Assignment #9: 密集期中考试周

Updated 1918 GMT+8 Nov 6, 2023

2023 fall, 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, 学号），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、作业评论有md或者doc。

3）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：Windows 11 | 22H2

Python编程环境：Spyder IDE 5.4.3 | Python 3.11.4 64-bit

1. 必做题目

OJ19943：图的拉普拉斯矩阵

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

思路：简单分开处理D和A矩阵

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

#42306098提交状态

查看 提交 统计 提问

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

OJ19942：二维矩阵上的卷积运算v0.2

matrix, http://cs101.openjudge.cn/practice/19942/

思路：逐个计算各位上的值

代码

```
# -*- coding: utf-8 -*-
"""
Created on Tue Nov  7 11:07:02 2023

@author: Lenovo
"""
```

```
m,n,p,q=map(int,input().split())
l,x=[],[]
for i in range(m):
    l.append([int(_) for _ in input().split()])
for i in range(p):
    x.append([int(_) for _ in input().split()])
ans=[[0]*(n+1-q) for _ in range(m+1-p)]
for i in range(m+1-p):
    for j in range(n+1-q):
        for a in range(p):
            for b in range(q):
                ans[i][j]+=x[a][b]*l[i+a][j+b]
        ans[i][j]=str(ans[i][j])
for _ in range(m+1-p):
    print(" ".join(ans[_]))
```

代码运行截图

#42306303提交状态

查看 提交 统计 提问

状态: Accepted

源代码

```
# -*- coding: utf-8 -*-
"""
Created on Tue Nov  7 11:07:02 2023

@author: Lenovo
"""

m,n,p,q=map(int,input().split())
l,x=[],[]
for i in range(m):
    l.append([int(_) for _ in input().split()])
for i in range(p):
    x.append([int(_) for _ in input().split()])
ans=[[0]*(n+1-q) for _ in range(m+1-p)]
for i in range(m+1-p):
    for j in range(n+1-q):
        for a in range(p):
            for b in range(q):
                ans[i][j]+=x[a][b]*l[i+a][j+b]
        ans[i][j]=str(ans[i][j])
for _ in range(m+1-p):
    print(" ".join(ans[_]))
```

基本信息

#: 42306303
题目: 19942
提交人: 23n2300011075（才疏学浅）
内存: 3680kB
时间: 26ms
语言: Python3
提交时间: 2023-11-07 11:16:27

CF313B: Ilya and Queries

dp/implementation, 1100, https://codeforces.com/contest/313/problem/B

思路对从头开始计算值，答案为两项之差

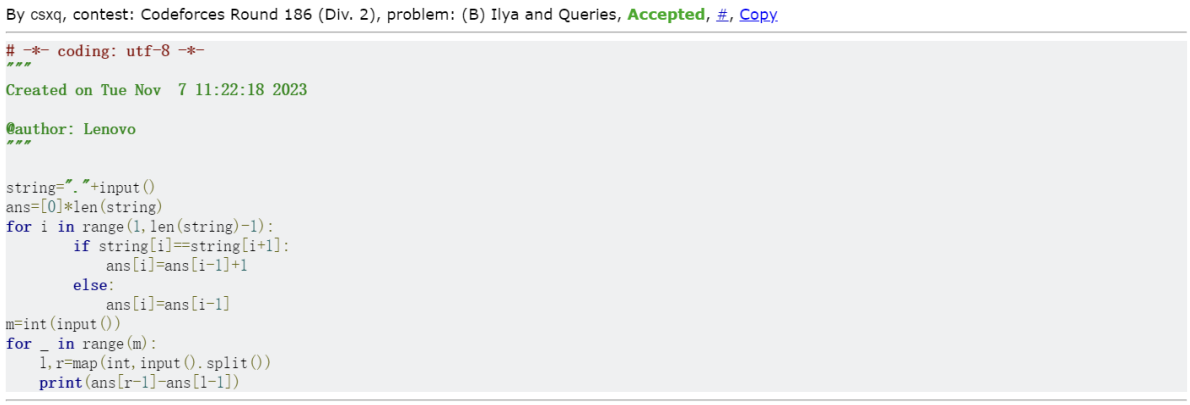
代码

```
# -*- coding: utf-8 -*-
"""
Created on Tue Nov  7 11:22:18 2023
```

```
@author: Lenovo
"""

string="."+input()
ans=[0]*len(string)
for i in range(1,len(string)-1):
        if string[i]==string[i+1]:
            ans[i]=ans[i-1]+1
        else:
            ans[i]=ans[i-1]
m=int(input())
for _ in range(m):
    l,r=map(int,input().split())
    print(ans[r-1]-ans[l-1])
```

代码运行截图



CF706B: Interesting drink

binary search/dp/implementation, 1100, https://codeforces.com/problemset/problem/706/B

思路：dp，对每种钱可以去的酒馆数记录

代码

```
# -*- coding: utf-8 -*-
"""
Created on Tue Nov  7 12:02:25 2023

@author: Lenovo
"""

n=int(input())
l=[int(_) for _ in input().split()]
l.sort()
ans=[0]*100001
cnt=0
```

```
i=1
while i<=100000:
    if i==l[cnt]:
        ans[i]=max(ans[i-1],ans[i])+1
        cnt+=1
    else:
        ans[i]=max(ans[i-1],ans[i])
        i+=1
    if cnt>=n:
        break
for i in range(l[cnt-1]+1,100001):
    ans[i]=ans[i-1]
q=int(input())
for i in range(q):
    m=int(input())
    if m<=l[-1]:
        print(ans[m])
    else:
        print(ans[l[-1]])
```

