**II B.Tech. – I Semester**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Sl. No.** | **Course Code** | **Course Title** | **Contact Periods per week** | | | | **C** | **Scheme of Examination**  **Max. Marks** | | |
| **L** | **T** | **P** | **Total** | **Int. Marks** | **Ext. Marks** | **Total Marks** |
| 1 | 19BT3BS01 | Numerical Methods, Probability and Statistics | 3 | 1 | - | 4 | 4 | 40 | 60 | 100 |
| 2 | 19BT30502 | Computer Organization | 3 | - | - | 3 | 3 | 40 | 60 | 100 |
| 3 | 19BT31201 | Discrete Mathematical Structures | 3 | - | - | 3 | 3 | 40 | 60 | 100 |
| 4 | 19BT31501 | Data Structures and Algorithms | 3 | 1 | - | 4 | 4 | 40 | 60 | 100 |
| 5 | 19BT31502 | Operating Systems | 3 | - | - | 3 | 3 | 40 | 60 | 100 |
| 6 | 19BT31503 | System Programming | 3 | - | - | 3 | 3 | 40 | 60 | 100 |
| 7 | 19BT31531 | Data Structures and Algorithms Lab | - | - | 2 | 2 | 1 | 50 | 50 | 100 |
| 8 | 19BT31532 | Operating Systems Lab | - | - | 2 | 2 | 1 | 50 | 50 | 100 |
| 9 | 19BT31533 | Workshop in Computer Science and Systems Engineering | - | - | 2 | 2 | 1 | 50 | 50 | 100 |
| **Total:** | | | **18** | **2** | **6** | **26** | **23** | **390** | **510** | **900** |
| 10 | 19BT315AC | Design Thinking | 2 | - | - | 2 | - | - | - | - |

**II B. Tech.** **- I Semester**

(19BT3BS01) **NUMERICAL METHODS, PROBABILITY AND STATISTICS**

(Common to CSE, CSE(AI), CSE(DS), CSSE, IT, ME and CE)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Int. Marks | Ext. Marks | Total Marks |  | L | T | P | C |
| 40 | 60 | 100 |  | 3 | 1 | - | 4 |

**PRE-REQUISITES**: -

**COURSE DESCRIPTION**: Numerical solutions of equations; interpolation; numerical differentiation and integration; random variables; mathematical expectations; probability distributions; test of hypothesis.

**COURSE OUTCOMES:** After successful completion of the course, students will be able to:

**CO1:** Analyse the data and develop skills to solve equations and integrals by applying numerical methods.

**CO2:** Demonstrate knowledge in statistics and analyse the data for validations by applying statistical testing methods and distributions.

**DETAILED SYLLABUS**

**UNIT-I: NUMERICAL SOLUTIONS OF EQUATIONS AND INTERPOLATION**

**(8 Periods)**

Solutions of algebraic and transcendental equations: Regula-falsi method, Newton-Raphson method; Interpolation: Forward and backward differences, interpolation using Newton’s forward and backward difference formulae, Lagrange’s interpolation formula, partial fractions using Lagrange’s interpolation formula.

**UNIT- II: NUMERICAL DIFFERENTIATION AND INTEGRATION (9 Periods)**

Numerical differentiation using Newton’s forward and backward interpolation formulae; Numerical integration using Simpson’s 1/3rd and 3/8th rules; Numerical solutions of first order ordinary differential equations using Taylor’s series method, fourth order Runge-Kutta method.

**UNIT-III: RANDOM VARIABLES AND MATHEMATICAL EXPECTATIONS (8 Periods)**

Random Variables: Discrete and continuous random variables, distribution function of random variable, properties, probability mass function, probability density function; mathematical expectation, properties of mathematical expectation, mean and variance.

**UNIT-IV: PROBABILITY DISTRIBUTIONS (9 Periods)**

Discrete probability distributions: Binomial, Poisson- mean, variance, standard deviation (without derivations); Continuous probability distributions: Normal, uniform and exponential distributions- mean, variance, standard deviation (without derivations), area under the normal curve.

**UNIT-V: TEST OF HYPOTHESIS (11 Periods)**

Population and sample, parameter and statistic, null and alternative hypothesis, Type I and Type II errors, level of Significance, critical region, degrees of freedom; Large sample test: Tests of significance for proportions and means; Small sample test: Student’s t-test- single mean, difference of means; F-test for equality of population variance; Chi-Square test for independence of attributes.

**Total Periods: 45**

***Topics for self-study are provided in the lesson plan***

**TEXT BOOKS**:

1. T. K. V. Iyengar, B. Krishna Gandhi, S. Ranganatham and M. V. S. S. N. Prasad, *Mathematical Methods,* S. Chand & Company, 5th edition, 2016.
2. T. K. V. Iyengar, B. Krishna Gandhi, S. Ranganatham and M. V. S. S. N. Prasad, *Probability and Statistics,* S. Chand & Company, 5th edition, 2016.

**REFERENCE BOOKS**:

1. B. S. Grewal, *Higher Engineering Mathematics*, Khanna Publishers, 44th edition, 2017.
2. P. Kandasamy, K. Thilagavathy, K. Gunavathi, *Numerical Methods*, S. Chand and Company, 2nd edition, Reprint 2012.
3. S. C. Gupta and V. K. Kapoor, *Fundamentals of Mathematical Statistics*, Sultan Chand and Sons Publications, 11th edition, 2012.

