Assignment #8: 图论：概念、遍历，及 树算

Updated 1150 GMT+8 Apr 8, 2024

2024 spring, Complied by ==黄源森，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：W11

Python编程环境：Spyder IDE 5.2.2,

1. 题目

19943: 图的拉普拉斯矩阵

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

思路：

代码

```
#
n,m=map(int,input().split())
l=[[0]*n for _ in range(n)]
p=[[0]*n for _ in range(n)]
for _ in range(m):
    a,b=map(int,input().split())
    l[a][a]+=1
    l[b][b]+=1
    p[a][b]+=1
    p[b][a]+=1
for i in range(n):
    for j in range(n):
        print(l[i][j]-p[i][j],end=' ')
```

```
    print()
```

代码运行截图 ==（至少包含有"Accepted"）==

状态: Accepted

源代码

```
n,m=map(int,input().split())
l=[[0]*n for _ in range(n)]
p=[[0]*n for _ in range(n)]
for _ in range(m):
    a,b=map(int,input().split())
    l[a][a]+=1
    l[b][b]+=1
    p[a][b]+=1
    p[b][a]+=1
for i in range(n):
    for j in range(n):
        print(l[i][j]-p[i][j],end=' ')
    print()
```

基本信息

#:	42305360
题目:	19943
提交人:	23n2300011031
内存:	3652kB
时间:	27ms
语言:	Python3
提交时间:	2023-11-07 10:42:14

18160: 最大连通域面积

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

思路:

代码

```
#
import sys
sys.setrecursionlimit(2000000)
c=0
def dfs(i,j):
    global c
    if flag[i][j]==1:
        return
    flag[i][j]=1
    c+=1
    for t in range(i-1,i+2):
        for u in range(j-1,j+2):
            if 0<=t<=n-1 and 0<=u<=m-1 and (not flag[t][u]) and l[t][u]=='W':
                dfs(t,u)


for _ in range(int(input())):
    n,m=map(int,input().split())
    l=[]
    flag=[]
    for __ in range(n):
        l.append(list(input()))
```

```
        flag.append([0]*m)
    ans=[]
    for p in range(n):
        for q in range(m):
            if flag[p][q]==0 and l[p][q]=='W':
                c=0
                dfs(p,q)
                ans.append(c)
    if ans:    print(max(ans))
    else:print(0)
```

代码运行截图 ==（至少包含有"Accepted"）==

状态: Accepted

源代码

```
import sys
sys.setrecursionlimit(2000000)
c=0
def dfs(i,j):
    global c
    if flag[i][j]==1:
        return
    flag[i][j]=1
    c+=1
    for t in range(i-1,i+2):
        for u in range(j-1,j+2):
            if 0<=t<=n-1 and 0<=u<=m-1 and (not flag[t][u]) and l[t][u]=
                dfs(t,u)


for _ in range(int(input())):
    n,m=map(int,input().split())
    l=[]
    flag=[]
    for __ in range(n):
        l.append(list(input()))
        flag.append([0]*m)
    ans=[]
    for p in range(n):
        for q in range(m):
            if flag[p][q]==0 and l[p][q]=='W':
                c=0
                dfs(p,q)
                ans.append(c)
    if ans:    print(max(ans))
    else:print(0)
```

基本信息

#:	42801201
题目:	18160
提交人:	23n2300011031
内存:	3696kB
时间:	132ms
语言:	Python3
提交时间:	2023-11-28 11:15:23

sy383: 最大权值连通块

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

思路:

代码

```
#
from collections import defaultdict
def dfs(k):
    global c
```

```
    c+=l[k]
    for u in dic[k]:
        if u not in vis:
            dfs(u)
n,m=map(int,input().split())
l=list(map(int,input().split()))
vis=set()
dic=defaultdict(list)
for _ in range(m):
    a,b=map(int,input().split())
    dic[a].append(b)
ans=0
for i in range(n):
    if i not in vis:
        c=0
        dfs(i)
        ans=max(c,ans)
print(ans)
```

代码运行截图 ==（AC代码截图，至少包含有"Accepted"）==

03441: 4 Values whose Sum is 0

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

思路:

代码

```
#include <iostream>
#include <stdio.h>
#include <vector>
#include <cstring>
#include <algorithm>
using namespace std;
int main()
{
    int n,i;
    scanf("%d",&n);
    int a[4005],b[4005],c[4005],d[4005],sum=0;
    for (i=0;i<n;i++)
        cin>>a[i]>>b[i]>>c[i]>>d[i];
    int j,t=0;
    vector <int> sum_ab;
    for (i=0;i<n;i++)
        for (j=0;j<n;j++)
        {
            sum_ab.push_back( a[i] + b[j]);
            t++;
        }
    stable_sort(sum_ab.begin(),sum_ab.end());
    int lp,rp,mid;
    for (i=0;i<n;i++)
    {
        for (j=0;j<n;j++)
        {
            int sum_cd=c[i]+d[j];
            lp=0;
            rp=t-1;
            while (lp<rp)
            {
                mid=(lp+rp)/2;
                if (sum_ab[mid]+sum_cd==0)
                {
                    sum++;
                    for (int q=mid-1;q>=0;q--)
                    {
                        if (sum_ab[q]+sum_cd==0)
                            sum++;
                        else
                            break;
```

```
                    }
                    for (int q=mid+1;q<t;q++)
                    {
                        if (sum_ab[q]+sum_cd==0)
                            sum++;
                        else
                            break;
                    }
                    lp=mid+1;
                    rp=mid;
                }
                else if (sum_ab[mid]+sum_cd>0)
                    rp=mid;
                else if (sum_ab[mid]+sum_cd<0)
                    lp=mid+1;
            }
        }
    }
        printf("%d\n",sum);
    return 0;
}
```

