II B. Tech. - I Semester

(19BT3BS01) NUMERICAL METHODS, PROBABILITY AND STATISTICS

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

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 $1/3^{rd}$ and $3/8^{th}$ 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

(19BT31201) DISCRETE MATHEMATICAL STRUCTURES

(Common to CSE, CSSE and 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 successful completion 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 (9 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 (7 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 (9 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

(19BT31502) **OPERATING SYSTEMS**

(Common to CSE and CSSE)

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

CO1. Analyze performance of CPU scheduling algorithms.

CO2. Design solutions for process synchronization problems by using semaphores and monitors.

CO3. Devise solutions for deadlocks using deadlock handling mechanisms.

CO4. Solve memory management problems using page replacement and disk scheduling algorithms.

CO5. Identify efficient file allocation methods for optimal disk utilization.

CO6. 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 BOOK(S):

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

(19BT30501) **COMPUTER GRAPHICS**

(Common to CSE and 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: Introduction to computer graphics; Output primitives; 2D geometric transformations and viewing; 3D geometric transformations and viewing; 3D object representation; Illumination models; Visible surface detection methods and rendering methods.

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

CO1. Analyze raster scan and random scan systems by applying knowledge on graphical interactive devices.

CO2. Design algorithms for output primitives such as lines, circles and filled area primitives to fill specified area.

CO3. Analyze the concepts of geometrical transformations, representations, and viewing for 2D and 3D objects.

CO4. Apply appropriate techniques for visible surface detection, illumination models and rendering methods.

DETAILED SYLLABUS:

UNIT- I: INTRODUCTION AND OUTPUT PRIMITIVES (9 Periods)

Introduction to Graphics Systems: Video display devices, Raster-scan systems, Random-scan systems, Graphics monitors and workstations, Input devices.

Output Primitives: Points and lines, Line drawing algorithms, Midpoint circle algorithm.

UNIT- II: FILLED AREA PRIMITIVES AND 2D GEOMETRIC TRANSFORMATIONS (9 Periods)

Filled Area Primitives: Scan-line polygon fill algorithm, Boundary-fill algorithm and Flood-fill algorithm.

2D Geometric Transformations: Transformations – translation, scaling, rotation, reflection and shear; Homogeneous coordinates, Composite transformations, Transformations between coordinate systems.

UNIT- III: 2D VIEWING AND 3D OBJECT REPRESENTATIONS (9 Periods)

2D Viewing: The viewing pipeline, Viewing coordinate reference frame, Window-to-viewport coordinate transformation, Cohen-Sutherland line clipping algorithm, Sutherland-Hodgeman polygon clipping algorithm.

3D Object Representations: Polygon surfaces, Quadric surfaces, Spline representations, Hermite curve, Bezier curves and surfaces.

UNIT- IV: 3D GEOMETRIC TRANSFORMATIONS AND VIEWING (7 Periods)

3D Geometric Transformations: Translation, Rotation, Scaling, Reflection and Shear.

3D Viewing: Viewing pipeline, Viewing coordinates, Projections and Clipping.

UNIT-V:ILLUMINATION MODELS, VISIBLESURFACE DETECTION AND SURFACERENDERING METHODS (11 Periods)

Illumination Models: Ambient Light and Diffuse Reflection.

Visible Surface Detection Methods: Classification, Back-face detection, Depth-buffer method, Scan-line method, Depth-sorting method and BSP-tree method.

Surface Rendering methods: Gouraud shading and Phong shading.

Total Periods: 45

Topics for self-study are provided in the lesson plan

TEXT BOOK(S):

1. Donald D. Hearn and M. Pauline Baker, *Computer Graphics C version*, Pearson Education, Second Edition, 2006.

REFERENCE BOOKS:

1. Fabio Ganovelli, Massimiliano Corsini, Sumanta Pattanaik, Marco Di Benedetto, *Introduction to Computer Graphics: A Practical Learning Approach*, CRC Press, 2014.
2. Apurva A. Desai, *Computer Graphics*, PHI Learning, 2008.
3. William M. Newman and Robert F. Sproull, *Principles of Interactive Computer Graphics*, McGraw-Hill, Second Edition, 2005.

