04

RAJARATA UNIVERSITY OF SRI LANKA
FACULTY OF APPLIED SCIENCES

B.Sc. (General) Degree in Applied Sciences
Second Year Semester I Examination – September/October 2019

COM 2307 – Data Structures and Algorithms

Time Allowed: Three (03) Hours

Answer **ALL** the questions

7

1. a) Define terms "Big O", "Big Ω" and "Big Θ" related to analysis of algorithms. (06 marks)

   b) What are static and dynamic data structures? Explain using examples. (04 marks)

   c) Explain the factors that you have to consider when measuring the running time of an algorithm. (04 marks)

   d) Following equations represent the running time of two algorithms. Find the "Big O" values (asymptotic upper bound) of them.

      i. $T(n)=T(n-1) + 1; T(1)=1$
      ii. $T(n)=T(n-1) + T(n-2); T(1)=T(2)=1$ (06 marks)

2. a) Using examples, explain why you should avoid super-polynomial algorithms. (04 marks)

   b) What are the advantages and disadvantages of array-based lists? (03 marks)

c) Assume that there are set of students in a class who follow 8 subjects throughout the year. Term examinations are conducted per each term of the year and marks obtained by each student are recorded for 3 terms per year. An individual student record must contain name, index no, and marks for 3 terms for 8 subjects.

i. Suggest a suitable structure for a student record and write its definition using C. (04 marks)

ii. Using the structure defined in i. above define a structure for the class. (02 marks)

iii. Devise an algorithm to add a student to your structure suggested in ii. above. (03 marks)

iv. Write a function to remove a given student from the structure using C. (04 marks)

3. a) Compare and contrast queues and stacks. (04 marks)

b) Explain the two main operations, enqueue and dequeue, of queues using examples. (02 marks)

c) Explain how to use an array to implement a linear queue. (04 marks)

d) Wastage of memory space is a drawback of linear array-based queues. Suggest a mechanism to overcome this problem. (04 marks)

e) Suppose there are two dynamic queues Q1 and Q2 available with data. A queue node is comprised of 2 integers named key and value. Write a C function to create a new queue by concatenating Q1 and Q2. Followings are some steps and rules to consider;

i. Take two nodes N1, N2 from Q1 and Q2 respectively.

ii. Add the node with smaller key to the new queue. If two keys are equal, add the node with highest value. If values are equal, add N1.

iii. Take a node, N1, from Q1 if N1 is added to the new queue, else take a node, N2, from Q2. Continue to step ii. (06 marks)

4. a) List four applications of stacks. (02 marks)

b) Explain how to use a stack to convert an infix expression to a postfix. Illustrate the conversion of the expression $A * B + (C - D/E) * F + G$ in to a postfix expression. (06 marks)

c) Write a few lines of C codes to define a fixed size stack. Assume that the elements in the stack are integers. (04 marks)

d) Write C functions to implement push() and pop() operations of the above defined stack. (04 marks)

e) Consider the following algorithm;

```
printTree(TREE T) ⊳ T − a binary search tree
    Declare Q as a Queue ⊳ Q − a FIFO queue
    Q ← T
    While Q is not empty
        Node=dequeue(Q)
        print Node->value
        if (Node->left) not empty
            enqueue(Node->left)
        if (Node->right) not empty
            enqueue(Node->right)
```

Explain the expected output of the algorithm using a sample binary search tree. (04 marks)

5. a) Explain the following terms related to binary search trees;

i. Siblings

ii. Skewed tree

iii. Complete tree (03 marks)

b) Illustrate how to insert the items 10, 12, 18, 8, 14, 6, 20, 25, 4, 3 into a binary search tree. (05 marks)

c) Write the output of In-order, Preorder and Post-order traversals of the tree above. (03 marks)

d) Write a C function to insert an item into a binary search tree. Assume that the tree and tree node structure are already defined and a tree node contains only an integer data item. (04 marks)

e) Explain the cases you have to consider when deleting an item from a binary search tree. (05 marks)

- END -