代码运行截图

By csxq, contest: Codeforces Round 367 (Div. 2), problem: (B) Interesting drink, **Accepted**, #, Copy

```
# -*- coding: utf-8 -*-
"""
Created on Tue Nov  7 12:02:25 2023

@author: Lenovo
"""

n=int(input())
l=[int(_) for _ in input().split()]
l.sort()
ans=[0]*100001
cnt=0
i=1
while i<=100000:
    if i==l[cnt]:
        ans[i]=max(ans[i-1],ans[i])+1
        cnt+=1
    else:
        ans[i]=max(ans[i-1],ans[i])
        i+=1
    if cnt>=n:
        break
for i in range(l[cnt-1]+1,100001):
    ans[i]=ans[i-1]
q=int(input())
for i in range(q):
    m=int(input())
    if m<=l[-1]:
        print(ans[m])
    else:
        print(ans[l[-1]])
```

2. 选做题目

如果耗时太长，直接看解题思路，或者源码

CF466C: Number of Ways

binary search/brute force/data structures/dp/two pointers, 1700

https://codeforces.com/problemset/problem/466/C

思路：当且仅当sum为3的倍数且2s/3和s/3的和均能出现且s/3先出现时情况成立，记录s/3出现数目，当2s/3出现是则前面所有s/3数目都能成立

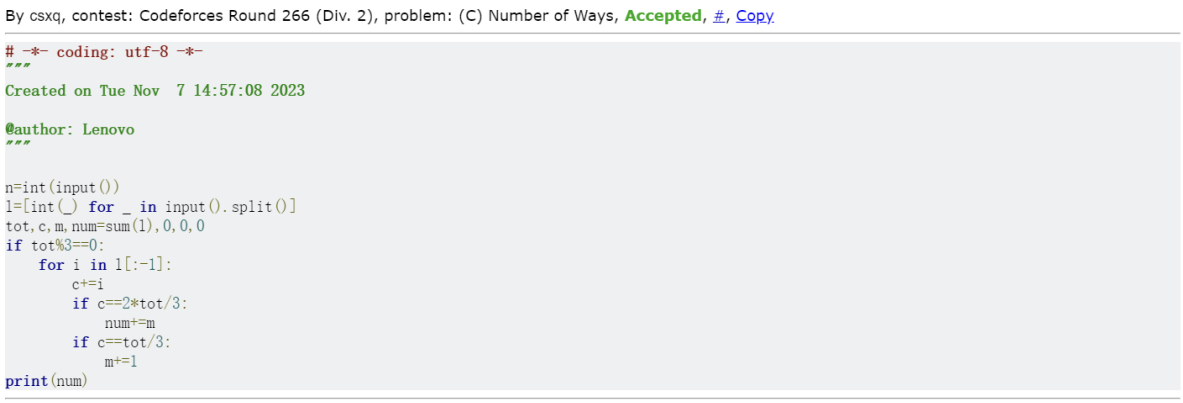
代码

```
# -*- coding: utf-8 -*-
"""
Created on Tue Nov  7 14:57:08 2023

@author: Lenovo
"""

n=int(input())
l=[int(_) for _ in input().split()]
tot,c,m,num=sum(l),0,0,0
if tot%3==0:
    for i in l[:-1]:
        c+=i
        if c==2*tot/3:
            num+=m
        if c==tot/3:
            m+=1
print(num)
```

代码运行截图



CF1443C: The Delivery Dilemma

binary search/greedy/sortings, 1400,

https://codeforces.com/problemset/problem/1443/C
提示： 1）结果要一起输出，不要分次print，会超时。2）用zip函数。

思路：最差情况是每个都自己取，对每家店讨论是否可以让店进行配送

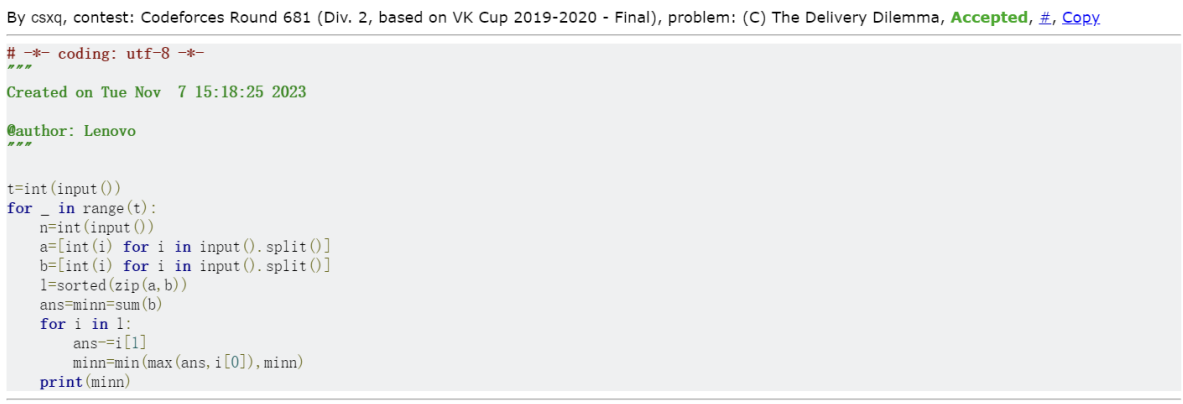
代码

```
# -*- coding: utf-8 -*-
"""
Created on Tue Nov  7 15:18:25 2023

@author: Lenovo
"""

t=int(input())
for _ in range(t):
    n=int(input())
    a=[int(i) for i in input().split()]
    b=[int(i) for i in input().split()]
    l=sorted(zip(a,b))
    ans=minn=sum(b)
    for i in l:
        ans-=i[1]
        minn=min(max(ans,i[0]),minn)
    print(minn)
```

代码运行截图



3. 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“每日选做”中每天推出的2题目、CF、LeetCode、洛谷等网站题目。==

还在练习dp，尝试使用线段树结构