ADDITIONAL LEARNING RESOURCES:

1. Sumanta Guha, *Computer Graphics through OpenGL*, CRC Press, Second Edition, 2015.
2. https://nptel.ac.in/courses/106/106/106106090/

II B. Tech. - I Semester

(19BT30502) **COMPUTER ORGANIZATION**

(Common to CSE, CSSE and 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:

UNIT-I: COMPUTER ARITHMETIC,REGISTER TRANSFER ANDMICROOPERATIONS (9 Periods)

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

Register Transfer and Microoperations: Register transfer, Bus and memory transfers, Arithmetic microoperations, Logic microoperations, Shift microoperations, Arithmetic logic shift unit.

UNIT- II: 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.

UNIT-III: MICROPROGRAMMED CONTROL AND INPUT-OUTPUT ORGANIZATION (10 Periods)

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

UNIT- IV: 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.

UNIT-V: PIPELINE AND VECTOR PROCESSING, MULTIPROCESSORS, MULTICORE COMPUTERS (9 Periods)

Pipeline and 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 3rd Edition, Pearson Education, 2017.
2. Carl Hamacher, Zvonko Vranesic, SafwatZaky, Naraig Manjikian, *Computer Organization and Embedded Systems*, 6th Edition, McGraw Hill, 2012.

REFERENCE BOOKS:

1. William Stallings, *Computer Organization and Architecture: Designing for Performance*, 11th Edition, Pearson Education, 2018.
2. Andrew S. Tanenbaum, Todd Austin, *Structured Computer 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

(19BT30503) **DATA STRUCTURES**

(Common to CSE and IT)

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

PRE-REQUISITES: A course on "Object Oriented Programming through Java"

COURSE DESCRIPTION: Introduction to data structures; Linked Lists; Types of lists; Stacks; Queues; Trees; Binary search trees; AVL trees; Red-Black Trees; Searching algorithms; Sorting algorithms; Graphs; Minimum spanning trees; Hashing.

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

CO1. Analyze linear data structures such as arrays, linked lists, stacks, queues for efficient data organization and manipulation.

CO2. Analyze data structures such as trees, graphs, hash tables for efficient search and retrieval of data.

CO3. Select and apply appropriate techniques for searching and sorting problems.

CO4. Apply knowledge to select appropriate data structures for modeling information in data.

DETAILED SYLLABUS:

UNIT- I: LINKED LISTS (7 Periods)

Overview of data structures and algorithms, Linear and non-linear data structures, Big O notation, Linked lists – Definition, Operations; Single linked lists, Circular linked lists, Doubly linked lists, Sorted lists, Linked lists efficiency, Applications of linked lists.

UNIT- II: STACKS AND QUEUES (9 Periods)

Stacks: Definition, Operations, Implementation using arrays and linked lists, Applications – Reversing a word, Delimiter matching, Parsing arithmetic expressions.

Queues: Definition, Operations, Applications, Implementation using arrays and linked lists, Circular queue, Double-ended queues, Priority queues.

UNIT- III: BINARY TREES AND SEARCH TREES (11 Periods)

Tree terminology, Binary trees, Trees represented as arrays, Binary search trees - Concepts, Advantages, Operations, Finding maximum and minimum values, Efficiency; Balanced and unbalanced trees, AVL search trees – Concepts, Operations; Red-Black trees – Concepts, Rotations, Inserting a node, Efficiency.

UNIT- IV: SEARCHING AND SORTING (9 Periods)

Searching: Linear search, Binary search.

Sorting: Bubble sort, Selection sort, Insertion sort, Sorting objects, Shell sort, Partitioning, Quick sort, Merge sort, Heap sort.

UNIT- V: GRAPHS AND HASHING (9 Periods)

Graphs: Concepts, Representation, Operations, Depth-first search, Breadth-first search, Minimum spanning trees.

Hashing: Introduction, Open addressing, Separate chaining, Characteristics of good hash functions – Quick computation, Random and Non-random keys, Folding; Hashing efficiency.

