CS-13101: DATA STRUCTURES
B.Tech (CS+IT) III Semester

Time : 3 Hrs. **M.M. : 60**

Note : All questions are compulsory. All the subparts of a question are to be attempted together.

Q1. Answer the following with proper justification.

a. Let S be a sorted array of n integers. State the time complexity, with justification, of the most efficient algorithm to determine if there are two elements with sum less than 99 in S or not. **[1]**

b. Consider a full binary tree of height h with root at level 0. Assume this full binary tree is the binary representation of a Forest. State the number of trees present in this forest. Also state the number of nodes present in each tree of the forest. **[2]**

c. Consider a binary tree T in which left subtree contains the maximum number of nodes possible in the AVL tree of height h and right subtree contains twice of the minimum number of nodes possible in the complete binary tree of height h. How many nodes are present in T? **[2]**

d. If t is the level of the smallest element in a binary search tree (BST), what may be the possible levels of a k-th smallest element? Show the analysis. Note that the root is at level 0. **[3]**

Q2. Insert the following numbers, in the given sequence, in an empty AVL Tree: **[7]**

1, 26, 2, 25, 3, 24, 4, 23, 5

Display the tree after every insertion. Also state the minimum number of nodes which are required to construct AVL tree of height 7 (Note that the root is at level 0).

Q3. Insert the following numbers, in the given sequence, in an empty B Tree of order 5 and display the tree at every split. **[8]**

17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1

Now delete the following elements from the tree, in the given sequence, and display the tree at every merge.

1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17

Q4. Sort the following numbers in ascending order using Heapsort. Show step by step analysis. **[5]**

25, 57, 48, 37, 12, 92, 86, 33

Q5. The frequency for the following alphabets is given in the table below **[5]**

Alphabet	Frequency
R	5
T	6
Y	2
C	16
S	3
L	60
A	8

Construct the Huffman tree in such a way that for every node its left child should be smaller than its right child and in case of a tie the left subtree should be lighter than the right subtree.

a. Write down the Huffman codes for each alphabet

b. Decode the following message 000100010100101101010111

c. Compute the average number of bits/symbol using Shannon's entropy

Q6. Show at every step the contents of the hash table after inserting the keys in the order **[4]**

23, 11, 4, 17, 84, 22, 33

The hash function is given as

$$h(key) = key \ \% \ 11$$

The hash table has a space for 11 keys only. Resolve collision using

a. Linear probing

b. Quadratic probing

Q7. Let $C_x < C_y$ represents that course C_x is a pre-requisite of course C_y. Apply topological sort on the given data showing the values of *count[k], top[k], succ* and *next* at every step. **[5]**

Course Number	Pre-requisite
C_1	None
C_2	C_1
C_3	C_1, C_2
C_4	C_2, C_3
C_5	C_3, C_4
C_6	C_4, C_5
C_7	C_5, C_6

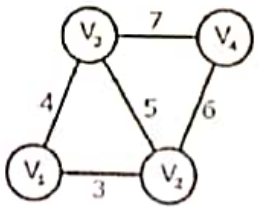
Q8. Write a function in C to implement Quick Sort for doubly linked list. The declaration of the function is **[8]**

void *quick_sort* **(DNODE** ***head)**

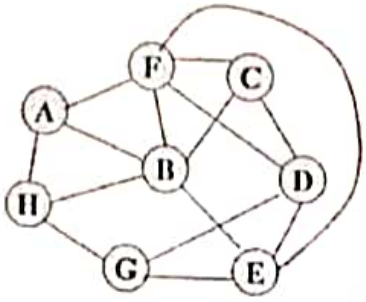
Q9. With proper justification answer the following:

a. Let **G** be a directed graph whose vertex set is the set of numbers from 1 to 100. There is an edge from a vertex i to a vertex j if and only if either $j = i + 1$ or $j = 3i$. The minimum number of edges in a path in **G** from vertex 1 to vertex 100 is **[2]**

b. An undirected graph **G** contains n ($n > 2$) nodes named v_1 , v_2 ,....v_n. Two nodes v_i and v_j are connected if and only if $0 < |i - j| <= 2$. Each edge (v_i, v_j) is assigned a weight $i + j$. A sample graph with $n = 4$ is shown below. What will be the cost of the minimum spanning tree (MST) of such a graph with n nodes? **[3]**



c. For the graph given below write down the Depth First Search and Breadth First Search traversals. Starting vertex shall be *A* and adjacent vertices shall be picked in lexicographic order only. **[2]**



d. Draw all distinct binary trees with 3 nodes (ignore the information of nodes). How many such binary trees will be there for n nodes? **[3]**