Assignment #8: 图论：概念、遍历，及 树算

Updated 1150 GMT+8 Apr 8, 2024

2024 spring, Complied by 苏王捷 工学院

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：Windows11

Python编程环境：Spyder IDE 5.5.3

1. 题目

19943: 图的拉普拉斯矩阵

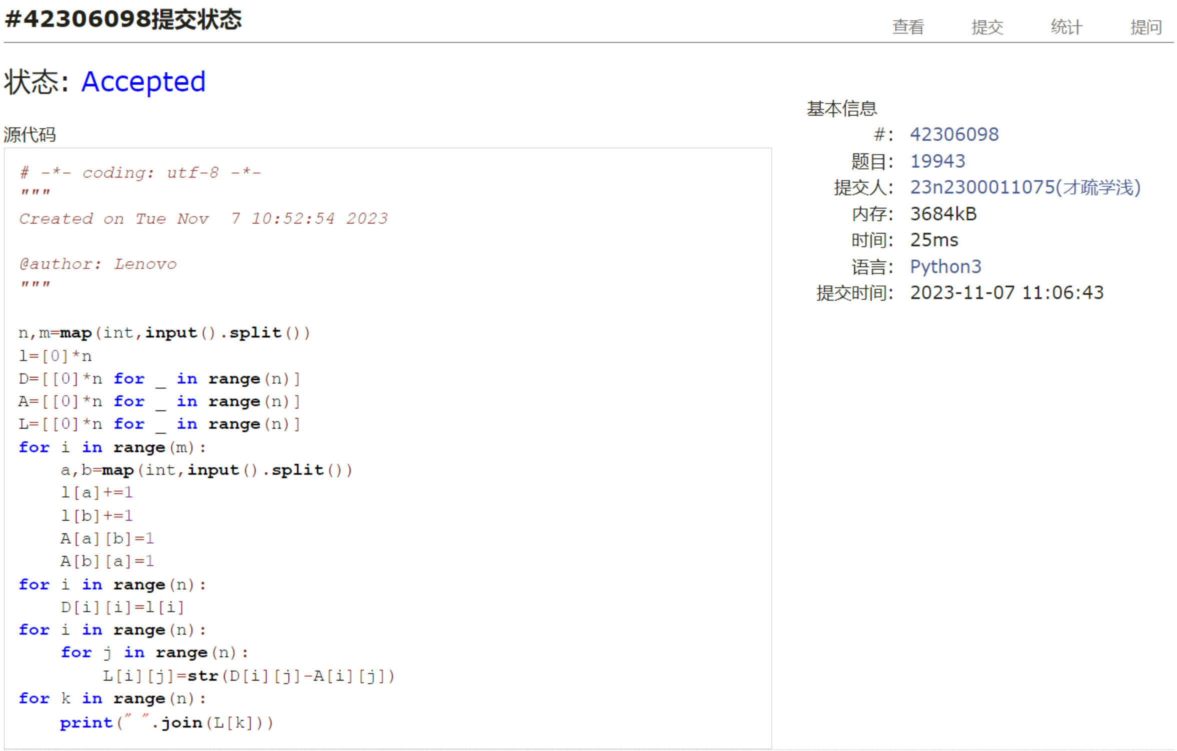
matrices, http://cs101.openjudge.cn/practice/19943/

思路：记录边和节点度数，给出D和A，求L

代码

```
# # -*- coding: utf-8 -*-
"""
Created on Tue Nov  7 10:52:54 2023

@author: Lenovo
"""


n,m=map(int,input().split())
l=[0]*n
D=[[0]*n for _ in range(n)]
A=[[0]*n for _ in range(n)]
L=[[0]*n for _ in range(n)]
for i in range(m):
    a,b=map(int,input().split())
    l[a]+=1
    l[b]+=1
    A[a][b]=1
    A[b][a]=1
for i in range(n):
    D[i][i]=l[i]
for i in range(n):
    for j in range(n):
        L[i][j]=str(D[i][j]-A[i][j])
for k in range(n):
    print(" ".join(L[k]))
```

