Birla Institute of Technology & Science, Pilani
Work-Integrated Learning Programmes Division
First Semester 2018-2019

Comprehensive Examination
(EC-3 Regular)

Course No.	: SS ZG519
Course Title	: DATA STRUCTURES AND ALGORITHMS DESIGN
Nature of Exam	: Open Book
Weightage	: 45%
Duration	: 3 Hours
Date of Exam	: 24/11/2018 (AN)

No. of Pages = 2
No. of Questions = 8

Note:

1. Please follow all the *Instructions to Candidates* given on the cover page of the answer book.
2. All parts of a question should be answered consecutively. Each answer should start from a fresh page.
3. Assumptions made if any, should be stated clearly at the beginning of your answer.

Q.1 (a) Two of the most common Divide and Conquer sorting algorithms are quicksort and mergesort.in practice quicksort is often used for sorting data in main storage rather than merge sort. Give a reason why quicksort is likely to be the preferred sorting algorithm for this application. [2]

Q.1 (b) Quicksort's worst-case running time is O(n^2), but it has an expected running time of O(nlogn) if the partition function works well. What needs to be true about the partition function in order for the running time to be O(nlogn)? In practice how can we ensure that this happens? [4]

Q.2. Suppose that you want to get from vertex s to vertex t in an unweighted graph G=(V,E) but you would like to stop by vertex u if it is possible to do so without increasing the length of your path by more than a factor of α. Describe an efficient algorithm that would determine an optima s-t path given your preference for stopping at u along the way if doing so is not prohibitively costly (it should either return the shortest path from s to t or the shortest path from s to t containing u, depending on the situation). [6]

Q.3. The graph below(fig-1) shoes the cities P, Q, R, S, T, U, V and W represented by the vertices and the rail connections between them represented by edges. The number on edges are the times in hours it takes to travel by train between each of the cities.

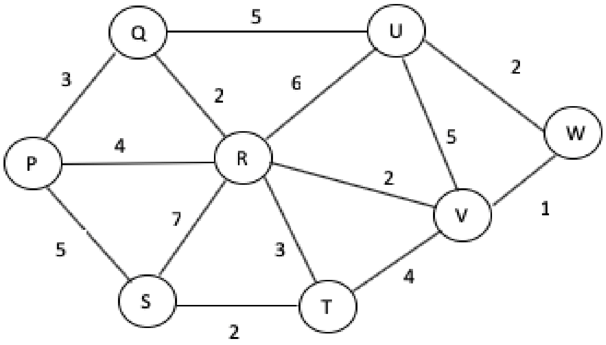

Fig-1

(a) Draw the minimum spanning tree for the graph of rail connections. [all step's needed to be shown and use any MST method]. [4]

(b) Write down the total weight of the minimum spanning tree. [2]

Q.4. Consider the following directed, weighted graph (fig-2). Step through Dijkstra's algorithm to calculate the single-source shortest path from A to every other vertex. Show all steps in detail. [8]

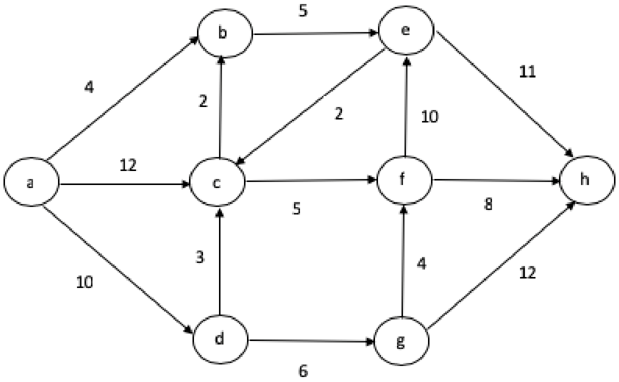

Fig-2

Q.5. Show a 3-Vertex example of a graph on which Dijkstra's algorithm always fails. [please clearly identify which vertex is the source]. [5]

Q.6. Show the result of inserting 10,12,1,14,6,5,8,15,3,9,7,4,11,13 and 2.one at a time, into an initially empty binary heaps [all step's needed]. Show what the actual memory(array) looks like. [5]

Q.7. Let us consider an algorithm which solves problems by dividing them into five subproblems of half the size, recursively solving each subproblems, and then combining the solutions in linear time. What is the complexity of this algorithm? [5]

Q.8. Justify you answer with appropriate evidences [answer in the same order].

(a) Let A_a, A_2 and A_3 be three sorted arrays of n real numbers (all distinct). In the comparison model constructing a balanced binary search tree of the set $A_a \cup A_2 \cup A_3$ requires Ω(nlogn) time. [2]

(b) Let T be a complete binary tree with n-node finding a path from the root of T to a given vertex v $\in$ T using breadth-first search taken O(logn) time. [2]

****