## Computer Graphics (3 – 1 – 2)

**Evaluation:**

| | Theory | Practical | Total |
|---|---|---|---|
| Sessional | 30 | 20 | 50 |
| Final | 50 | - | 50 |
| Total | 80 | 20 | 100 |

**Course Objectives:**

1. To provide the knowledge of basic techniques used in Computer Graphics Systems.
2. To provide the knowledge of 2D and 3D algorithms used in Computer Graphics Systems.

**Course Contents:**

**Unit 1: Introduction** **(2 hrs)**

1.1 Introduction
1.2 History of Computer Graphics
1.3 Application of Computer Graphics

**Unit 2: Graphics Hardware** **(6 hrs)**

2.1 Interactive Input Devices
2.2 Display Devices and Hard Copy Devices
2.3 Raster and Random Systems and Architectures
2.4 Video Controller
2.5 Use of Digital to Analog Converter and Frame Buffer Organization
2.6 Color Monitors

**Unit 3: Two Dimensional Algorithms** **(7 hrs)**

3.1 Line Drawing Algorithms
3.1.1 DDA
3.1.2 Bresenham's Algorithm
3.2 Circle Generation Algorithm
3.3 Ellipse Generation Algorithms
3.4 Area Filling-Scan Line Algorithm
3.5 Boundary Fill Techniques
3.6 Flood Fill Techniques

**Unit 4: Two Dimensional Geometric Transformations and Viewing** **(8 hrs)**

4.1 Basic Transformations
4.2 Other Transformations
4.3 Homogeneous Co-ordinate systems
4.4 Composite Transformations
4.5 Windowing Concepts
4.6 Viewing Pipeline
4.7 Window to View port Transformation

4.8 Line Clipping Algorithm: Cohen-Sutherland
4.9 Polygon Clipping: Sutherland-Hodgeman

**Unit 5: Three Dimensional Graphics Systems** **(7 hrs)**

5.1 3D Co-ordinate System and 3D Transformations
5.2 3D Representations
5.3 Polygon Surfaces
5.4 Cubic Spline and Beizer Curve
5.5 Non-Planer Surface: Bezier Surface
5.6 Fractal Geometry Method
5.7 3D Viewing Transformation
5.8 Projection Methods: Parallel and Perspective
5.9 Clipping in 3D

**Unit 6: Visible Surface Detection** **(5 hrs)**

6.1 Hidden Surfaces and their Removal Techniques
6.2 Back-Face Detection
6.3 Depth Buffer Method
6.4 A- buffer method
6.5 Scan Line Method
6.6 Area Subdivision Method
6.7 Depth Sorting Method

**Unit 7: Illumination and Shading** **(6 hrs)**

7.1 Illumination Theory
7.2 Ambient Light
7.3 Reflections: Diffuse, Specular
7.4 Surface Shading methods
 7.4.1 Constant Shading
 7.4.2 Gouraud Shading
 7.4.3 Phong Shading
 7.4.4 Fast Phong Shading
7.5 Color Models: RGB, CMYK

**Unit 8: Graphical Languages** **(4 hrs)**

8.1 Need for Machine Independent Graphical Languages
8.2 Graphical Languages: PHIGS, GKS
8.3 Graphics Software Standard
8.4 Overview of Graphics File Formats
8.5 Data Structure in Computer Graphics
8.6 Introduction to OpenGL

**Laboratory:**

Implementation of various 2D and 3D graphics algorithms covered in the course using C / C++ and OpenGL.

**Text Book:**

Donald Hearn and M. Pauline Baker: *Computer Graphics*, Prentice-Hall.

**References:**


1. James D. Foley, Andries van Dam, Steven K. Feiner, John F. Hughes, *Computer Graphics: Principles and Practice in C*, Addison-Wesley.
2. Mason Woo, Jackie Neider, Tom Davis, Dave Shreiner, *Open GL Programming Guide* Third Edition, The Official Guide to Learning OpenGL, Version 1.2, OpenGL Architecture Review Board, LPE Pearson Edition Asia.

