一、题目说明

题目226. Invert Binary Tree，翻转一个二叉树。难度是Easy!

二、我的解答

这个题目，和二叉树的遍历类似。用递归方法（前、中、后序遍历，按层遍历都可以）：

```
class Solution{
    public:
        TreeNode* invertTree(TreeNode* root){
            if(root ==NULL) return root;
            TreeNode * p = root->left;
            root->left = root->right;
            root->right = p;
            root->left = invertTree(root->left);
            root->right = invertTree(root->right);
            return root;
        }
};
```

性能如下：

```
Runtime: 4 ms, faster than 65.13% of C++ online submissions for Invert Binary
Tree.
Memory Usage: 10 MB, less than 5.45% of C++ online submissions for Invert Binary
Tree.
```

三、优化措施

非递归的算法，下面用广度优先遍历实现：

```
class Solution{
    public:
        //non-recursive using level
        TreeNode* invertTree(TreeNode* root){
            if(root == NULL) return root;
            queue<TreeNode*> q;
            TreeNode* tmp,*cur;
            q.push(root);
            while(! q.empty()){
                cur = q.front();
                q.pop();
                tmp = cur->left;
                cur->left = cur->right;
                cur->right = tmp;
                if(cur->left !=NULL ){
                    q.push(cur->left);
                }
                if(cur->right != NULL){
                    q.push(cur->right);
                }
            }
```

```
            return root;
        }
};
```

性能如下：

```
Runtime: 0 ms, faster than 100.00% of C++ online submissions for Invert Binary
Tree.
Memory Usage: 10.2 MB, less than 5.45% of C++ online submissions for Invert
Binary Tree.
```