代码运行截图 （至少包含有"Accepted"）



#42306098提交状态

状态: Accepted

源代码

```
# -*- coding: utf-8 -*-
"""
Created on Tue Nov  7 10:52:54 2023

@author: Lenovo
"""

n,m=map(int,input().split())
l=[0]*n
D=[[0]*n for _ in range(n)]
A=[[0]*n for _ in range(n)]
L=[[0]*n for _ in range(n)]
for i in range(m):
    a,b=map(int,input().split())
    l[a]+=1
    l[b]+=1
    A[a][b]=1
    A[b][a]=1
for i in range(n):
    D[i][i]=l[i]
for i in range(n):
    for j in range(n):
        L[i][j]=str(D[i][j]-A[i][j])
for k in range(n):
    print(" ".join(L[k]))
```

基本信息

#:	42306098
题目:	19943
提交人:	23n2300011075(才疏学浅)
内存:	3684kB
时间:	25ms
语言:	Python3
提交时间:	2023-11-07 11:06:43

18160: 最大连通域面积

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

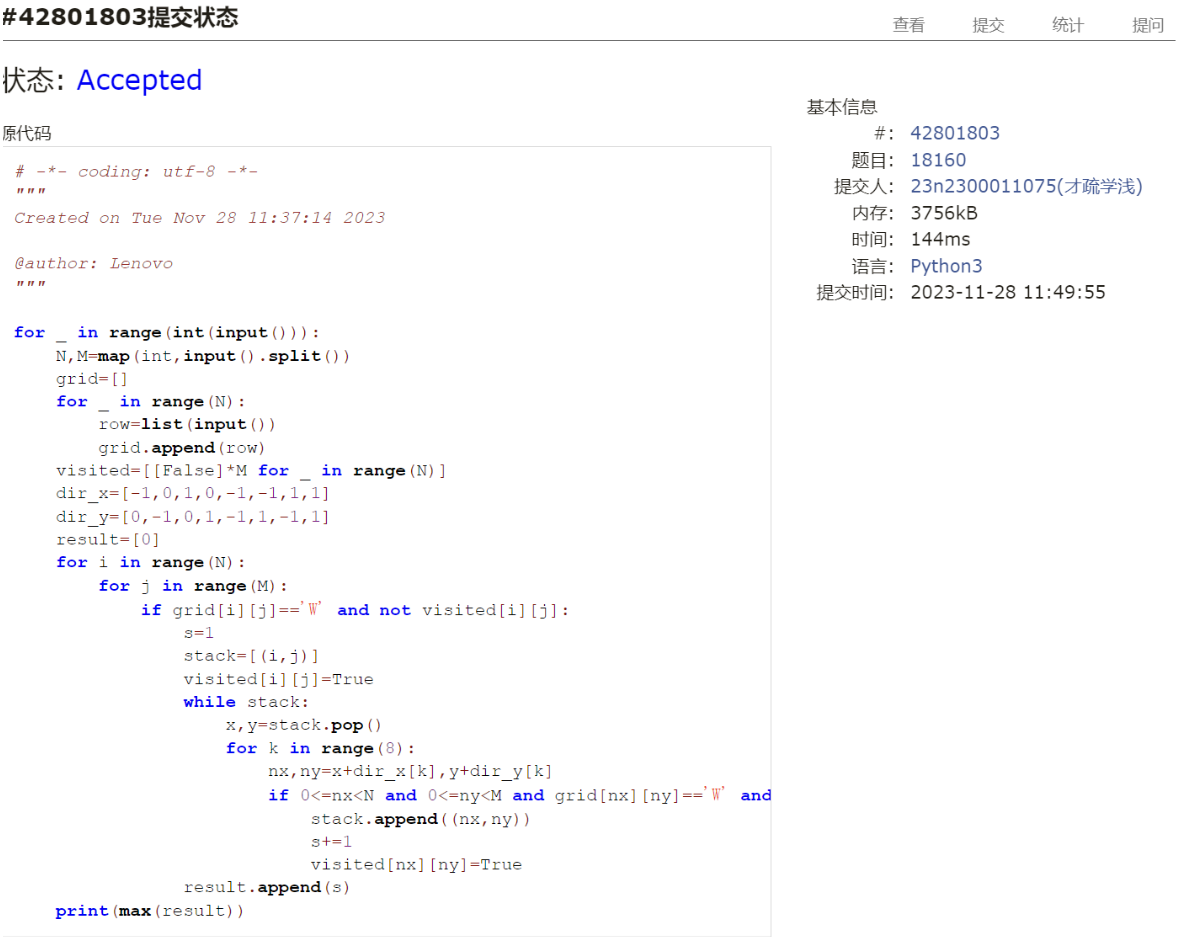
思路：bfs找到所有连通域记录面积，max找最大值

代码

```
# # -*- coding: utf-8 -*-
"""
Created on Tue Nov 28 11:37:14 2023

@author: Lenovo
```

```
"""

for _ in range(int(input())):
    N,M=map(int,input().split())
    grid=[]
    for _ in range(N):
        row=list(input())
        grid.append(row)
    visited=[[False]*M for _ in range(N)]
    dir_x=[-1,0,1,0,-1,-1,1,1]
    dir_y=[0,-1,0,1,-1,1,-1,1]
    result=[0]
    for i in range(N):
        for j in range(M):
            if grid[i][j]=='W' and not visited[i][j]:
                s=1
                stack=[(i,j)]
                visited[i][j]=True
                while stack:
                    x,y=stack.pop()
                    for k in range(8):
