

```
/*
二叉树的层平均值
Category    Difficulty Likes Dislikes
algorithms Easy (60.75%) 74 -
Tags
tree
```

Companies
facebook

给定一个非空二叉树，返回一个由每层节点平均值组成的数组。

示例 1:

输入:

```
      3
     /\
    9 20
   /\
  15 7
```

输出: [3, 14.5, 11]

解释:

第0层的平均值是 3， 第1层是 14.5， 第2层是 11。 因此返回 [3, 14.5, 11]。

注意:

节点值的范围在32位有符号整数范围内。

```
*/
```

分析：

- 常规的层次遍历即可，然后在每层求均值即可。

方法一:C++_辅助队列迭代bfs

```
class Solution
{
public:
    vector<double> averageOfLevels(TreeNode* root)
    {
        vector<double>    ret_val        ;
        queue<TreeNode*>   qt             ;
        int                num_1    = 0    ;
        int                num_2    = 0    ;
        TreeNode*         temp_tn = NULL  ;
        double             temp_db = 0    ;
        int                i          = 0    ;
        if(root!=NULL)
        {
            qt.push(root);
            num_2 = 1;
            num_1 = 0;

            while(!qt.empty())
```

```

    {
        num_1 = num_2 ;
        num_2 = 0 ;
        temp_db = 0 ;

        for(i = 0; i < num_1 ; i++)
        {
            temp_tn = qt.front();
            temp_db += temp_tn->val;
            if(temp_tn->left!=NULL)
            {
                qt.push(temp_tn->left);
                num_2++;
            }
            if(temp_tn->right!=NULL)
            {
                qt.push(temp_tn->right);
                num_2++;
            }

            qt.pop();
        }
        ret_val.push_back(temp_db/num_1);
    }
}
return ret_val;
};

```

/*

执行结果:

通过

[显示详情](#)

执行用时 :28 ms, 在所有 C++ 提交中击败了72.10% 的用户

内存消耗 :21.7 MB, 在所有 C++ 提交中击败了83.54%的用户

*/