

Задача 1.

Если на полотне четное количество клеток, то крестиков и пустых клеток поровну.
Если нечетное — то крестиков на один больше.

```
n = int(input())
m = int(input())
if n * m % 2 == 0:
    print(n * m // 2)
else:
    print(n * m // 2 + 1)
```

Задача 2.

Простое решение (не работает на больших числах).

Будем моделировать календарь данной планеты. Заметим, что если пронумеровать дни недели от 1 до 6 и воскресенье считать 0-ым, то для получения дня недели 13-го числа у следующего месяца, необходимо к текущему номеру дня недели (с учетом перенумерации) прибавить 30 и взять остаток от деления на 7.

```
N = int(input())
k = int(input())
result = 0
k = (k + 12) % 7
for i in range(N):
    if k == 5:
        result += 1
        k = (k + 30) % 7
print(result)
```

Продвинутое решение.

Заметим, что числа 7 и 30 — взаимно простые. Это означает, что последовательность дней, на которые выпадают 13 числа месяца будет являться циклом длинны 7: пятница, воскресенье, вторник, четверг, суббота, понедельник, среда. Таким образом необходимо найти первый месяц с пятницей 13, а после добавить количество оставшихся месяцев в этом году, деленное на 7.

```
N = int(input())
k = int(input())
k = (k + 12) % 7
i = 1
while i <= N and k != 5:
    i += 1
    k = (k + 30) % 7
if i > N:
    print(0)
else:
    print(1 + (N - i) // 7)
```

Задача 3.

Заменяем в строке все символы на символ S, а все числа на символ N. Очевидно, что после замены идентификаторы туристов будут выглядеть как SNNNSS, менеджеров-администраторов как SSNNN и так далее.

Осталось только проверить полученную строку на нужное соответствие и вывести ответ.

```
for i in range(3):
    S = input()
    T = ""
    for char in T:
        if "0" <= char <= "9":
            T += "N"
        else:
            T += "S"
    if T == "NSSSNN":
        print(1)
    elif T == "SSNNNN":
        print(2)
    elif T == "SSNNNNN":
        print(3)
    elif T == "NNNNSS":
        print(4)
    else:
        print(0)
```