Total Periods: 45

Topics for self-study are provided in the lesson plan

TEXT BOOK(S):

1. Robert Lafore, *Data Structures & Algorithms in Java*, 2nd Edition, Pearson, 2007.

REFERENCE BOOKS:

1. Goodrich, Tamassia, Goldwasser, *Data structures & Algorithms in Java*, 6th Edition, Wiley, 2014.
2. John R. Hubbard, *Programming with Java*, 2nd Edition, McGraw Hill, 2009.
3. Debasis Samanta, *Classic Data Structures*, 2nd Edition, Prentice Hall, 2009.

ADDITIONAL LEARNING RESOURCES:

- http://www.nptel.ac.in/courses/106102064
- http://nptel.ac.in/courses/106106127/
- https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-006-introduction-to-algorithms-fall-2011/lecture-videos

II B. Tech. – I Semester

(19BT31532) **OPERATING SYSTEMS LAB**

(Common to CSE and 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: *After successful completion of the course, students will be able to:*

CO1. Analyze process scheduling problems by applying contextual knowledge on CPU scheduling algorithms.

CO2. Apply memory management and disk scheduling algorithms to attain optimal solutions.

CO3. Devise solution for deadlock avoidance using banker's algorithm.

CO4. Design solutions for process synchronization problems using semaphores and monitors.

CO5. Apply file allocation strategies to achieve optimal disk utilization.

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

LIST OF EXERCISES:

1. Simulate the following:
   a)Process System Callsb) 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.

SOFTWARE/TOOLS USED:

- Software: J2SDK 1.7
  - Eclipse or Net bean
- Java compatible web browser

II B. Tech. - I Semester

(19BT30531) DATA STRUCTURES LAB

(Common to CSE and IT)

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

PRE-REQUISITES:A course on "Data Structures"

COURSE DESCRIPTION: Hands on practice on implementation of Linked lists; Arrays; Stacks; Queues; Search algorithms; Sorting algorithms; Binary search tree representation and operations; Graph representation and operations; Hashing functions.

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

CO1. Implement linear data structures such as arrays, linked lists, stacks, queues for efficient data organization and manipulation.

CO2. Develop solutions using data structures such as trees, graphs, heaps, hash tables for efficient search and retrieval of data.

CO3. Select and apply appropriate techniques for searching and sorting problems.

CO4. Work independently and communicate effectively in oral and written forms.

LIST OF EXERCISES:

1. A college has N number of students and the following details of all the students are maintained – register number, name, branch, phone number. Write a program to store the details of the students using a singly linked list. Develop functions to perform the following operations on the data.
    a) Insert new student's details
    b) Display the details of the students
    c) Display the total number of students
    d) Delete a given student's information
2. Department of CSE has readers club named 'Aalochana'. Students can be granted membership in readers club on their request. Similarly, one may cancel their membership of the club. Members of the club can rent books from the club. Write a program to create data structure to maintain readers club members information (Hall ticket number, name) using singly linked list. In singly linked list, the header node should store details of head of readers club and last node should store details of in-charge of readers club. Develop functions to perform the following operations on the data.
    a) Store details of head and in-charge of the readers club
    b) Grant and cancel memberships of students
    c) Display total number of members
    d) Display the details of the members
    e) Display the sorted list of details of the members (sort based on their names in alphabetical order)
3. A company has N number of employees and it maintains the following details of each of its employees: ID, department, salary, phone number. Develop a menu driven

program using doubly linked list to store the employees' data. Develop functions to perform the following operations on the data.

a) Add and delete employees
b) Display total number of employees
c) Display details of employees with salary more than Rs. 50,000
d) Display the phone number of the employee given the ID

4. a) Develop a menu driven program to perform the following operations on a stack of integers (Array and linked list implementations of stack with maximum size MAX)

i) Push an element
ii) Pop an element
iii) Display the status
iv) Demonstrate overflow and underflow situations (in array implementation)

b) Write a program to check whether a string is palindrome or not using stack data structure.

c) Mostly syntax errors in a computer program arise due to unbalanced braces (such as (), {}, []). Write a program using stack to check whether a given expression has balanced braces or not.

5. a) Develop a menu driven program to perform the following operations on a queue of characters (Array and linked list implementations of queue with maximum size MAX)

i) Insert an element
ii) Delete an element
iii) Display the status
iv) Demonstrate overflow and underflow situations (in array implementation)

b) A restaurant based on its human resources can accept a maximum of N number of food orders. The food orders are served in first come first serve basis. The food orders once placed cannot be cancelled. Write a program to simulate the food ordering and serving system in the restaurant using circular queue.

