



ШКОЛКОВО

ЕГЭ

# Информатика

открытый курс

Вариант 7. Уровень: НОРМ. Решения

**Задача 1.**

Между населёнными пунктами  $A, B, C, D, E, F$  построены дороги, протяжённость которых приведена в таблице. Отсутствие числа в таблице означает, что прямой дороги между пунктами нет.

	A	B	C	D	E	F
A				4	7	3
B			10		8	
C		10		5		
D	4		5			
E	7	8				2
F	3				2	

Определите длину кратчайшего пути между пунктами  $C$  и  $E$  (при условии, что передвигаться можно только по построенным дорогам).

**Ответ.**

14

**Решение.**

Построим возможные пути из  $C$  в  $E$ :

1)  $C \rightarrow B \rightarrow E = 10 + 8 = 18$

2)  $C \rightarrow D \rightarrow A \rightarrow E = 5 + 4 + 7 = 16$

3)  $C \rightarrow D \rightarrow A \rightarrow F \rightarrow E = 5 + 4 + 3 + 2 = 14$

Длина кратчайшего пути 14.

## Задача 2.

Логическая функция  $F$  задаётся выражением  $(x \rightarrow w) \wedge (\bar{z} \rightarrow y) \wedge (y \rightarrow x)$ . На рисунке приведён фрагмент таблицы истинности функции  $F$ . Определите, какому столбцу таблицы истинности функции  $F$  соответствует каждая из переменных  $x, y, z, w$ .

?	?	?	?	F
0	0	0	1	1
0	0	1	1	1
0	1	1	1	1

В ответе напишите буквы  $x, y, z, w$  в том порядке, в котором идут соответствующие им столбцы. Буквы в ответе пишите подряд, никаких разделителей между буквами ставить не нужно.

**Ответ.**

uxwz

**Решение.**

Напишем программу:

```
print('x y z w')
for x in [0, 1]:
    for y in [0, 1]:
        for z in [0, 1]:
            for w in [0, 1]:
                p = (x <= w) and ((not z) <= y) and (y <= x)
                if p == 1:
                    print(x, y, z, w)
```

Программа выдаст следующий результат:

```
x y z w
0 0 1 0
0 0 1 1
1 0 1 1
1 1 0 1
1 1 1 1
```

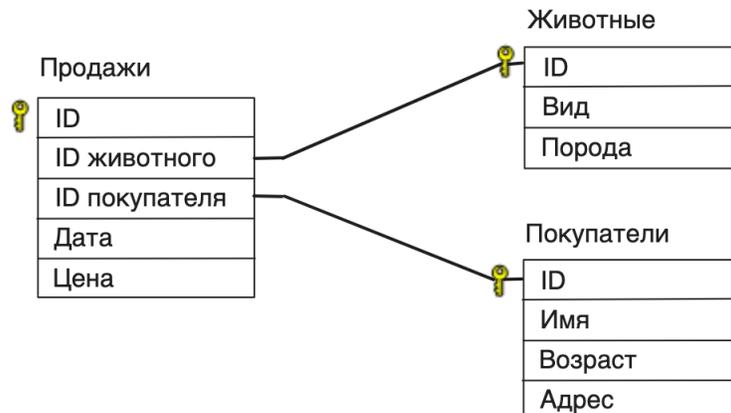
Видим, что только одна строка имеет три цифры 0 и одну 1, значит, в исходной таблице последний столбец – это  $z$ . Две цифры 0 и две цифры 1 также имеет только одна строка и из них правый столбец мы уже выяснили, значит, третий столбец –  $w$ . Обратим внимание, что у нас две строки с тремя цифрами 1 и одним 0, однако в одной из строк мы уже выяснили, чему равен столбец с нулем, значит, остается  $y$ , который является первым столбцом в исходной таблице, а оставшийся столбец –  $x$ .



### Задача 3.

Задание выполняется с использованием прилагаемых файлов.

