Birla Institute of Technology & Science, Pilani
Work-Integrated Learning Programmes Division
Second Semester 2019-2020
M.Tech (Data Science and Engineering)
End-Semester Test (EC-3 Regular)

Course No.	: DSECF ZG519
Course Title	: DATA STRUCTURE ALGORITHMS AND DESIGN
Nature of Exam	: Closed Book
Weightage	: 40%
Duration	: 150 Min
Date of Exam	: 22-09-2019[AN]

No. of Pages = 3
No. of Questions = 9

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.

Answer All the Questions (Only in the pages mentioned against questions. If you need more pages. Continue remaining answers from page 16 onwards, we have provided extra space at each question)

Question 1: [3M] [to be answered only in page 2]

Q.1. Show how to implement a queue using two stacks. Analyze the running time of the queue operations.

Question 2: [2M] [to be answered only in page 3]

Q.2. How many instructions are executed when we multiply n × m matrix A with m × r matrix B?

Question 3: [2+3 =5M] [to be answered only in pages 4-5]

Q.3. Question based on fig-1,Based on AVL-Tree in the fig: 1 below:

(a) Draw the resulting tree after 5 is removed, but before any rebalancing takes place. Label each node in the resulting tree with its balance factor.

(b) Rebalance the tree that results from (a). Draw a new tree for each rotation that occurs when rebalancing the AVL Tree.

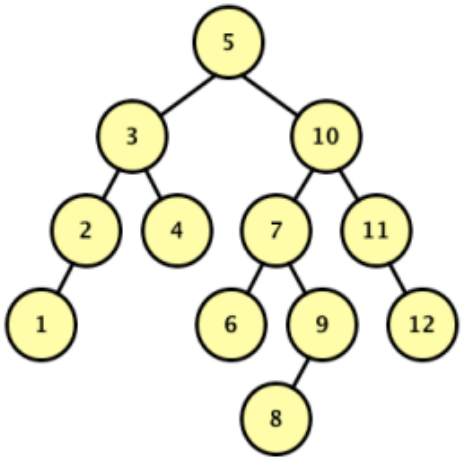

Fig-1

Question 4: [5M] [to be answered only in pages 6-7]

Q.4. Question based on fig:2, For the given Graph G. Using Floyd-Warshall algorithm compute the shortest-path weights matrix $D^{(k)}$.

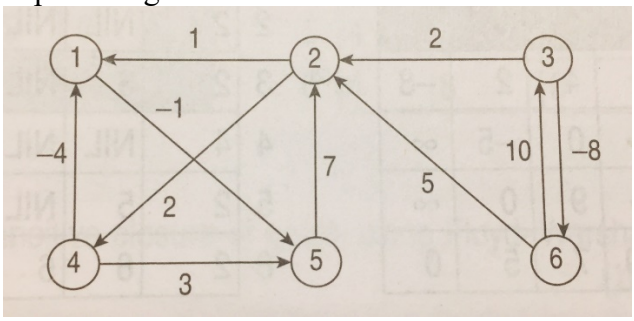

Fig-2

Question 5: [5M] [to be answered only in pages 8-9]

Q.5. Question based on fig:3, For the given graph G, find the shortest path from the node A to all other nodes.

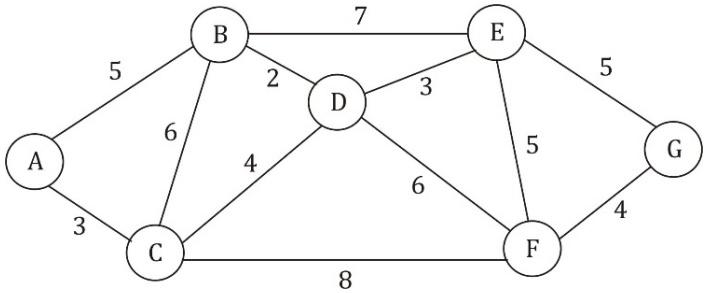

Fig-3

Question 6: [3+3 =6M] [to be answered only in pages 10-11]

Q.6. Suppose we're using quick sort to process data that we're receiving from a connection in a networked system. We want to cover our system from the possibility of being "sabotaged" by hostile connections — we could receive data that is specifically crafted to cause quick sort to have its worst- case performance and thus make our system consume excessive resources and time (rendering it unable to efficiently respond to other connections).

(a) Assuming that quick sort simply chooses the first element as the pivot (instead of the median of first, last, and middle), what is the arrangement of data that produces the worst-case performance in quick sort? (that is, if you were the attacker trying to sabotage the system, what data would you have to send?)

(b) Suggest a simple strategy (hopefully requiring no more than linear time) to avoid the problem. That is, a strategy to guarantee that quick sort will run in O(n log n) most of the time, regardless of input data, even if this input data is maliciously created. Notice that in the context of this question, you're not allowed to change the way quick sort selects the pivot (in fact, you will hopefully suggest a strategy that works regardless of how quick sort selects the pivot).

Question 7: [5M] [to be answered only in pages 12]

Q.7. Question based on fig-4, List the nodes of the tree below in preorder, postorder, and breadth-first order.

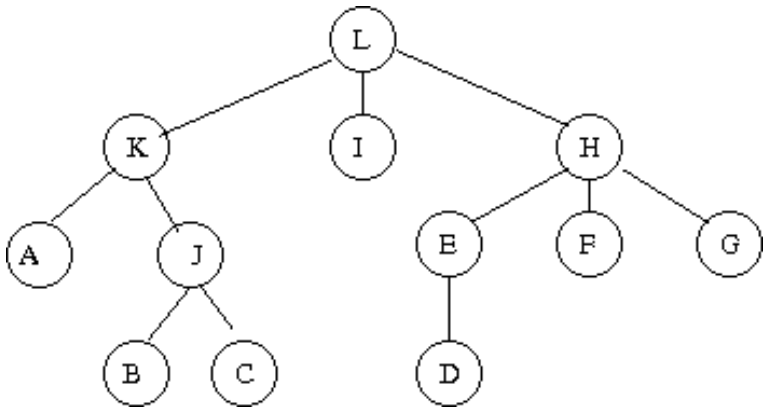

Fig-4.

Question 8: [5M] [to be answered only in pages 13-14]

Q.8. There are four pages. Page A contains a link to page B, a link to page C, and a link to page D. Page B contains one single link to page D. Page C points to pages A and D, and page D points to pages A and C. Using page rank algorithm find the page rank. [Hint: the page are initial ranked 1, use the standard damping factor, find page rank till 5 iteration]

Question 9: [4M] [to be answered only in pages 14-15]

Q.9. Construct K-D tree for the following data (3,6), (17,15), (13,15), (6,12), (9,1), (2,7), (10,19).

*** ALL THE BEST ***