University of Swaziland

Department of Computer Science

Supplementary Examination

JULY 2018

Title of paper : Data Structures and Algorithms

Course number : CSC311/CS342

Time Allowed : Three(3) hours

Instructions :

- *Each question carries 25 marks*
- *Answer any four (4) questions from questions 1 to 6.*

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

Question 1

(a) Write an algorithm that finds the smallest value in a given stack of integers. What is the running time complexity of this algorithm? [7]

(b) With the aid of a sample array, trace the execution of the merge-sort algorithms. What is the running time complexity of this algorithm?. Is there a better sorting algorithm than merge-sort? Explain [6]

(c) What is the main cause of collisions when organizing a hash table? Explain the two broad strategies for resolving collisions. [7]

(d) With the aid of an example, distinguish between the separate chaining and coalesced chaining in organizing a hash table and resolving collisions. [5]

Question 2

(a) List and describe the operations of the list data structure. [5]

(b) Compare an array-based implementation and linked-list implementation of a list. Which implementation would you recommend as a better implementation? Justify your answer.

[5]

(c) Using Java notation, write a Java **List** class that uses a linked-list to implement a List. Provide code for the constructor function, inserting and removing values from the list. [15]

Question 3

(a) List and describe the operations of the queue data structure. [5]

(b) Using Java notation, write a generic Java **Queue** class that uses a circular array implement a queue structure. Provide code for the constructor, adding and removing a value from the queue. [12]

(c) Analyze the Big-Oh time-complexity for each queue operation implemented in (b) above. [3]

(d) Based on your implementation in (c) above, write a function that swaps the contents of two queues. What is the running time complexity of this function?[5]

Question 4

(a) Draw a binary search tree (BST) of size 10 and depth 4. Label the nodes 1, 2,..., 10 according to the post-order traversal sequence. [6]

(b) Using the sample BST from (a) above, trace the execution of the iterative level order traversal algorithm. [5]

(c) Using C# or Java notation, write a function that searches for a given key value in a BST tree. It should return a sub-tree whose root contains the key value. Otherwise it should return an empty tree. [7]

(d) With the aid of examples, distinguish between an (a,b)-tree and a B-tree. Which structure would you recommend? Explain why. [7]

Question 5

(a) With the aid of a sample graph containing at least 6 nodes and at least 7 edges, distinguish between depth-first search and breadth first search. Clearly explain the differences and running time complexities. [7]

(b) Using the sample graph in (a) above, show a trace of the breadth-first search traversal algorithm. [6]

(c) Using the sample graph in (a) above, distinguish between adjacency matric and adjacency-list representation of a graph. Which representation would you recommend and why? [7]

(d) Using the sample graph in (a) above, explain Dijkstra's shortest path algorithm. [5]