Construct an AVL tree for the following keys

34,30,32,15,13,20,31,25,28

Consider a double linked list is already created. Write C function for the following:

   (a) Find the number of occurrences of a given key
   (b) Delete the last occurrence of the key
   (c) Print the contents of double linked list in reverse order
   (d) Create a header node, which will be the first node of double linked list. The header node will store number of nodes in double linked list.

For the graph given below depict the DFS and BFS traversal with starting node as S. Select the edge/vertex in alphabetical order:
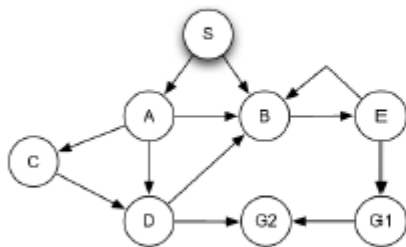
Write a C function for BFS traversal.

Define the following terms

   1) Strongly connected graph
   2) Cyclic graph
   3) Degree of a node in directed graph

Write a C program to construct a binary search tree. Create a function to find the inorder successor of a given node.

Consider a circular linked list is already created, write a function to perform the following operations :—

(1) Delete the first node.

(2) Count the number of nodes in the linked list.   5(CO2)

The following questions use a linked list consisting of nodes as :

```
typedef struct Node {
   char  data ;
   struct  Node  *next ;
} Node ;
Variable Node  *head ;  holds the first node of the list.
```

Compute FindSpecialChar(str) for str = "merge", "quick", and "recursion" to get an idea for what the function FindSpecialChar(str) does. State the output

(return value) of FindSpecialChar(str) in words for a given str. Explain how did you reach to the conclusion.

```
# include <stdlib.h>
# include <string.h>

char FindSpecialChar(chr *str) {
    Node  *head = (Node  *) malloc (sizeof (Node)) ;
    Node  *p = head ;
    Node  *q = 0,  *r = 0 ;
    int  i,  n = strlen (str) ;
    // Form  a  list  with  the  characters  of  the  string
    for (i = 0 ;  i < n – 1 ;  ++i) {
        p → data = str[i] ;
        p → next = (Node  *)  malloc(sizeof (Node)) ;
        p = p → next ;
        p → next = NULL ;
    }
```

```
        p → data = str[n − 1] ;
        // Find  a  special  character
        q = p = head ;
        while  (p) {
                p = p → next ;
                if  (p) {
                        p = p → next ;
                        q = q → next ;
                }
        }
        return  q → data ;
}
```

(a)   Sort the following numbers in increasing order with the help of heap sort :

$$30, \quad 60, \quad 51, \quad 40, \quad 15, \quad 95, \quad 89$$

Show the contents of array alongwith individual heaps after each number is placed at its correct position.   5(CO3)

(b)   Write a function to perform binary search. Also discuss the complexity of binary search.   5(CO3)

(a)   Consider a binary search tree and write a function to perform the following :—

   (i)   Count number of nodes in a binary tree.

   (ii)   Print the data of nodes in increasing order.   6(CO4)

(b)   Construct a B+ Tree of Order 4

        11, 21, 31, 3, 6, 8, 15, 27, 32, 34   4(CO4)

What is collision in hashing ? Explain the best collision resolution technique. Why is it the best ? Explain the technique with proper example.
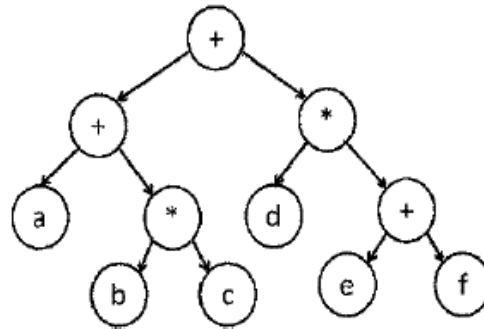   4(CO4)

Consider an array of integers : A = [37, 19, 42, 31, 54, 23, 67, 58].

 (a)   Write a Merge Sort algorithm to sort the given array.

 (b)   Provide a step-by-step trace of the Merge Sort algorithm, demonstrating how the array is sorted.   5(CO2)

(a)   What is binary tree ? Give the preorder, inorder and postorder traversals of the given binary tree :

4(CO3)

Consider a binary search tree and write C-functions to :

(A) Create the tree.

(B) Locate the node with the smallest key value.

(C) Count the number of parent nodes (excluding the root).

