IIITD&M Kancheepuram

Data Structures and Algorithms

Course code: CS1004

CS23I1027

Date: 9/5/2024
Duration: 3 Hours.

Total Marks: 50

Instructions

1. Do not write anything on the question paper. Please attach the question paper with answer sheet.
2. Answering the same question more than once on the answer sheet will lead to negative marking.
3. All parts of a question should be written on the same place on the answer sheet.
4. Negative marks will be provided for not providing reason for the answer to any question

Section A: Each question carries 3 Marks **(30 Marks)**

1. Let S be a stack of size n >=1. Starting with the empty stack, suppose we push the first n natural numbers in sequence and then perform n pop operations. Assume that push and pop operations take X seconds each, and Y seconds elapse between the end of the one such stack operation and the start of the next operation. For m>=1, define the stack life of m as the time elapsed from the end of push(m) to the start of the pop operation that removes m from S. The average stack-life of an element of this stack is ?
2. Let P be a singly linked list. Let be Q the pointer to an intermediate node in the list. What is the worst-case time complexity of the best-known algorithm to delete the node from the list and explain the algorithm in steps? (not expecting program)
3. The five items: A, B, C, D, and E are pushed in a stack, one after other starting from A. The stack is popped four items and each element is inserted in a queue. The two elements are deleted from the queue and pushed back on the stack. Now one item is popped from the stack. The popped item is? Write the functions of push and pop where the stack is implemented using array?
4. Explain all the asymptotic notations with an example?
5. Rank the following functions by increasing order of growth (i.e., the slowest-growing first, the fastest-growing last)

   $(\log \log n)^2$, $\log n!$, $n!$, $n^{1.1}$, $n \log n$, 2^n , $10 \log n$, $2^{\log n}$

   where all the logarithms are to the base 2. If two functions have equal orders of growth then list them grouped together.
6. Consider an AVL tree consisting of integers. You have performed a series of insertions and ended up with an unbalanced AVL tree that requires rebalancing. The sequence of integers inserted into an initially empty AVL tree is as follows: 20, 15, 30, 10, 18, 25, 40, 5, 12
   After inserting the last integer, the tree has become unbalanced.

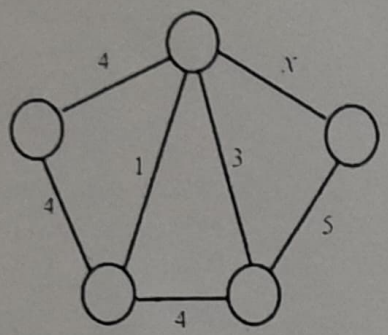
(a) Determine which node became unbalanced after the last insertion.

(b) Identify the type of rotation required to rebalance the tree

(c) Perform the appropriate rotation(s) and provide the new balanced AVL tree.

7. List out various types of binary trees and explain them with an example?

8. Consider the following undirected graph, choose a value for X that will maximize the number of minimum weight spanning trees (MWSTs) of G. The number of MWSTs of G for this value of X is



9. Consider a double hashing scheme in which the primary hash function is h1(k) = k mod 23, and the secondary hash function is h2(k) = 1+(k mod 19). Assume that the table size is 23. Then the address returned by probe 1 in the probe sequence (assume that the probe sequence begins at probe 0) for key value k = 90 is?

10. The following C function takes a singly-linked list as input argument. It modifies the list by moving the last element to the front of the list and returns the modified list. Some part of the code is left blank. Please write the code which is required in the blank to perform the above operation?

```
typedef struct node {
    int value;
    struct node *next;
} Node;
Node *move_to_front(Node *head) {
      Node *p, *q;
      if ((head = = NULL: || (head->next = = NULL)) return head;
      q = NULL; p = head;
      while (p-> next !=NULL) {
                        q=P;
                        p=p->next;
}
_____________________________
return head;
}
```

Section B: Each question carries 5 marks (20 Marks)

1. Write a C program to perform **Binary search** using an array?. List out the best case, average case, and worst case time complexity?

2. Perform the **Quick sort** on the following elements 51, 36, 62, 13, 16, 7, 5, 24. Explain each step in detail. Find out the number of comparisons and number of swapping/shifting in each pass/iteration.

3. Consider the following infix expression. **[2+2+1 Marks]**

   2+3*5+50/5^2 (^ is power operator)

   a) Convert the above expression into postfix form using stack and show each and every step in order.

   b) Evaluate the obtained postfix expression using stack, show the necessary steps and find the value of the expression.

   c) What is the value stored at top of the stack after the second '+' operation's evaluation.

4.

a. Consider the following statements and specify each statement is either true or false [2 Marks]

1. The smallest element in a max-heap is always at a leaf node
2. The second largest element in a max-heap is always a child of a root node
3. A max-heap can be constructed from a binary search tree in $\theta(n)$ time
4. A binary search tree can be constructed from a max-heap in $\theta(n)$ time

b. Provide the data structures used by the breadth first search and depth first search [1 Mark]

c. List out the essential books and supplementary books (any two) mentioned for the data structures and algorithms course syllabus
[2 Marks]