В файле приведён фрагмент базы данных «Зоомагазин» о доступных животных, покупателях и совершенных продажах. База данных состоит из трёх таблиц. Таблица «Продажи» содержит записи о проданных животных различным покупателям, а также информацию о дате каждой продажи и финальной цене. Таблица «Животные» содержит информацию о характеристиках каждого животного. Таблица «Покупатели» содержит данные о покупателях магазина. На рисунке приведена схема указанной базы данных.



Используя информацию из приведённой базы данных, определите, сколько людей старше 50 лет купили собак породы «Лабрадор».

**Ответ.**

3

**Решение.**

Найдем ID собаки породы "Лабрадор" на листе "Животные 39".

Вернем возраст на лист "Продажи" по каждому ID покупателя с помощью формулы ВПР. Запишем в ячейку F2 формулу =ВПР(C2; Покупатели!A:D; 3; 0).

Далее запишем формулу, которая будет возвращать 1, если возраст человека старше 50 и если порода собаки – Лабрадор. =ЕСЛИ(И(F2>50; B2=39); 1; 0). Кол-во ячеек, в которых функция вывела 1 – 3.



#### Задача 4.

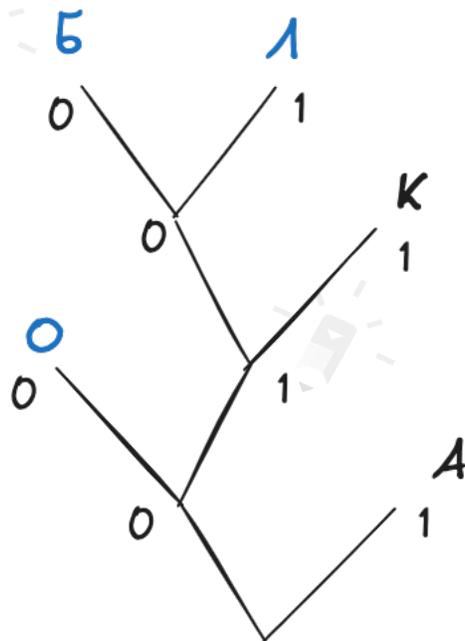
Для кодирования информационных сообщений, состоящих из букв А, Б, К, Л, О, используется неравномерный двоичный код, удовлетворяющий обратному условию Фано. Для букв А, К использовали соответственно кодовые слова: 1, 110. Какое наименьшее количество двоичных разрядов потребуется для кодирования слова ОБЛАКО?

*Примечание:* Обратное условие Фано означает, что никакое кодовое слово не является концом другого кодового слова. Соблюдение этого условия достаточно для однозначного декодирования информации.

**Ответ.**

16

**Решение.**



$$2 + 4 + 4 + 1 + 3 + 2 = 16$$

### Задача 5.

Автомат получает на вход семизначное число. По этому числу строится новое число по таким правилам:

1. Складываются квадраты цифр, стоящих на нечетных позициях;
2. Складываются квадраты цифр, стоящих на четных позициях;
3. Затем в порядке убывания записываются эти суммы.

Укажите наибольшее число, при вводе которого автомат выдает число 1512.

**Ответ.**

9161503

**Решение.**

```
for i in range(9999999, 1000000 - 1, -1):
    x = str(i)
    nechet = 0
    chet = 0
    for j in range(len(x)):
        if j % 2 == 0:
            chet += int(x[j]) ** 2
        else:
            nechet += int(x[j]) ** 2
    res = str(max(chet, nechet)) + str(min(chet, nechet))
    if res == "1512":
        print(i)
        break
```

### Задача 6.

Исполнитель Черепаха действует на плоскости с декартовой системой координат. В начальный момент Черепаха находится в начале координат, её голова направлена вдоль положительного направления оси ординат, хвост опущен. При опущенном хвосте Черепаха оставляет на поле след в виде линии. В каждый конкретный момент известно положение исполнителя и направление его движения. У исполнителя существует 4 команды: **Вперёд  $n$**  (где  $n$  – целое число), вызывающая передвижение Черепахи на  $n$  единиц в том направлении, куда указывает её голова; **Назад  $n$**  (где  $n$  – целое число), вызывающая передвижение в противоположном голове направлении; **Направо  $m$**  (где  $m$  – целое число), вызывающая изменение направления движения на  $m$  градусов по часовой стрелке, **Налево  $m$**  (где  $m$  – целое число), вызывающая изменение направления движения на  $m$  градусов против часовой стрелки.

