Count and Height of a Tree

```c
#include <stdio.h>
#include <stdlib.h>
#include "Queue.h"
#include "Stack.h"

struct Node *root=NULL;

void Treecreate()
{
    struct Node *p,*t;
    int x;
    struct Queue q;
    create(&q,100);

    printf("Eneter root value ");
    scanf("%d",&x);
    root=(struct Node *)malloc(sizeof(struct Node));
    root->data=x;
    root->lchild=root->rchild=NULL;
    enqueue(&q,root);

    while(!isEmpty(q))
    {
        p=dequeue(&q);
        printf("eneter left child of %d ",p->data);
        scanf("%d",&x);
        if(x!=-1)
        {
            t=(struct Node *)malloc(sizeof(struct Node));
            t->data=x;
            t->lchild=t->rchild=NULL;
            p->lchild=t;
            enqueue(&q,t);
        }
        printf("eneter right child of %d ",p->data);
        scanf("%d",&x);
        if(x!=-1)
        {
            t=(struct Node *)malloc(sizeof(struct Node));
            t->data=x;
            t->lchild=t->rchild=NULL;
            p->rchild=t;
            enqueue(&q,t);
        }
    }
```

```
}

int count(struct Node *root)
{
    if(root)
        return count(root->lchild)+count(root->rchild)+1;
    return 0;
}
int height(struct Node *root)
{
    int x=0,y=0;
    if(root==0)
        return 0;
    x=height(root->lchild);
    y=height(root->rchild);
    if(x>y)
        return x+1;
    else
        return y+1;

}


int main()
{
    Treecreate();

    printf("%d",count(root));
    printf("%d",height(root));


    return 0;
}
```