Course: Data Structures

Unit 1: Basic of Data Structures

1. What is Data Structure and explain various operations on data structures.
2. What is Data Structure and explain various types of data structures.
3. What is algorithm? Explain its characteristics.
4. What is algorithm? Write an algorithm to find factorial of given number.
5. What is algorithm? Write an algorithm to find greater number among three numbers.
6. Explain the difference between static and dynamic data structures.
7. Differentiate between linear and non-linear data structures with examples
8. Define algorithm complexity and explain best case, average case and worst case complexity with examples.
9. What is the difference between time complexity and space complexity? Provide examples.
10. How to calculate time complexity of an algorithm. Explain with the help of example.

Unit 2: Searching and Sorting Techniques

1. Write an algorithm for linear search and explain its working using example.
2. Write an algorithm for binary search and explain its working using example.
3. Differentiate between linear and Binary Search
4. Let DATA be sorted array as follows,
   DATA:11,22,30,33,40,44,55,60,66,77,80,88,99
   Use appropriate searching technique to search ITEM=60 and ITEM= 12 in given DATA array. [Write step by step explanation]
5. Let DATA be sorted array as follows,
   DATA: 15,29,34,41,54,67,78,86,91,96 Use appropriate searching technique to search ITEM=67 and ITEM= 12 in given DATA array. [Write step by step explanation]
6. Write algorithm for binary search. Apply the algorithm on the array A containing 9, 17, 23,38,45,50,57,76,90,100 to search items 10 and 100.
7. What do you mean by sorting? List all sorting techniques. Write an algorithm of bubble sort.
8. Write algorithm for linear search. Apply the algorithm on the array A containing 56, 43, 21, 18, 65, 88, 51,95, 27, 39 to search item 13 and 88.
9. Write an algorithm for bubble sort with example.
10. Explain working of bubble sort with example.
11. Apply bubble sort algorithm on array having 5 elements as follows: 45, 32, 28, 5, 11. Write all the steps.
12. Write an algorithm for selection sort with example.
13. Explain working of selection sort with example.
14. Apply selection sort algorithm on array having 9 elements as follows: 89, 57, 74, 63, 51, 36, 27, 11, 45. Write all the steps.
15. Write an algorithm for insertion sort with example.
16. Explain working of insertion sort with example.
17. Apply insertion sort algorithm on array having 7 elements are as follows: 45, 22, 32, 28, 5, 11. Write all the steps.
18. Write an algorithm for merging two sorted array.

19. Apply merging operation on array A having elements as 23, 46, 55, 89 and array B having elements as 18, 32, 65, 77, 84, 91, 98
20. Write an algorithm for merge sort.
21. Explain working of merge sort with example.
22. Apply merge sort algorithm on array having 11 elements are as follows: 45, 18, 22, 56, 32, 76, 28, 5, 11, 87, 16. Write all the steps.
23. Write an algorithm for quick sort.
24. Explain working of quick sort with example.
25. Apply reduction/partition step of quick sort algorithm on array having 12 elements are as follows: 45, 98, 18, 22, 56, 32, 76, 28, 5, 11, 87, 16. Write all the steps.
26. Explain radix sort. Apply it on 4512, 3251, 2354, 8974, 5461, 8799, 5674, 2723.
27. Write an algorithm for radix sort and explain it with example.
28. What is hashing? Explain hash functions.
29. What do you mean by collision? List all the collision resolving techniques and explain open hashing.
30. Explain closed hashing technique with the help of example.
31. Suppose the table T has 11 memory locations T[1], T[2], …… , T[11] and suppose the file F consists of 8 records A, B , C, D, E, X, Y and Z with the following hash addresses.

Record:	A	B	C	D	E	X	Y	Z
H(k)	4	8	2	11	4	11	5	1

Apply linear probing and find average number successful and unsuccessful probes/comparison.

32. Suppose the table T has 11 memory locations T[1], T[2], …… , T[11] and suppose the file F consists of 8 records A, B , C, D, E, X, Y and Z with the following hash addresses.

Record:	A	B	C	D	E	X	Y	Z
H(k)	4	8	2	11	4	11	5	1

Apply chaining (close hashing).

