circuit Switching, Packet Switching, Message Switching.

Unit 4: Logical Link Control Sublayer (LLC) (08)

Design issues, services, functions, Framing, Error Control and Flow Control, Error Control-Parity Bits, Hamming Code & CRC, Flow Control-Unrestricted Simplex Protocol, Stop and Wait, Sliding Window Protocol.

Unit 5: Medium Access Control Sublayer (MAC) (08)

Channel Allocation-Static and Dynamic, Multiple Access Protocols: CSMA, , IEEE

802.Standards and Frame Formats, CSMA/CD, CSMA/CA.

Unit 6: Connecting Devices and Virtual LAN (04)

Various Network Devices NIC, Switches, Hub, Routers, Repeaters, Bridge and Access Point. Virtual LANS : membership, configuration, communication between switches, advantages.

Text Books:

1. Fourauzan B., "Data Communications and Networking", 5th Edition, Tata McGraw- Hill, Publications, 2006.
2. William Stallings “Data and computer communication”, Pearson, 8th Edition, ISBN: 0-13-243310-9

References:

1. Kurose, Ross “Computer Networking a Top Down Approach Featuring the Internet”,6th edition (March 5, 2012), Pearson , ISBN-10: 0132856204.
2. Andrew S. Tenenbaum, “Computer Networks”, 5th Edition, PHI, ISBN 81-23-2175-8.

List of the Tutorial Assignments:

Every Student should perform 12-14 tutorials which will cover topics of all units mentioned in the syllabus of Fundamentals of Computer Network.

Tutorial assignments will help students learn and explore the subject in greater detail. Students will be able to recall and practically apply the concepts learnt. Students will emulate algorithms to get insight of the strategies used for flow control.

1. Basic concepts of Computer Networking.
2. Execute and understand basic Networking Commands.
3. Study and discuss various Network components, devices and Structured Cabling components.
4. Problems on Line Coding techniques- POLAR (RZ, NRZ)
5. Problems on Line Coding techniques- Polar Biphase: Manchester and Differential Manchester encoding
6. Problem solving on Error Control coding through Hamming code technique.
7. Problem solving based on CRC technique.
8. Problem solving on basic Flow Control strategies: Sliding Window protocol (Go Back N).
9. Problem solving on basic Flow Control strategies: Sliding Window protocol (Selective repeat).
10. Create Peer to Peer network and LAN network to share files within the created network.

CE 2203 OPERATING SYSTEMS

Teaching Scheme
Lectures : 3 Hrs/week

Examination Scheme
In Semester : 50 Marks
End Semester : 50 Marks
Credits : 3

Prerequisites:

1. Fundamentals of Programming Languages – II (ES 1202)
2. Digital Systems and Computer Organization (CE 2104)

Course Objectives:
To facilitate the learner -

1. To understand basic concepts of Operating Systems.
2. To understand process life-cycle and scheduling algorithms.
3. To analyze memory management strategies.
4. To understand File System concepts.
5. To learn operating system for managing resources such as I/O, CPU, memory etc.
6. To understand Inter-process Communication and deadlock concepts.

Course Outcomes:
By taking this course, the learner will be able -

1. To Build the basic knowledge of operating system.
2. To Apply the CPU scheduling algorithms using process concepts.
3. To Apply the memory management strategies.
4. To Apply the file attributes and different access modes on various types of file.
5. To Make use of the knowledge of storage devices for disk management.
6. To Apply the concepts of Inter-process Communication.

Unit 1: Introduction to Operating Systems (06)

Introduction to Operating System (OS), Evolution of OS, Functions of OS, Types of OS, OS Concepts, Process, Files, Shell and its types, Kernel and its types, System Calls, Virtual Machine, Case Study of UNIX Operating System.

Unit 2: Process and CPU Scheduling (08)

Process Concept, Operations On Processes, Creation, Termination, States, Transition and Context Switching, Scheduling Criteria, Scheduling Algorithm, First-Come First-Serve (FCFS), Shortest Job First (SJF), Round-Robin (RR), Introduction to Threads and Benefits, Case Study of Unix Process Management.