**II B. Tech. -I Semester**

(19BT30502) **COMPUTER ORGANIZATION**

(Common to CSE, CSBS, CSE(AI), CSE(DS), CSSE, IT)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Int. Marks | Ext. Marks | Total Marks |  | L | T | P | C |
| 40 | 60 | 100 |  | 3 | - | - | 3 |

**PRE-REQUISITES:**  A course on “Digital Logic Design”

**COURSE DESCRIPTION:**

Basic structure and operation of a digital computer; Organization and functional principles of the arithmetic and logic unit, control unit, memory unit and I/O unit; Concepts of pipelining and parallel processing techniques; Multicore computers.

**COURSE OUTCOMES:**

*After successful completion of the course, students will be able to:*

**CO1:** Analyze computer arithmetic algorithms for fixed-point and floating-point binary operations.

**CO2:** Analyze the architecture, organization and functions of the components of a digital computer.

**CO3:** Design digital circuits for the given functional description of micro operations and memory elements.

**CO4:** Investigate the performance of memory systems, I/O systems, pipelined processors and multiprocessors to evaluate the cost-performance trade-offs.

**DETAILED SYLLABUS:**

**UNITI -COMPUTER ARITHMETICANDREGISTER TRANSFER& MICROOPERATIONS (9 periods)**

**Computer Arithmetic:** Fixed point representation, Floating point representation, Addition and subtraction, Binary multiplication algorithms

**Register Transfer & Micro operations:** Register transfer, Bus and memory transfers, Arithmetic microoperations, Logic microoperations, Shift microoperations, Arithmetic logic shift unit.

**UNITII - BASIC COMPUTER ORGANIZATION AND DESIGN (8 periods)**

Instruction codes, Computer registers, Computer instructions, Instruction formats, Addressing modes, Timing and control, Instruction cycle, Input-Output and Interrupt.

**UNITIII - MICRO PROGRAMMED CONTROL AND INPUT-OUTPUT ORGANIZATION (10 periods)**

**Micro Programmed Control:** Control memory, Address sequencing, Design of control unit, Hardwired control, Microprogrammed control.

**Input-Output Organization:** Peripheral devices, Input-Output interface, Modes of transfer, Priority interrupt – Daisy chaining priority, Parallel priority interrupt, Priority encoder; Direct Memory Access, Input-Output Processor – CPU-IOP communication; PCI Express - PCI physical and logical architecture.

**UNITIV -THE MEMORY SYSTEM (9 periods)**

Semiconductor RAM memories – Internal organization, Static memories, Dynamic RAMs, Synchronous and Asynchronous DRAMs, Structure of larger memories; Read-only memories, Cache memories – Mapping functions; Nonvolatile Solid-State Memory Technologies, Solid state drives.

**UNITV - PIPELINE AND VECTOR PROCESSING, MULTIPROCESSORS, MULTICORE COMPUTERS (9 periods)**

**Pipeline & Vector Processing:** Parallel processing, Pipelining, Instruction pipeline, Vector processing, Array processors.

**Multiprocessors:** Characteristics of multiprocessors, Interconnection structures, Inter-processor arbitration.

**Multicore Computers**: Hardware performance issues, Software performance issues, Multicore organization, Intel Core i7-990X.

**Total Periods: 45**

***Topics for self-study are provided in the lesson plan***

**TEXT BOOKS:**

1. M. Morris Mano, Rajib Mall, *Computer System Architecture*, Revised 3rdEdition,Pearson Education, 2017.

2. Carl Hamacher, Zvonko Vranesic, Safwat Zaky,Naraig Manjikian,*Computer Organization and Embedded Systems*,6th Edition, McGrawHill, 2012.

**REFERENCE BOOKS:**

1. William Stallings, *Computer Organization and Architecture: Designing for Performance*, 11th Edition, Pearson Education, 2018.

2. Andrew S. Tanenbaum, Todd Austin, *StructuredComputer Organization*, 6th Edition, Pearson, 2016.

**ADDITIONAL LEARNING RESOURCES:**

* https://nptel.ac.in/courses/106105163/
* Bilkent Online Courses, Bilkent University, Lectures by William Sawyer, https://www.youtube.com/watch?v=CDO28Esqmcg

**II B. Tech. I Semester**

(19BT31201) **DISCRETE MATHEMATICAL STRUCTURES**

(Common to CSE, CSE(AI), CSE(DS), CSSE & IT)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Int. Marks | Ext. Marks | Total Marks |  |  |  |  |  |  |  |  |  |  | L | T | P | C |
| 40 | 60 | 100 |  |  |  |  |  |  |  |  |  |  | 3 | - | - | 3 |

**PRE-REQUISITES:** A Course on Transformation Techniques and Linear Algebra.

**COURSE DESCRIPTION:** Mathematical Logic; Predicates; Relations; Algebra Structures; Mathematical Reasoning; Recurrence Relations; Graphs; Graph Theory and its Applications.

**COURSE OUTCOMES:** After successfulcompletion of the course,students will be able to:

**CO1:** Demonstrate knowledge on mathematical logic and predicates.

**CO2:** Analyze sets using functions and relations.

**CO3:** Analyze properties of different algebraic structures.

**CO4:** Apply mathematical reasoning, recurrence relations, permutations and combinations to solve computational problems.