**Probability and Statistics (3 – 2 – 0)**

**Evaluation:**

| | Theory | Practical | Total |
|---|---|---|---|
| Sessional | 50 | - | 50 |
| Final | 50 | - | 50 |
| Total | 100 | - | 100 |

**Course Objective:**

This course is designed to familiarize the students with basic knowledge about probability and statistics. After successful completion of this course students would be able to understand and apply the concept ofpresentation and summarization of data, probability and probability distributions,sampling and estimation, hypothesis testing, simple regression and correlation.

**Course Contents:**

1. **Introduction of Statistics and Presentation of Data** **(4 hrs)**
   1.1 Introduction of statistics
   1.2 Application of statistics in engineering
   1.3 Variables, types of variable: numerical and categorical variable
   1.4 Sources of data: primary and secondary source
   1.5 Presentation and classification ofdata: stem- and-leaf displays
   1.6 Frequencydistribution
   1.7 Diagrammatic and graphical presentation of data: Pareto diagram
   1.8 Pie-diagram, histogram, frequency curve and frequency polygon
   1.9 Cumulativefrequency curve or ogive curve

2. **Summarizing and Describing the Numerical Data** **(4 hrs)**
   2.1 Measure of central tendency (mean ,median, mode), partition values
   2.2 Measure of variation: range, inter-quartile range, standard deviation
   2.3 Coefficient of variation
   2.4 Box and whisker plot

3. **Probability** **(5 hrs)**
   3.1 Random experiment, sample space, event and types of events, counting rule
   3.2 Various approaches to probability
   3.3 Laws of probability-additive, multiplicative
   3.4 Conditional-probability and independence
   3.5 Baye's theorem

4. **Random Variable and Probability Distribution** **(12 hrs)**
   4.1 Random variable: discrete and continuous random variable
   4.2 Probability mass function
   4.3 Expectation, laws of expectation (addition and product law)
   4.4 Discrete probability distribution: Binomial distribution, Poisson distribution,Hyper Geometric distribution and Negative binomial distribution

4.5 Probability density function,cumulative distribution functions, expected values ofcontinuous random variables
4.6 Continuous probability distribution: rectangular distribution, exponentialdistribution, Gamma distribution, Beta distribution, Normal distribution, Log-Normal distribution

## 5. Bi-variate Random Variables and Joint Probability Distribution (3 hrs)

5.1 Joint probability mass function,joint probability density function,joint probability distribution function
5.2 Marginal probability mass function, marginal probability density function,conditional probability mass function
5.3 Sums and average of random variables

## 6. Sampling and Estimation (5 hrs)

6.1 Population and samples
6.2 Sampling distribution of mean
6.3 Types of sampling: probability and non-probability sampling
6.4 Determination of sample size
6.5 Central limit theorem and its application
6.6 Estimation: concept of point and interval estimation, criteria of good estimator, interval estimation, maximum likelihood estimation
6.7 Confidence interval for population mean and population proportion

## 7. Testing of Hypothesis (7 hrs)

7.1 Null and alternative hypothesis, level of significance, type I and type II error, critical value, P-value, one and two tailed test, steps involved in hypothesis testing
7.2 One Sample test for mean and proportion
7.3 Two sample test for mean (independent and dependent) and proportion

## 8. Simple Linear Regression and Correlation (5 hrs)

8.1 Simple correlation and its properties
8.2 Concept of simple regression analysis, estimation of regression coefficient by using least square estimation method
8.3 Standard error, coefficient of determination.

**Text Book:**

Johnson, Richard A. *Probability and Statistics for Engineers* (8th edition). New Delhi: PHI learning private limited. 2011.

**References:**

1. Devore, Jay L. *Probability and Statistics for Engineering and the Sciences* (8th edition). New Delhi: Cengage learning.
2. Sheldom, M. Ross. *Probability and Statistics for Engineers and Scientist* (4th edition). New Delhi: Cengage Learning.
3. Shrestha, Hridya B. *Statistics and Probability* (2nd edition).Kathmandu: Ekata Books Distributer Pvt. Ltd.

