

Assignment #5: "树"算：概念、表示、解析、遍历

Updated 2124 GMT+8 March 17, 2024

2024 spring, Compiled by 余汶青 生命科学学院

说明：

1) The complete process to learn DSA from scratch can be broken into 4 parts:

Learn about Time complexities, learn the basics of individual Data Structures, learn the basics of Algorithms, and practice Problems.

2) 请把每个题目解题思路（可选），源码Python, 或者C++（已经在Codeforces/Openjudge上AC），截图（包含Accepted），填写到下面作业模版中（推荐使用 typora <https://typoraio.cn>，或者用 word）。AC 或者没有AC，都请标上每个题目大致花费时间。

3) 提交时候先提交pdf文件，再把md或者doc文件上传到右侧“作业评论”。Canvas需要有同学清晰头像、提交文件有pdf、“作业评论”区有上传的md或者doc附件。

4) 如果不能在截止前提交作业，请写明原因。

编程环境

操作系统：版本 Windows 11 家庭中文版

版本 22H2

安装日期 2023/7/18

操作系统版本 22621.2283

序列号 5CD323PJKL

体验 Windows Feature Experience Pack 1000.22662.1000.0

Python编程环境：* Spyder version: 5.4.3 (conda)

- Python version: 3.11.4 64-bit
- Qt version: 5.15.2
- PyQt5 version: 5.15.7
- Operating System: Windows 10

1. 题目

27638: 求二叉树的高度和叶子数目

<http://cs101.openjudge.cn/practice/27638/>

思路：

代码

```

leaf=0
class treeNode:
    def __init__(self):
        self.right=None
        self.left=None

def depth(a):
    global leaf
    if a.left==None and a.right==None:
        leaf+=1
        return 0
    l=0
    r=0
    if a.left!=None:
        l=depth(a.left)
    if a.right!=None:
        r=depth(a.right)
    return max(l,r)+1

n=int(input())
tree=[treeNode() for i in range(n)]
root=[0]*n
for i in range(n):
    l,r=map(int,input().split())
    if l!=-1:
        tree[i].left=tree[l]
        root[l]=1
    if r!=-1:
        tree[i].right=tree[r]
        root[r]=1
rootnode=root.index(0)
print(depth(tree[rootnode]),leaf)

```

代码运行截图

#44307341提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: **Accepted**

源代码

```

leaf=0
class treeNode:
    def __init__(self):
        self.right=None
        self.left=None

def depth(a):
    global leaf

```

基本信息

#: 44307341
 题目: 27638
 提交人: 23n2300012265
 内存: 3676kB
 时间: 22ms
 语言: Python3
 提交时间: 2024-03-19 23:19:34

24729: 括号嵌套树

<http://cs101.openjudge.cn/practice/24729/>

思路:

代码

```
num=0
a=0
class treeNode:
    def __init__(self):
        self.leaf=[]
        self.name=''
    def printt(self):
        print(self.name,self.leaf)

def treebuild(n):
    global a,num
    tree[num].name=s[n]
    i=num
    num+=1
    a+=1
    if a<len(s) and s[a]==')':
        return
    if a<len(s) and s[a]=='(':
        a+=1
        tree[i].leaf.append(num)
        treebuild(a)
        while s[a]!=')':
            a+=1
            tree[i].leaf.append(num)
            treebuild(a)
        a+=1

def preprint(n):
    print(tree[n].name,end='')
    for i in tree[n].leaf:
        preprint(i)
def postprint(n):
    for i in tree[n].leaf:
        postprint(i)
    print(tree[n].name,end='')

s=input()
tree=[treeNode() for i in range(26)]
treebuild(0)
#for i in range(num):
#    tree[i].printt()
preprint(0)
print()
```

```
postprint(0)
```

代码运行截图

#44307895提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: **Accepted**

源代码

```
num=0
a=0
class treeNode:
    def __init__(self):
        self.leaf=[]
        self.name=''
    def printt(self):
```

基本信息

#: 44307895
题目: 24729
提交人: 23n2300012265
内存: 3672kB
时间: 25ms
语言: Python3
提交时间: 2024-03-20 00:10:30

02775: 文件结构“图”

<http://cs101.openjudge.cn/practice/02775/>

思路:

代码

```
class treeNode:
    def __init__(self,value):
        self.dir=[]
        self.file=[]
        self.name=value
    def printt(self,n):
        for _ in range(n):
            print('|',end='')
        print(self.name)
        for i in self.dir:
            i.printt(n+1)
        self.file.sort()
        for i in self.file:
            for _ in range(n):
                print('|',end='')
            print(i,sep='\n')

root=treeNode('ROOT')
stack=[root]
a=input()
n=1
while a!='#':
    if a=='*':
        print("DATA SET %d:"%(n))
        root.printt(0)
```

```

print()
root=treenode('ROOT')
stack=[root]
n+=1
elif a==']':
    stack.pop()
elif a[0]=='f':
    stack[-1].file.append(a)
elif a[0]=='d':
    node=treenode(a)
    stack[-1].dir.append(node)
    stack.append(node)
a=input()

