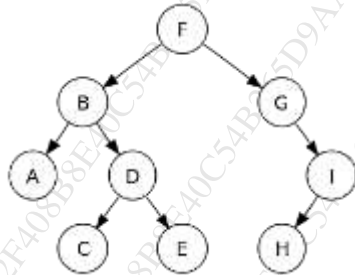


(3 Hours)

Total Marks: 80

N.B: (1) Question No. 1 is compulsory**(2) Attempt any three questions out of the remaining five questions**

- Q.1** (a) Define ADT. Write ADT for Queue data structure. [05]
 (b) Find the in-order, pre-order, post-order traversal [05]



- (c) Differentiate between Linked list and Array [05]
 (d) Explain application of Binary tree [05]

- Q.2** (a) Apply Huffman coding for following examples. Determine the code for the following characters. "CONSTRUCTION" [10]

- (b) Consider a hash table with size = 10. Using Linear probing, insert the keys 28, 55, 71, 67, 11, 10, 90, 44 into the table. [10]

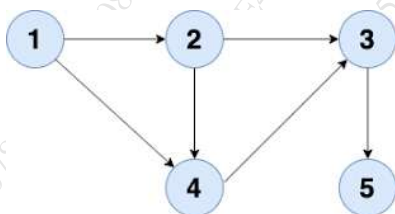
- Q.3** (a) Write an C program to check the well-formedness of parenthesis in an algebraic expression using the Stack data structure. [10]

- (b) Construct AVL for the given elements 27,25,23,29,35,33,34 [10]

- Q.4** (a) Write a program to perform the following operations on the Doubly linked list: [10]

- Insert a node at the end
- Delete a node from the beginning
- Search for a given element in the list
- Display the list

- (b) Write DFS algorithm. Show DFS traversal for the following graph with all the steps. [10]



- Q.5** (a) Define Data Structure. Explain its type with an example [10]
(c) Explain B tree. Draw the B-tree of order 3 created by inserting the following data arriving in sequence: 50, 25, 10, 5, 7, 3, 30, 20, 8, 15 [10]

- Q.6** (a) Draw the Stack structure in each case when the following operations are performed on an empty stack. [10]
i. PUSH A, B, C, D, E, F
ii. POP two letters
iii. PUSH G
iv. POP H
v. POP four letters
vi. PUSH I, J
vii. POP one letter
(b) Write a C program for polynomial addition using a Linked-list. [10]