## Computer Architecture (3-1-1)

**Evaluation:**

| | Theory | Practical | Total |
|---|---|---|---|
| Sessional | 30 | 20 | 50 |
| Final | 50 | - | 50 |
| Total | 80 | 20 | 100 |

**Course Objectives:**

- To acquaint the students with the fundamentals of computer systems.
- To apprise the students with the architectural and associated components of computer systems.
- To aware the students about the architecture of the computer systems available in the market.

**Course Contents:**

**1. Introduction** **4 hrs**

1.1. Computer Organization and Computer Architecture
1.2. Review of Evolution of Computer System
1.3. Basic Structure of Computer System
1.4. Examples of Computer Families
1.5. Future Trends in Computer
1.6. Review of Instruction Sets, Addressing Modes and Instruction Formats

**2. Register Transfer and Micro Operations** **2 hrs**

2.1. Register Transfer and RTL
2.2. Micro operations
2.3. Data Transfer Micro operations
2.4. Arithmetic and Logical Micro operations
2.5. Shift Micro operations
2.6. Introduction to HDL and VHDL

**3. Central Processing Unit** **3 hrs**

3.1. CPU Organization/Structure
3.2. Register Organization and Data Paths
3.3. Instruction Cycle
3.4. Arithmetic and Logical Unit
3.5. Design Principles for Modern Systems

**4. Computer Arithmetic** **6 hrs**

4.1. Integer Representation
4.2. Integer Arithmetic
4.3. Unsigned Binary Addition and Subtraction
4.4. Unsigned Binary Multiplication Algorithm

4.5. Booth's Algorithm
4.6. Unsigned Binary Division Algorithm
4.7. Floating Point Representation
4.8. BCD Arithmetic Unit
- BCD Adder

4.9. Arithmetic Pipelining

**5. Control Unit** **6 hrs**

5.1. Control of the Processor
5.2. Hardwired Control Unit
- Control Unit Inputs
- Control Unit Logic

5.3. Micro programmed Control Unit
- Micro Instructions and Its Types
- Architecture of Micro programmed Control Unit

5.4. Micro Instruction Sequencing
5.5. Micro Instruction Execution
5.6. Applications of Hardwired and Micro programmed Control Units

**6. Memory Organization** **6 hrs**

6.1. Memory Hierarchy
6.2. Main Memory
- RAM and ROM

6.3. Auxiliary Memory
- Magnetic Disks and Tapes
- Optical Disks
- Flash Drives
- Review of RAID

6.4. Associative Memory
- Hardware Organization
- Address Matching Logic
- Read/Write Operations

6.5. Cache Memory
- Cache Initialization
- Mapping Cache Memory
    - Direct, Associative and Set Associative Memory Mapping
- Write Policy
- Replacement Algorithms

**7. Input Output Organization** **4 hrs**

7.1. External Devices
7.2. I/O Module Structure
7.3. Review of Programmed I/O and Interrupt Driven I/O
7.4. Review of DMA, I/O Channels and I/O Processors

7.5. External Interfaces

**8. Reduced Instruction Set Computers** **5 hrs**

8.1. RISC VS. CISC
8.2. RISC Pipelining
8.3. Instruction Pipelining
8.4. Conflicts in Instruction Pipelining and their Solutions
8.5. Introduction to Register Windows and Register Renaming

**9. Introduction to Parallel Processing** **6 hrs**

9.1. Parallelism in Uniprocessor System
9.2. Multiprocessor Systems and their Characteristics
9.3. Flynn's Classification
9.4. Interconnection Structures in Multiprocessors
9.5. Cache Coherence
9.6. Introduction to Vector Processing and Array Processors
9.7. Introduction to Multithreaded Architecture

**10. Multicore Computers** **3 hrs**

10.1. Hardware Performance Issues
- Increase in Parallelism
- Alternative Chip Organizations
- Power Consumption

10.2. Software Performance Issues
- Software on Multicore

10.3. Multicore Organization
10.4. Dual Core and Quad Core Processors
10.5 Power efficient Processors

**Laboratory**

The individual student should develop a project or perform a case study on Computer Architecture. The topic could be either initiated by the student or selected from a list provided by the instructor. An oral presentation with a demonstration in case of project should be part of the laboratory. Reports must be prepared.

