Name: KAUSHAL VASHISTH Roll Number: 18ETCS002147

Laboratory 9

1. Questions

- 1. Write a C program to construct a binary search tree and perform the Preorder, post order and in order
- 2. Write a C program to implement a linked list to construct a tree and count the number of leaves in a tree.

2. Program

Q1) program to construct a binary search tree and perform the Pre order, post order and in order traversal:-

```
1
   = #include <stdio.h>
   #include <stdlib.h>
 2
 3
     struct node
 4
   □ {
 5
         int info;
 6
          struct node *left, *right;
   L };
 7
     struct node *newnode(int element)
 8
 9
   □ {
10
          struct node *temp=(struct node *)malloc(sizeof(struct node));
          temp->info=element;
11
          temp->left=temp->right=NULL;
12
13
     struct node *insert(struct node *root, int key)
14
   □ {
15
16
          if(root==NULL)
17
             return newnode (key);
          if(key<root->info)
18
              root->left=insert(root->left, key);
19
20
          if(key>root->info)
21
              root->right=insert(root->right, key);
22
          return root;
23
```

Name: KAUSHAL VASHISTH

Roll Number: 18ETCS002147

```
24
      void inorder(struct node *root)
   - {
25
26
          if(root==NULL) return;
27
          inorder(root->left);
          printf(" %d", root->info);
28
          inorder (root->right);
29
30
31
      void postorder(struct node *root)
32
33
          if(root==NULL) return;
34
          postorder(root->left);
          postorder(root->right);
35
          printf(" %d", root->info);
36
37
      void preorder(struct node *root)
38
39
40
          if (root==NULL) return;
          printf(" %d", root->info);
41
42
          preorder(root->left);
          preorder(root->right);
43
44
45
   □ int main(int argc, char** argv) {
          struct node *root=NULL;
46
47
   int a[7] = \{10, 9, 11, 4, 5, 6, 8\};
48
          int i;
          root=insert(root,a[0]);
49
50
          for(i=1;i<7;i++)</pre>
51
   52
              insert(root, a[i]);
53
          printf("preorder= ");
54
55
          preorder(root);
          printf("\n");
56
          printf("postorder= ");
57
58
          postorder(root);
          printf("\n");
59
          printf("inorder= ");
60
          inorder(root);
61
62
          return (EXIT SUCCESS);
63
```

Name: KAUSHAL VASHISTH Roll Number: 18ETCS002147

Q2) to implement a linked list to construct a tree and count the number of leaves in a tree:-

```
1 = #include <stdio.h>
 2
   #include <stdlib.h>
     struct node
 3
  □ {
 4
 5
         int data;
 6
         struct node* left;
 7
         struct node* right;
 8
    └ };
     unsigned int LeafCount(struct node* node)
 9
10
     if(node == NULL)
11
12
          return 0;
13
     if(node->left == NULL && node->right==NULL)
14
          return 1;
15
     else
         return LeafCount(node->left)+
16
              LeafCount(node->right);
17
18
19
```

Name: KAUSHAL VASHISTH

```
20
     struct node* newNode(int data)
21
   22
     struct node* node = (struct node*)
23
                          malloc(sizeof(struct node));
24
     node->data = data;
25
     node->left = NULL;
26
     node->right = NULL;
27
28
     return (node);
   L }
29
30
31
     int main()
   = {
32
33
     struct node *root = newNode(1);
34
     root->left = newNode(2);
     root->right = newNode(3);
35
36
    root->left->left = newNode(4);
     root->left->right = newNode(5);
37
38
     printf("Leaf count of the tree = %d", LeafCount(root));
```

Roll Number: 18ETCS002147

3. Presentation of Results

return 0;

Q1:-

40 41

```
preorder= 10 9 4 5 6 8 11
postorder= 8 6 5 4 9 11 10
inorder= 4 5 6 8 9 10 11
RUN SUCCESSFUL (total time: 145ms)
```

```
Leaf count of the tree = 3
RUN SUCCESSFUL (total time: 71ms)
```

Name: KAUSHAL VASHISTH Roll Number: 18ETCS002147

4. Conclusions

All the programs have been executed successfully.

1. We have learned to to construct a binary search tree and perform the Preorder, post order and in order

2. And we have learned to implement a linked list to construct a tree and count the number of leaves in a tree.