1. (6min)

34B. Sale

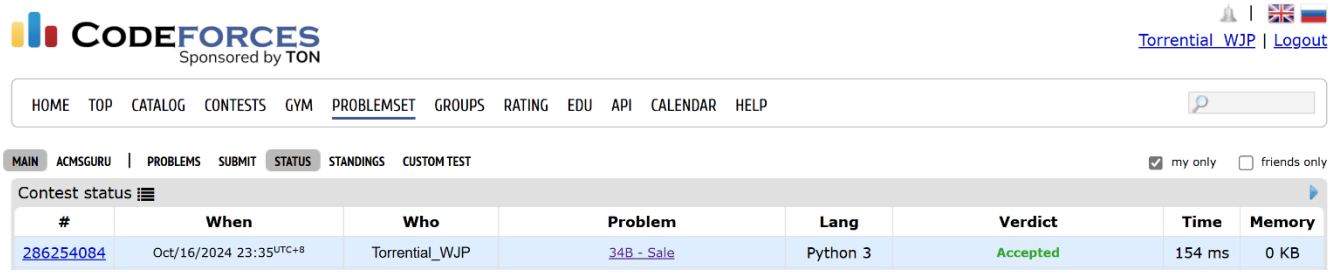
https://codeforces.com/problemset/problem/34/B

思路：排序后对小于零部分求和

代码：

```
m,n=map(int,input().split())
goods=list(map(int,input().split()))
goods=sorted(goods)
i=0
maxi=0
while i<n and goods[i]<0:
    maxi-=goods[i]
    i+=1
print(maxi)
```

运行：



2. (8min)

160A. Twins

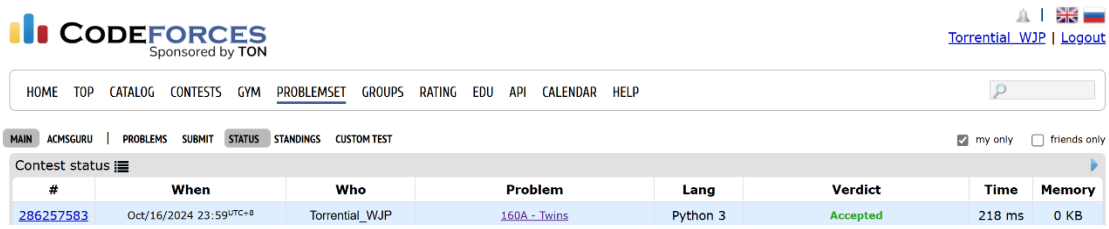
https://codeforces.com/problemset/problem/160/A

思路：排序后找临界指标

代码：

```python
n=int(input())
coin=list(map(int,input().split()))
coin=sorted(coin)
l=0
t=0
m=sum(coin)
for i in range(n-1,-1,-1):
    l+=coin[i]
    t+=1
    if l>m/2:
        break
print(t)
```

运行:



3. (10min)

1879B. Chips on the Board

https://codeforces.com/problemset/problem/1879/B

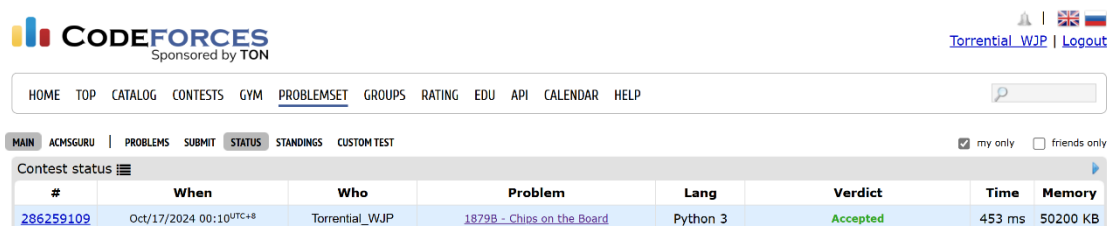
思路: 经过简单的分析我们可以发现“每行都有格子”和“每列都有格子”至少之一成立,, 而如果是前者, 我们只要将列指标取成最小的, 如果是后者, 我们只要将行指标取成最小的, 再在两者中取较小的即可。