                        nx,ny=x+dir_x[k],y+dir_y[k]
                        if 0<=nx<N and 0<=ny<M and grid[nx][ny]=='W' and not visited[nx][ny]:
                            stack.append((nx,ny))
                            s+=1
                            visited[nx][ny]=True
                result.append(s)
    print(max(result))
```

代码运行截图 （至少包含有"Accepted"）



#42801803提交状态

状态: Accepted

原代码

```python
# -*- coding: utf-8 -*-
"""
Created on Tue Nov 28 11:37:14 2023

@author: Lenovo
"""

for _ in range(int(input())):
    N,M=map(int,input().split())
    grid=[]
    for _ in range(N):
        row=list(input())
        grid.append(row)
    visited=[[False]*M for _ in range(N)]
    dir_x=[-1,0,1,0,-1,-1,1,1]
    dir_y=[0,-1,0,1,-1,1,-1,1]
    result=[0]
    for i in range(N):
        for j in range(M):
            if grid[i][j]=='W' and not visited[i][j]:
                s=1
                stack=[(i,j)]
                visited[i][j]=True
                while stack:
                    x,y=stack.pop()
                    for k in range(8):
                        nx,ny=x+dir_x[k],y+dir_y[k]
                        if 0<=nx<N and 0<=ny<M and grid[nx][ny]=='W' and
                            stack.append((nx,ny))
                            s+=1
                            visited[nx][ny]=True
                result.append(s)
    print(max(result))
```

基本信息

#:	42801803
题目:	18160
提交人:	23n2300011075(才疏学浅)
内存:	3756kB
时间:	144ms
语言:	Python3
提交时间:	2023-11-28 11:49:55

