

剑指Offer（五十九）：按之字顺序打印二叉树

🕒 2018年1月29日 10:48:25 🗨 6 🌡 3,583 °C 🛠 编辑



一、前言

本系列文章为《剑指Offer》刷题笔记。

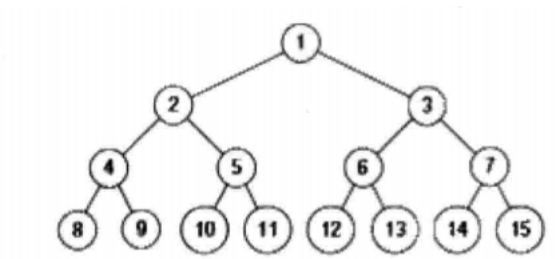
刷题平台：[牛客网](#)

书籍下载：[共享资源](#)

二、题目

请实现一个函数按照之字形打印二叉树，即第一行按照从左到右的顺序打印，第二层按照从右至左的顺序打印，第三行按照从左到右的顺序打印，其他行以此类推。

1、思路



按之字顺序打印上图二叉树，打印顺序为：

```
1
3 2
4 5 6 7
15 14 13 12 12 10 9 8
```

为了达到这样打印的效果，我们需要使用两个栈。我们在打印某一行结点时，把下一层的子结点保存到相应的栈里。如果当前打印的是奇数层（第一层、第三层等），则先保存左子树结点再保存右子树结点到第一个栈里。如果当前打印的是偶数层（第二层、第四层等），则则先保存右子树结点再保存左子树结点到第二个栈里。

步骤	操作	Stack1 中的结点	Stack2 中的结点
1	打印结点 1	2, 3	
2	打印结点 3	2	7, 6
3	打印结点 2		7, 6, 5, 4
4	打印结点 4	8, 9	7, 6, 5
5	打印结点 5	8, 9, 10, 11	7, 6
6	打印结点 6	8, 9, 10, 11, 12, 13	7
7	打印结点 7	8, 9, 10, 11, 12, 13, 14, 15	

详细步骤，如上图所示。

2、代码

C++:

C+

```
1  /*
2  struct TreeNode {
3      int val;
4      struct TreeNode *left;
5      struct TreeNode *right;
6      TreeNode(int x) :
7          val(x), left(NULL), right(NULL) {
8      }
9  };
10 */
11 class Solution {
12 public:
13     vector<vector<int> > Print(TreeNode* pRoot) {
14         vector<vector<int> > result;
15         if(pRoot == NULL){
16             return result;
17         }
18         stack<TreeNode* > s[2];
19         s[0].push(pRoot);
20         while(!s[0].empty() || !s[1].empty()){
21             vector<int> v[2];
22             // 偶数行
23             while(!s[0].empty()){
24                 v[0].push_back(s[0].top()->val);
25                 if(s[0].top()->left != NULL){
26                     s[1].push(s[0].top()->left);
27                 }
28                 if(s[0].top()->right != NULL){
29                     s[1].push(s[0].top()->right);
30                 }
31                 s[0].pop();
32             }
33             if(!v[0].empty()){
34                 result.push_back(v[0]);
35             }
36             // 奇数行
37             while(!s[1].empty()){
38                 v[1].push_back(s[1].top()->val);
39                 if(s[1].top()->right != NULL){
40                     s[0].push(s[1].top()->right);
41                 }
42                 if(s[1].top()->left != NULL){
43                     s[0].push(s[1].top()->left);
44                 }
45                 s[1].pop();
46             }
47             if(!v[1].empty()){
48                 result.push_back(v[1]);
49             }
50         }
51         return result;
52     }
53 };
```

Python:

感谢@小小毛提供python代码及本地测试用例。

Pytho

```
1  #- coding:utf-8 -*-
2  class TreeNode:
3      def __init__(self, x):
4          self.val = x
5          self.left = None
6          self.right = None
7  class Solution:
8      """docstring for Solution"""
```

```
9 def Print(self, pRoot):
10     resultArray = []
11     if not pRoot:
12         return resultArray
13     curLayerNodes = [pRoot]
14     isEvenLayer = True
15     while curLayerNodes:
16         curLayerValues = []
17         nextLayerNodes = []
18         isEvenLayer = not isEvenLayer
19         for node in curLayerNodes:
20             curLayerValues.append(node.val)
21             if node.left:
22                 nextLayerNodes.append(node.left)
23             if node.right:
24                 nextLayerNodes.append(node.right)
25         curLayerNodes = nextLayerNodes
26         resultArray.append(curLayerValues[::-1]) if isEvenLayer else resultArray.append(curLayerValues)
27     return resultArray
28
29 if __name__ == '__main__':
30     A1 = TreeNode(1)
31     A2 = TreeNode(2)
32     A3 = TreeNode(3)
33     A4 = TreeNode(4)
34     A5 = TreeNode(5)
35     A6 = TreeNode(6)
36     A7 = TreeNode(7)
37
38     A1.left=A2
39     A1.right=A3
40     A2.left=A4
41     A2.right=A5
42     A3.left=A6
43     A3.right=A7
44
45     solution = Solution()
46     ans=solution.Print(A1)
47     print(ans)
```



微信公众号

分享技术，乐享生活：微信公众号搜索

「JackCui-AI」关注一个在互联网摸爬滚

打的潜行者。

芝兰生于幽林，不以无人而不芳；君子修道立德，不为穷困而改节。--- 孔子