

```

1  #include<stdio.h>
2  #include<stdlib.h>
3
4  struct BinaryNode
5  {
6      struct BinaryNode *left;
7      int data;
8      struct BinaryNode *right;
9  };
10
11 struct BinaryNode * createBinaryNode(int value)
12 {
13     struct BinaryNode *B=(struct BinaryNode *)malloc(sizeof(struct BinaryNode));
14     B->left=NULL;
15     B->data=value;
16     B->right=NULL;
17     return B;
18 };
19 struct BinaryNode* FindMax(struct BinaryNode *);
20
21 struct BinaryNode *deleteNode(struct BinaryNode *root, int value)
22 {
23
24     if(root==NULL)
25     {
26         printf("Cant Delete");
27         return NULL;
28     }
29
30     if(value<root->data)
31         root->left=deleteNode(root->left,value);
32     else if(value>root->data)
33         root->right=deleteNode(root->right,value);
34     else
35     {
36         if(root->left&&root->right)
37         {
38             struct BinaryNode *temp=FindMax(root->left);
39             root->data=temp->data;
40             root->left=deleteNode(root->left,root->data);
41
42         }
43         else
44         {
45             if(root->right==NULL)
46                 root=root->left;
47             else if(root->left==NULL)
48                 root=root->right;
49             else
50                 root=NULL;
51         }
52     }
53     return root;
54
55
56
57 };
58
59 struct BinaryNode * FindMax(struct BinaryNode *root)
60 {
61     if(root==NULL)
62         return NULL;
63     if(root->right==NULL)
64         return root;
65     else
66         return FindMax(root->right);

```

```

67 };
68
69 void inOrderTraversal(struct BinaryNode *root)
70 {
71     if(!root)
72         return NULL;
73     inOrderTraversal(root->left);
74     printf("%d ", root->data);
75     inOrderTraversal(root->right);
76 }
77
78 void main()
79 {
80     struct BinaryNode *root=createBinaryNode(50);
81     root->left=createBinaryNode(30);
82     root->right=createBinaryNode(60);
83     root->left->left=createBinaryNode(20);
84     root->left->right=createBinaryNode(35);
85
86
87     inOrderTraversal(root);
88
89     deleteNode(root,35);
90
91     printf("\n\nAFTER DELETION\n\n");
92
93     inOrderTraversal(root);
94
95 }

```