

Assignment #8: 图论：概念、遍历，及树算

Updated 1919 GMT+8 Apr 8, 2024

2024 spring, Compiled by ==同学的姓名、院系==

说明：

- 1) 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。
- 2) 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。
- 3) 如果不能在截止前提交作业，请写明原因。

编程环境

==（请改为同学的操作系统、编程环境等）==

操作系统：macOS Ventura 13.4.1 (c)

Python编程环境：Spyder IDE 5.2.2, PyCharm 2023.1.4 (Professional Edition)

C/C++编程环境：Mac terminal vi (version 9.0.1424), g++/gcc (Apple clang version 14.0.3, clang-1403.0.22.14.1)

1. 题目

19943: 图的拉普拉斯矩阵

matrices, <http://cs101.openjudge.cn/practice/19943/>

代码

```
n,m=map(int,input().split())
l=[[int(i-i) for i in range(n)] for i in range(n)]
for _ in range(m):
    a,b=map(int,input().split())
    l[a][b]=-1
    l[b][a]=-1
for i in range(n):
    t=l[i]
    x=sum(t)
    l[i][i]=-x
for _ in l:
    k=map(lambda x:str(x),_)
    print(" ".join(k))
##用的以前的代码，还是比较直接的
```

状态: Accepted

源代码

```
n,m=map(int,input().split())
l=[int(i-i) for i in range(n)] for i in range(n)]
for _ in range(m):
    a,b=map(int,input().split())
    l[a][b]=-1
    l[b][a]=-1
for i in range(n):
    t=l[i]
    x=sum(t)
    l[i][i]=-x
for _ in l:
    k=map(lambda x:str(x),_)
    print(" ".join(k))
```

基本信息

#: 44667003
题目: 19943
提交人: 22n2200011800
内存: 7404kB
时间: 29ms
语言: Python3
提交时间: 2024-04-15 20:13:21

18160: 最大连通域面积

matrix/dfs similar, <http://cs101.openjudge.cn/practice/18160>

代码

```
n=int(input())
q=[]
for _ in range(n):
    x,y=map(int,input().split())
    l=[["0" for i in range(y+2)]]
    for _ in range(x):
        e=["0"]+list(input())+"0"
        l.append(e)
    l.append(["0" for i in range(y+2)])
    r=[[0 for i in range(y+2)] for i in range(x+2)]
    ans=0
    for i in range(1,x+1):
        for j in range(1,y+1):
            if l[i][j]=="w":
                k=[str(i)+" "+str(j)]
                g=[str(i)+" "+str(j)]
            else:
                k=[]
                g=[]
            while len(g)>0:
                a,b=map(int,g[0].split())
                if l[a][b]=="." or r[a][b]!=0:
                    g=[]
                    continue
                if "w"==l[a][b+1] and str(a)+" "+str(b+1) not in k:
                    k.append(str(a)+" "+str(b+1))
                    g.append(str(a)+" "+str(b+1))
                if "w"==l[a+1][b] and str(a+1)+" "+str(b) not in k:
                    k.append(str(a+1)+" "+str(b))
                    g.append(str(a+1)+" "+str(b))
                if "w"==l[a][b-1] and str(a)+" "+str(b-1) not in k:
                    k.append(str(a)+" "+str(b-1))
                    g.append(str(a)+" "+str(b-1))
```

```

        if "W"==l[a-1][b] and str(a-1)+" "+str(b) not in k:
            k.append(str(a-1)+" "+str(b))
            g.append(str(a-1)+" "+str(b))
        if "W"==l[a+1][b+1] and str(a+1)+" "+str(b+1) not in k:
            k.append(str(a+1)+" "+str(b+1))
            g.append(str(a+1)+" "+str(b+1))
        if "W"==l[a-1][b+1] and str(a-1)+" "+str(b+1) not in k:
            k.append(str(a-1)+" "+str(b+1))
            g.append(str(a-1)+" "+str(b+1))
        if "W"==l[a+1][b-1] and str(a+1)+" "+str(b-1) not in k:
            k.append(str(a+1)+" "+str(b-1))
            g.append(str(a+1)+" "+str(b-1))
        if "W"==l[a-1][b-1] and str(a-1)+" "+str(b-1) not in k:
            k.append(str(a-1)+" "+str(b-1))
            g.append(str(a-1)+" "+str(b-1))
        g.pop(0)
        r[a][b]=1

    if len(k)>ans:
        ans=len(k)

    q.append(ans)
for y in q:
    print(y)