Запись **Повтори  $k$  [Команда 1 Команда 2 ... Команда S]** означает, что последовательность из  $S$  команд повторится  $k$  раз. Черепахе был дан для исполнения следующий алгоритм:

**Повтори 4 [Повтори 4 [Вперёд 10 Направо 90] Вперёд 3 Направо 90]**

Определите количество точек на линиях, заданных данным алгоритмом.

**Ответ.**

128

**Решение.**

Напишем программу:

```
from turtle import *
```

```
k = 20  
tracer(0)
```

```
for i in range(4):  
    for j in range(4):  
        forward(10 * k)  
        right(90)  
        forward(3 * k)  
        right(90)  
up()
```

```
for x in range(-20 * k, 20 * k, k):  
    for y in range(-20 * k, 20 * k, k):  
        goto(x, y)  
        dot(3, "red")  
done()
```

Количество точек на каждом из "угловых" квадратов – 28. Значит, всего на них 112 точек. И остается еще  $4 \cdot 4$  точек. Всего  $112 + 16 = 128$  точек.



### Задача 7.

Камера делает фотографии и передаёт их по каналу связи в виде сжатых изображений размером  $1280 \times 720$  пикселей с разрешением 12 битов. Пропускная способность канала позволяет передать ровно 10 фотографий в секунду. Камеру заменили на новую, которая передаёт фотографии размером  $1920 \times 1080$  пикселей с разрешением 16 бит, при этом коэффициент сжатия изображений не изменился. Сколько фотографий сможет полностью передать новая камера за одну секунду, если в полтора раза увеличить пропускную способность канала связи?

**Ответ.**

5

**Решение.**

Найдем размер одного изображения со старой камеры:  $1280 \cdot 720 \cdot 12$ . Так как камера делает 10 фотографий в секунду, то тратится  $1280 \cdot 720 \cdot 12 \cdot 10$  бит памяти.

Если в полтора раза увеличить пропускную способность, значит, можно передавать  $1.5 \cdot 1280 \cdot 720 \cdot 12 \cdot 10$  бит памяти.

Если одна фотография весит  $1920 \cdot 1080 \cdot 16$  бит, а всего  $1.5 \cdot 1280 \cdot 720 \cdot 12 \cdot 10$  бит памяти, то можно сделать  $\frac{1.5 \cdot 1280 \cdot 720 \cdot 12 \cdot 10}{1920 \cdot 1080 \cdot 16} = 5$  фотографий.



### Задача 8.

Все 6-буквенные слова, составленные из букв слова ШКОЛКОВО, записаны в алфавитном порядке и пронумерованы.

Вот начало списка:

1. ВВВВВВ
2. ВВВВВК
3. ВВВВВЛ
4. ВВВВВО
5. ВВВВВШ

...

Сколько существует слов с чётными номерами, которые начинаются с буквы Л и содержат не менее двух букв О?

**Ответ.**

386

**Решение.**

```
letters = "ВКЛОШ"
ans = 0
num = 1
for q in letters:
    for w in letters:
        for e in letters:
            for r in letters:
                for t in letters:
                    for y in letters:
                        word = q + w + e + r + t + y
                        if num % 2 == 0 and word[0] == "Л" and word.count("О") >= 2:
                            ans += 1
                            num += 1
print(ans)
```

### **Задача 9.**

*Задание выполняется с использованием прилагаемых файлов.*

В файле электронной таблицы в каждой строке содержатся пять натуральных чисел. Определите количество строк таблицы, содержащих числа, для которых выполнено ровно одно условие:

- разность максимального и минимального чисел не менее **100**;
- произведение трёх других чисел не более **10000**.

**Ответ.**

**1667**

**Решение.**

Отсортируем числа в ячейки F1-J1 с помощью формулы =НАИБОЛЬШИЙ(): первым аргументом подаем диапазон, а вторым какой наибольший по убыванию. В ячейку F1 запишем =НАИБОЛЬШИЙ(A1:E1; 1), правее =НАИБОЛЬШИЙ(A1:E1; 2) и т.д.