**CO5:** Apply concepts of graph theory and trees to implement computer applications.

**DETAILED SYLLABUS:**

**UNIT–I: MATHEMATICAL LOGIC AND PREDICATES (10 periods)**

**Mathematical Logic**: Statements and notations, Connectives, Well formed formulae, Truth tables, Tautology, Equivalence of formulae, Normal forms.

**Predicates:** Predicate calculus, Free and Bound variables, Rules of inference, Consistency, Proof of contradiction and Automatic Theorem Proving.

**UNIT-II: FUNCTIONS AND RELATIONS (09 periods)**

**Relations:** Properties of binary relations, Equivalence relations, Compatibility relations, Partial ordering relations, Hasse diagram and related applications.

**Functions:** Inverse Functions, Composition of functions, Recursive functions, Lattice and its Properties.

**UNIT-III: ALGEBRAIC STRUCTURES (07 periods)**

**Algebraic System:** Examples and General Properties, Semi Groups and Monoids, Groups, Subgroups, Homomorphism and Isomorphism.

**UNIT-IV: MATHEMATICAL REASONING AND RECURRENCE RELATIONS**

**(10 periods)**

**Mathematical Reasoning**: Methods of Proof, Mathematical Induction, Basics of counting, The Inclusion-Exclusion Principle, The Pigeon hole principle, Permutations and Combinations, Generalized Permutations and Combinations.

**Recurrence Relations**: Generating Functions of Sequences, Calculating coefficients of generating function, Recurrence relation, solving recurrence relations by substitution and Generating functions, Methods of Characteristic Roots, Solutions of Inhomogeneous Recurrence Relation.

**UNIT-V: GRAPH THEORY AND ITS APPLICATION (09 periods)**

**Graphs:** Introduction to Graphs, Types of Graphs, Graph basic terminology and Special types of simple graphs, Representation of Graphs and Graph Isomorphism, Euler Paths and Circuits, Hamiltonian Paths and Circuits, Planar Graphs, Euler's Formula and Graph Coloring.

**Trees:** Introduction to Trees, Properties of Trees, Applications of Trees, Spanning Trees, Counting trees, Depth-First Search, Breadth-First Search, Minimum Spanning Trees, Kruskal's Algorithm and Prim's Algorithm.

**Total Periods 45**

***Topics for self-study are provided in the lesson plan***

**TEXT BOOKS:**

1. J.P. Trembly and R. Manohar, *Discrete Mathematical Structures with Applications to Computer Science,* Tata McGraw Hill, Thirty Seventh Edition, 2017
2. Kenneth H. Rosen, *Discrete Mathematics and its Applications,* Tata McGraw Hill, Sixth Edition, 2007.

**REFERENCE BOOKS:**

1. Joe L. Mott and Abraham Kandel, *Discrete Mathematics for Computer Scientists and Mathematicians,* Prentice Hall of India Private Limited, Second Edition, 2004.
2. Ralph P. Grimaldi and B.V.Ramana, *Discrete and Combinatorial Mathematics - an Applied Introduction,* Pearson Education, Fifth Edition, 2006.

**ADDITIONAL LEARNING RESOURCES:**

1. <https://www.maa.org/press/ebooks/resources-for-teaching-discrete-mathematics>.
2. <https://www.quora.com/>

**II B. Tech. – I Semester**

(19BT31501) **DATA STRUCTURES AND ALGORITHMS**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Int. Marks | Ext. Marks | Total Marks |  | L | T | P | C |
| 40 | 60 | 100 |  | 3 | 1 | - | 4 |

**PRE-REQUISITES:** A Course on “Object Oriented Programming”

**COURSE DESCRIPTION:**

Algorithm Analysis; Linked Lists; Stacks and Queues; Trees; Binary search trees; AVL trees; Heaps; Multiway search trees; Graphs; Sorting and Searching; Hashing

**COURSE OUTCOMES:** After successful completion of this course, the students will be able to:

**CO1:** Understand the fundamental concepts of data structures, asymptotic notations and Algorithm analysis techniques to measure the performance of an algorithm.

**CO2:** Analyze performance of sorting and searching algorithms by making use of time and space complexity.

**CO3:** Design algorithms to solve societal problems by applying contextual knowledge on linked lists

**CO4:** Solve computational problems by using stacks and queues

**CO5:** Apply suitable data structure to perform operations on trees and graphs

**CO6:** Construct hash tables by using Hash functions and relevant collision resolution technique.

**DETAILED SYLLABUS:**

**UNIT I**– **Introduction, Sorting and Searching** **(11 periods)**

**Introduction:** Introduction to data structures, Introduction to Algorithm, Performance Analysis- Space Complexity, Time Complexity, Asymptotic Notation- Big Oh, Omega, Theta notations, Guidelines for Asymptotic Analysis, Algorithms Analysis: Problems & Solutions.

**Sorting:** Bubble Sort**,** Insertion sort, Selection Sort, Shell Sort, Radix sort and their performance analysis

**Searching:** Linear Search, Binary Search and their performance analysis

**UNIT II – Linked List****(8 periods)**

Single Linked List, Circular Linked List, Double Linked List, Circular Double Linked List, Applications of Linked List- Sparse Matrix Representation and its performance analysis, Addition of Polynomials and its performance analysis

**UNIT III – Stacks and Queues** **(8 periods)**

**Stacks:** Introduction, Definition, Implementation of stacks using arrays, Implementation of stacks using linked list, Applications of Stacks

**Queues:** Introduction, Definition, Implementation of queues using arrays, Implementation of queues using linked list, Circular Queue, Deque, Priority Queue, Applications of Queues

**UNIT IV – Trees, Search Trees and Heaps****(9 periods)**

**Trees:** Basic Terminologies, binary trees, Properties of binary tree, Representation of Binary Tree, Binary tree traversals.