6. Write a program to perform the following operations on the binary search tree.

a) Construct binary search tree by inserting the values {6, 9, 5, 2, 8, 15, 24, 14, 7, 8, 5, 2} in the given order.
b) Display the nodes of the tree using inorder, preorder and postorder traversal techniques.
c) Display the smallest number stored in the tree.
d) Search the tree for a given number.

7. There are train paths between cities. If there is a train between city A and city B then there is a route between the cities. The cost of the route is the distance between city A and city B. Represent the train travel route information as a graph. The node can be represented by the name of the city. Write a program to perform the following operations.

a) Store the details of train travel route information using adjacency list or adjacency matrix representation.

b) Traverse the graph and display the details of all trains between the cities along with the cost using breadth-first method.

c) Traverse the graph and display the details of all trains between the cities along with the cost using depth-first method.

8. Store register numbers of students who attended placement training program in a random order in an array. Write a function to search whether a student has attended placement training program or not using
   a) Linear search
   b) Binary search

9. Write a program to sort a given set of integers using
   a) Quick sort
   b) Shell sort

10. a) Write a program to sort a given set of integers using merge sort.

    b) Write a program to read the marks obtained by students in a mathematics examination and store the data using a heap data structure. Find out the maximum and minimum marks obtained by the students.

11. Write a program to implement the following hashing functions.
    a) Separate Chaining Method
    b) Open Addressing Method

12. Consider an online movie ticket booking system through which customers can book tickets to watch movies at theatres. The database stores the details of each transaction of ticket booking with the details - ID, customer name, customer phone number, movie name, theatre name, date of show, time of show, number of tickets booked, starting seat number, total amount. Write a menu driven program to perform create the database and given an ID, display a client's phone number. Use a hash table implementation to quickly search through the database.

REFERENCE BOOKS:

1. Robert Lafore, *Data Structures & Algorithms in Java*, 2nd Edition, Pearson, 2007.
2. Goodrich, Tamassia, Goldwasser, *Data structures & Algorithms in Java*, 6th Edition, Wiley, 2014.

SOFTWARE/TOOLS USED:

- Software: J2SDK 1.7
  - Eclipse or Net bean
- Java compatible web browser

II B. Tech. - I Semester

(19BT30532) **WORKSHOP IN COMPUTER SCIENCE**

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 sessions on Google cloud productivity and collaboration tools; AI Tools for speech recognition, language detection and object detection; Introduction to GitHub; Programming in C language.

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

CO1. Use Google productivity and collaboration tools for effective documentation and data sharing.

CO2. Use AI tools for speech recognition, language detection and object detection.

CO3. Use online code hosting platforms such as GitHub for hosting and collaborating software projects.

CO4. Develop modular programs in C programming language to solve engineering problems.

CO5. Work independently and communicate effectively in oral and written forms.

LIST OF EXERCISES:

Cloud Based Productivity and Collaboration Tools:

1. Introduction to Google drive, Google Backup & Sync, Google Docs, Google Forms.
   a) Store, sync, and share files in the cloud using Google Drive, Google Backup & Sync tools.
   b) Create and edit text documents online using Google Docs.
   c) Create Google forms to manage event registrations/surveys/quizzes and analyze responses.

AI Tools:

2. Speech Recognition: Read an audio file with Python and use the Google speech recognition API to perform conversion of Speech to Text.
3. Language Detection: Detect the language of the text using language detection library (langdetect) ported from Google's language-detection.
4. Object Detection: Detect multiple objects present in an image using Detectron-Facebook's API.

GitHub:

5. a) Introduction to GitHub basic concepts and Flow.
   b) GitHub Installation and Setup.

c) Create a "Hello World" project on GitHub by performing the following operations
- Create and use a repository
- Create and manage a new branch
- Perform and commit changes to a file
- Open and merge pull requests

Programming in C:

The C Character set, Identifiers and Keywords, Data types, Constants, Variables, Declarations, Expressions, Statements, Operators, Datatypes, Input-Output statements, Control statements, Arrays, Strings, Functions, Structures.