```

代码运行截图

#44312939提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: Accepted

源代码

```

class treenode:
    def __init__(self,value):
        self.dir=[]
        self.file=[]
        self.name=value
    def printt(self,n):
        for _ in range(n):
            print(' | ',end='')

```

基本信息

#: 44312939
 题目: 02775
 提交人: 23n2300012265
 内存: 3640kB
 时间: 23ms
 语言: Python3
 提交时间: 2024-03-20 15:59:18

25140: 根据后序表达式建立队列表达式

<http://cs101.openjudge.cn/practice/25140/>

思路:

代码

```

class treenode:
    def __init__(self,value):
        self.left=None
        self.right=None
        self.name=value

n=int(input())
for _ in range(n):
    s=input()
    stack=[]
    for i in range(len(s)):
        if ord(s[i])>=97:

```

```

        node=treenode(s[i])
        stack.append(node)
    if ord(s[i])<97:
        node=treenode(s[i])
        r=stack.pop()
        l=stack.pop()
        node.left=l
        node.right=r
        stack.append(node)
root=stack.pop()
queue=[]
queue.append(root)
ans=[]
while queue:
    a=queue.pop(0)
    if a.left:
        queue.append(a.left)
    if a.right:
        queue.append(a.right)
    ans.append(a.name)
ans.reverse()
print(*ans,sep='')

```

代码运行截图

#44322357提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: Accepted

源代码

```

class treenode:
    def __init__(self,value):
        self.left=None
        self.right=None
        self.name=value
n=int(input())

```

基本信息

#: 44322357
 题目: 25140
 提交人: 23n2300012265
 内存: 3696kB
 时间: 29ms
 语言: Python3
 提交时间: 2024-03-21 11:51:49

24750: 根据二叉树中后序序列建树

<http://cs101.openjudge.cn/practice/24750/>

思路:

代码

```

class treenode:
    def __init__(self,value):
        self.left=None
        self.right=None
        self.name=value
    def printt(self):

```

```

        print(self.name, end='')
        l=self.left
        r=self.right
        if l:
            l.printt()
        if r:
            r.printt()

def build(ino,post):
    if len(post)==0:
        return
    if len(post)==1:
        node=treenode(post[0])
        return node
    root=treenode(post[-1])
    l,r=ino.split(post[-1])
    root.left=build(l,post[0:len(l)])
    root.right=build(r,post[len(l):-1])
    return root

inorder=input()
postorder=input()
root=build(inorder,postorder)
root.printt()

```

代码运行截图

#44322978提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: **Accepted**

源代码

```

class treenode:
    def __init__(self,value):
        self.left=None
        self.right=None
        self.name=value
    def printt(self):
        print(self.name, end='')

```

基本信息

#: 44322978
 题目: 24750
 提交人: 23n2300012265
 内存: 3656kB
 时间: 25ms
 语言: Python3
 提交时间: 2024-03-21 13:40:41

22158: 根据二叉树前中序序列建树

<http://cs101.openjudge.cn/practice/22158/>

思路:

代码

```

class treenode:
    def __init__(self,value):

```

```

        self.left=None
        self.right=None
        self.name=value
    def printt(self):
        l=self.left
        r=self.right
        if l:
            l.printt()
        if r:
            r.printt()
        print(self.name,end='')

def build(pre,ino):
    if len(pre)==0:
        return
    if len(pre)==1:
        node=treenode(pre[0])
        return node
    root=treenode(pre[0])
    l,r=ino.split(pre[0])
    root.left=build(pre[1:len(l)+1],l)
    root.right=build(pre[len(l)+1:],r)
    return root

while True:
    try:
        preorder=input()
        inorder=input()
        root=build(preorder,inorder)
        root.printt()
        print()
    except EOFError as e:
        break

```

代码运行截图

#44323107提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: **Accepted**

源代码

```

class treenode:
    def __init__(self,value):
        self.left=None
        self.right=None
        self.name=value
    def printt(self):
        l=self.left

```

基本信息

#: 44323107
 题目: 22158
 提交人: 23n2300012265
 内存: 3624kB
 时间: 21ms
 语言: Python3
 提交时间: 2024-03-21 13:57:13

2. 学习总结和收获

很有收获的一次作业，学到了很多树的知识，建树可以用栈和递归，输出则用递归。

由于都是模板题，所以写起来不算特别困难，感觉自己还有很多不足的地方，和大佬们的差距越来越大了🤔