sy383: 最大权值连通块

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

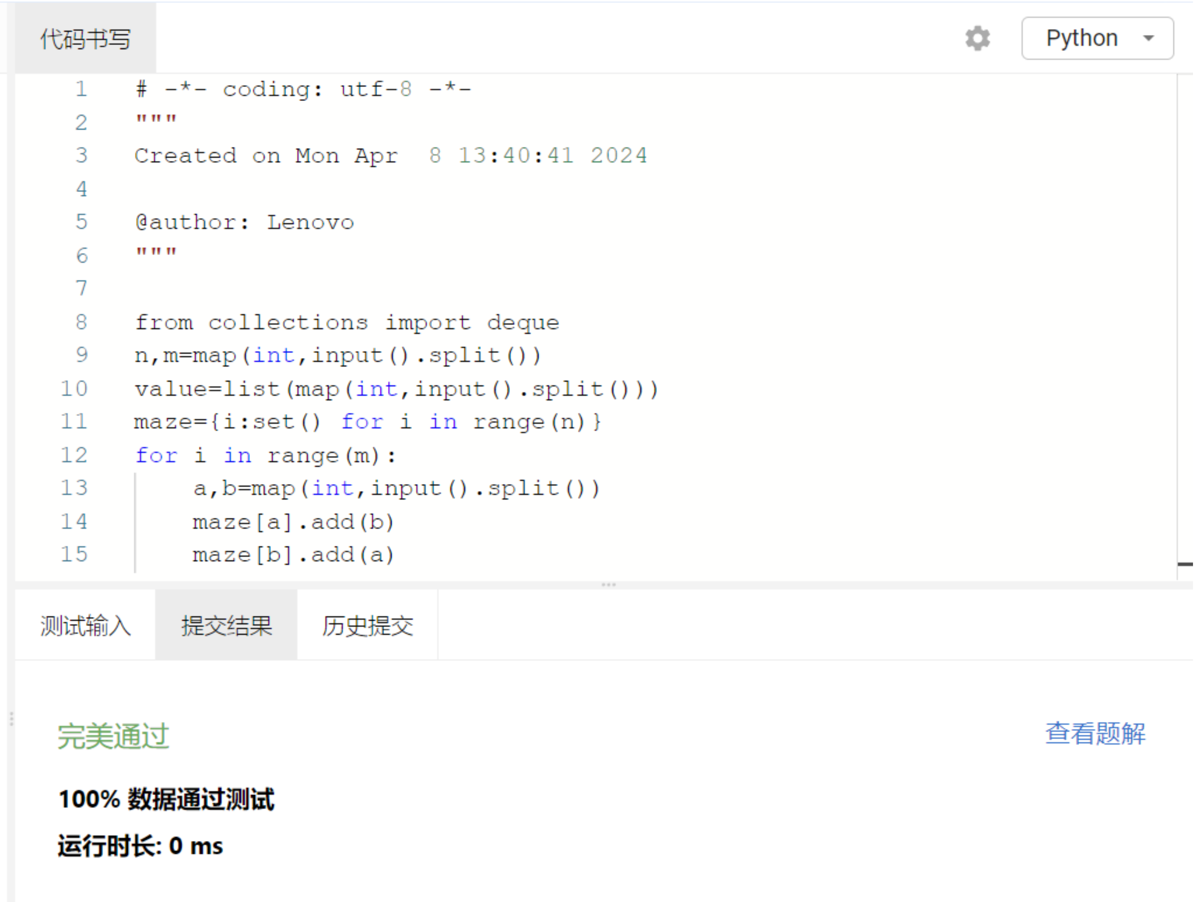
思路：一样的bfs

代码

```python
# # -*- coding: utf-8 -*-
"""
Created on Mon Apr  8 13:40:41 2024

@author: Lenovo
"""

from collections import deque
n,m=map(int,input().split())
value=list(map(int,input().split()))
maze={i:set() for i in range(n)}
for i in range(m):
    a,b=map(int,input().split())
    maze[a].add(b)
    maze[b].add(a)
vis=[0]*n
res=[]
```

```
for i in range(n):
    if vis[i]:
        continue
    q=deque([i])
    ans=value[i]
    vis[i]=1
    while q:
        now=q.popleft()
        for Next in maze[now]:
            if not vis[Next]:
                vis[Next]=1
                q.append(Next)
                ans+=value[Next]
    res.append(ans)
print(max(res))
```

代码运行截图 （AC代码截图，至少包含有"Accepted"）



```
# -*- coding: utf-8 -*-
"""
Created on Mon Apr  8 13:40:41 2024

@author: Lenovo
"""

from collections import deque
n,m=map(int,input().split())
value=list(map(int,input().split()))
maze={i:set() for i in range(n)}
for i in range(m):
    a,b=map(int,input().split())
    maze[a].add(b)
    maze[b].add(a)
```