```

##仍然用的以前的代码，现在看起来有点想笑疯狂if，没有什么美感

#44667037提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: **Accepted**

源代码

```

n=int(input())
q=[]
for _ in range(n):
    x,y=map(int,input().split())
    l=["0" for i in range(y+2)]
    for _ in range(x):
        e=["0"]+list(input())+"0"
        l.append(e)
    l.append(["0" for i in range(y+2)])
    r=[0 for i in range(y+2) for i in range(x+2)]
    ans=0
    for i in range(1,x+1):
        for j in range(1,y+1):
            if l[i][j]=="W":
                k={str(i)+" "+str(j)}
                g={str(i)+" "+str(j)}
            else:
                k=[]
                g=[]
            while len(g)>0:
                a,b=map(int,g[0].split())
                if l[a][b]=="." or r[a][b]!=0:
                    g=[]
                    continue
                if "W"==l[a][b+1] and str(a)+" "+str(b+1) not in k:
                    k.append(str(a)+" "+str(b+1))
                    g.append(str(a)+" "+str(b+1))
                if "W"==l[a+1][b] and str(a+1)+" "+str(b) not in k:
                    k.append(str(a+1)+" "+str(b))
                    g.append(str(a+1)+" "+str(b))
                if "W"==l[a][b-1] and str(a)+" "+str(b-1) not in k:
                    k.append(str(a)+" "+str(b-1))
                    g.append(str(a)+" "+str(b-1))
                if "W"==l[a-1][b] and str(a-1)+" "+str(b) not in k:
                    k.append(str(a-1)+" "+str(b))
                    g.append(str(a-1)+" "+str(b))
                if "W"==l[a+1][b+1] and str(a+1)+" "+str(b+1) not in k:
                    k.append(str(a+1)+" "+str(b+1))

```

基本信息

#: 44667037
 题目: 18160
 提交人: 22n2200011800
 内存: 3884kB
 时间: 205ms
 语言: Python3
 提交时间: 2024-04-15 20:16:02

sy383: 最大权值连通块

<https://sunnywhy.com/sfbj/10/3/383>

代码

```
class Node():
    def __init__(self,value,weight,visit):
        self.value=value
        self.weight=weight
        self.children=[]
        self.visit=visit

def dfs(node):
    source=[node]
    answer=0
    while source:
        subject=source.pop()
        if not subject.visit:
            subject.visit=True
            answer+=subject.weight
            source+=subject.children[::-1]
    return answer

n,m=map(int,input().split())
node_list=[Node(i,0,False) for i in range(n)]
weight_list=list(map(int,input().split()))

for i in range(n):
    node_list[i].weight=weight_list[i]

for _ in range(m):
    a,b=map(int,input().split())
    node_list[a].children+=node_list[b],
    node_list[b].children+=node_list[a],

max_mass=0
for node in node_list:
    if not node.visit:
        max_mass=max(max_mass, dfs(node))

print(max_mass)
```

##第一次定义dfs函数写，感觉会简洁很多，实现上细节处参考了同学的代码

```
1 class Node():
2     def __init__(self,value,weight,visit):
3         self.value=value
4         self.weight=weight
5         self.children=[]
6         self.visit=visit
7
8     def dfs(node):
9         source=[node]
10        answer=0
11        while source:
12            subject=source.pop()
13            if not subject.visit:
14                subject.visit=True
15                answer+=subject.weight
16                source+=subject.children[::-1]
17        return answer
```

完美通过

[查看题解](#)

100% 数据通过测试

运行时长: 0 ms

03441: 4 Values whose Sum is 0

data structure/binary search, <http://cs101.openjudge.cn/practice/03441>

代码

```
from collections import Counter
from itertools import product

A,B,C,D=[],[],[],[]

for i in range(int(input())):
    a,b,c,d=map(int,input().split())
    A.append(a)
    B.append(b)
    C.append(c)
    D.append(d)

ab_sum_counter=Counter(map(sum,product(A, B)))
cn=0
for cd_sum in map(sum,product(C,D)):
    cn+=ab_sum_counter.get(-cd_sum,0)
```

```
print(cn)
```

##超内存了好多次，最后参考了同学的代码才知道有Counter，随学习，于是ac

#44667356提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: Accepted

源代码

```
from collections import Counter
from itertools import product

A,B,C,D=[],[],[],[]

for i in range(int(input())):
    a,b,c,d=map(int,input().split())
    A.append(a)
    B.append(b)
    C.append(c)
    D.append(d)

ab_sum_counter=Counter(map(sum,product(A, B)))
cn=0
for cd_sum in map(sum,product(C,D)):
    cn+=ab_sum_counter.get(-cd_sum,0)

print(cn)
```

基本信息

#: 44667356
题目: 03441
提交人: 22n2200011800
内存: 171824kB
时间: 4049ms
语言: Python3
提交时间: 2024-04-15 20:33:43

©2002-2022 POJ 京ICP备20010980号-1

[English](#) [帮助](#) [关于](#)