**Search Trees:** Binary Search Trees, Operations on Binary Search Trees, AVL Trees and Operations on AVL trees

**Heap:** Heap Trees, Implementation of Heap Trees, Applications of Heap – Heap Sort and Its performance Analysis

**UNIT V – Multi way Trees, Graphs and hashing****(9 periods)**

**Multiway Trees:** M-way search trees, B-trees, Operations on B-trees, B+-trees

**Graphs:** Introduction, Basic Terminologies, Representation of Graphs, Breadth First Search and its Complexity Analysis, Depth First Search and its Complexity Analysis

**Hashing:** Introduction, Hash Table Structure, Hash Functions, Linear Open Addressing, Chaining and their performance analysis.

**Total Periods: *45***

***Topics for self-study are provided in the lesson plan***

**TEXT BOOKS:**

1. Debasis Samanta, Classic Data Structures, PHI Learning private limited, Second Edition, 2017

2. Narasimha Karumanchi, Data Structures and Algorithms made easy, Career Monk, 5th Edition, 2017

**REFERENCE BOOKS:**

1. G A V Pai, Data Structures and Algorithms: Concepts, Techniques and Applications,Mc graw Hill Edition

2. Satraj Sahani, Data Structures, Algorithms and Applications in Java, Universities Press, Second Edition, 2008

3. Michael T. Goodrich, Roberto Tamassia, Data Structures and Algorithms in java, Wiley India, Second Edition, 2007

**ADDITIONAL LEARNING RESOURCES:**

1. https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-006-introduction-to-algorithms-fall-2011/lecture-videos

2. http://nptel.ac.in/courses/106106127/

3. http://www.nptel.ac.in/courses/106102064

**II B. Tech. – I Semester**

(19BT31502) **OPERATING SYSTEMS**

(Common to CSE, CSE(AI), CSE(DS), CSSE and CSBS)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Int. Marks | Ext. Marks | Total Marks |  | L | T | P | C |
| 40 | 60 | 100 |  | 3 | - | - | 3 |

**PRE-REQUISITES:**- A course on “Computer Organization”

**COURSE DESCRIPTION:** Operating Systems Operations; Process Scheduling; Process Synchronization, Deadlocks; Paging and Segmentation, Disk Scheduling; File Concepts, I/O Interface; Concepts of Protection and Security.

**COURSE OUTCOMES:**

*On successful completion of this course, the students will be able to:*

1. Analyze performance of CPU scheduling algorithms.
2. Design solutions for process synchronization problems by using semaphores and monitors.
3. Devise solutions for deadlocks using deadlock handling mechanisms.
4. Solve memory management problems using page replacement and disk scheduling algorithms.
5. Identify efficient file allocation methods for optimal disk utilization.
6. Analyze services of I/O subsystems and mechanisms of security & protection.

**DETAILED SYLLABUS:**

**UNIT I: INTRODUCTION TO OPERATING SYSTEM AND PROCESS MANAGEMENT**

**(8 periods)**

**INTRODUCTION:** Definition, Operating System Structure and Services: Layered approach, Hybrid Systems :Mac OSX, iOS, Android, System Calls.

**PROCESS MANAGEMENT:** Process Scheduling, Process Control Block, Inter Process Communication, Threads, Multithreading Models, CPU Scheduling Criteria, Scheduling Algorithms, Multiprocessor Scheduling.

**UNIT II: PROCESS SYNCHRONIZATION AND DEADLOCKS (10 periods)**

**PROCESS SYNCHRONIZATION:** Critical Section Problem, Peterson’s Solution, Synchronization Hardware, Semaphores, Synchronization Problems, Monitors.

**DEADLOCKS:** System Model, Deadlock characterization, Methods for handling deadlocks, Prevention, Detection, Avoidance, Recovery from deadlock.

**UNIT III:MEMORY MANAGEMENT AND SECONDARY STORAGE (10 periods)**

**MEMORY MANAGEMENT:** Swapping, Contiguous Allocation, Paging, Segmentation, Segmentation with Paging.

**VIRTUAL MEMORY:** Demand Paging, Page Replacement Algorithms, Copy-on-Write, Thrashing.

**SECONDARY STORAGE STRUCTURE:** Overview of Mass Storage Structure, Disk Structure, Disk Scheduling, Disk Management.

**UNIT IV: File and I/O Systems (8 periods)**

**FILE SYSTEM:** File concept, Access Methods, Directory Structure, File System Structure,   
i-node, File System Implementation, Directory Implementation, Allocation Methods.

**I/O SYSTEM:** I/O Hardware, Application I/O Interface, Kernel I/O subsystem

**UNIT V – PROTECTION AND SECURITY (9 periods)**

**PROTECTION:** Goals, Principles, Domain of Protection, Access Matrix, Implementation of Access Matrix, Access Control, Revocation of Access Rights.