**Text Books:**


1. Stallings, W., "*Computer Organization and Architecture*", Eighth Edition, 2011, Pearson.
2. Mano, M. M., "*Computer Systems Architecture*", Third Edition, 2011, Pearson.


**References:**


1. Tanenbaum, A.S., "*Structured Computer Organization*", Fourth Edition, 2003, Pearson Education.
2. Carpinelli, J.D., "*Computer Systems Organization and Architecture*", 2012, Pearson.
3. Rajaraman, V. et all, "*Computer Organization and Architecture*", 2011, PHI.
4. Sima, D. et all, "*Advanced Computer Architecture*", 2000, Addison Wesley.

**Numerical Methods (3-1-3)**

**Evaluation:**

| | Theory | Practical | Total |
|---|---|---|---|
| Internal | 30 | 20 | 50 |
| Final | 50 | - | 50 |
| Total | 80 | 20 | 100 |

**Course Objectives:**

1. To introduce numerical methods for interpolation, regressions, and root finding to the solution of problems.
2. To solve elementary matrix arithmetic problems analytically and numerically.
3. To find the solution of ordinary and partial differential equations.
4. To provide knowledge of relevant high level programming language for computing, implementing, solving, and testing of algorithms.

**Course Contents:**

1. **Solution of Nonlinear Equations** **(10 hrs)**
   - 1.1 Review of calculus and Taylor's theorem
   - 1.2 Errors in numerical calculations
   - 1.3 Bracketing methods for locating a root, initial approximation and convergence criteria
   - 1.4 False position method, secant method and their convergence, Newton's method and fixed point iteration and their convergence.

2. **Interpolation and Approximation** **(7 hrs)**
   - 2.1 Lagrangian's polynomials
   - 2.2 Newton's interpolation using difference and divided differences
   - 2.3 Cubic spline interpolation
   - 2.4 Curve fitting: least squares lines for linear and nonlinear data

3. **Numerical Differentiation and Integration** **(5 hrs)**
   - 3.1 Newton's differentiation formulas
   - 3.2 Newton-Cote's, Quadrature formulas
   - 3.3 Trapezoidal and Simpson's Rules
   - 3.4 Gaussian integration algorithm
   - 3.5 Romberg integration formulas.

4. **Solution of Linear Algebraic Equations** **(10 hrs)**
   - 4.1 Matrices and their properties
   - 4.2 Elimination methods, Gauss Jordan method, pivoting
   - 4.3 Method of factorization: Dolittle, Crout's and Cholesky's methods
   - 4.4 The inverse of a matrix
   - 4.5 Ill-Conditioned systems
   - 4.6 Iterative methods: Gauss Jacobi, Gauss Seidel, Relaxation methods
   - 4.7 Power method.

5. **Solution of Ordinary Differential Equations** **(8 hrs)**
   - 5.1 Overview of initial and boundary value problems
   - 5.2 The Taylor's series method
   - 5.3 The Euler Method and its modifications
   - 5.4 Huen'smethod
   - 5.5 Runge-Kuttamethods
   - 5.6 Solution of higher order equations
   - 5.7 Boundary Value problems: Shooting method.

6. **Solution of Partial Differential Equations** **(5 hrs)**
   - 6.1 Review of partial differential equations
   - 6.2 Elliptical equations, parabolic equations, hyperbolic equations and their relevantexamples.

**Laboratory:**

Use of Matlab/Math-CAD/C/C++or any other relevant high level programming language for applied numerical analysis. The laboratory experiments will consist of program development and testing of:

1. Solution of nonlinear equations
2. Interpolation, extrapolation, and regression
3. Differentiation and integration
4. Linear systems of equations
5. Ordinary differential equations (ODEs)
6. Partial differential equations (PDEs)

**Text Books:**


1. Gerald, C. F. & Wheatly, P. O. *Applied Numerical Analysis* (7th edition).New York: Addison Wesley Publidhing Company.
2. Guha, S. & Srivastava, R. *Numerical Methods: For Engineers and Scientists*. Oxford University Press.
3. Grewal, B. S. & Grewal, J. S. *Numerical Methods in Engineering & Science (*8th edition). New Delhi: Khanna publishers. 2010.
4. Balagurusamy, E. *Numerical Methods*. New Delhi: TataMcGraw Hill. 2010.