完美通过

100% 数据通过测试

运行时长: 0 ms

03441: 4 Values whose Sum is 0

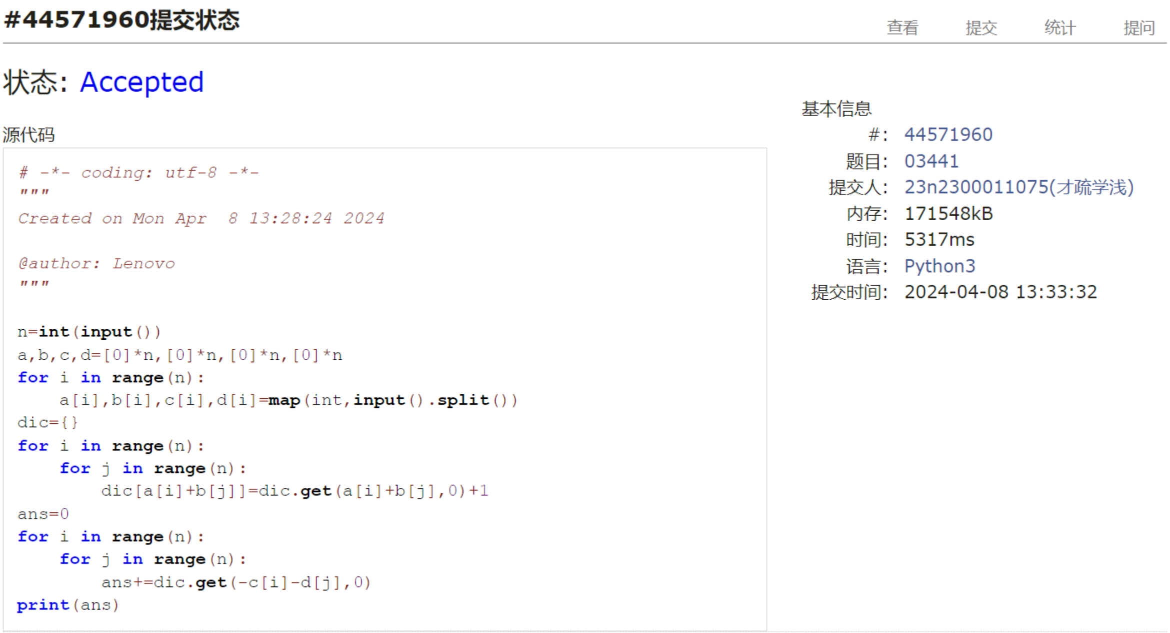
data structure/binary search, http://cs101.openjudge.cn/practice/03441

思路：两两分组遍历，只要找到相反数，就能找出组数

代码

```python
# # -*- coding: utf-8 -*-
"""
Created on Mon Apr  8 13:28:24 2024

@author: Lenovo
"""

n=int(input())
a,b,c,d=[0]*n,[0]*n,[0]*n,[0]*n
for i in range(n):
    a[i],b[i],c[i],d[i]=map(int,input().split())
dic={}
for i in range(n):
    for j in range(n):
        dic[a[i]+b[j]]=dic.get(a[i]+b[j],0)+1
ans=0
for i in range(n):
    for j in range(n):
        ans+=dic.get(-c[i]-d[j],0)
print(ans)
```

代码运行截图 （AC代码截图，至少包含有"Accepted"）



#44571960提交状态

状态: Accepted

源代码

```python
# -*- coding: utf-8 -*-
"""
Created on Mon Apr  8 13:28:24 2024

@author: Lenovo
"""

n=int(input())
a,b,c,d=[0]*n,[0]*n,[0]*n,[0]*n
for i in range(n):
    a[i],b[i],c[i],d[i]=map(int,input().split())
dic={}
for i in range(n):
    for j in range(n):
        dic[a[i]+b[j]]=dic.get(a[i]+b[j],0)+1
ans=0
for i in range(n):
    for j in range(n):
        ans+=dic.get(-c[i]-d[j],0)
print(ans)
```

