University of Swaziland

Department of Computer Science

Final Supplementary Examination

JUNE 2019

Title of paper : Data structures

Course number : CS311

Time Allowed : Three(3) hours

Instructions :

- *Each question carries 25 marks*
- *Answer question 1.*
- *Answer any other three (3) questions from questions 2 to 5.*

This paper may not be opened until permission has been granted by the invigilator

QUESTION 1

(a) State precisely/formally the meaning of the statement: **f(n) is O(g(n))** [2]

(b) Suppose that items A, B, C, D and E are pushed, in that order, onto an initially empty stack S. S is then popped four times. As each item is popped off, it is inserted into an initially empty queue. If two items are then removed from the queue, what is the *next* item that will be removed from the queue? [3]

(c) A graph implementation that uses a adjacency-matrix to represent the edges would be most reasonable for which of the following cases? Explain you answer. [3]

A. 1000 nodes, 1200 edges

B. 100 nodes, 4000 edges

C. 1000 nodes, 10000 edges

D. 10 nodes, 20 edges

E. none of these, since a graph can only be represented by a linked structure.

(d) A binary tree is constructed of nodes that are instances of the following class:

```
public class Node {
        public int val;
        public Node left;
        public Node right;
}
```

Consider the following method:

```
public static Node mystery(Node root) {
        if (root.right == null)
                return root;
        else
                return mystery(root.right);
}
```

You consult three supposedly tech-savvy consultants, and you get the following three opinions about what the method does when passed a reference to the root node of a binary tree:

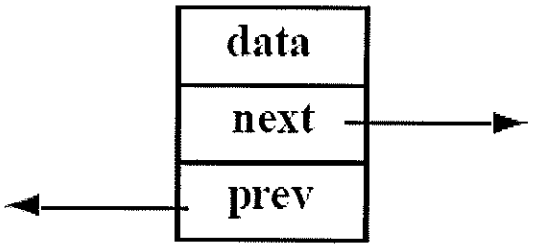
I. It returns the last node visited by an in-order traversal

II. It returns the last node visited by a post-order traversal

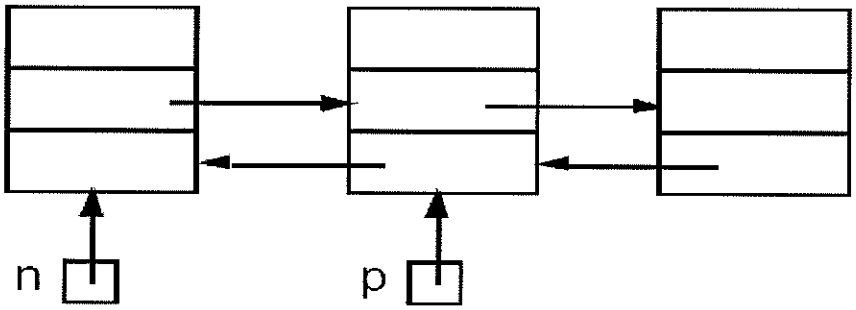
III. It returns the last node visited by a level-order traversal

Which of these opinions are correct regardless of the contents of the tree? [4]

(e) Nodes for a doubly linked list are defined to have the following structure:



The next instance variable stores a reference to the **next** node in the list, and the **prev** instance variable refers to the previous node in the list. Below is a list of three of these nodes, along with two reference variables, **n** and **p**, that refer to specific nodes in the list.



Which of the following expressions does *not* refer to the third node in the list? [2]

A. p.next **B.** n.next.next **C.** p.prev.next **D.** p.next.prev.next

E. n.next.next.prev.next

(f) A police department wants to maintain a database of up to 1800 license-plate numbers of people who receive frequent tickets so that it can be determined very quickly whether or not a given license plate is in the database. Speed of response is very important; efficient use of memory is also important, but not as important as speed of response. Which of the following data structures would be most appropriate for this task? Explain your answer [4]

A. a sorted linked list

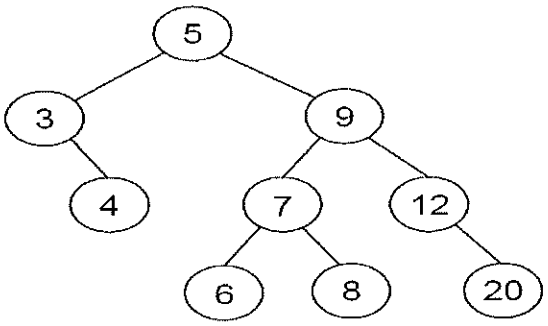
B. a sorted array with 1800 entries

C. a hash table using open addressing with 1800 entries

D. a hash table using open addressing with 3600 entries

E. a hash table using open addressing with 10000 entries

(g) The binary search tree shown below was constructed by inserting a sequence of items into an empty tree.



