Total Number of Pages: 02   **Course: B.Tech**
Sub_Code: RAE4G003/RAU4G003/REC4G002/ RME4G003/
RMF4G003/RMM4G003/RMN4G003/RPL4G003

4th Semester Back Examination: 2024-25
SUBJECT: Data Structure
BRANCH(S): AME, AUTO, ECE, ETC, MECH, MANUTECH, METTA, MME, MINING, PLASTIC
Time: 3 Hours
Max Marks: 100
Q.Code: S232
Answer Question No.1 (Part-I) which is compulsory, any eight from Part-II and any two from Part-III.
**The figures in the right hand margin indicate marks.**

Part-I

Q1   **Answer the following questions:**   **(2 x 10)**
a)   How many stacks are required to implement Queue?
b)   Write the condition to test "Queue is Empty and "Queue is full".
c)   Compute the postfix equivalent of the following expression: 3*log (x + 1) – a/2.
d)   Define a complete binary tree and strictly binary tree with suitable example.
e)   Which data structure are applied when dealing with a recursive function.
f)   Show that for any non-empty binary tree, if T0 is the number of Leaf nodes and T2 is the number of full nodes, then T0 = T2 + 1.
g)   The preorder traversal of a binary search tree T is: < 40, 20, 10, 30, 60, 50, and 70 >. Convert it into the inorder traversals of T.
h)   Illustrate the representation of doubly linked list. Give an example of double linked list with 4 nodes.
i)   Classify the hashing function based on the various methods by which key value is found
j)   Explain about threaded binary tree with an example.
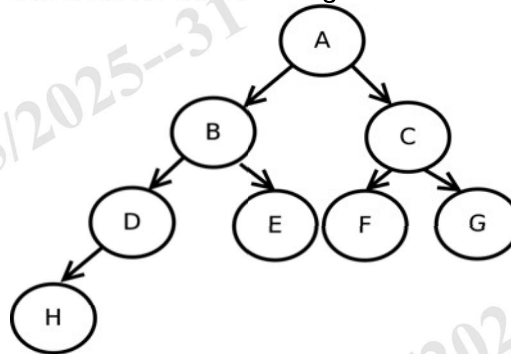
Part-II

Q2   **Only Focused-Short Answer Type Questions- (Answer Any Eight out of Twelve)**   **(6 x 8)**
a)   Construct the spanning tree using the Breadth First Search for the following graph, starting from node 1
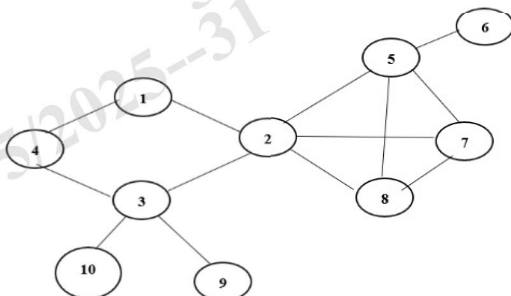
b)   Differentiated between array and linked list.
c)   Write a Program to reverse a String using Stack.
d)   Differentiate Binary tree and B-Tree. Construct a B-Tree of order 4 from the given list of elements:
12, 75, 60, 20, 115, 42, 85, 138, 100, 55, 185, 99, 168, 240, 30

e) Discuss the advantage and average efficiency of quick sort? Apply Quick sort on the following data and show the contents of the array every pass:
48, 7, 26, 4, 13, 23, 98, 57, 10, 5, 32

f) Convert the following list of keys into a Max heap and sort them using the Heap Sort technique.
8, 20, 9, 4, 15, 10, 7, 22, 3, 12

g) Write function for push, pop operation on the stack and add and delete functions on the queue.

h) Differentiated between linear and nonlinear data structure. List out few application of data structure.

i) Let A is the array of the following elements **A = {2, 4, 6, 8, 9, 10, 12, and 13}**. Search the element **12** using binary search technique. Write its time complexity.

j) Write the recursive procedure for different tree traversal techniques and perform the preorder, in order and post order traversal for the following tree:

k) Define a Sparse Matrix. Specify an Efficient Representation of Sparse Matrix.

l) Write a program to arrange the list of numbers in ascending order using bubble sort

Part-III

Only Long Answer Type Questions (Answer Any Two out of Four)   (16 x 2)

Q3 **a)** Write a Program to perform insertion and Deletion from a Linear Array.   **(8 x 2)**

b) Write a Program to Search an Element in a List using Binary Search Technique. Specify the Time Complexity of Binary Search. If a List has 2048 Elements, compute the number of comparisons required in Linear Search and Binary Search.

Q4 **a)** Convert the expression ((A + B) – C)/ (D – E) * (F + G) to equivalent Postfix notations using a Stack.   **(8 x 2)**

b) Describe the Algorithm for the Evaluation of Postfix Expression. Apply the postfix evaluation algorithm to evaluate the expression 1 5 7 3 – 2 ^ * +

Q5 **a)** Describe the different collision resolution techniques. Apply chaining and linear probing to store the following values in a hash table of size 7:
25, 42, 96, 101, 102, 162, 197   **(8 x 2)**

b) Arrange the following values in ascending order using insertion sort and selection sort.
23, 11, 37, 28, 15, 19, 55, 9

Q6 **a)** Write an algorithm/program to perform the following operations in a Doubly linked list: **(8 x 2)**
   i) To delete the last node.
   ii) To insert a node after a given node.

b) Insert the following nodes in an AVL tree.
21, 26, 30, 9, 4, 14, 28, 18, 15, 10,2, 3