基本信息

#:	44571960
题目:	03441
提交人:	23n2300011075(才疏学浅)
内存:	171548kB
时间:	5317ms
语言:	Python3
提交时间:	2024-04-08 13:33:32

04089: 电话号码

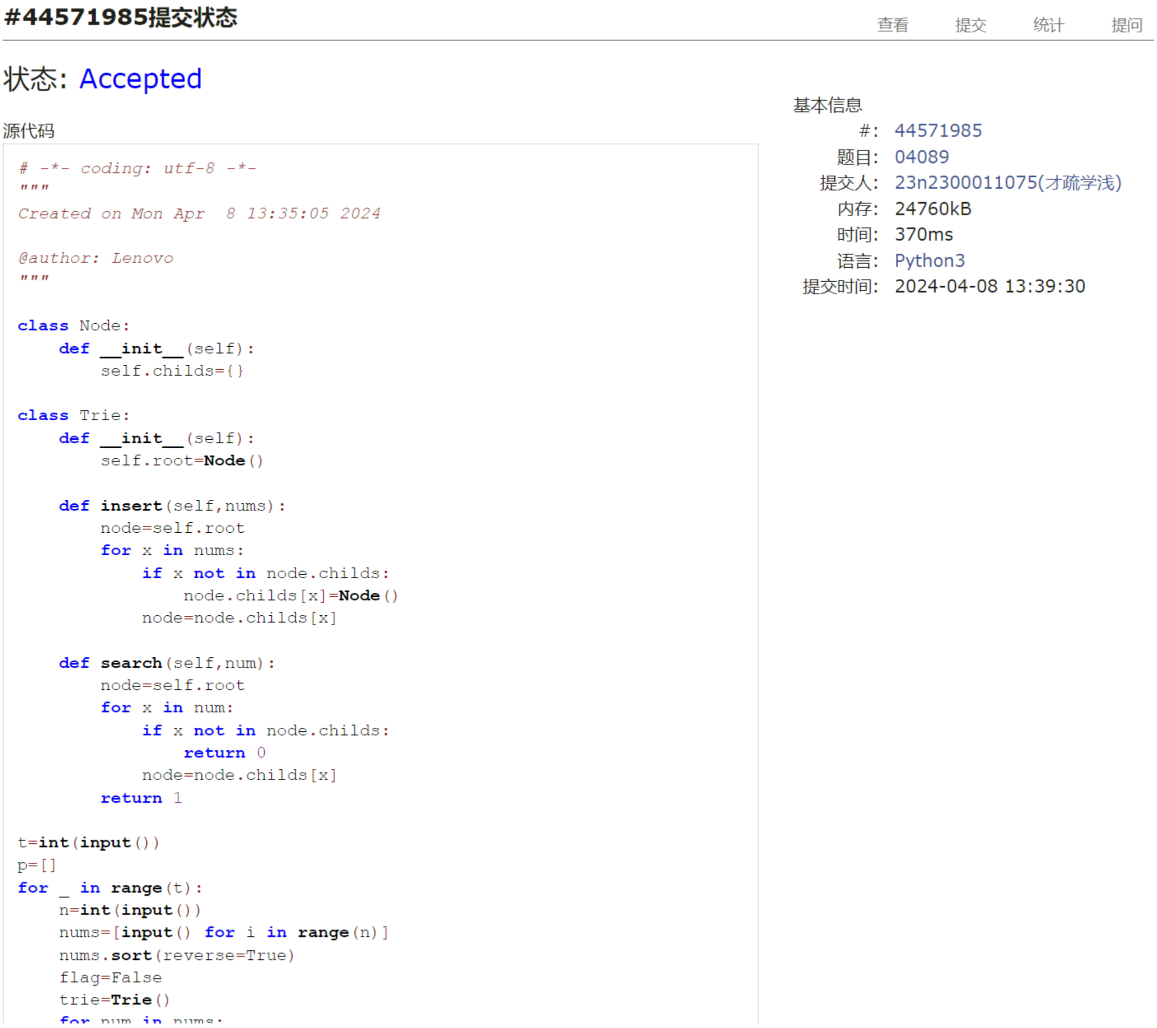
trie, http://cs101.openjudge.cn/practice/04089/