**SECURITY:** Security Problem, Program Threats, System and Network Threats, User Authentication, Implementing Security Defenses, Firewalling to Protect Systems and Networks, Computer-Security Classifications.

**Total Periods: 45**

***Topics for self-study are provided in the lesson plan***

**TEXT BOOKS**:

1. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, *Operating System Concepts*, Wiley India Edition, 9th Edition, 2016.

**REFERENCE BOOKS**:

* 1. William Stallings, *Operating Systems, Internals and Design Principles*, Pearson Education, 7th Edition, 2013.
  2. Andrew S. Tanenbaum, *Modern Operating Systems*, PHI, 3rd Edition, 2009.

**II B. Tech. – I Semester**

(19BT31503) **SYSTEM PROGRAMMING**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Int. Marks | Ext. Marks | Total Marks |  | L | T | P | C |
| 40 | 60 | 100 |  | 3 | - | - | 3 |

**PRE-REQUISITES:**- -

**COURSE DESCRIPTION:** Synchronized I/O, Direct I/O, Buffered I/O, Scatter/Gather I/O, I/O Schedulers, Device Drivers, Kernel Modules, Debugging System

**COURSE OUTCOMES:**

*On successful completion of this course, the students will be able to:*

1. Apply system programming APIs for performing operations on files.
2. Implement buffer through streams for interactive I/O management.
3. Analyze I/O performance using synchronous and asynchronous operations
4. Identify classes of device drivers and its security issues in kernel development
5. Build running modules for setting up a system using module parameters in the user space
6. Analyze different I/O operations and debugging techniques in the design of char drivers.

**DETAILED SYLLABUS:**

**UNIT I: FILE I/O (8 periods)**

Systems Programming, APIs, ABIs, Standards, Opening Files, Reading via read(),Writing with write(),Synchronized I/O, Direct I/O, Closing Files, Seeking with lseek(),Positional Reads and Writes, Truncating Files, Multiplexed I/O, Kernel Internals

**UNIT II: BUFFERED I/O (9 periods)**

User-Buffered I/O, Standard I/O, Opening Files, Opening a Stream via File Descriptor, Closing Streams, Reading from a Stream, Writing to a Stream, Sample Program Using Buffered I/O, Seeking a Stream, Flushing a Stream, Errors and End-of-File, Obtaining the Associated File Descriptor, Controlling the Buffering, Thread Safety, Critiques of Standard I/O

**UNIT III: ADVANCED FILE I/O (8 periods)**

Scatter/Gather I/O, The Event Poll Interface, Mapping Files into Memory, Advice for Normal File I/O, Synchronized, Synchronous, and Asynchronous Operations, I/O Schedulers and I/O Performance

**UNIT IV:DEVICE DRIVERS AND RUNNING MODULES (10 periods)**

The role of Device Driver, Splitting the kernel, Classes of Devices and Modules, Security Issues. Building and Running Modules: Kernel Modules versus Applications,The Kernel Symbol Table, Preliminaries, Initialization and shutdown, Doing it in user space.

**UNIT V: CHAR DRIVERS AND DEBUGGING TECHNIQUES (10 periods)**

The Design of Scull, Some Important Data Structures, Char Device Registration, Open and Release, Scull's Memory Usage, Read and Write. Debugging Techniques: Debugging support in the Kernel, Debugging System faults, Debuggers and Related Tools.

**Total Periods: 45**

***Topics for self-study are provided in the lesson plan***

**TEXT BOOKS**:

* 1. Robert Love, Linux System Programming, O'Reilly Media, 2007.
  2. Jonathan Corbet, Alessandro Rubini, Greg Kroah-Hartman, LINUX Device Drivers, O'Reilly Media, Year: 2005

**REFERENCE BOOKS**:

1.Dayanand Ambawade, Deven Shah, "Linux Labs and Open Source Technologies", Dreamtech Press, 2014.

2.Venkateswaran S, “Essential Linux Device Drivers”, Pearson Education, 2008

**II B. Tech. – I Semester**

(19BT31531) **DATA STRUCTURES AND ALGORITHMS LAB**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Int. Marks | Ext. Marks | Total Marks |  | L | T | P | C |
| 50 | 50 | 100 |  | - | - | 2 | 1 |

**PRE-REQUISITES:** A course on “Data Structures and Algorithms”

**COURSE DESCRIPTION:**

Sorting and Searching; Linked Lists; Stacks and Queues; Binary Search Trees; AVL trees; Graph Traversing Techniques; Collision Resolution Techniques

**COURSE OUTCOMES:** After successful completion of this course, the students will be able to:

**CO1:** Implement sorting and searching algorithms using suitable data structure.

**CO2:** Develop algorithms to solve real time problems using Linked lists

**CO3:** Solve computational problems using stacks and queues

**CO4**: Develop algorithms to perform operations on trees and graphs

**CO5**: Build solution for collisions in hash tables using suitable data structure

**CO6**: Work independently and in team to solve problems with effective communication

**List of Exercises/List of Experiments:**

**1.** Implement following sorting algorithms

**a)** Bubble Sort

**b)** Insertion sort

**c)** Selection sort

**2.** Store roll numbers of students who attended placement training program in random order in an array.

**a)** Write a program to search whether a particular student attended training or not using linear search

**b)** Write a program to search whether a particular student attended training or not using binary search

**3.** **a)** Department of CSSE has readers club named 'Prerana'. Students of all years can be granted membership on request and they can get books. Similarly one may cancel the membership of club. First node is reserved for head of readers club and last node is reserved for in-charge of readers club. The student’s information in each node consisting of name of the student and roll no of the student. Develop a program to perform following operations on readers club member‘s information using singly linked list.