Unit 3: Memory Management (08)

Contiguous and Non-Contiguous Memory, Swapping, Paging, Segmentation, Virtual Memory, Demand Paging, Page Replacement Algorithms- First-In First-Out (FIFO), Least Recently Used (LRU), Optimal, Allocation of Frames and Trashing.

Unit 4: Introduction to the File System (06)

File Concepts, File Attributes, File Operations, File Types, File Sharing, File Structure, Mounting and Un-Mounting, Directory Overview, Types of Directories, Types of Users, Access Modes, Free space management, Case Study of UNIX File Structure.

Unit 5: I/O Management and Disk Scheduling (06)

I/O Devices, Organization of I/O Functions, Operating System Design Issues Related to I/O, I/O Buffering, Disk Scheduling - First Come-First Serve (FCFS), SCAN, Circular SCAN (C- SCAN), Shortest Seek Time First (SSTF).

Unit 6: Inter-Process Communication (IPC) (08)

Critical Section Problem, Hardware Support for Mutual Exclusion, Semaphores, Classical Problems of Synchronization, Monitors, Deadlocks, Methods of Handling Deadlock, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection and Recovery from Deadlock.

Text Books:

1. William Stallings, "Operating System-Internals and Design Principles ", Prentice Hall India,(5/e) ISBN: 81-297-0 1 094-3.
2. Silberschatz, Galvin, Gagnes, "Operating System Concepts", John Wiley & Sons, (6/e), ISBN: 9971-51-388-9.
3. Maurice J. Bach, “The Design of the Unix Operating System”, Pearson Education, ISBN: 81-7758-770-6.

References:

1. Evi Nemeth, Garth Snyder, Tren Hein, Ben Whaley, “Unix and Linux System Adminstration Handbook”, (4/e), ISBN: 978-81-317-6177-9. (2011).
2. Milan Milenkovic, “Operating Systems”, TMH,(2/e), ISBN: 0-07-044700-4.
3. Andrew S. Tanenbaum, "Modern Operating Systems", Prentice Hall India, (2/e), ISBN: 81-203-2063-8.

CE 2204 MICROPROCESSOR ARCHITECTURE

Teaching Scheme:	**Examination Scheme:**
Lectures: 3 hrs./Week	In-Semester: **50** Marks
Tutorial: 1 hr./Week	End-Semester: **50** Marks
	Credits: 4

Prerequisite:

1. Digital Systems and Computer Organization (CE 2104)

Course Objectives:

To facilitate the learners

1. To understand basic architecture of 8086 microprocessor.
2. To understand and analyze the basic interfacing techniques.
3. To understand pipelined and superscalar architecture of Pentium.
4. To understand, apply and analyze x86 microprocessor instructions to the assembly language programming.

Course Outcomes:

By taking this course, the learner will be able to

1. Demonstrate the knowledge of basic 8086 microprocessor concepts.
2. Select the different components and interfacing peripherals associated with microprocessor architectures.
3. Apply the programming concepts using x86 assembly level language.
4. Relate the advanced features of Pentium microprocessor.
5. Infer the advanced microprocessor architectures.

Unit – 1: BASIC MICROPROCESSOR ARCHITECTURE (06)

8086 Architecture, Pin diagram of 8086, Programmers' model of 8086, pin Diagram, Addressing Modes, Instruction Set, Memory architecture of 8086, Segmentation, even and odd memory banks, address mapping.

Unit – 2: BASIC INTERFACING TECHNIQUES (06)

Block diagram, control words, operating modes, programs of Parallel peripheral interface with 8255 (Programmable Peripheral Interface), Block diagram, control words, operating modes, programs of Serial peripheral interface with 8251 (USART), Block diagram, control words, operating modes, programs of Timing and control signals handling using 8253 (Programmable Interval Timer).

Unit – 3: SUPERSCALAR ARCHITECTURE IN PENTIUM MICROPROCESSOR (06)

Pentium Architecture, Pipeline stages, Superscalar pipeline issues, Instruction paring rules, Branch prediction, Memory organization with Instruction and Data caches Pentium programmers' model, Register set, Addressing modes and instructions (other than 8086).