04089: 电话号码

trie, <http://cs101.openjudge.cn/practice/04089/>

代码

```
class TrieNode:
    def __init__(self):
        self.child={}

class Trie:
    def __init__(self):
        self.root=TrieNode()

    def insert(self,nums):
        current=self.root
        for x in nums:
            if x not in current.child:
                current.child[x]=TrieNode()
            current=current.child[x]

    def search(self,num):
        current=self.root
        for x in num:
            if x not in current.child:
                return 0
            current=current.child[x]
        return 1

for _ in range(int(input())):
    nums=[]
    for _ in range(int(input())):
```

```

        nums.append(str(input()))
    nums.sort(reverse=True)
    s=0
    trie=Trie()
    for num in nums:
        s+=trie.search(num)
        trie.insert(num)
    if s>0:
        print('NO')
    else:
        print('YES')

```

##一开始写的时候没有什么思路，后面看到群里讨论了trie以及看了一下题解里用到了这个，随学习

#44667436提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: Accepted

源代码

```

class TrieNode:
    def __init__(self):
        self.child={}

class Trie:
    def __init__(self):
        self.root=TrieNode()

    def insert(self,nums):
        current=self.root
        for x in nums:
            if x not in current.child:
                current.child[x]=TrieNode()
            current=current.child[x]

    def search(self,num):
        current=self.root
        for x in num:
            if x not in current.child:
                return 0
            current=current.child[x]
        return 1

for _ in range(int(input())):
    nums=[]
    for _ in range(int(input())):
        nums.append(str(input()))
    nums.sort(reverse=True)
    s=0
    trie=Trie()
    for num in nums:
        s+=trie.search(num)
        trie.insert(num)
    if s>0:
        print('NO')
    else:
        print('YES')

```

基本信息

#: 44667436
 题目: 04089
 提交人: 22n2200011800
 内存: 24648kB
 时间: 390ms
 语言: Python3
 提交时间: 2024-04-15 20:38:29

©2002-2022 POJ 京ICP备20010980号-1

[English](#) [帮助](#) [关于](#)

04082: 树的镜面映射

<http://cs101.openjudge.cn/practice/04082/>

代码

```

from collections import deque

class TreeNode:
    def __init__(self, x):
        self.x=x
        self.children=[]

def create_node():

```

```

return TreeNode('')

def build_tree(tempList,index):
    node=create_node()
    node.x=tempList[index][0]
    if tempList[index][1]=='0' and node.x!='$':
        index+=1
        child,index=build_tree(tempList,index)
        node.children.append(child)
        index+=1
        child,index=build_tree(tempList,index)
        node.children.append(child)
    return node,index

def print_tree(p):
    Q=deque()
    s=deque()
    while p is not None:
        if p.x!='$':
            s.append(p)
            p=p.children[1] if len(p.children)>1 else None
    while s:
        Q.append(s.pop())
    while Q:
        p = Q.popleft()
        print(p.x,end=' ')
        if p.children:
            p = p.children[0]
            while p is not None:
                if p.x!='$':
                    s.append(p)
                p = p.children[1] if len(p.children)>1 else None
            while s:
                Q.append(s.pop())

n=int(input())
tempList=input().split(' ')
root, _=build_tree(tempList,0)
print_tree(root)

```

##难难难，时间用得很久了就直接看了题解，看了挺久的看懂了，但感觉让自己从头写还是有点费劲

状态: Accepted

源代码

```
from collections import deque

class TreeNode:
    def __init__(self, x):
        self.x=x
        self.children=[]

def create_node():
    return TreeNode('')

def build_tree(tempList,index):
    node=create_node()
    node.x=tempList[index][0]
    if tempList[index][1]!='0' and node.x!='$':
        index+=1
        child,index=build_tree(tempList,index)
        node.children.append(child)
        index+=1
        child,index=build_tree(tempList,index)
        node.children.append(child)
    return node,index

def print_tree(p):
    Q=deque()
    s=deque()
    while p is not None:
        if p.x!='$':
            s.append(p)
            p=p.children[1] if len(p.children)>1 else None
        while s:
            Q.append(s.pop())
        while Q:
            p = Q.popleft()
            print(p.x,end=' ')
            if p.children:
                p = p.children[0]
                while p is not None:
                    if p.x!='$':
                        s.append(p)
```

基本信息

#: 44667539
题目: 04082
提交人: 22n2200011800
内存: 3728kB
时间: 29ms
语言: Python3
提交时间: 2024-04-15 20:44:56

2. 学习总结和收获

忙于期中考试这周没怎么额外学习，连作业里好几题碰壁之后没想很久就去学习参考题解和同学的代码了。不过这次感触最深的一点是虽然第一遍做不出来，但是通过学习补充知识还是能把问题解决的，比如用到的Counter和Trie，闭门造车我自己一定是很难独立想出来的，还是要多多向大佬还有各种资料学习。