Для первого условия пропишем следующую формулу: =ЕСЛИ(F1-J1>=100; 1; 0), для второго следующую: =ЕСЛИ(G1\*N1\*I1<=10000; 1; 0).

Чтобы проверить, что выполнено ровно одно, проверим, что сумма ячеек равна 1: =ЕСЛИ(K1+L1=1; 1; 0).

**Задача 10.**

*Задание выполняется с использованием прилагаемых файлов.*

Определите сколько раз в романе-антиутопии Д. Оруэлла "1984" встречается слово "противоположность" в разных формах и падежах не в составе сносков.

**Ответ.**

6

**Решение.**

Нажмем Ctrl+F и введем слово "противоположность". Далее прокликаем все варианты и посмотрим, чтобы они были не в сносках.



### **Задача 11.**

При регистрации в компьютерной системе каждому пользователю выдаётся пароль, состоящий из 15 символов. В качестве символов используют прописные и строчные буквы латинского алфавита (в нём 26 букв), а также десятичные цифры и символы #, @, %. В базе данных для хранения сведений о каждом пользователе отведено одинаковое и минимально возможное целое число байт. При этом используют посимвольное кодирование паролей, все символы кодируют одинаковым и минимально возможным количеством бит. Кроме собственно пароля, для каждого пользователя в системе хранятся дополнительные сведения, для чего выделено 5 байт на одного пользователя. В компьютерной системе выделено 2 Кб для хранения сведений о пользователях. О каком наибольшем количестве пользователей может быть сохранена информация в системе? В ответе запишите только целое число – количество пользователей.

**Ответ.**

107

**Решение.**

Все в пароле используют алфавит из  $26 + 26 + 10 + 3 = 65$  символов, значит, нужно 7 бит на один символ. В пароле используется 15 символов, т.е. под один пароль отведено  $15 \cdot 7 = 105$  бит, т.е. 14 байт. Если выделено еще 5 байт доп. сведений, то всего на каждого пользователя выделено 19 байт.

Значит, информация может быть сохранена о  $\frac{2048}{19} = 107$  пользователях.

## Задача 12.

Исполнитель Редактор получает на вход строку цифр и преобразует её. Редактор может выполнять две команды, в обеих командах  $v$  и  $w$  обозначают цепочки цифр.

А) заменить ( $v, w$ ).

Эта команда заменяет в строке первое слева вхождение цепочки  $v$  на цепочку  $w$ . Например, выполнение команды

заменить (111, 27)

преобразует строку 05111150 в строку 0527150.

Если в строке нет вхождений цепочки  $v$ , то выполнение команды заменить ( $v, w$ ) не меняет эту строку.

Б) нашлось ( $v$ ).

Эта команда проверяет, встречается ли цепочка  $v$  в строке исполнителя Редактор. Если она встречается, то команда возвращает логическое значение «истина», в противном случае возвращает значение «ложь». Строка исполнителя при этом не изменяется.

Цикл

ПОКА условие

    последовательность команд

КОНЕЦ ПОКА

выполняется, пока условие истинно.

Дана программа для редактора:

НАЧАЛО

    ПОКА нашлось(46)

        заменить(46, 6)

        заменить(65, 5)

    КОНЕЦ ПОКА

КОНЕЦ

В результате работы программы к строке  $\underbrace{4\dots 4}_{20} \underbrace{6\dots 6}_{?} \underbrace{5\dots 5}_{35}$  получилась строка, в которой содержится 15 цифр 6. Сколько цифр 6 было в изначальной строке?

**Ответ.**

35

**Решение.**

```
for i in range(1000):
    n = '4' * 20 + '6' * i + '5' * 35
    while '46' in n:
        n = n.replace('46', '6', 1)
        n = n.replace('65', '5', 1)
    if n.count('6') == 15:
        print(i)
```

### Задача 13.

В терминологии сетей TCP/IP маска сети – это двоичное число, меньшее  $2^{32}$ ; в маске сначала (в старших разрядах) стоят единицы, а затем с некоторого места нули. Маска определяет, какая часть IP-адреса узла сети относится к адресу сети, а какая – к адресу самого узла в этой сети. Обычно маска записывается по тем же правилам, что и IP-адрес – в виде четырех байт, причем каждый байт записывается в виде десятичного числа. Адрес сети получается в результате применения поразрядной конъюнкции к заданному IP-адресу узла и маске.