**References:**


1. Moin, Parviz. *Fundamentals of Engineering Numerical Analysis*. Cambridge University Press, 2001.
2. Lindfield, G. R. & Penny, J. E. T. *Numerical Methods: Using MATLAB*. Academic Press. 2012.
3. Schilling, J. & Harris, S.L. *Applied Numerical Methods for Engineers using MATLAB and C*. Thomson publishers, 2004.
4. Sastry, S. S. *Introductory Methods of Numerical Analysis* (3rd edition). New Delhi: Prentice Hall of India. 2002.
5. Rao, S. B. & Shantha, C. K. *Numerical Methods with Programs in Basic, Fortran and Pascal.* Hyderabad: Universities Press. 2000.
6. Pratap, Rudra. *Getting Started with MATLAB*. Oxford University Press. 2010.
7. Vedamurthy, V.N. & Lyengar, N. *Numerical Methods*. Noida: Vikash Publication House. 2009.

**Theory of Computation (3-1-0)**

**Evaluation:**

| | Theory | Practical | Total |
|---|---|---|---|
| Sessional | 50 | - | 50 |
| Final | 50 | - | 50 |
| Total | 100 | - | 100 |

**Course Objectives:**

To provide basic knowledge of the theory of automata, formal languages, and computational complexity.

**Course Contents:**

1. **Introduction** **(2 hrs)**
   1.1 Brief review of set, relation and functions
   1.2 Alphabet and language

2. **Finite Automata and Regular Expression** **(8 hrs)**
   2.1 Deterministic finite automata, Non-deterministic finite automata,
   2.2 Regular expressions, equivalence of regular language and fininte automata
   2.3 Properties of regular language
   2.4 The pumping lemma for regular sets
   2.5 Closure properties of regular sets
   2.6 Decision algorithms for regular sets.

3. **Context-free Language** **(8 hrs)**
   3.1 Context-free grammar
   3.2 Derivative trees and simplification of context- fee grammars
   3.3 Normal forms

4. **Pushdown Automata** **(10 hrs)**
   4.1 Introduction
   4.2 Equivalence of pushdown automata and context-free grammars
   4.3 Properties of Context-free languages (CFL)
   4.4 The pumping lemma for CFL's
   4.5 Closure properties of CF's
   4.6 Decision algorithms for CFLs

5. **Turing Machines** **(8 hrs)**
   5.1 Introduction to Turing machine
   5.2 Computing with Turing machine
   5.3 Extensions of Turing machines
   5.4 Computable languages and functions,

6. **Undecidability** **(5 hrs)**
   6.1 Church's Thesis
   6.2 Halting problem
   6.3 Universal Turing machines
   6.4 Undecidable problems about Turing machines
   6.5 Recursive function theory
   6.6 Properties of recursive and recursively enumerable languages

7. **Computational Complexity Theory** **(4 hrs)**
   7.1 Computable languages and functions
   7.2 Class P and class NP problems
   7.3 NP-complete problems

**References:**

1 R. McNaughton, Elementary Computability, Formal languages and Automata, Prentice Hall of India.

2 H.R Lewis, and C.H Papadimitriou, Element of the Theory of Computation, Eastern Economy Edition, Pretice Hall of India.

3 E. Engeler, Introduction to the Theory of Computation, Academic Press.

**Operating System (3 – 0 – 2)**

**Evaluation:**

| | Theory | Practical | Total |
|---|---|---|---|
| Sessional | 30 | 20 | 50 |
| Final | 50 | - | 50 |
| Total | 80 | 20 | 100 |

**Course Objectives:**

1. To provide the students with the knowledge of basics design principles of operating systems.
2. To gain knowledge about the Operating Systems concepts such as process, memory, resource, I/O management, CPU and disk scheduling etc.

