

Aim:

Experiment No. 08 (Implementation of various operations on tree like – copying tree, mirroring a tree,

counting the number of nodes in the tree, etc.)

SOURCE CODE

```
1// C program to convert a binary tree
2// to its mirror
3#include <stdio.h>
4#include <stdlib.h>
5
6/* A binary tree node has data, pointer
7to left child and a pointer to right child */
8struct Node
9{
10    int data;
11    struct Node* left;
12    struct Node* right;
13};
14
15/* helper function that allocates a new node with the
16given data and NULL left and right pointers. */
17struct Node* newNode(int data)
18{
19    struct Node* node = (struct Node*)
20        malloc(sizeof(struct Node));
21    node->data = data;
22    node->left = NULL;
23    node->right = NULL;
24    return(node);
25}
26
27/* Change a tree so that the roles of the left and
28right pointers are swapped at every node.
29So the tree...
30    4
31   / \
32  2   5
33 / \
34 1  3
35
36is changed to...
37    4
38   / \
39  5   2
40 / \
41 3  1
42
43*/
44void mirror(struct Node* node)
45{
46    if (node == NULL)
47        return;
48    else
49    {
50        struct Node* temp;
51        /* do the subtrees */
52        mirror(node->left);
53        mirror(node->right);
54        /* swap the pointers in this node */
55        temp = node->left;
56        node->left = node->right;
57        node->right = temp;
58    }
59}
60
61int countnodes(struct Node*root)
```

```
Activities Text Editor Oct 1 09:46 exp8.c Save
61 node->left = node->right;
62 node->right = temp;
63 }
64 }
65 int countnodes(struct Node*root)
66 {
67     int count=0;
68     if(root !=NULL)
69     {
70         countnodes(root->left);
71         count++;
72         countnodes(root->right);
73     }
74 }
75 return count;
76 }
77 // return a tree which is exact copy of past tree
78 struct Node *cloneBinaryTree(struct Node *root)
79 {
80     if(root==NULL)
81         return NULL;
82     //create a copy of root node
83     struct Node *NNew=newNode(root->data);
84     // recursively create clone of left and right sub tree
85     NNew->left = cloneBinaryTree(root->left);
86     NNew->right= cloneBinaryTree(root->right);
87     //return root of cloned tree
88     return NNew;
89 }
90 /* Helper function to print Inorder traversal.*/
91 void inOrder(struct Node* node)
92 {
93     if (node == NULL)
94         return;
95 }
96 inOrder(node->left);
97 printf("%d ", node->data);
98 inOrder(node->right);
99 }
100
101
102 /* Driver program to test mirror() */
103 int main()
104 {
105     struct Node *clone;
106     struct Node *root = newNode(1);
107     root->left = newNode(2);
108     root->right = newNode(3);
109     root->left->left = newNode(4);
110     root->left->right = newNode(5);
111 }
112 /* Print inorder traversal of the input tree */
113 printf("Inorder traversal of the constructed
114        tree is \n");
115 inOrder(root);
116
117 /* Convert tree to its mirror */
118 mirror(root);
119
120 /* Print inorder traversal of the mirror tree */
121 printf("\nInorder traversal of the mirror tree"
122        " is \n");
123 inOrder(root);
124 clone =cloneBinaryTree(root);
125 printf("\nInorder traversal of the clone tree"
126        " is \n");
127 inOrder(clone);
128 printf("\n number of nodes in tree = %d \n",countnodes(root));
129
130 return 0;
131 }
132
133 printf("\nInorder traversal of the mirror tree"
134        " is \n");
135 inOrder(root);
136 clone =cloneBinaryTree(root);
137 printf("\nInorder traversal of the clone tree"
138        " is \n");
139 inOrder(clone);
140 printf("\n number of nodes in tree = %d \n",countnodes(root));
141
142 return 0;
143 }
```

OUTPUT

```
Activities Terminal Oct 1 10:05 dl416@itadm1n: -  
dl416@itadm1n:~$ ./a.out  
Inorder traversal of the constructed tree is  
4 2 5 1 3  
Inorder traversal of the mirror tree is  
3 1 5 2 4  
Inorder traversal of the clone tree is  
3 1 5 2 4  
number of nodes in tree = 1  
dl416@itadm1n:~$ gedit exp8.c  
dl416@itadm1n:~$ gedit exp8.c  
dl416@itadm1n:~$ gcc exp8.c  
dl416@itadm1n:~$ ./a.out  
Inorder traversal of the constructed tree is  
4 2 5 1 3  
Inorder traversal of the mirror tree is  
3 1 5 2 4  
Inorder traversal of the clone tree is  
3 1 5 2 4  
number of nodes in tree = 5  
dl416@itadm1n:~$ gedit exp8.c
```