Trace your functions when a BST will be constructed for the following keys in the specified sequence : 65, 35, 55, 90, 75, 86, 98, 40, 20, 80.   6(CO3)

(a)   Use mid square method to store the list, H = {853, 951, 125, 342, 396, 456, 765, 901, 567, 547} in a hash table of size 11. Enlist the merits and limitations of mid square method.   5(CO3)

(b)   Find the minimum cost spanning tree for the below undirected graph using Prim's approach and show all steps.

5(CO3)

❖
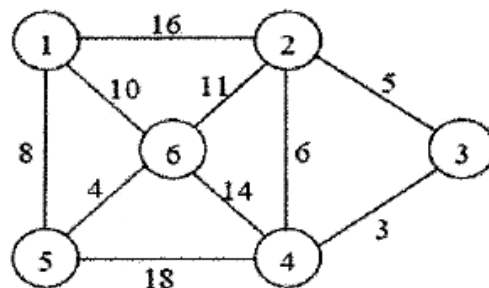
(a)   Convert expression A + (B* C- (D/E^ F)* G) *H into its equivalent reverse polish notation form. Evaluate the converted expression by considering A = 2, B = 3, C = 4, D = 4, E = 2, F = 2, G = 3, H = 2. Show stack contents on each stage of conversion and evaluation.   5(CO2)

(b)   A Double- Ended Queue, DQUE is to be implemented using an array of size N.Write C functions.

    (1) To implement DQUE as an output restricted deque.

    (2) To implement DQUE as an input restricted deque.

    Write conditions for queue empty and queue full.   5(CO2)

Solve any **Two** :—

(a) Write C program to insert a node at immediate left of jth node in a singly linked list. Your program should first create a singly linked list with 'n' nodes and then should carry out specified insertion operation (using function).
5(CO2)

(b) Give a linked list sorted in ascending sequence according to a key field in the nodes. Write C function to delete all the nodes in the list which have key values between two k1 and k2.
5(CO2)

(c) Build C functions to relize a linked stack in constant time. Write a program using these functions to simulate Stack ADT. The program should perform basic operations on ADT.
5(CO2)

Solve any **Two** :—

(a) Write an algorithm to order a list using modified bubble sort. Your algorithm should ensure non-quadratic running time when the input has all keys equal.
5(CO3)

(b) Write pseudocode to implement recursive MERGESORT. Demonstrate the recursive Merge Sort algorithm to sort the given data {38, 27, 43, 3, 9, 82, 10, 12, 19, 22, 89}. Also draw recursion tree and indicate the order in which steps are processed.
5(CO3)

(c) Consider given list L = {68, 66, 71, 69, 65,72, 67, 70}. Sort the list in descending order using a tree based sorting method. Show intermediate configurations while carrying out the sort. 5(CO3)

(a) If you have given the post-order and in-order of nodes, show that the binary tree structure can be constructed. Write C function to achieve the construction of binary tree as above. 5(CO3,4)

(b) Show the results of inserting the keys F, S, Q, K, C, L, H, T, V, W, M, R, N, P, A, B, X, Y, D, Z, E in the given sequence into an empty B+ – tree with each node storing maximum of 3 Keys, Draw the configurations of the tree before some node get split, and also draw the final configuration. Also, show configurations of B+Tree while removing the keys V and T. 5(CO3,4)

(a) For a given graph, determine shortest path cost matrix and path matrix by considering all pairs of vertices as source and destination. Name the algorithm used for solving the above problem and also, write pseudo-code.
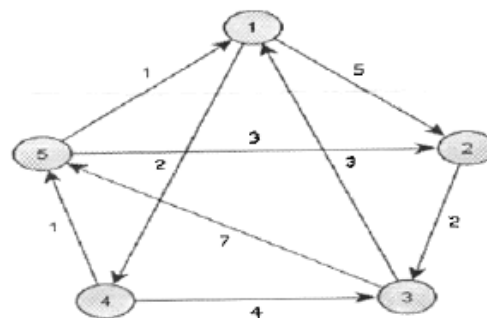
5(CO3,4)

(b) List out factors considered for good hash function. Explain various hash functions with example. For the set of keys {1015, 2025, 3025, 2028, 4049, 5052, 8094, 4421, 4068, 1017, 9080, 7523, 6192}, obtain a hash table of size 13 using quadratic probing by considering division hash function. 5(CO3,4)