Например, если IP-адрес узла равен 131.32.255.131, а маска равна 255.255.240.0, то адрес сети равен 131.32.240.0.

Для узла с IP-адресом 141.78.142.4 адрес сети равен 141.78.128.0. Чему равно наибольшее возможное значение третьего слева байта маски? Ответ запишите в виде десятичного числа.

**Ответ.**

240

**Решение.**

Запишем третий слева байт IP-адреса в двоичной системе счисления:  $142_{10} = 10001110_2$ .

Аналогично запишем третий слева байт адреса сети, также в двоичной системе счисления:  $128_{10} = 10000000_2$ .

Учитывая, что адрес сети получается в результате применения поразрядной конъюнкции к заданному IP-адресу узла и маске, запишем возможный вариант маски, обозначив символом  $x$  те места, где маска может иметь неоднозначное значение:  $1xxx0000$ .

Так как значение должно быть наибольшее, то вместо символов  $x$  ставим 1. Получаем значение третьего слева байта маски:  $11110000_2 = 240_{10}$ .



#### Задача 14.

Операнды арифметического выражения записаны в системе счисления с основанием 21.

$$F0xH1_{21} + ABx23_{21}$$

В записи чисел переменной  $x$  обозначена неизвестная цифра из алфавита 21-ричной системы счисления. Определите наибольшее значение  $x$ , при котором значение данного арифметического выражения в десятичной системе счисления кратно 43. Для найденного значения  $x$  вычислите частное от деления значения арифметического выражения на 43 и укажите его в ответе. Основание системы счисления в ответе указывать не нужно.

**Ответ.**

115777

**Решение.**

```
for x in range(21):
    p1 = 15 * 21 ** 4 + x * 21 ** 2 + 17 * 21 ** 1 + 1
    p2 = 10 * 21 ** 4 + 11 * 21 ** 3 + x * 21 ** 2 + 2 * 21 ** 1 + 3
    p = p1 + p2
    if p % 43 == 0:
        print(p // 43)
```

### Задача 15.

Обозначим через  $\text{ДЕЛ}(n, m)$  утверждение «натуральное число  $n$  делится без остатка на натуральное число  $m$ ». Для какого наименьшего натурального числа  $A$  формула

$$(\text{ДЕЛ}(x, A) \wedge \neg \text{ДЕЛ}(x, 17)) \rightarrow (\text{ДЕЛ}(x, 19) \vee \text{ДЕЛ}(x, 17))$$

тождественно истинна (то есть принимает значение 1 при любом натуральном значении переменной  $x$ )?

**Ответ.**

17

**Решение.**

```
for A in range(1, 1000):
    flag = 1
    for x in range(1, 1000):
        p = ((x % A == 0) and (x % 17 != 0)) <= ((x % 19 == 0) or (x % 17 == 0))
        if p == 0:
            flag = 0
            break
    if flag == 1:
        print(A)
        break
```



### Задача 16.

Алгоритм вычисления значения функции  $F(n)$ , где  $n$  – целое число, задан следующими соотношениями:

$$F(n) = n, \text{ при } n < 3,$$

$$F(n) = F(n - 2) + 1, \text{ когда } n \geq 3 \text{ и чётное,}$$

$$F(n) = F(2 \cdot n + 1), \text{ когда } n \geq 3 \text{ и нечётное.}$$

Назовите количество значений  $n$  на отрезке  $[1; 567]$ , для которых  $F(n)$  определено.

**Ответ.**

284

**Решение.**

Заметим, что при нечетных  $n$  у нас число как было нечетное, так им и останется, а значит, оно будет бесконечно увеличиваться и так и не завершит работу. При четных же все хорошо.

