

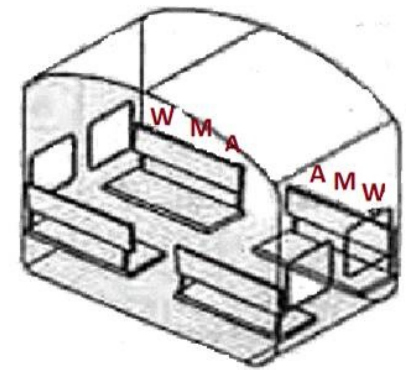
# Ингеройнен Алексей Андреевич

Балл: 120

## Задача №1

### Условие

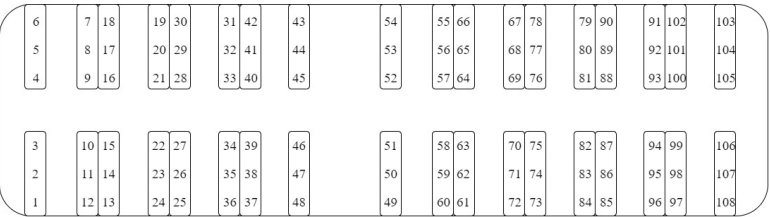
Витя и Маша очень любят путешествовать. В основном они путешествуют по железной дороге. Однажды они ехали в вагоне, где были только сидячие места, и заинтересовались расположением сидений в своем купе. Купе выглядело примерно так



Тип места обозначается следующим образом:

- Сиденье у окна: W
- Среднее место: M
- Место у прохода: A

Нумерация мест в вагоне приведена на следующем рисунке



Ребятам стало интересно узнать номер и тип сиденья, находящегося напротив.

Напишите программу, которая поможет ребятам решить эту задачу.

### Входные данные

На вход подается одно целое число N (1 <= N <= 108), обозначающее номер места.

### Выходные данные

Целое число, и большая латинская буква W, M или A, разделенные одним пробелом, которые будут соответствовать номеру и типу сиденья напротив.

### Примеры

| Входные данные | Выходные данные |
|----------------|-----------------|
| 19             | 18 W            |
| 29             | 32 M            |

### Исходный код

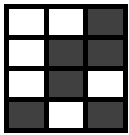
```
n = int(input())
A = [3, 4, 9, 10, 15, 16, 21, 22, 27, 28, 33, 34, 40, 39, 45, 46, 51, 52, 57, 58, 63, 64, 69, 70, 75, 81, 82, 88, 87, 93, 87, 94, 99, 100, 105, 106]
M = [2, 5, 8, 11, 17, 20, 29, 32, 41, 44, 53, 56, 65, 68, 77, 80, 89, 92, 101, 104, 107, 98, 95, 86, 83, 74, 71, 62, 59, 90, 47, 38, 35, 26, 23, 14]
z = 0
if n % 6 == 0:
    z = n + 1
elif n % 6 == 1:
    z = n + 11
elif n % 6 == 2:
    z = n + 9
elif n % 6 == 3:
    z = n + 7
elif n % 6 == 4:
    z = n + 5
elif n % 6 == 5:
    z = n + 3
if n % 12 > 6 or n % 12 == 0:
    z = z - 12
print(z, end=' ')
if z in A:
    print('A')
elif z in M:
```

```
print('M')
else:
    print('W')
```

Задача №2

Условие

Петя недавно узнал, что для записи чисел достаточно использовать только нолики и единички. Числа получаются длиннее, зато их можно изображать, закрашивая клеточки тетради, используя только карандаш или только ручку. Петя решил, что он будет закрашивать клеточку, когда встречается цифра один, и оставлять пустой, если цифра ноль. Например, вот такой рисунок Пети



означает число  $001\ 011\ 010\ 101_2 = 512 + 128 + 64 + 16 + 4 + 1 = 725$ .

Петя считает, что каждый рисунок должен обладать разнообразностью, т.е. не должно быть рисунков, в которых сильно преобладает какой-то один цвет. Лучше, когда пустых и закрашенных клеточек приблизительно поровну. Теперь Петя задался вопросом, сколько разных чисел можно изобразить такими картинками.