**Course Contents:**

**1. Types and Structure of Operating Systems** **(4 hrs)**

1.1. Introduction and History of Operating System
1.2. Operating System Concepts and Functionalities
  1.2.1 Processes
  1.2.2. Files
  1.2.3. System Calls
  1.2.4. The Shell
1.3. Operating System Structure
  1.3.1 Monolithic Systems
  1.3.2 Layered
  1.3.3 Virtual Machines
  1.3.4 Client-Server
1.4 Types and Evolution of Operating Systems

**2. Processes and Threads** **(16 hrs)**

1.1.Process Concepts
  1.1.1. Introduction
  2.1.2. Definition of Process
  2.1.3. Process States and Transition
  2.1.4. PCB (Process Control Block)
  2.1.5. Concurrent Process, Parallel Processing
2.2. IPC (Inter-Process Communication)
  2.2.1. Critical Regions and Conditions
  2.2.2. Mutual Exclusion
  2.2.3. Mutual Exclusion Primitives and Implementation
    2.2.3.1. Dekker's Algorithm
    2.2.3.2. Peterson's Algorithm
    2.2.3.3. TSL (Test and Set Lock)
    2.2.3.3. Locks
  2.2.4. Producer and Consumer problem

2.2.5. Monitors
2.2.6. Message Passing
2.2.7. Classical IPC Problems
2.2.7.1. The Dining Philosophers Problem
2.2.7.2. The Readers and Writers Problem
2.2.7.3. The Sleeping Barber Problem
2.3. Deadlock and Indefinite Postponement:
2.3.1 Introduction
2.3.1.1 Preemptable and Nonpreemptable Resources
2.3.1.2 Conditions for Deadlock
2.3.1.3 Deadlock Modeling
2.3.2. Deadlock Avoidance
2.3.3. Deadlock Detection and Recovery
2.3.4. Deadlock Prevention
2.3.5. Issues Related to Deadlocks
2.3.5.1. Two Phase Locking
2.3.5.2. Non resource Deadlocks
2.3.5.3. Starvation
2.4. Threads
2.4.1. Introduction
2.4.2. Thread Model
2.4.3. Thread Usage
2.4.4. Advantages of Threads
2.4.5 User Space and Kernel Space Threads
2.4.6. Multithreading Model
2.5. Differences between Threads and Processes

**3. Kernel** **(2 hrs)**

3.1. Introduction and Architecture of a Kernel
3.2. Types of Kernels
3.3. Context Switching (Kernel mode and User mode)
3.4. First Level Interrupt Handling
3.5. Kernel Implementation of Processes

**4. Scheduling** **(4 hrs)**

4.1. Introduction:
4.2. Scheduling Levels
4.2.1. Scheduling Objectives and Criteria
4.2.2 Quantum Size
4.3. Preemptive Versus No Preemptive Scheduling
4.4. Scheduling techniques:
4.4.1. Priority Scheduling
4.4.2. Deadline Scheduling
4.4.3. First-In-First-Out Scheduling
4.4.4. Round Robin Scheduling
4.4.5. Shortest-Job-First(SJF) Scheduling

4.4.6. Shortest-Remaining-Time(SRT) Scheduling
4.4.7. Highest-Response-Ration-Next(HRN) Scheduling
4.4.8. Multilevel Feedback Queues

**5. Memory Management** **(6 hrs)**

5.1. Introduction,
 5.1.1. Storage Organization, Hierarchy and Management
 5.1.2. Storage Allocation
 5.1.3. Contiguous versus Noncontiguous Storage Allocation
 5.1.4. Logical and Physical Memory
5.2. Fragmentation
5.3. Fixed Partition Multiprogramming
5.4. Variable Partition Multiprogramming
5.5. Relocation and Protection
5.6. Coalescing and Compaction
5.7. Virtual Memory:
 5.7.1. Introduction
 5.7.2. Paging
 5.7.3. Page Tables
 5.7.4. Block Mapping
 5.7.5. Direct Mapping
 5.7.6. TLB(Translation Look aside Buffers)
 5.7.7. Page Fault
 5.7.8. Thrashing
5.8. Page Replacement Algorithms
 5.8.1. Optimal Page Replacement Algorithm
 5.8.2. Not Recently Used Page Replacement Algorithm
 5.8.3. First-In-First-Out Algorithm
 5.8.4. Second Chance Page Replacement Algorithm
 5.8.5. Least Recently Used Replacement Algorithm
 5.8.6. Clock Page Replacement Algorithm
 5.8.7. Working Set Page Replacement Algorithm
 5.8.8. WS Clock Page Replacement Algorithm
5.9. Segmentation
 5.9.1. Implementation of Pure Segmentation
 5.9.2. Segmentation with Paging

