

2020 年 8 月 15 日

1.P1087 FBI树

算法思路：

建树:将字符串按左子树和右子树递归，当字符串长度为1时，即为叶子结点，如果是'1'，则结点值为'I'，否则为'B'，而非叶子结点的结点则看左右儿子的值，若左右儿子的值相同则父亲结点的值同儿子结点一样，否则为'F'。最后后序输出。

代码：

```
#include <iostream>
#include <cstring>
using namespace std;

struct TREE
{
    char value;
    TREE *left , *right;
};

TREE* digui(string s,int n){
    if(s.length() == 1){
        TREE *node = new TREE;
        node->value = (s[0] == '1') ? 'I' : 'B';
        node->left = node->right = NULL;
        return node;
    }

    TREE *node = new TREE;
    node->left = digui(s.substr(0,n/2),n/2);
    node->right = digui(s.substr(n/2),n/2);
    node->value = (node->left->value == node->right->value) ? node->left-
>value : 'F';
    return node;
}

void print(TREE *node){
    if(node == NULL) return;
    print(node->left);
    print(node->right);
    cout<< node->value;
}
```

```
int main(){
    int n;
    cin >> n;
    string s;
    cin >> s;
    TREE *root = digui(s,s.length());
    print(root);
    return 0;
}
```

Accepted截图：



wongsiyoung

所属题目	P1087 FBI树
评测状态	Accepted
评测分数	100
提交时间	2020-08-15 14:47:33

2.P3884 [JLOI2009]二叉树问题

算法思路：

那只好建了树再dfs咯，但我不想改了嘿嘿嘿

代码：

```
#include <iostream>
#include <cstring>
using namespace std;
int d[101],w[101],tree[101][101];
int main(){
    int n,x,y,depth = 0,width = 0;
    cin >> n;
    d[1] = 1;
    memset(tree,63,sizeof(tree)); // !!!这里的63约等于1e9

    /*前提是输入按顺序*/
    for(int i = 0; i < n-1; i++){
        cin >> x >> y;
        d[y] = d[x] + 1; //该结点的深度 = 父亲结点+1
        depth = max(depth,d[y]);
        w[d[y]]++; //该结点所在层宽度+1
        width = max(width,w[d[y]]);
    }
```

```

        tree[x][y] = 1;
        tree[y][x] = 2;
    }
    cout << depth << endl << width << endl;
    cin >> x >> y;
    for (int i = 1; i <= n; i++)
        for (int j = 1; j <= n; j++)
            for (int k = 1; k <= n; k++)
                tree[j][k] = min(tree[j][k], tree[j][i] + tree[i][k]);
    cout << tree[x][y];
    return 0;
}

```

Accepted截图：


wongsiyoung

所属题目	P3884 [JLOI2009]二叉树问题
评测状态	Accepted
评测分数	100
提交时间	2020-08-15 15:29:03

备注：

- wow，原来你是这样的memset：
 - memset以字节赋值，63 = 00111111，int类型是4个字节，它就会把每个int都赋值成00111111001111110011111100111111 $\approx 1e9$ 。127约等于int最大值，这里取63是约等于int最大值一半，为了防止后面Floyd算法int相加爆int。
 - memset只可以把0和-1按你想象的样子赋值。(1也不可)
 - char数组可以按想的那样赋值，因为char只有一个字节
 - 真要每个数组复制63：fill(a, a+n, 63)，不过只适用于一维数组，二维的话就for循环