



MAULANA ABUL KALAM AZAD UNIVERSITY OF TECHNOLOGY, WEST BENGAL

Paper Code : PC-ECS301/PCC- CS301/PCC- CSBS301/PCC-CS301/PCCCS301 Data Structure & Algorithms

UPID : 003443

CSE/IT/ CSBS/ECE/ Cys/DS/IOT/ACML

Time Allotted : 3 Hours

Full Marks : 70

The Figures in the margin indicate full marks.

Candidate are required to give their answers in their own words as far as practicable

Group-A (Very Short Answer Type Question)

1/ Answer any ten of the following :

[1 x 10 = 10]

- (i) What is the disadvantage of array data structure?
 - a) The amount of memory to be allocated should be known beforehand.
 - b) Elements of an array can be accessed in constant time.
 - c) Elements are stored in contiguous memory blocks.
 - d) Multiple other data structures can be implemented using arrays.
- (ii) Which of the following data structures allow insertion and deletion from both ends?
 - a) Stack b) Deque c) Queue d) Strings
- (iii) Which of the following sorting algorithms provide the best time complexity in the worst-case scenario?
 - a) Merge Sort
 - b) Quick Sort
 - c) Bubble Sort
 - d) Selection Sort
- (iv) Worst case time complexity to access an element in a BST can be?
 - a) $O(n)$ b) $O(n * \log n)$ c) $O(1)$ d) $O(\log n)$
- (v) How are String represented in memory in C?
 - a) An array of characters.
 - b) The object of some class.
 - c) Same as other primitive data types.
 - d) LinkedList of characters.
- (vi) Which of the following data structures can be used to implement queues?
 - a) Stack b) Arrays c) Linked List d) Both b and c
- (vii) Which of the following is a Divide and Conquer algorithm?
 - a) Bubble Sort b) Selection Sort c) Heap Sort d) Merge Sort
- (viii) What is the best case time complexity of the binary search algorithm?
 - a) $O(1)$ b) $O(n)$ c) $O(\log 2n)$ d) $O(n^2)$
- (ix) What is the time complexity to insert an element to the front of a LinkedList (head pointer given)?
 - a) $O(n)$ b) $O(1)$ c) $O(\log n)$ d) $O(n * \log n)$
- (x) In a B+ tree, both the internal nodes and the leaves have keys.
 - a) True b) False
- (xi) What is the time complexity of an infix to postfix conversion algorithm?
 - a) $O(N \log N)$ b) $O(N)$ c) $O(N^2)$ d) $O(M \log N)$
- (xii) What is a hash table?
 - a) A structure that maps values to keys.
 - b) A structure that maps keys to values.
 - c) A structure used for storage.
 - d) A structure used to implement stack and queue.

Group-B (Short Answer Type Question)

Answer any three of the following :

[5 x 3 = 15]

2. Write an algorithm to create linear linked list with n nodes. [5]
3. Write an algorithm to insert an element in Circular Queue. [5]
4. Justify the statement with proper example: All Binary Search Trees are Binary Tree but All Binary Trees are not Binary Search Tree. [5]
5. Find the time complexity of recursive algorithm of Tower of Hanoi. [5]

6. Convert the given infix expression to postfix expression
 $A+B \cdot C-D \cdot E/F/H$

[5]

Group-C (Long Answer Type Question)

Answer any three of the following :

[15 x 3 = 45]

7. Create a AVL tree by inserting the following numbers in the order in which they are given:
17, 25, 19, 23, 75, 6. Explain it step wise with suitable diagram.
What is Threaded Binary Tree? What are its advantages?
8. Define queue and its applications. Write an algorithm for insertion and deletion from queue.
9. a) Define Circular Linked List and Double Linked List.
b) Write an algorithm to delete an element from Double Linked List.
[Consider three cases: 1st node deletion, last node deletion and any intermediate node deletion]
10. a) Write a quick sort algorithm to sort a list. Explain with an example.
b) Prove that the best case time complexity for quick sort is $O(n \log n)$ for input size of n .
11. Write short notes on any three from following:
- a) AVL Tree
b) Dequeue and Priority Queue
c) Collision Resolution Techniques in Hashing
d) BFS

[8+3+4]

[5+5+5]

[3+4+4+4]

[9+6]

[15]

*** END OF PAPER ***

<https://www.makaut.com>

Whatsapp @ 9300930012

Send your old paper & get 10/-

अपने पुराने पेपर्स भेजे और 10 रुपये पायें,

Paytm or Google Pay से