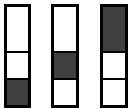
На вход программе подаются три натуральных числа ширина и высота картинка в клеточках, а также количество клеточек, которые должны быть закрашены, чтобы картинка обладала разнообразностью. Общее количество клеточек в изображении не превышает 30, количество клеточек, которые должны быть закрашены меньше общего количества клеточек.

Выведите, сколько разных чисел можно изобразить заданным образом.

Пример

| Входные данные | Выходные данные |
|----------------|-----------------|
| 1 3 1          | 3               |

Пояснение к примеру. С одной закрашенной клеточкой разных рисунков размера 3 x 1 может быть только три:



Исходный код

```
def fact(a):
    z = 1
    for i in range(1, a + 1):
        z *= i
    return z

q, w, n = map(int, input().split())
S = q * w
print(fact(S) / fact(n) / fact(S - n))
```

Задача №3

Условие

Исполнитель Цифропоглотитель получает на вход натуральное число  $X$  (не превышающее  $10^6$ ). По этому числу строится трёхзначное восьмеричное число  $Y$  по следующим правилам.

- 1. Первая цифра числа  $Y$  – остаток от деления  $X$  на 2.
- 2. Вторая цифра числа  $Y$  – остаток от деления  $X$  на 3.
- 3. Третья цифра числа  $Y$  – остаток от деления  $X$  на 5.

Пример. Исходное число: 35. Остаток от деления на 2 равен 1; остаток от деления на 3 равен 2; остаток от деления на 5 равен 0. Результат работы автомата:  $120_8$ .

(Замечание:  $120_8=80_{10}$ )

Напишите программу, которая будет находить наибольшее десятичное число, которое можно получить на заданном интервале  $[A,B]$ , и наибольшее число, на котором оно достигается (например, число 80 можно получить из чисел 5 и 35, наибольшее из них 35, оно и пойдет в ответ).

На вход программы подаётся два целых числа  $A$  и  $B$  ( $1 \leq A \leq B \leq 1\ 000\ 000$ ), записанных через пробел.

Программа должна вывести одно целое число - вычисленное значение.

| Входные данные | Вывод | Примечание |
|----------------|-------|------------|
|----------------|-------|------------|

|       |       |   |
|-------|-------|---|
| 34 40 | 80 35 | 34 -> 12<br>35 -> 80<br>36 -> 1<br>37 -> 74<br>38 -> 19<br>39 -> 68<br>40 -> 8<br><br>Наибольшее значение, получаемое на заданном интервале 80. Оно получается из числа 35. |
|-------|-------|---|

#### Исходный код

```

q, w = map(int, input().split())
z = Z = a = b = c = P = o = 0
for i in range(q, w + 1):
    a = i % 2
    b = i % 3
    c = i % 5
    P = a * 100 + b * 10 + c
    o = int(str(P), 8)
    if o > Z:
        Z = o
        z = i
    elif o == Z:
        if z < i:
            z = i
print(Z, z)

```

#### Задача №4

##### Условие

Станция связи принимает сообщения от нескольких зондов. Каждый зонд посылает сообщение в виде нескольких идущих подряд пакетов фиксированной длины. Пакет состоит из символов. Длина пакета для каждого зонда своя. Из-за ошибки в программе станция связи обрабатывает каждый пакет в течение одной секунды.

Напишите программу, которая найдет, сколько секунд станция обрабатывала самую долгую передачу.

##### Формат ввода

В первой строке вводится сначала целое число *n* – количество пакетов в цепочке (*n* ≤ 1000), затем в *n* следующих строках записано по слову. Слова состоят только из строчных латинских букв.

##### Формат вывода

Вывести одно целое число – сколько секунд станция обрабатывала самую долгую передачу.