思路：建立字典树，找到是否有重合路径

代码

```
# # -*- coding: utf-8 -*-
"""
Created on Mon Apr  8 13:35:05 2024

@author: Lenovo
"""

class Node:
    def __init__(self):
        self.childs={}

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

    def insert(self,nums):
        node=self.root
        for x in nums:
            if x not in node.childs:
                node.childs[x]=Node()
            node=node.childs[x]

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

t=int(input())
p=[]
for _ in range(t):
    n=int(input())
    nums=[input() for i in range(n)]
    nums.sort(reverse=True)
    flag=False
    trie=Trie()
    for num in nums:
        flag=(flag or trie.search(num))
        trie.insert(num)
    print("NO" if flag else "YES")
```

代码运行截图 （AC代码截图，至少包含有"Accepted"）



#44571985提交状态

状态: Accepted

源代码

```
# -*- coding: utf-8 -*-
"""
Created on Mon Apr  8 13:35:05 2024

@author: Lenovo
"""

class Node:
    def __init__(self):
        self.childs={}

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

    def insert(self,nums):
        node=self.root
        for x in nums:
            if x not in node.childs:
                node.childs[x]=Node()
            node=node.childs[x]

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

t=int(input())
p=[]
for _ in range(t):
    n=int(input())
    nums=[input() for i in range(n)]
    nums.sort(reverse=True)
    flag=False
    trie=Trie()
    for num in nums:
```

基本信息

#:	44571985
题目:	04089
提交人:	23n2300011075(才疏学浅)
内存:	24760kB
时间:	370ms
语言:	Python3
提交时间:	2024-04-08 13:39:30