Т.е. нам подходят числа 1, 2 и все четные выше, посчитаем их программой:

```
c = 2
for i in range(3, 567 + 1):
    if i % 2 == 0:
        c += 1
print(c)
```

### Задача 17.

Задание выполняется с использованием прилагаемых файлов.

В файле содержится последовательность целых неотрицательных чисел. Определите количество пар последовательности, в которых оба числа двузначные, и сумма элементов пары кратна максимальному двузначному числу последовательности, оканчивающемуся на 3. В ответе запишите два числа через пробел: сначала количество найденных пар, затем максимальную из сумм элементов всех пар.

В данной задаче под парой подразумевается два соседних элемента последовательности.

**Ответ.**

1 8673

**Решение.**

```
f = open("17.txt")
a = [int(i) for i in f]
mx3 = max([i for i in a if i % 10 == 3 and 10 <= i <= 99])
c = 0
mx = 0
for i in range(len(a) - 1):
    if 10 <= a[i] <= 99 and 10 <= a[i + 1] <= 99 and (a[i] + a[i + 1]) % mx3 == 0:
        c += 1
    mx = max(mx, a[i] + a[i + 1])
print(c, mx)
```

### **Задача 18.**

*Задание выполняется с использованием прилагаемых файлов.*

Квадрат разлинован на  $15 \times 15$  клеток. Исполнитель Робот может перемещаться по клеткам, выполняя за одно перемещение одну из двух команд: вправо или вверх. По команде вправо Робот перемещается в соседнюю правую клетку, по команде вверх – в соседнюю верхнюю. При попытке выхода за границу квадрата Робот разрушается. Перед каждым запуском Робота в каждой клетке квадрата лежит монета достоинством от 1 до 100. Посетив клетку, Робот забирает монету с собой; это также относится к начальной и конечной клетке маршрута Робота. На пути у Робота могут встретиться ямки, они обозначены красным цветом. Попадая в ямку Робот не может из нее выбраться и от горя ломается. Между некоторыми клетками находятся стены, проходить сквозь стены Робот не может.

Откройте файл. Определите максимальную денежную сумму, которую может собрать Робот, пройдя из левой нижней клетки в правую верхнюю. В ответ запишите только одно число – искомую сумму.

**Ответ.**

1962

**Решение.**

Перенесем левую угловую ячейку на поле вниз, например, в A32. Тогда формулы по левому краю будут выглядеть так: =A32+A14, а по нижнему так: =A32+B15. В ячейки с ямами поставим вместо чисел поставим для максимума очень маленькие значения, чтобы даже если Робот их посетил, не учитывал: т.е. поставим -1000000. В ячейку B31 запишем формулу =МАКС(A31;B32)+B14 и растянем. Осталось учесть уголки: над стенками можем прийти только из ячеек слева, а справа от стенок только из ячеек снизу.



### Задача 19.

Два игрока, Петя и Ваня, играют в следующую игру. Перед игроками лежит куча камней. Игроки ходят по очереди, первый ход делает Петя. За один ход игрок может добавить в кучу два камня, добавить три камня или увеличить количество камней в куче в два раза. При этом нельзя повторять ход, который этот же игрок делал на предыдущем ходу. Повторять чужие ходы и свои более старые ходы разрешается. Чтобы делать ходы, у каждого игрока есть неограниченное количество камней. Игра завершается, когда количество камней в куче становится не менее 45. Победителем считается игрок, сделавший последний ход, то есть первым получивший кучу, в которой будет 45 или больше камней. В начальный момент в куче было  $S$  камней,  $1 \leq S \leq 43$ .

Укажите наименьшее значение  $S$ , при котором Петя не может выиграть за один ход, но у Пети есть выигрышная стратегия, позволяющая ему выиграть вторым ходом.