1. Add and delete the members as well as head or even in-charge.
2. Compute total number of members in readers club
3. Display members in readers club
4. Display list in reverse order using recursion
5. Sort the list using name and display it.

**b)** A Company has N employees and it maintains each employee data with the following attributes like: emp\_id, emp-dept,emp\_sal, emp\_mobileno. Use a menu driven Program to perform following operations on employee’s data using DoublyLinked List (DLL).

1. Create a DLL of N Employees Data by using end insertion.
2. Display the status of DLL and count the number of nodes in it
3. Perform Insertion and Deletion at End of DLL
4. Perform Insertion and Deletion at Front of DLL
5. Perform Insertion and Deletion at any user specified position of DLL
6. Exit

**4**. **a)** Implement a menu driven Program for the following operations on stack using arrays.

1. Push an Element on to Stack
2. Pop an Element from Stack
3. Demonstrate how Stack can be used to check Palindrome
4. Display the elements of a Stack
5. Exit

**b)** Develop a menu driven program to implement queue operations using arrays

**5. a)** Write a program to implement stack using linked list

**b)** Write a program to implement queue using linked list

**6. a)** Develop a program to convert an infix expression to postfix expression using stack

**b)** Write a program to evaluate given postfix expression using stack

**7.** Develop a menu driven program to perform the following operations on a binary search tree

**a)** Create a binary search tree

**b)** Insert an element into a binary search tree

**c)** Delete an element from binary search tree

**d)** Traverse the binary search tree in Inorder, Preorder and post order

**8.** Write a program to perform the following operations on AVL tree

**a)** Insert an element into AVL tree

**b)** Delete an element from AVL tree

**c)** Display the elements of AVL tree in ascending order

**9**. **a)** Develop a program to implement Breadth first search traversal.

**b)** Develop a program to implement Depth first search traversal.

**10.** Write a program to implement hashing with

**a)** Separate Chaining Method

**b)** Open Addressing Method

**REFERENCE BOOKS/LABORATORY MANUALS**:

1. Debasis Samanta, Classic Data Structures,PHI Learning private limited, Second Edition, 2017

2. Robert Lafore, Data Structures & Algorithms in Java, Second Edition, Pearson Education (2008)

**SOFTWARE/Tools used:**

**Software:** JDK 1.8

**Operating System:** Windows/ Linux

**ADDITIONAL LEARNING RESOURCES:**

1. https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-006-introduction-to-algorithms-fall-2011/lecture-videos

2. http://nptel.ac.in/courses/106106127/

3. http://www.nptel.ac.in/courses/106102064

**II B. Tech. – I Semester**

(19BT31532) **OPERATING SYSTEMS LAB**

(Common to CSE, CSBS, CSE(AI), CSE(DS), CSSE)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Int. Marks | Ext. Marks | Total Marks |  | L | T | P | C |
| 50 | 50 | 100 |  | - | - | 2 | 1 |

**PRE-REQUISITES:** A course on “Operating Systems.”

**COURSE DESCRIPTION:** Hands-on practice in simulating algorithms for CPU Scheduling, Memory Management, I/O Management, Deadlock Handling mechanisms; Implementing Synchronization problems;

**COURSE OUTCOMES:** *On successful completion of this course, the students will be able to:*

1. Analyze process scheduling problems by applying contextual knowledge on CPU scheduling algorithms.
2. Apply memory management and disk scheduling algorithms to attain optimal solutions.
3. Devise solution for deadlock avoidance using banker’s algorithm.
4. Design solutions for process synchronization problems using semaphores and monitors.
5. Apply file allocation strategies to achieve optimal disk utilization.
6. Work independently and in team to solve problems with effective communication

**LIST OF EXPERIMENTS:**

|  |  |
| --- | --- |
| 1 | Simulate the following  a) Process System Calls.  b) I/O System Calls. |
| 2 | Simulate multi-level queue scheduling algorithm by considering the following scenario. All the processes in the system are divided into two categories – system processes and user processes. System processes are to be given higher priority than user processes. The priority of each process ranges from 1 to 3. Use fixed priority scheduling for all the processes. |
| 3 | Demonstrate File Permissions. |
| 4 | Simulate the following CPU Scheduling Algorithms: |
|  | a) FCFS b) SJF (Preemptive) c) Priority d) Round Robin |
| 5 | Design solutions for the following synchronization problems: |
|  | a) Producer Consumer Problem b) Dining Philosophers Problem. |
| 6 | Design Banker’s Algorithm for Deadlock Avoidance. Find the safe sequence. If Maximum request of any one process is changed, detect whether a deadlock has occurred or not. Consider the number of resources are three and Jobs are five. |
| 7 | Simulate the following Algorithms: |
|  | a) First Fit b) Best Fit c) Worst Fit |
| 8 | Simulate the following Page Replacement Algorithms |
|  | a) FIFO b) LFU c) LRU d) Optimal |
| 9 | Simulate the following Disk Scheduling Algorithms |
|  | a) FCFS b)SSTF c) SCAN d) CSCAN |
| 10 | Simulate the following file allocation strategies: |
|  | a) Contiguous Allocation b) Linked Allocation |