Unit – 4: PROTECTED MODE ARCHITECTURE IN PENTIUM MICROPROCESSOR (06)

Real Mode vs. Protected mode, Memory management with segmentation and paging Protection mechanism in segmentation and paging, Virtual 8086 Mode (support registers, descriptors, privilege-level, protection, exclusive instructions, inter-privilege level, transfer control, Paging-support registers, Descriptor, linear to physical address translation, TLB, page level protection).

Unit – 5: MULTITASKING, INTERRUPTS, EXCEPTION AND INPUT/OUTPUT (06)

Multitasking, support registers, Descriptors, Task switching, Nested task, I/O handling in Pentium, I/O instructions, I/O Permission bit map, Interrupts and Exceptions structure in real, protected and virtual modes.

Unit–6: INTRODUCTION TO ADVANCED MICROPROCESSOR ARCHITECTURES (06)

Introduction to multicore architectures i3/i5/i7, Design Issues, Cache coherency Advanced Processor Architectures for Mobile Application, Embedded Application and Enterprise Application.

Text Books:

1. 8086 and peripherals – Intel Manual
2. Pentium Architecture – Intel Manual
3. Douglas Hall, **'Microprocessors & Interfacing'**, *McGraw Hill*, (Revised 2nd Edition), (2006)
4. James Antonakos, **'The Pentium Microprocessor'**, *Pearson Education,* (2nd Edition), (2004)

Reference Books:

1. Sivarama P. Dandamudi, **'Introduction to Assembly Language Programming For Pentium and RISC Processors'**, *Springer*,(2nd Edition), (2004)
2. Peter Abel, **'Assembly language programming'**, *Pearson Education*,(5th Edition),(2002)
3. John Uffenbeck, **'The 8086/88 Family: Design, Programming & Interfacing'**, *PHI*, (2nd Edition), (2002)
4. A. Ray, K.Bhurchandi, **'Advanced Microprocessors and peripherals: Architecture, Programming & Interfacing'**, *Tata McGraw Hill*, (2nd Edition), (2004)
5. Liu, Gibson, **'Microcomputer Systems:The8086/88Family'**,*PHI*,(2nd Edition),(2005)
6. Kip Irvine, **'Assembly language for IBM PC'**, *PHI*, (2nd Edition),(1993)

Web References:

1. NPTEL series – nptel.ac.in/courses/Webcourse-contents/IIT-KANPUR/
2. service.scs.carleton.ca/sivarama/org_book/org_book_web/slides/chap_1_versions/ch7_1.pdf

CE 2205 DATA STRUCTURE AND ALGORITHMS- II LABORATORY

Teaching Scheme
Practical : 4 Hrs/Week

Examination Scheme
In Semester : 25 Marks
Practical : 25 Marks
Credits : 2

Prerequisite:
CE2107 - Data Structures and Algorithm Laboratory I

Course Objectives:
To facilitate the learners

1. To choose and apply appropriate Data Structures for a given problem statement.
2. To design algorithmic solution for a given problem.
3. To analyze and compare algorithms.
4. To implement non linear data structures using Object Oriented Programming.

Course Outcome:
By taking this course, the learner will be able to

1. Develop a solution of the given problem using tree data structure.
2. Develop a solution of the given problem using graph data structure.
3. Apply hashing techniques to solve a given problem.
4. Make use of sequential file handling operations.
5. Design small application using non linear data structures.

List of Assignments
The laboratory assignments are designed in a set of group A, B and C such that students will be able to design and implement solution for a given problem. Group A assignments are designed in such a way that students will choose appropriate data structures to implement solution of a given problem. All the units of the syllabus of Data Structures and Algorithms II are covered in group B assignments. In group C assignments students will design an algorithmic solution for selected problem using concepts covered in the subject Data Structures and Algorithms II.
The laboratory assignments of group A and B are to be submitted by student individually using C++/JAVA object oriented programming language. Group C assignments may be performed in a group of 2 to 4 students from the same batch. For each assignment program code with sample output is to be submitted as a soft copy. Handwritten write up (Title, Objectives, Problem Statement, Algorithms, Outcomes) of each assignment is to be submitted by students.