#### Примеры

| Входные данные                     | Выходные данные |
|------------------------------------|-----------------|
| 3<br>abc<br>aab<br>abad            | 2               |
| 4<br>abmn<br>abdv<br>abcd<br>abcfd | 3               |

#### Исходный код

```

q = int(input())
Z = 0
T = 0
a = ''
A = []

for i in range(1, q + 1):
    a = input()
    A.append(len(a))

for i in A:
    T = A.count(i)
    if T > Z:
        Z = T

```

```
print(Z)
```

Задача №5

Условие

Миша собирает со своей команды данные о количестве тренировок. Игроки сообщают о тренировках, вписывая в форме опроса в одном окне сначала количество сыгранных игр, а потом – свой номер. Номер игрока представляет собой двузначное число. Собрав данные, Миша увидел, что все пробелы между количеством игр и номером игрока были удалены. Напишите программу, которая поможет Мише найти игрока с наибольшим количеством тренировок.

Формат ввода

В строке вводится сначала целое число *n* – количество записей (*n* ≤ 1000), затем *n* натуральных чисел, каждое из которых не менее 100. Все числа отделены друг от друга одним или несколькими пробелами.

Формат вывода

Вывести одно целое число – номер игрока, сыгравшего больше всего игр.

Пример

| Входные данные      | Выходные данные |
|---------------------|-----------------|
| 4 1125 496 882 1056 | 25              |

Исходный код

```
s = input()
while ' ' in s:
    s = s.replace(' ', ' ')
S = s.split()
n = int(S[0])
Z = 0
A = 0
q = ''
for i in range(1, n + 1):
    q = S[i][:2]
    if int(q) > Z:
        Z = int(q)
        A = S[i][2:] + S[i][-1]
print(A)
```

Задача №6

Условие

В цепи постоянного тока под напряжением U надо получить силу тока не больше, чем I, включив нагрузку. Получить это напряжение можно, включив в цепь один из нескольких предложенных резисторов. Для каждого резистора известны его сопротивление и цена. Найти резистор, включить который выйдет дешевле всего. Гарантируется, что хотя бы один подходящий резистор есть.

На вход программе в первой строке подаются натуральное число N – количество резисторов (N ≤ 20) и два вещественных положительных числа: U – требуемое напряжение, I – сила тока в цепи

Далее в N строках подается на вход пара вещественных неотрицательных чисел – r<sub>i</sub> и p<sub>i</sub>: сопротивление i-го резистора и цена за него соответственно.

Вывести одно натуральное число – номер самого дешевого подходящего резистора.

Количество теплоты рассчитывается по формуле

$U = IR$

где U – напряжение, I – сила тока, R – сопротивление.

Пример

| Ввод           | Вывод |
|----------------|-------|
| 2 380 0.001    | 1     |
| 1000000 1005.9 |       |
| 1000 1         |       |

Исходный код

```
n, u, Q = map(float, input().split())
Z = 0
n = int(n)
pm = 10 ** 20
rnow = pnow = 0
for j in range(1, n + 1):
    rnow, pnow = map(float, input().split())
    if u / rnow <= Q:
        if pnow < pm:
            pm = pnow
            Z = j
print(Z)
```

Задача №7

Условие

Автоматика, управляющая грузовым лифтом, оперирует только четырьмя командами: «Открыть двери», «Закрыть двери», «Поехать на второй этаж», «Поехать на первый этаж». Из-за недоработок в системе нельзя два раза подряд выполнить команду «Закрыть двери». Остальные попытки обрабатываются системой корректно.

Сколько существует последовательностей команд длиной  $n$ , не приводящих к ошибке и не содержащих двух последовательных перемещений между этажами, если лифт в начальный момент времени стоял на втором этаже?

На вход программе подается натуральное число  $n$  ( $N \leq 20$ ) — количество команд.

Вывести целое число — количество последовательностей команд длиной  $n$ , не приводящих к ошибке и не содержащих двух одинаковых команд подряд.

Пример

| Ввод | Вывод |
|------|-------|
| 2    | 7     |

Исходный код

```
q = int(input())
pa = 0
pb = 0
pc = 1
pd = 0
a = c = b = d = 0
for i in range(1, q + 1):
    a = pa + pb + pc + pd
    b = pa + pc + pd
    c = pa + pb
    d = pa + pb
    pa, pb, pc, pd = a, b, c, d
print(pa + pb + pc + pd)
```