Unit 3: Stack and Queue

1. What is Stack? Explain its operations with algorithm and example.
2. List applications of stack. Explain any one in detail.
3. What is Queue? Explain linear queue operations with algorithm and example.
4. List applications of queue. Explain any one in detail.
5. Write an algorithm for insertion and deletion in circular queue.
6. What is priority queue? Explain its types.
7. What is deque? Explain its types.
8. Explain priority queue and circular queue with examples.
9. What is Queue? Explain the types of Queues in detail.
10. Explain the working of circular queue with example.
11. Define the following with example
    i. Linear queue
    ii. Circular queue
    iii. Priority queue
    iv. Deque
12. Write an algorithm to evaluate postfix expression using stack and explain it with an example.

13. Write an algorithm to convert infix expression to postfix using stack and explain it with an example.
14. Write an algorithm to convert infix expression to prefix using stack and explain it with an example.
15. Evaluate following postfix expression with the help of stack:

    5, 3,+ ,6 ,2,/,*,3 ,5,*,+

    Show the contents of stack at every step of evaluation.
16. Apply infix to postfix conversion algorithm using stack to convert the following expression:

    (A+B)*C/D

    Show the contents of stack at every step of conversion.
17. Explain basic operations that can be performed on stack. Evaluate following postfix expression using stack.

    5, 6, 2, +, *, 12, 4, /, -

    Show the contents of stack at every step of evaluation.
18. Apply infix to postfix conversion algorithm using stack to convert and evaluate the following expression:

    A + B * (C + D) / F + D * E

    Where A=10, B=2, C=7, D=3, E=4 and F=5
19. Apply infix to postfix conversion algorithm using stack to convert and evaluate the following expression:

    A + (C + D) / F + B * E

    Where A=10, B=2, C=7, D=3, E=4 and F=5
20. Apply infix to prefix conversion algorithm using stack to convert the following expression:

    (A+B)*C/D

    Show the contents of stack at every step of conversion.
21. Apply infix to prefix conversion algorithm using stack to convert the following expression:

    (P+(Q*R)/(S-T))

    Show the contents of stack at every step of conversion.
22. Apply infix to prefix conversion algorithm using stack to convert the following expression:

    X+Y*Z/W+U

    Show the contents of stack at every step of conversion.
23. Apply infix to prefix conversion algorithm using stack to convert the following expression:

    (a+(b/c)*(d ^ e)-f)

    Show the contents of stack at every step of conversion.
24. Apply infix to prefix conversion algorithm using stack to convert and evaluate the following expression:

    (A + B) /C + D ^ E * F

    Where A=10, B=4, C=7, D=3, E=2 and F=5
25. Apply infix to prefix conversion algorithm using stack to convert and evaluate the following expression:

    A^ B + C / D - E * F

    Where A=10, B=3, C=8, D=2, E=2 and F=5

Unit 4: Linked List

1. What is linked list? Explain types of linked list with the help of example.

2. What is one-way (singly) lined list? Explain representation of singly linked list in memory with example.
3. Write an algorithm or pseudo code to perform following operations on singly Linked List
    a. Traversing a linked list
    b. Searching in unsorted list
    c. Searching in sorted list
    d. Insert at beginning
    e. Inserting after given node
    f. Insert at end
    g. Insert at any location
    h. Insert in sorted list
    i. Delete from end
    j. Deleting a node following a given node
    k. Deleting a node with given item of information
4. What is two-way (doubly) linked list?
5. Write an algorithm to perform following operations on doubly Linked List
    a. Traversing a linked list
    b. Searching in unsorted list
    c. Searching in sorted list
    d. Deleting a node following a given node
    e. Inserting in between two nodes with given location
6. What is queue? Write algorithms to perform enqueue (insert) and dequeue (delete) operations on linear queue using linked list.
7. What is stack? Write algorithms to perform push and pop operations on stack using linked list.
8. Write an algorithm to perform following operations on circular Linked List
    a. Create a linked list
    b. Insert into the list
    c. Delete from that list

Unit 5: Trees

1. What is AVL tree? Write all the rotation operations of it.
2. Construct Binary Search Tree of following data and apply all the tree traversal techniques on that tree. Data:- 11,6,8,19,4,10,5,17,43,49,31,60
3. Explain sequential (array) representation of binary tree. Explain preorder and inorder traversal in binary tree with example.
4. Write an algorithm to insert an element in heap tree. Construct heap tree (maxheap) using following data entered as a sequential set. 23, 55, 46, 35, 10.
5. Define AVL tree. With suitable example explain single rotations required for rebalancing in AVL trees after insertion operation.
6. What is Binary tree? Write algorithms for preorder, inorder and post order traversal in binary tree. Consider binary tree given in Fig.1, give preorder, inorder and postordertraversal.

