一、题目说明

题目124. Binary Tree Maximum Path Sum，给一个非空二叉树，求最大路径之和。

二、我的解答

这个题目，经过几个小时的思考，终于做出来了。一个树的最大路径，可能出现在“左子树”，“右子树”，或者包含“根节点”。其中dfs用来递归计算从根到叶的最大路径之和。

```
class Solution{
    public:
        int dfs(TreeNode* root){
            if(root==NULL) return INT_MIN;
            if(root->left==NULL && root->right==NULL){
                if(root->val>curMax) curMax = root->val;
                return root->val;
            }

            int left = dfs(root->left);
            int right = dfs(root->right);

            //当左右子树最大路径之和为正，则可能产生最大路径
            int cur = root->val;
            if(left>0 && right>0){
                cur += left;
                cur += right;
                if(cur>curMax) curMax = cur;
            }

            cur = root->val;
            int m = max(left,right);
            if(m>0){
                cur += m;
            }

            if(cur>curMax) curMax = cur;
            return cur;
        }
        int maxPathSum(TreeNode* root){
            if(root==NULL) return INT_MIN;
            if(root->left==NULL && root->right==NULL){
                return root->val;
            }
            curMax = INT_MIN;
            if(root->val>curMax) curMax = root->val;

            int left = dfs(root->left);
            int right = dfs(root->right);

            int cur = root->val;
            if(left>0 && right>0){
                cur += left;
                cur += right;
            }else if(left>0){
                cur += left;
            }else if(right>0){
                cur += right;
            }

            if(cur>curMax) curMax = cur;
            return curMax;
        }
    private:
        int curMax;
};
```

性能如下:

```
Runtime: 28 ms, faster than 86.43% of C++ online submissions for Binary Tree
Maximum Path Sum.
Memory Usage: 24.2 MB, less than 96.97% of C++ online submissions for Binary
Tree Maximum Path Sum.
```

三、优化措施

无