6. a) Write a C program to solve the logical implication P → Q = !P ^ Q. Consider various combinations of binary values for P and Q.

Hint:

P	Q	!P ^ Q
0	0	0
0	1	1
1	0	0
1	1	0

b) Mr. X is a superstitious person. He also likes to double everything he has been given and his lucky number is the one which is divisible by 2. Given a number, write a C program to find out whether the number is Mr. X's lucky number or not. Analyze the program carefully and find out test cases where the given input number is lucky and where the input number is not lucky.

7. a) Write a C program to evaluate the following expression.

$Y = 1.5X^2 + 5$ if $0 <= X <= 2$

$Y = 2X^2 + 5$ otherwise

b) The evaluation schema for a subject in an engineering course is as follows. There will be two mid-term examinations to be attended by the student. Student has to attempt an objective paper, essay paper and an assignment for each mid-term exam. The details of the division of marks for the exams is given in the following table. The final marks of the student in a subject is the sum of 80% of the best marks and 20% of the other out of the two mid-term examinations.

Mid-Term Examination	Examination Schema	Max. Marks		Final Marks in the subject
Mid-Term 1	Objective Paper	10M	25M	80% of the best mid-term + 20% of the other mid-term
	Essay Paper	10M		
	Assignment	05M		
Mid-Term 2	Objective Paper	10M	25M	
	Essay Paper	10M		
	Assignment	05M		

Given the marks of the student in each paper, write a C program to find out his final marks.

8. a) A pharmaceutical corporation is considering taking applicant interviews. The interviews will be offered to applicants who meet certain educational specifications. An applicant who meets any of the following criteria should be accepted for an interview:

   i) Graduates over 25 who studied chemistry and who did not graduate from X university.

   ii) Graduates from X University who studied chemistry.

   iii) Graduates from Y University who studied pharmacy and are not older than 28.

   iv) Graduates from X University who are over 25 and who didn't study chemistry

   Hint: Use menu-based strategy to know the user's qualifications.

   b) Write a C program to confirm whether a person is Lardy or Hardy or None depending upon his height and weight.

Height	**Weight**	**Person**
5' 5"	50kg to 60kg	Lardy
Less than 5' 5"	Greater than 90kg	Hardy
Otherwise		None

9. a) a) Write a C program that takes in a class ID of a ship and displays the equivalent string class description of the given class ID. Use the details given in the table below.

Class ID	**Ship Class**
B or b	Battle Ship
C or c	Cruiser
D or d	Destroyer
F or f	Frigate

   b) Given two rolling dices, write a C program to display all the possible output values when the dices are rolled. Hint: Each dice has values from 1 to 6.

10. Write a C program to read two strings and display their difference. Let the two strings be S1 and S2. Let the strings be of equal length 'n'. The difference between S1 and S2 is given as

$$\sum_{i=1}^{n} |S1_i - S2_i|$$

11. Department of Telecom is maintaining the following details for each of the customers: Customer Name, Telephone number, Phone Registration Address and Billing Address. Write a C program to read and store the details of all the customers. Develop a function to search for a given customer's details based on the telephone number.

12. Write a program in C to check whether two given strings are an anagram using functions.

REFERENCE BOOKS:

1. Pradip Dey and Manas Ghosh, *Programming in C*, 2nd Edition, Oxford University Press, 2007.

SOFTWARE/TOOLS USED:

- Google Cloud based productivity and collaboration tools
- Github
- Google speech recognition API, Google's language-detection API
- Detectron-Facebook's API
- GCC compiler

ADDITIONAL LEARNING RESOURCES:

- https://gsuite.google.com/learning-center/products/#!/
- https://pypi.org/project/langdetect/
- https://cloud.google.com/translate/docs
- https://ai.facebook.com/tools/detectron/

II B.Tech. - I Semester

(19BT315AC) **DESIGN THINKING**

(Audit Course)

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

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 the course, 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:

UNIT-I: 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

TEXT BOOKS:

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, Scott Swan , Abbie Griffin, *Design Thinking – New Product Essentials from PDMA*, Wiley, 2015.
2. Vijay Kumar, *101 Design Methods: A Structured Approach for Driving Innovation in Your Organization*, 2012.

ADDITIONAL LEARNING RESOURCES:

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