**REFERENCE BOOKS**:

1. Herbert Schildt, *Java the Complete Reference*, Ninth Edition, Oracle Press, 2014.

2. Sachin Malhotra and Saurab Choudhary, *Programming in Java*, Second Edition, Oxford University press, 2014.

**SOFTWARES/TOOLS USED:**

* Software:J2SDK1.7

-Eclipse or NetBeans IDE

* Java compatible web browser

**II B. Tech. – I Semester**

(19BT31533) **WORKSHOP IN COMPUTER SCIENCE AND SYSTEMS ENGINEERING**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Int. Marks | Ext. Marks | Total Marks |  | L | T | P | C |
| 50 | 50 | 100 |  | - | - | 2 | 1 |

**PRE-REQUISITES:** A course on “Programming for Problem Solving”

**COURSE DESCRIPTION:**

Hands-on practice PC Hardware; Installation of Operating System; Software and Hardware Troubleshooting; Microsoft Office- Word and Excel; C Language- Operators, Expressions, Decision Making Statements, Looping Statements, Arrays and Functions.

**COURSE OUTCOMES:**

*On successful completion of this course, the students will be able to:*

1. Devise solutions to the basic problems using C Language constructs.
2. Solve problems by applying functions, structures, dynamic memory allocation and pointers.
3. Develop, maintain and modify Web pages effectively using markdown.
4. Design personal portfolioin customized style by using git and Jekyll themes.
5. Build simple mobile applications using MIT App inventor.
6. Work independently and communicate effectively in oral and written forms.

**LIST OF EXPERIMENTS:**

**I: C Programming:**

|  |  |  |
| --- | --- | --- |
| **1** | **A** | Write a program to evaluate the following algebraic expressions after reading necessary values from keyword.  i) (ax+b)/(ax–b) ii) 2.5logx+Cos32o+|x2+y2|+2xy  iii) x5+10x4 +8x3+4x+2 iv) aekt |
|  | **B** | Write a program to calculate commission for the input value of sales amount. Commission is calculated as per the following rules:  i) Commission is **NIL** for sales amount Rs. 5000.  ii) Commission is 2% for sales when sales amount is >Rs. 5000 and <= Rs. 10000.  iii) Commission is 5% for sales amount >Rs. 10000. |
| **2** | **A** | Write a program to find the grace marks for a studentusing switch. The user should enter the class obtained bythe student and the number of subjects he has failed in.  Use the following rules:  i. If the student gets first class and the number of subjectsfailed is >3, then no grace marks are awarded. If thenumber of subjects failed is less than or equal to '3' thenthe grace is 5 marks per subject.  ii. If the student gets second class and the number of subjectsfailed in is >2, then no grace marks are awarded. If thenumber of subjects failed in less than or equal to '3' thenthe grace is 4 marks per subject.  iii. If the student gets third class and the number of subjectsfailed in is >1, then no grace marks are awarded. If thenumber of subjects failed in is equal to '1' then the graceis 5 marks per subject |
|  | **B** | Write a program to find the sum of individual digits of a positive integer using for loop |
| **3** | **A** | Write a program to generate all the prime numbers between 1 and N using while loop |
|  | **B** | Write a program to generate Fibonacci sequence for N numbers using do-while loop. |
| **4** | **A** | Write a program to perform the following: i) Addition of two matrices.  ii) Multiplication of two matrices. |
|  | **B** | Write a program to implement (i) Call by value (ii) Call by reference. |
| **5** | **A** | Write a program to find factorial of a given number using recursion. |
|  | **B** | Write a program that uses functions to perform the followingoperations:  Write a program to determine whether the given string ispalindrome or not. |
| **6** | Define a structure to store employee's data with the following specifications:  Employee-Number, Employee-Name, Basic pay, Date of Joining  i. Write a function to store 10 employee details.  ii. Write a function to implement the following rules while revising the basic pay.  If Basic pay<=Rs.5000 then increase it by 15%.  If Basic pay> Rs.5000 and <=Rs.25000 then it increase by 10%.  If Basic pay>Rs.25000 then there is no change in basic pay.  Write a function to print the details of employees who have completed 20 years of service from the date of joining. | |
| **7** | **A** | Write a Program to calculate the sum of n numbers entered by the user using dynamic memory allocation functions. |
|  | **B** | Write a Pointer Program to swap two numbers without using the 3rd variable. |

**II: Source Code Management Using Git and GitHub:**

|  |  |
| --- | --- |
| **8** | Installing Git, Configuring Git, Creating a Git repository, Creating and editing files, Adding files to Git repository, Making changes and tracking them, Synchronizing local Git repository with GitHub, Deleting and renaming files. |
| **9** | Markdown Syntax: Adding text of various styles and formats, Adding images along with text, Creating Ordered/Unordered list,  Adding videos/ pdfs to the markdown file, adding links in the markdown file. |
| **10** | GitHub Pages: Creation of personal portfolio site- Creating a GitHub Page using Markdown and Jekyll themes forrepositories. |

**III: Build Apps with MIT App Inventor:**

|  |  |
| --- | --- |
| **11** | Building the App “HelloCodi”: Select components to design app, Programming with the Blocks Editor, Playing the Sound, Packaging app. |
| **12** | Building the simple Game APP “BallBounce”. |
| **13** | Building the Drawing App “DigitalDoodle”. |

**REFERENCE BOOKS**:

**MIT App Inventor:**

1. <http://appinventor.mit.edu/>

**GIT Hub:**

1. Scott chacon, Ben Straub, “**Pro Git**”, Second Edition, APress open, 2014.

**C LANGUAGE:**

1. Behrouz A. Forouzan and Richard F. Gilberg, “**A Structured Programming Approach using C**,” Third Edition, Cengage Learning, New Delhi, 2007.
2. PradipDey and ManasGhosh, “**Programming in C**,” Second Edition, Oxford University Press, New Delhi, 2007.