**6. Input/Output** **(3 hrs)**

6.1. Introduction
6.2. Principles of I/O Hardware
 6.2.1. I/O Devices
 6.2.2. Device Controllers
 6.2.3. Memory-mapped I/O
 6.2.4. DMA (Direct Memory Access)
6.3. Principles of I/O Software
 6.3.1. Goals of I/O Software

6.3.2. Polled I/O versus Interrupt Driven I/O
6.3.3. Character User Interface and Graphical User Interface
6.3.4. Goals of I/O Software
6.3.5. Device Drivers
6.3.6. Device independent I/O software
6.3.7. User Space I/O Software
6.4. Disk
6.4.1. Disk hardware
6.4.2. Disk Scheduling
6.4.2.1. Seek Time
6.4.2.2. Transfer Time
6.4.2.3. Disk Arm Scheduling Algorithms
6.4.3. RAID (Redundant Array of Inexpensive Disks)
6.4.4. RAM Disks

**7. File Systems** **(3 hrs)**

7.1. File Organization
7.1.1. Blocking and Buffering
7.1.2. File Descriptor
7.1.3. File Naming
7.1.4. File Structure
7.1.5. File Types
7.1.6. File Attributes
7.1.7. File Operations
7.1.8. File Access Methods
7.2. Sharing
7.3. ACL (Access Control List)
7.4. Directories
7.4.1. Directory Hierarchy
7.4.2. Directory Operations
7.5. File System Implementation
7.5.1. Contiguous Allocation
7.5.2. Linked list Allocation
7.5.3. I-nodes
7.6. Security and Multi-media files

**8. Distributed Operating Systems** **(5 hrs)**

8.1. Introduction
8.1.1. Advantages and Disadvantages of Distributed Operating System
8.1.2. Network Architecture
8.1.3. Hardware and Software Concepts
8.2. Communication in Distributed Systems
8.2.1. ATM (Asynchronous Transfer Mode)
8.3. Client-Server Model
8.4. RPC (Remote Procedure Call)
8.5. Group Communication

8.6. Processes and Processors in Distributed System
8.7. Clock Synchronization
8.8. Scheduling in Distributed System.

**9. Case Studies** **(6 hrs)**

**DOS Operating System:** System Configurations, Filing and Disk Management, Graphical Capabilities, Memory management.

**Unix/Linux Operating System:** File Systems and Disk Management, Filters, Pipelining, Sockets, Shell, Memory Management, Networking feature, Multiprocessing Feature.

**Window 2000:** File System and Disk management, Networking, Security.

**AMOEBA**: The System Architecture of Amoeba, Memory, Process Management and Communication in AMOEBA

**Laboratory:**

1. Housekeeping in DOS and Windows.
2. Housekeeping in UNIX/LINUX
3. Shell management in UNIX/LINUX and Shell Programming
4. Implementation of Resource and Memory Management Schemes in UNIX/LINUX
5. Process and Thread Creation, Scheduling and Management
6. Implementation of Inter-Process Communication Using Buffers
7. Implementation of Deadlock Prevention Algorithms
8. Device Programming

**Text Books:**

1. A.S. Tanenbaum, Operating systems, Design and Implementation, Prentice hall India.
2. H.M. Dietel, An Introduction to Operating System, Addison Wesley

**References:**

1. A.S. Tanenbaum, Modern Operating System, Second Edition, Prentice hall India.
2. W. Stallings, Operating Systems, Prentice hall India.