



**RAJARATA UNIVERSITY OF SRI LANKA  
FACULTY OF APPLIED SCIENCES**

**B.Sc. (General) Degree in Information and Communication Technology**

**First Year Semester II Examination - September/October 2020**

**ICT 1305 – DATA STRUCTURES**

**Time: Three (03) hours**

---

**Instructions:**

Answer **ALL (05)** Questions.

The total marks allotted for each question are given in square brackets.

This paper consists of **three (03)** pages.

---

1. a. In which phase, the knowledge of Data Structures can be applied in the process of software engineering? (03 marks)
- b. “An algorithm is independent from both hardware and programming language”. Do you agree with this statement? Explain. (05 marks)
- c. Discuss the advantages of using an appropriate data structure in a computer programme. (05 marks)
- d. Suggest a suitable structure that facilitates to keep a student record with name, registration number, year, marks of three subjects for two years. Write the required lines of codes in C to implement your structure. Indicate your assumptions if any. (07 marks)

**[Total 20 marks]**

2. a. Compare and contrast queues and linked lists. (04 marks)
- b. As a result of several insertion and deletion operations, an array-based list may become full even though there are some vacant spaces available. Explain how this would happen and how to overcome this problem. (06 marks)
- c. Operations on doubly linked lists consume more memory and time when compared to singly linked lists. Explain the reasons for this with suitable examples. (05 marks)
- d. Given below is a C code to implement a node of a doubly linked list.

```
struct dNode {
    int id, capacity;
    char grade[2];
    struct dNode next, previous;
}
```

Find the error(s) in above code, if any, and rewrite the corrected version. (02 marks)

- e. To initialize a node of the above structure, write a function which takes the values of *id*, *capacity* and *grade* as parameters. (05 marks)
- f. Write a few lines of C codes to delete a node of a doubly linked list created using the nodes defined in part d. above. (04 marks)

**[Total 26 marks]**

3. a. Represent the expression  $(A + B) * C - (D - E^2) * (F/2 + G)$  in prefix and postfix forms. (04 marks)
- b. Illustrate how to use a stack to convert a given infix expression into a postfix expression. (05 marks)
- c. What is a priority queue? (04 marks)
- d. Insert the given list of integers [15, 12, 4, 20, 18, 2, 10, 5] in to an array-based priority queue. Illustrates the steps clearly. (05 marks)
- e. Explain stable and unstable sorting algorithms using examples. (04 marks)
- f. Illustrate how to sort the given list of integers [10, 16, 14, 21, 9, 4, 15, 7] using *insertion sort* algorithm. (04 marks)

**[Total 26 marks]**

4. a. Compare and contrast array based Binary Search trees and dynamic Binary Search trees. (04 marks)
- b. Write a few lines of C codes to define a node of a dynamic Binary Search tree that stores a real number and an integer. (04 marks)
- c. Find the relationship between the number of nodes and the height of a Binary search tree. (04 marks)
- d. Write an algorithm or a C function to find the height of a Binary tree. (04 marks)

**[Total 16 marks]**

5. a. Compare and contrast AVL trees and Binary Search Trees. (03 marks)
- b. Explain the single rotation and double rotation operations of AVL trees using examples. (06 marks)
- c. AVL trees are suitable for applications that perform more lookups than insertions/deletions operation. Explain reasons for this. (03 marks)

**[Total 12 marks]**

**-- End --**