04082: 树的镜面映射

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

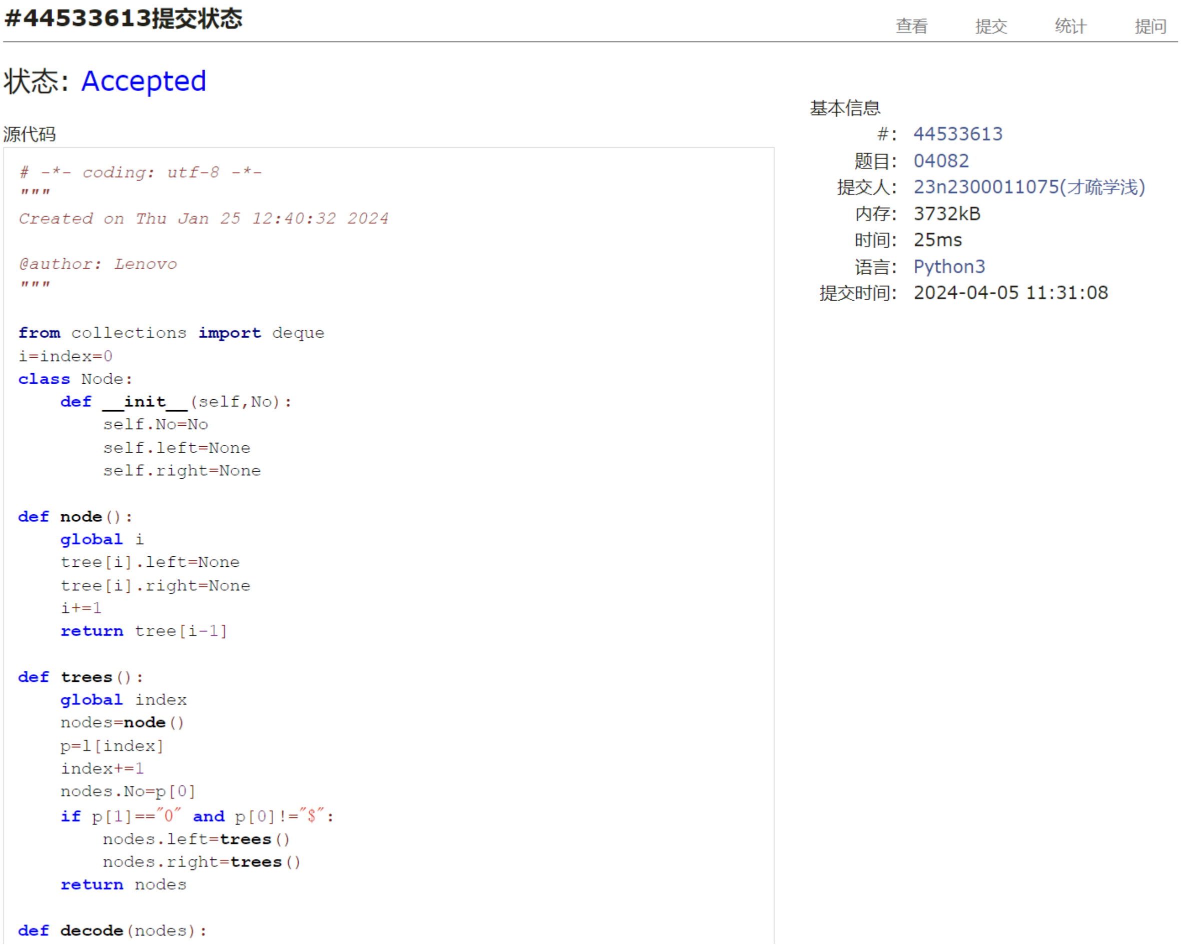
思路：按前序建树，层序遍历输出

代码

```
# # -*- coding: utf-8 -*-
"""
Created on Thu Jan 25 12:40:32 2024

@author: Lenovo
"""

from collections import deque
i=index=0
class Node:
    def __init__(self,No):
        self.No=No
        self.left=None
        self.right=None
```

```python
def node():
    global i
    tree[i].left=None
    tree[i].right=None
    i+=1
    return tree[i-1]

def trees():
    global index
    nodes=node()
    p=l[index]
    index+=1
    nodes.No=p[0]
    if p[1]=="0" and p[0]!="$":
        nodes.left=trees()
        nodes.right=trees()
    return nodes

def decode(nodes):
    s,q=deque(),deque()
    while nodes is not None:
        if nodes.No!="$":
            s.append(nodes)
        nodes=nodes.right
    while s:
        q.append(s.pop())
    while q:
        nodes=q.popleft()
        print(nodes.No,end=" ")
        if nodes.left is not None:
            nodes=nodes.left
            while nodes is not None:
                if nodes.No!="$":
                    s.append(nodes)
                nodes=nodes.right
            while s:
                q.append(s.pop())

n=int(input())
l=list(input().split())
tree=[Node("")for i in range(n)]
i=index=0
Tree=trees()
decode(Tree)
```

代码运行截图 （AC代码截图，至少包含有"Accepted"）



#44533613提交状态

状态: Accepted

源代码

```
# -*- coding: utf-8 -*-
"""
Created on Thu Jan 25 12:40:32 2024

@author: Lenovo
"""

from collections import deque
i=index=0
class Node:
    def __init__(self,No):
        self.No=No
        self.left=None
        self.right=None

def node():
    global i
    tree[i].left=None
    tree[i].right=None
    i+=1
    return tree[i-1]

def trees():
    global index
    nodes=node()
    p=l[index]
    index+=1
    nodes.No=p[0]
    if p[1]=="0" and p[0]!="$":
        nodes.left=trees()
        nodes.right=trees()
    return nodes

def decode(nodes):
```

基本信息

#: 44533613
题目: 04082
提交人: 23n2300011075(才疏学浅)
内存: 3732kB
时间: 25ms
语言: Python3
提交时间: 2024-04-05 11:31:08

2. 学习总结和收获

如果作业题目简单，有否额外练习题目，比如：OJ“2024spring每日选做”、CF、LeetCode、洛谷等网站题目。

图来，图搜索从四面八方来！