Which of the following input sequences will *not* produce this binary search tree? [2]

A. 5 3 4 9 12 7 8 6 20

B. 5 9 3 7 6 8 4 12 20

C. 5 9 7 8 6 12 20 3 4

D. 5 9 7 3 8 12 6 4 20

E. 5 9 3 6 7 8 4 12 20

(h) Using the binary tree from the previous question above, trace the iterative level-order and iterative in-order traversal algorithms. Show all steps of the trace. [6]

QUESTION 2

(a) Draw a picture of a sample directed graph G with 13 nodes and 21 edges. Each node must have at least 2 but not more than 3 neighbors. Each edge must have a weight.[3]

(b) Using the graph from (a) above as an example, compare the adjacency list and adjacency matrix representation. Which representation would you say is more efficient? Justify/explain your answer. [4]

(c) Using the Java Collection Framework classes, define a suitable structure to represent a graph using an adjacency list. Where necessary, clearly state and assumptions you are making. [5]

(d) Based on your type definition in (c.) above, write Java functions/method that would perform the following: In each case, estimate the running time.

(i) Determine if any two given nodes are neighbors. [3]

(ii) Add an edge between two nodes. [4]

(iii) Display all nodes [3]

(iv) Display all edges [3]

QUESTION 3

(a) Suppose the keys on the first row of a standard keyboard (QWERTYUIOP) are inserted in succession into an initially empty 2-3 tree. Draw the 2-3_tree after the sequence of insertions. Show all the steps of the insertion. What is the running time of inserting into a 2-3_tree? [6]

(b) With the aid of an example, distinguish between a B-tree and B+-tree, and discuss any practical application of B-tree and B+-trees. [4]

(c) Distinguish between open-addressing and closed-addressing [4]

(d) You are given an empty hash table of size 7 that uses open addressing. The following sequence of keys is to be inserted: 15 17 8 23 3 5. Insert these keys using each of the following approaches. If overflow occurs, say so,

(i) h(x) = x % 7; linear probing [3]

(ii) h(x) = x % 7; quadratic probing [3]

(iii) h(x) = x % 7; double hashing with h2(x) = x / 7 + 1 (integer division) [5]

QUESTION 4

(a) Using Java notation, write a **Queue** Module that uses a circular-linked implementation of a queue. Your code must define an appropriate interface and suitable classes to implement the interface. Provide code for all required member functions/methods. [15]

(b) Consider the language

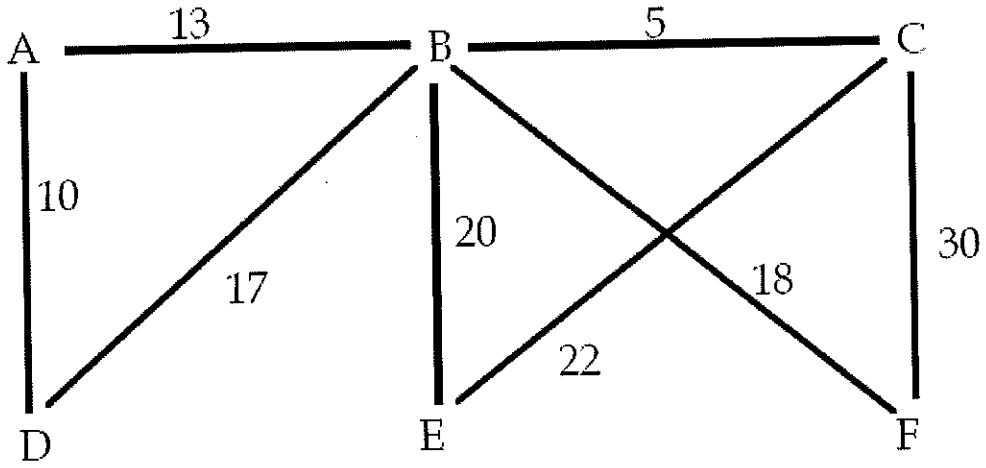
$L = \{w\$w' : w \text{ is a string of characters other than } \$, w' = reverse(w)\}$

Using Java collections Framework classes, write a recognition function (determines strings in the language) that uses both a stack and a queue. Thus, as you read the input string, you insert each character of w into a queue and each character of w' into a stack. The function returns true if the input string is in the language, and false otherwise. Assume the input string contains only one $.

[10]

QUESTION 5

Consider the following graph,



(a) Trace the execution of the depth-first traversal of the graph shown above, starting with vertex C. Select the smallest edge first when appropriate. In the space below, list the vertices in the order in which they are visited. [5]

(b) Trace the execution of the breadth-first traversal of the graph shown above, starting with vertex C. Select the smallest edge first when appropriate. In the space below, list the vertices in the order in which they are visited [5]

(c) With the aid of the given graph, trace the execution Prim's minimum spanning tree algorithm starting from vertex F. How does this algorithm compare with other (at least 1 other) minimum spanning tree algorithms? [7]

(d) With the aid of an example, trace Dijkstra's shortest path algorithm to find a shortest path from vertex D to vertex E. How does this algorithm compare with other (at least 2 other) shortest path algorithm? [8]