代码运行截图 ==（AC代码截图，至少包含有"Accepted"）==

状态: Accepted

源代码

```
#include <iostream>
#include <stdio.h>
#include <vector>
#include <cstring>
#include <algorithm>
using namespace std;
int main()
{
    int n,i;
    scanf("%d",&n);
    int a[4005],b[4005],c[4005],d[4005],sum=0;
    for (i=0;i<n;i++)
        cin>>a[i]>>b[i]>>c[i]>>d[i];
    int j,t=0;
    vector <int> sum_ab;
    for (i=0;i<n;i++)
        for (j=0;j<n;j++)
        {
            sum_ab.push_back( a[i] + b[j]);
            t++;
        }
    stable_sort(sum_ab.begin(),sum_ab.end());
    int lp,rp,mid;
    for (i=0;i<n;i++)
    {
        for (j=0;j<n;j++)
        {
            int sum_cd=c[i]+d[j];
            lp=0;
```

基本信息

#: 44576017
题目: 03441
提交人: 23n2300011031
内存: 37056kB
时间: 1248ms
语言: G++
提交时间: 2024-04-08 19:16:32

04089: 电话号码

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

思路:

代码

```
#
class Node:
    def __init__(self):
        self.children=dict()
        self.is_end_of_word=0
class Trie:
    def __init__(self):
        self.root=Node()
    def insert(self,word):
        c=0
        cur=self.root
        for char in word:
            if char not in cur.children:
                cur.children[char]=Node()
            else:
                if cur.children[char].is_end_of_word:
                    return 0
                c+=1
            cur=cur.children[char]

        cur.is_end_of_word=1
        if c==len(word):
            return 0
        return 1
def f():
    f=1
    for i in range(n):
        x=input()
        if not s.insert(x):
            f=0
    if f==1:
        print('YES')
    else:
        print('NO')
    return

for _ in range(int(input())):
    n=int(input())
    s=Trie()
    f()
```

代码运行截图 ==（AC代码截图，至少包含有"Accepted"）==

状态: Accepted

源代码

```
class Node:
    def __init__(self):
        self.children=dict()
        self.is_end_of_word=0
class Trie:
    def __init__(self):
        self.root=Node()
    def insert(self,word):
        c=0
        cur=self.root
        for char in word:
            if char not in cur.children:
                cur.children[char]=Node()
            else:
                if cur.children[char].is_end_of_word:
                    return 0
                c+=1
            cur=cur.children[char]

        cur.is_end_of_word=1
        if c==len(word):
            return 0
        return 1
def f():
    f=1
    for i in range(n):
        x=input()
        if not s.insert(x):
            f=0
    if f==1:
        print('YES')
    else:
        print('NO')
    return

for _ in range(int(input())):
    n=int(input())
    s=Trie()
    f()
```

基本信息

#:	44575509
题目:	04089
提交人:	23n2300011031
内存:	25664kB
时间:	417ms
语言:	Python3
提交时间:	2024-04-08 18:30:12

04082: 树的镜面映射

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

思路:

代码

```
#
n=int(input())
l=list(input().split())
al=[[] for i in range(50)]
h=0
for elem in l:
    if elem[0]!='$':
        al[h].append(elem[0])
    if elem[1]=='0':
        h+=1
    else:
        h-=1
ans=[]
for u in al:
    ans.extend(list(reversed(u)))
```

```
print(*ans)
```

代码运行截图 ==（AC代码截图，至少包含有"Accepted"）==

状态：Accepted

源代码

```
n=int(input())
l=list(input().split())
al=[[] for i in range(50)]
h=0
for elem in l:
    if elem[0]!='$':
        al[h].append(elem[0])
    if elem[1]=='0':
        h+=1
    else:
        h-=1
ans=[]
for u in al:
    ans.extend(list(reversed(u)))
print(*ans)
```

基本信息

#:	44536687
题目:	04082
提交人:	23n2300011031
内存:	3648kB
时间:	25ms
语言:	Python3
提交时间:	2024-04-05 16:29:50

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。==

挺像计概的，很多题做过，和为0那题不知道为什么用c++的multiset之类过不了一定要用二分，感觉不都是n**2复杂度吗