Group A (Mandatory)

1. Create a Dictionary that stores keywords and its meanings, using appropriate data structure. Implement its operations such as add, delete, display, search and update its values.
2. Create a reasonably balanced tree to maintain names and telephone numbers of all the customers of a shopkeeper and perform operations on it. Test your program for at least 10 names.
3. A news paper delivery boy every day drops news paper in a society having many lanes and houses. Design a program to provide different paths that he could follow. Solve the problem by suggesting appropriate data structures. Design necessary class.
4. Write a program to create telephone book database of N clients. Make use of a hash table implementation to quickly look up client's telephone number.

Group B (At-least Six)

1. Create a binary tree and perform inorder ,preorder and postorder traversals.
2. Implement Binary Search Tree as Abstract Data Type and perform operations on it.
3. Write a program to create a binary tree if inorder and preorder or inorder and postorder any two traversals are given.
4. Create inorder threaded binary tree and perform its traversals.
5. Company wants to lease phone lines to connect its offices of different cities, with each other. Company charges different amounts of money to connect different pairs of offices. Solve the problem by suggesting appropriate data structures to connect all offices of a company with a minimum cost.
6. Write a modular program to implement primitive operations on Min/Max Heap.
7. Write a program to implement Symbol Table as an ADT.
8. Use sequential file to maintain student information. Write algorithm to add, delete and search student information from the file.
9. Implement hash table ADT and handle the collision using linear probing and chaining (with or without replacement). Perform operations on it.

Group C

Create a small application using appropriate data structures to process stock data / organization's data / college data.

CE 2206 MICROPROCESSOR ARCHITECTURES LABORATORY

Teaching Scheme:	**Examination Scheme:**
Practical: 2 hrs./Week	In Semester – 25 marks
	Credit(s): 1

Prerequisite:

1. Digital Systems and Computer Organization (CE 2104)

Course Objectives:

To facilitate the learners

1. To understand and apply x86 instructions to write assembly language program.
2. To learn, apply and analyze microprocessor and peripherals interfacing techniques.
3. To learn and use the interfacing of assembly language and higher level language.
4. To able to solve moderately complex problems using modular assembly language programming.
5. To understand and use privileged instructions.

Course Outcomes:

By taking this course, the learner will be able to

1. Choose x86 instructions to write assembly language programs.
2. Build a small system using microprocessor interfacing techniques.
3. Solve a given problem using advanced assembly language methods.
4. Apply the modular programming using assembly level language.

The Microprocessor Architectures laboratory assignments are designed using assembly language programming as well as hardware interfacing techniques. The laboratory work also covers the assembly language interface with higher level language like 'C'. The students are introduced to advanced protected mode instructions.

Group A Assignments (Perform all assignments)

1. Write ALP to perform basic arithmetic operations and check the output in debugger.
2. Write ALP to accept a string and display it on the screen.
3. Write ALP to accept a signed number and check if it is positive or negative. Display appropriate message.
4. Write 8086 ALP to interface DAC and generate following waveforms on oscilloscope
   (i) Square wave – Variable Duty Cycle and frequency.
   (ii) Ramp wave – Variable direction
   (iii) Trapezoidal wave
   (iv) Stair case wave
   (v) Temple wave
   (vi) Sine wave – using look up table
5. Write 8086 ALP to program 8251 for serial communication between two 8251s.
6. Write 8086 ALP to program 8253 to observe outputs of different modes using counter display.
7. Write ALP using STRING instructions to accept a string from user and perform following operations
   (a) Convert a string to uppercase / lowercase

(b) Toggle the case of the string
(c) Concatenation of another string
(d) Find if it is palindrome
(e) Find a substring
(For this assignment make a group of 4 students, each one performing each task and then combine all functions to apply modular programming.)

Group B Assignments (Perform any two)

1. Write ALP to perform following using command line arguments to simulate TYPE or COPY command.
2. Write ALP to find the largest number from an array using PUBLIC/GLOBAL and EXTERN.
3. Write a C/ inline program for PC to PC communication.
4. Write ALP for Mouse interface.
5. Write inline code to perform file operations.
6. Write ALP for floating point operations.

Group C Assignments (Perform any one)

1. Write ALP for to read GDTR/LDTR and IDTR and display the table content pointed by GDTR and IDTR.
2. Write ALP to implement multitasking using Pentium programming.