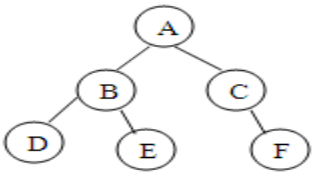


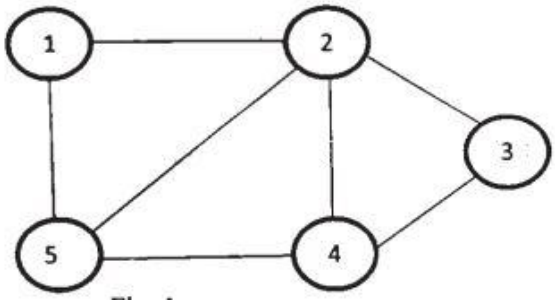
Fig. 1

7. What is Heap tree? Construct heap tree (maxheap) using following data entered as a sequential set. 44, 30, 50, 22, 60, 55, 77, 55. (6)
8. Construct the binary search tree (BST) from the following elements:45, 20, 80, 40, 10, 90, 70 Also, show pre-order and post-order traversal for the same.
9. What is AVL tree? Explain all the rotations in AVL tree. Construct AVL tree for the following data: 1, 2, 3, 4, 5, 6
10. Construct the binary search tree from the following elements:10, 60 40, 28, 14, 50, 6.
11. Construct the binary search tree from the following elements :5, 2, 8, 4, 1, 9, 7 Also show preorder, inorder and postorder traversal for the same.
12. Define Binary Tree. What are its types? Explain with suitable figures.
13. Define the following terms with respect to Trees:
    i) Root
    ii) Subtree
    iii) Level of node
    iv) Depth of Tree
    v) Siblings
14. Explain with suitable example how binary tree can be represented using:
    i) Array
    ii) Linked List
15. What is binary search tree? How to binary search tree in array.
16. Explain following types of tree:
    a. Binary tree
    b. Full binary Tree
    c. Complete Binary Tree
    d. Strict Binary Tree

Unit 6: Graph

1. Draw the picture of the directed graph specified below:
   G=(V,E)
   V(G)={1,2,3,4,5,6}
   E(G)={(1,2),(2,3),(3,4),(5,1),(5,6),(2,6),(1,6),(4,6),(2,4)}
   Obtain the following from the directed graph:
   i) Adjacency matrix
   ii) Adjacency list
2. Explain any one following term of Graph with help of suitable example.
   i) Breadth First Search
   ii) Depth First Search
3. What is graph? Explain all types of graph with the help of example.
4. What is graph? Explain depth first traversal (DFS) in graph with example. (6)

5. Explain representation of graph using adjacency matrix and adjacency list with example. List applications of graph.
6. Explain Breadth First Search (BFS) traversal in graph with example. (6)
7. What is Graph? With suitable example describe how graph is represented using adjacency list.
8. What do you mean by adjacency matrix and adjacency list? Give the adjacency matrix and adjacency list for the graph shown below:



10. Define with an example:
    i) Undirected Graph
    ii) Directed Graph
    iii) Weighted Graph

11. What is graph? Explain following terminology of graph with example:
    a. Degree of node
    b. Connected graph
    c. Complete graph
12. Draw the picture of the directed graph specified below:
    G=(V,E)
    V(G)={A, B, C, D, E)
    E(G)={(A, B), (A, C), (A, D), B, C), (D, C), (D,E), (E,C)}
    Obtain the following from the directed graph:
    i)Adjacency matrix
    ii) Adjacency list
13. What is sparse matrix? Explain it in details.
14. Explain linked representation of sparse matrix.