Masters of Computer Applications

MCAC-201: Data Structures

Unique Paper Code: 223401201

Semester II

June -2023

Year of Admission: 2021

Duration: 3hours | Max Marks: 70

Note: Attempt all questions. Write neatly and absolutely to the point. Assumptions (if any) should be clearly stated.

1. Give the time-complexity for each of the following operations in the worst case **(1*5=5)**
   (a) Inserting an element in an Unsorted Array.
   (b) Deleting a given element from a Binary Search Tree.
   (c) Returning the maximum element present in a Doubly-Linked List.
   (d) Searching an element in a Sorted array.
   (e) Searching an element in a Sorted Singly-Linked List.

2. Given head pointers $head1, head2$ of two singly linked lists (the length of these lists is unknown), give a linear time algorithm to append the longer list (one with more number of nodes) at the end of the shorter list. The algorithm should take no more than constant extra space. **(5)**

3. Construct a suffix tree for the text $BANANAS$. Show the steps, without explanation. **(5)**

4. In a directed graph $G(V, E)$, a node v is said to be a sink node if and only if no edge is going out of v. Given a graph G (represented as adjacency list), write a function to check if G contains a sink node or not. **(5)**

5. Give an example for each of the following **(2*5=10)**
   (a) Binary Search Tree that is not an AVL Tree
   (b) Red Black tree that is not an AVL Tree
   (c) Heap that is also a Binary search Tree
   (d) Sorted Array that is also a Heap
   (e) 2-dimensional array that represents a Tree.

6. Assuming hash function $H(x) = x\%7$, Insert the keys $10, 17, 4, 18, 24, 25, 5$ in the same sequence assuming

(a) Open addressing with Linear Probing (3)

(b) Chaining (3)

For each of the above case, show the contents of the hash table after deleting 24. Also calculate the number of comparisons to access the key in each case. (2+2=4)

7. Suppose you have 2 stacks $Stack1$ and $Stack2$. The only available operations are $push(key)$, $pop()$ and $isempty()$. Write a pseudocode to implement a Queue using these two stacks as underlying data structure. You are required to write the pseudocodes only for $enqueue(key)$, $dequeue()$ and $isempty()$ functions for your Queue. (5)

8. Write a pseudocode to implement a Min-Priority-Queue (only $enqueue(key, priority)$, $dequeue()$ and $isempty()$ functions) using a sorted array as the underlying data structure. Give the running time of your operations. (5)

9. The height of a tree is the number of edges in the tree from the root to the deepest node. Given a pointer to the root of a tree, write a recursive function to return the height of the tree. (5)

10. Given k different sorted arrays $A_1, A_2, \ldots A_k$ each containing n integers, give an efficient algorithm to merge these k sorted arrays into a single sorted array. Analyze the time complexity of your algorithm. (5)

11. Construct a Binary Search Tree from a given Pre-Order Traversal:
25, 17, 22, 40, 30, 45, 42. (5)

12. Answer the following questions briefly. (2+2+1=5)

(a) What are the advantages of using an array over a linked list.

(b) How many different Binary Search Trees are possible for the set of keys $\{1, 2, 3\}$. Show without explaining.

(c) What is the maximum and the minimum number of edges possible in a Tree with n nodes?

Best wishes