代码:

```python
t=int(input())
for i in range(t):
    n=int(input())
    lis1=list(map(int,input().split()))
    lis2=list(map(int,input().split()))
    lis1=sorted(lis1)
```

```
    lis2=sorted(lis2)
    a=min(sum(lis1)+lis2[0]*n,sum(lis2)+lis1[0]*n)
    print(a)
```

运行：



4. (10min)

158B. Taxi

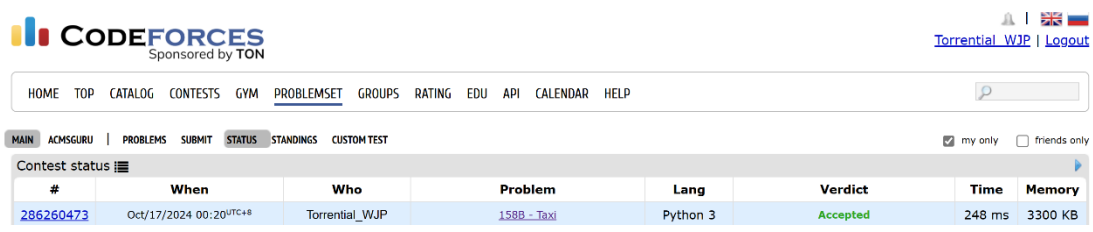
https://codeforces.com/problemset/problem/158/B

思路：1 先尽量配给 3，2 可自己和自己配对，最后如果还有剩下的 1 和 2 再配对

代码：

```
import math
n=int(input())
grp=list(map(int,input().split()))
a=[0,0,0,0]
mini=0
for i in grp:
    a[i-1]+=1
if a[0]<=a[2]:
    mini=a[3]+a[2]+(a[1]+1)//2
else:
    a[0]=a[0]-a[2]
    mini=a[3]+a[2]+math.ceil((2*a[1]+a[0])/4)
print(mini)
```

运行：



5. (40min)

230B. T-primes（选做）

http://codeforces.com/problemset/problem/230/B

思路：欧拉筛

代码：

```
import math
n=int(input())
lis=list(map(int,input().split()))
a=1
for x in lis:
    if math.sqrt(x)==math.floor(math.sqrt(x)) and
math.sqrt(x)>a:
        a=math.floor(math.sqrt(x))
integer=[True]*a
prime=[]
integer[0]=False
for i in range(1,a):
    if integer[i]:
        prime.append(i+1)
        s=0
        while s<=len(prime)-1 and (i+1)*prime[s]<=a:
            integer[(i+1)*prime[s]-1]=False
            s+=1
    else:
        t=0
        while (t==0 or (i+1)%prime[t-1]!=0) and
t<=len(prime)-1 and (i+1)*prime[t]<=a:
            integer[(i+1)*prime[t]-1]=False
            t+=1
    i+=1
for num in lis:
    if math.sqrt(num)!=math.floor(math.sqrt(num)):
        print('NO')
    else:
        if integer[int(math.sqrt(num))-1]:
            print('YES')
        else:
            print('NO')
```

运行:

6. (10min)

12559: 最大最小整数 （选做）

http://cs101.openjudge.cn/practice/12559

思路： 类似于条件型的冒泡排序

代码:

```
n=int(input())
lis=list(input().split())
lis_=lis
m=''
l=''
for i in range(n):
    for j in range(n-1,i,-1):
        if lis[j-1]+lis[j]<lis[j]+lis[j-1]:
            lis[j-1],lis[j]=lis[j],lis[j-1]
    m+=lis[i]
for i in range(n):
    for j in range(n-1,i,-1):
        if lis_[j-1]+lis_[j]>lis_[j]+lis_[j-1]:
            lis_[j-1],lis_[j]=lis_[j],lis_[j-1]
    l+=lis[i]
print(m,l)
```

运行：

总结和收获：

1. 有关第五题：

   我一开始的想法是每次输入一个数，判断其平方根（如果是整数的话）是不是素数，但 TLE 了。遂改成欧拉筛的思路，过了。但是我的室友似乎只在我的第一种思路上做了一些优化（比如质数只能是 6k 加减 1 型的）就过了，甚至还比我的代码快。仔细想了想，欧拉筛的优势在于当数据中有很多平方数时，不需要做重复的计算；而当输入数据中平方数比较少时，使用我一开始的思路反而能省去一些无用的计算，因为它只需要判断特定的几个数是不是质数。

2. 正在补上之前的每日选做，感觉独立完成一道比较复杂的题目还是很有成就感的。