C0-R4.B3: DATA STRUCTURE THROUGH JAVA

NOTE:

1. **Answer question 1 and any FOUR from questions 2 to 7.**
2. **Parts of the same question should be answered together and in the same sequence.**

Time: 3 Hours **Total Marks: 100**

1.

a) What do you men by Extreme programming?

b) Write an algorithm for finding solution to the Tower's of Hanoi problem. Explain the working of your algorithm (with 4 disks) with diagrams.

c) Can a Queue be represented by circular linked list with only one pointer pointing to the tail of the queue? Substantiate your answer using an example.

d) What is polymorphism in object oriented programming? What are the advantages of it? How Java support polymorphism?

e) Explain the difference between depth first and breadth first traversing techniques of a graph.

f) Differentiate between system defined data types and Abstract data types supported by java with suitable examples.

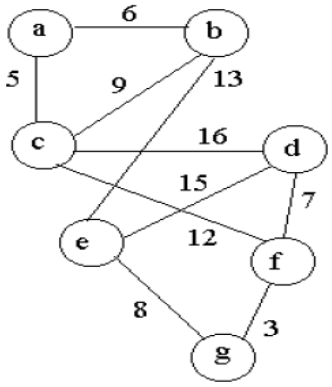
g) Write an algorithm for selection sort. Describe the behaviors of selection sort when the input is already sorted.

(7x4)

2.

a) Suppose a queue is maintained by a circular array QUEUE with N = 12 memory cells. Find the number of elements in QUEUE if
   i) Front = 4, Rear =8.
   ii) Front = 10, Rear = 3.
   iii) Front = 5, Rear = 6 and then two elements are deleted.

b) What is the difference between Prims algorithm and Kruskal's algorithm for finding the minimum-spanning tree of a graph? Execute Kruskal's algorithms on the following graph.



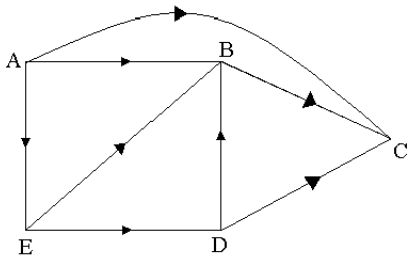
(9+9)

3.

a) How do you find the complexity of an algorithm? What is the relation between the time and space complexities of an algorithm? Justify your answer with an example.

b) Taking a suitable example explain how a general tree can be represented as a Binary Tree. How many Binary Trees can be formed wing 3 nodes 'A', 'B' & 'C'.

c) What are the different ways of representing a graph? Represent the following graph using those ways.



(6+6+6)

4.

a) Derive the asymptotic time complexity of a non recursive, binary search algorithm.

b) How do you rotate a Binary Tree? Explain right and left rotations with the help of an example.

c) In java classes, constructors, methods and fields are regulated using access modifiers. What is access modifier and explain various access modifiers available in java.

(4+6+8)

5.

a) Write an O(1) algorithm to delete a node p in a singly linked list. Can we use this algorithm to delete every node? Justify.

b) What is an AVL tree? Explain how a node can be inserted into an AVL tree.

c) What are priority Queues? How can priority queues be implemented? Explain in brief.

(9+4+5)

6.

a) What is circular queue? Write down java code for inserting and deleting elements from a circular queue implemented using arrays.

b) What is a Binary Search Tree (BST)? Make a BST for the following sequence of numbers.
45, 36, 76, 23, 89, 115, 98, 39, 41, 56, 69, 48
Traverse the tree in Preorder, Inorder and Postorder.

(8+10)

7.

a) Execute quick sort algorithm for two passes using the following list as input:
66, 33, 40, 20, 50, 88, 60, 11, 77, 30, 45, 65. Use first element as Pivot for each iteration. Describe the behavior of Quick sort when the input is already sorted.

b) Two linked lists contain information of the same type in ascending order. Write a Java module to merge them to a single linked list that is sorted.

(9+9)