

```

1  #include<stdio.h>
2  #include<stdlib.h>
3
4  struct BinaryTree
5  {
6      struct BinaryTree *left;
7      int data;
8      struct BinaryTree *right;
9  };
10
11 struct BinaryTree * CreateBinaryNode(int value)
12 {
13     struct BinaryTree *B=malloc(sizeof(struct BinaryTree));
14     B->left=NULL;
15     B->right=NULL;
16     B->data=value;
17     return B;
18 };
19
20 int pathExistence(struct BinaryTree *root,int sum)
21 {
22     int remaining_sum;
23     if(!root)
24         return 0;
25     remaining_sum=sum-root->data;
26     if(remaining_sum==0)
27         return 1;
28
29     if(root->left || root->right)
30     {
31         int temp= pathExistence(root->left,remaining_sum);
32         if(temp!=1)
33             return pathExistence(root->right,remaining_sum);
34         else return temp;
35     }
36     else
37         return 0;
38 }
39
40 int main()
41 {
42     struct BinaryTree *root;
43     root=CreateBinaryNode(10);
44     root->left=CreateBinaryNode(20);
45     root->left->left=CreateBinaryNode(30);
46     root->right=CreateBinaryNode(40);
47     root->right->right=CreateBinaryNode(50);
48     printf("Path Exists with given sum 100: %d",pathExistence(root,100));
49
50
51
52
53
54
55 }

```