02733: 判断闰年

http://cs101.openjudge.cn/practice/02733/

思路: 能被3200整除的非闰年,能被100整除但是不能被400的整除的也非闰年,余下能被4整除的为闰年,其他为平年

代码

```
year=int(input())
if year%3200==0:
    print('N')
elif year%400!=0 and year%100==0:
    print('N')
elif year%4==0:
    print('Y')
else:
    print('N')
```

代码运行截图 (至少包含有"Accepted")



02750: 鸡兔同笼

http://cs101.openjudge.cn/practice/02750/

思路:如果脚的总数能被4整除,那么最少情况即全为兔子,如果不能被4整除但能被2整除,最少情况需至少有一只鸡

代码

```
a=int(input())
if a%4==0:
    print(int(a/4),int(a/2))
elif a%2==0:
    print(int((a+2)/4),int(a/2))
else: print("0 0")
```

代码运行截图 (至少包含有"Accepted")



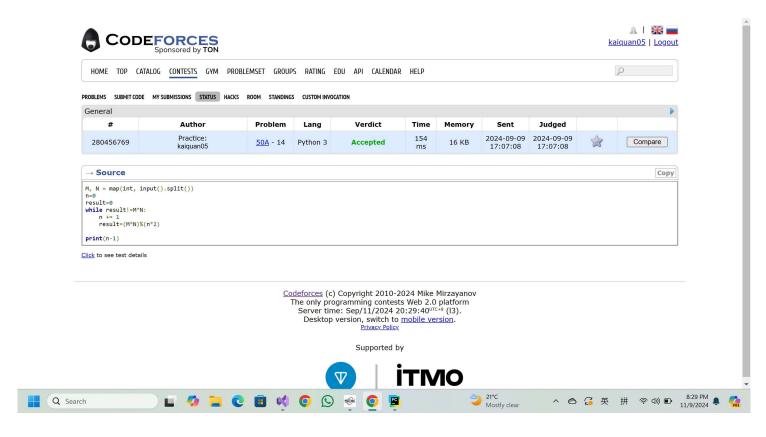


50A. Domino piling

greedy, math, 800, http://codeforces.com/problemset/problem/50/A

思路:观察发现,若M*N大小小于n块dominoes大小,答案为n-1

```
M, N = map(int, input().split())
n=0
result=0
while result!=M*N:
    n += 1
    result=(M*N)%(n*2)
print(n-1)
```



1A. Theatre Square

math, 1000, https://codeforces.com/problemset/problem/1/A

思路:了解题目, 计算行数和列数除以一块flagstones边长,分别向上取整

代码

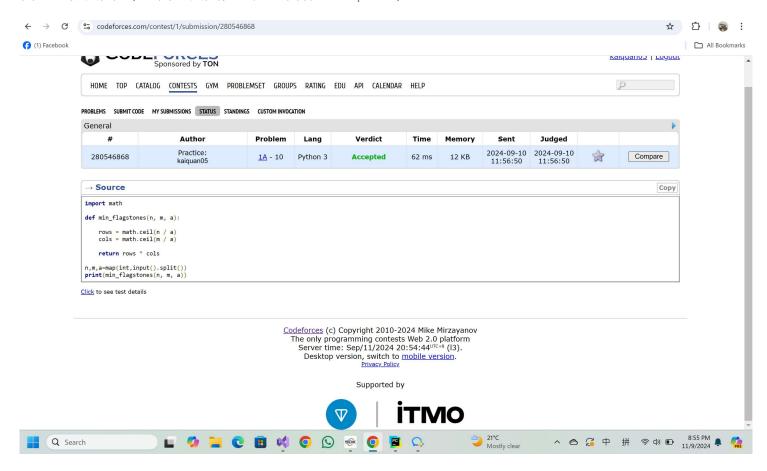
```
import math

def min_flagstones(n, m, a):
```

```
rows = math.ceil(n / a) #向上取整
cols = math.ceil(m / a)

return rows * cols

n,m,a=map(int,input().split())
print(min_flagstones(n, m, a))
```



112A. Petya and Strings

implementation, strings, 1000, http://codeforces.com/problemset/problem/112/A

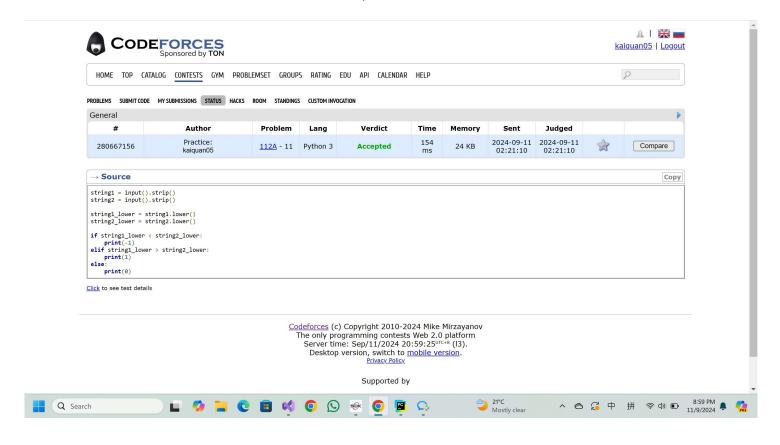
思路: 照题目要求完成, 全部转换成大/小写进行比较

代码

```
string1 = input().strip()
string2 = input().strip()
```

```
string1_lower = string1.lower()
string2_lower = string2.lower()

if string1_lower < string2_lower:
    print(-1)
elif string1_lower > string2_lower:
    print(1)
else:
    print(0)
```



231A. Team

bruteforce, greedy, 800, http://codeforces.com/problemset/problem/231/A

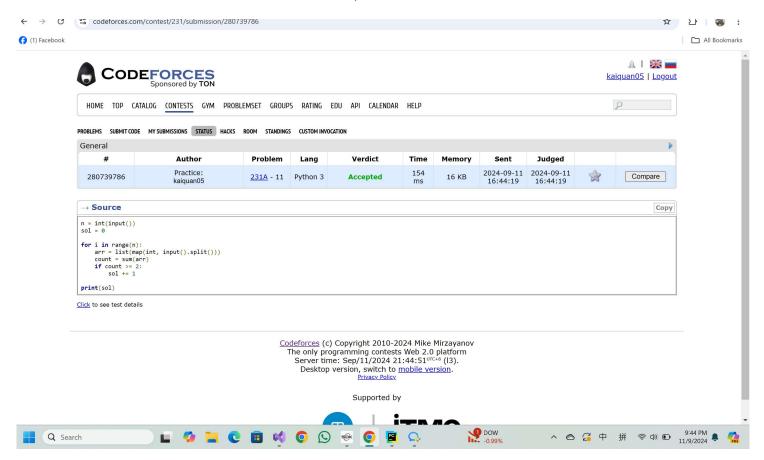
思路:读取,计算,总结,反馈

代码

```
n = int(input())
sol = 0
```

```
for i in range(n):
    arr = list(map(int, input().split()))
    count = sum(arr)
    if count >= 2:
        sol += 1

print(sol)
```



2. 学习总结和收获

这几道题其实算是我学Python以来解答的首六道,一开始还不习惯,会先用C语言写一遍,然后转换成Python,但是渐渐地也能把Python语法掌握,希望未来我能将Python用得更顺手!另外,我的思考速度还是有点太慢,太习惯先推演一遍,花了太长时间,检讨后觉得下次可以直接动手写代码试看运行,也是一种推演,会更快!