Text Books:

1. 8086 and peripherals – Intel Manual
2. Pentium Architecture – Intel Manual
3. Douglas Hall, **'Microprocessors & Interfacing'**, *McGraw Hill*, (Revised 2nd Edition), (2006)
4. James Antonakos, **'The Pentium Microprocessor'**, *Pearson Education*, (2nd Edition), (2004)

Reference Books:

5. Sivarama P. Dandamudi, **'Introduction to Assembly Language Programming For Pentium and RISC Processors'**, *Springer*,(2nd Edition), (2004)
6. Peter Abel, **'Assembly language programming'**, *Pearson Education*,(5th Edition),(2002)
7. John Uffenbeck, **'The 8086/88 Family: Design, Programming & Interfacing'**, *PHI*, (2nd Edition), (2002)
8. A.Ray, K.Bhurchandi, **'Advanced Microprocessors and peripherals: Architecture, Programming & Interfacing'**, *Tata McGraw Hill*, (2nd Edition), (2004)
9. Liu, Gibson, **'Microcomputer Systems:The8086/88Family'**,*PHI*,(2nd Edition),(2005)
10. Kip Irvine, **'Assembly language for IBM PC'**, *PHI*, (2nd Edition),(1993)

Web References:

1. NPTEL series – nptel.ac.in/courses/Webcourse-contents/IIT-KANPUR/
2. service.scs.carleton.ca/sivarama/org_book/org_book_web/slides/chap_1_versions/ ch7_1.pdf

CE 2207 OPERATING SYSTEMS LABORATORY

Teaching Scheme
Practical : 4 Hrs/week

Examination Scheme
In Semester : 25 Marks
Oral : 25 Marks
Credits : 2

Prerequisites:

1. Data Structures and Algorithms-I (CE2102)
2. Fundamentals of Programming Language Lab-II (ES1206)
3. Digital Systems and Computer Organization(CE2104)

Laboratory Objectives:
To facilitate the learners -

1. To understand the fundamentals of Operating Systems.
2. To understand shell scripting to automate operating system operations.
3. To understand the operations performed by Operating System as a resource manager.
4. To apply the concepts of Operating System for Process and Memory management.
5. To analyze various scheduling algorithms.
6. To understand the communication among the processes.

Laboratory Outcomes:
By taking this course, the learner will be able -

1. To choose Unix/Linux Commands for Shell Programming.
2. To make use of different CPU scheduling algorithms.
3. To apply Memory Management algorithms.
4. To apply various disk scheduling algorithms.
5. To examine the Inter-Process Communication concepts.

Every student should perform 9-10 assignments in this laboratory which will cover topics of all units mentioned in the syllabus of Operating Systems. Following is the list of assignments that can be considered as guideline for designing assignments and give basic knowledge of operating systems and its services. The choice of the assignments for each student is given in such a way that all topics should be distributed and covered amongst all batches.

List of Assignments:
Group A: (Mandatory)

1. Write a shell script to implement mount and un-mount commands to mount device and un-mount it.
2. Exploration of Unix/Linux Commands (File, Directory and Process commands).
3. Write a program to implement Banker's Algorithm for deadlock handling.
4. Write a program to implement Reader-Writer problem using semaphores.

Group B: (Any four)

5. Write a program to implement following Non- Pre-emptive scheduling algorithms : First Come First Serve (FCFS), Shortest Job First (SJF).
6 .Write a program to implement following Pre-emptive scheduling algorithms: Round-Robin (RR), Shortest Remaining Time First (SRTF)
7. Write a program to implement following memory allocation strategies: First Fit, Best Fit and Worst Fit.
8. Write a program to implement following Page replacement algorithms: a) First-In-First-Out (FIFO). b) Least Recently Used (LRU)   c) Optimal page replacement.
9. Write a shell script for adding users / groups and modifying permissions of file / directory accordingly.
10.Write a program to implement following disk scheduling algorithms: First Come First Serve (FCFS), SCAN, Circular - SCAN(C-SCAN), Shortest Seek Time First (SSTF).

Group C: (Any one)

1. Installation of Linux Operating System.
2. Implement producer-consumer algorithm using multi-threading concept.