**SOFTWARES/Tools used:**

System Software:C Compiler/Code Blocks, MIT App Inventor, Git hub.

**II B.Tech. I Semester**

(19BT315AC)**DESIGN THINKING**

(Common to CSE, IT, CSSE, CSBS, CSE (AI), CSE (DS))

(Audit Course)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Int. Marks | Ext. Marks | Total Marks |  | L | T | P | C |
| - | - | - |  | 2 | - | - | - |

**PRE-REQUISITES: -**

**COURSE DESCRIPTION:** Design thinking process, Design thinking phases, empathy tools; Idea generation, visualizing and empathizing; Fidelity for prototypes, prototyping; prototyping for physical products.

**COURSE OUTCOMES:** After successful completion of this course, the students will be able to:

**CO1:** Analyze design thinking concepts and principles to perform human centered design process for creative problem solving.

**CO2:** Create empathy maps to visualize user attitudes and behavior for gaining insights of customers.

**CO3:** Develop innovative products or services for a customer base using ideation techniques.

**CO4:** Build prototypes for complex problems using gathered user requirements.

**CO5:** Apply design thinking tools techniques to produce good design and relevant products or services for a specific target market.

**CO6:** Improve prototype by testing it with a specific set of users for making it sustainable by following ethics.

**DETAILED SYLLABUS:**

**UNITI: INTRODUCTION TO DESIGN THINKING (6 Periods)**

Design Thinking Process: Types of the thinking process, Common methods to change the human thinking process, Design thinking: Definition, Origin of design thinking, Importance of design thinking, Design vs Design thinking, Problem solving, Understanding design thinking and its process model, Design thinking tools.

**UNIT II: EMPATHIZE (6 Periods)**

Design thinking phases, How to empathize, Role of empathy in design thinking, purpose of empathy maps, Things to be done prior to empathy mapping, Activities during and after the session, Understanding empathy tools : Customer Journey Map, Personas.

**UNIT III: IDEATION** **(6 Periods)**

Challenges in idea generation, need for systematic method to connect to user, Visualize, Empathize, and Ideate method, Importance of visualizing and empathizing before ideating, Applying the method, Ideation Tools: How Might We? (HMW), Story board, Brainstorming.

**UNITI V: PROTOTYPING** **(6 Periods)**

What is a prototype? - Prototyping as a mindset, prototype examples, prototyping for products; Why we prototype? Fidelity for prototypes, Process of prototyping- Minimum Viable prototype

**UNIT V: TESTING PROTOTYPES (6 Periods)**

Prototyping for digital products: What’s unique for digital, Preparation; Prototyping for physical products: What’s unique for physical products, Preparation; Testing prototypes with users.

**Total Periods: 30**

***Topics for self-study are provided in the lesson plan***

**TEXTBOOKS:**

1. S.Salivahanan, S.Suresh Kumar, D.Praveen Sam, “Introduction to Design Thinking”,Tata Mc Graw Hill, First Edition,2019.
2. Kathryn McElroy, “Prototyping for Designers: Developing the best Digital and Physical Products”, O’Reilly,2017.

### REFERENCE BOOKS:

1. [Michael G. Luchs](https://www.amazon.com/s/ref=dp_byline_sr_book_1?ie=UTF8&field-author=Michael+G.+Luchs&text=Michael+G.+Luchs&sort=relevancerank&search-alias=books), [Scott Swan](https://www.amazon.com/s/ref=dp_byline_sr_book_2?ie=UTF8&field-author=Scott+Swan&text=Scott+Swan&sort=relevancerank&search-alias=books) , [Abbie Griffin](https://www.amazon.com/s/ref=dp_byline_sr_book_3?ie=UTF8&field-author=Abbie+Griffin&text=Abbie+Griffin&sort=relevancerank&search-alias=books),”Design Thinking – New Product Essentials from PDMA”, Wiley, 2015.
2. Vijay Kumar, “[101 Design Methods: A Structured Approach for Driving Innovation in Your Organization](http://www.amazon.com/101-Design-Methods-Structured-Organization/dp/1118083466)”, 2012.

**ADDITIONAL LEARNING RESOURCES:**

1. <https://www.interaction-design.org/literature/article/5-stages-in-the-design-thinking-process>
2. <https://www.ibm.com/design/thinking/page/toolkit>
3. <https://www.interaction-design.org/literature/article/define-and-frame-your-design-challenge-by-creating-your-point-of-view-and-ask-how-might-we>
4. <https://www.culturepartnership.eu/en/article/ten-tools-for-design-thinking>
5. <https://nptel.ac.in/courses/109/104/109104109/>
6. <https://nptel.ac.in/courses/110106124/>