**Ответ.**

18

**Решение.**

```
from functools import lru_cache

def moves(h):
    a, prelast, last = h
    m = []
    if prelast != "+2":
        m.append((a + 2, last, "+2"))
    if prelast != "+3":
        m.append((a + 3, last, "+3"))
    if prelast != "*2":
        m.append((a * 2, last, "*2"))
    return m

@lru_cache(None)
def game(h):
    if h[0] >= 45:
        return "END"
    elif any(game(x) == "END" for x in moves(h)):
        return "WIN1"
    elif all(game(x) == "WIN1" for x in moves(h)):
        return "LOSE1"
    elif any(game(x) == "LOSE1" for x in moves(h)):
        return "WIN2"

for i in range(1, 43 + 1):
    h = i, "", ""
    if game(h) == "WIN2":
        print(i)
```



### Задача 20.

Для игры, описанной в задании 19, найдите наименьшее значение  $S$ , при котором одновременно выполняются два условия:

- у Вани есть выигрышная стратегия, позволяющая ему выиграть первым или вторым ходом при любой игре Пети;
- у Вани нет стратегии, которая позволит ему гарантированно выиграть первым ходом.

**Ответ.**

15

**Решение.**

```
from functools import lru_cache
```

```
def moves(h):  
    a, prelast, last = h  
    m = []  
    if prelast != "+2":  
        m.append((a + 2, last, "+2"))  
    if prelast != "+3":  
        m.append((a + 3, last, "+3"))  
    if prelast != "*2":  
        m.append((a * 2, last, "*2"))  
    return m
```

```
@lru_cache(None)
```

```
def game(h):  
    if h[0] >= 45:  
        return "END"  
    elif any(game(x) == "END" for x in moves(h)):  
        return "WIN1"  
    elif all(game(x) == "WIN1" for x in moves(h)):  
        return "LOSE1"  
    elif any(game(x) == "LOSE1" for x in moves(h)):  
        return "WIN2"  
    elif all(game(x) == "WIN1" or game(x) == "WIN2" for x in moves(h)):  
        return "LOSE2"
```

```
for i in range(1, 43 + 1):  
    h = i, "", ""  
    if game(h) == "LOSE2":  
        print(i)
```



### Задача 21.

Для игры, описанной в задании 19, найдите наименьшее значение  $S$ , при котором одновременно выполняются два условия:

– у Вани есть выигрышная стратегия, позволяющая ему выиграть первым или вторым или третьим ходом при любой игре Пети;

– у Вани нет стратегии, которая позволит ему гарантированно выиграть первым или вторым ходом.

**Ответ.**

11

**Решение.**

```
from functools import lru_cache
```

```
def moves(h):
    a, prelast, last = h
    m = []
    if prelast != "+2":
        m.append((a + 2, last, "+2"))
    if prelast != "+3":
        m.append((a + 3, last, "+3"))
    if prelast != "*2":
        m.append((a * 2, last, "*2"))
    return m
```

```
@lru_cache(None)
```

```
def game(h):
    if h[0] >= 45:
        return "END"
    elif any(game(x) == "END" for x in moves(h)):
        return "WIN1"
    elif all(game(x) == "WIN1" for x in moves(h)):
        return "LOSE1"
    elif any(game(x) == "LOSE1" for x in moves(h)):
        return "WIN2"
    elif all(game(x) == "WIN1" or game(x) == "WIN2" for x in moves(h)):
        return "LOSE2"
    elif any(game(x) == "LOSE2" for x in moves(h)):
        return "WIN3"
    elif all(game(x) == "WIN1" or game(x) == "WIN2" or game(x) == "WIN3" for x in moves(h)):
        return "LOSE3"
```

```
for i in range(1, 43 + 1):
    h = i, "", ""
    if game(h) == "LOSE3":
        print(i)
```



## Задача 22.

Задание выполняется с использованием прилагаемых файлов.

В файле содержится информация о совокупности  $N$  вычислительных процессов, которые могут выполняться параллельно или последовательно. Будем говорить, что процесс  $B$  зависит от процесса  $A$ , если для выполнения процесса  $B$  необходимы результаты выполнения процесса  $A$ . В этом случае процессы могут выполняться только последовательно. Информация о процессах представлена в файле в виде таблицы. В первом столбце таблицы указан идентификатор процесса (ID), во втором столбце таблицы – время его выполнения в миллисекундах, в третьем столбце перечислены с разделителем «;» ID процессов, от которых зависит данный процесс. Если процесс является независимым, то в таблице указано значение 0.

Типовой пример организации данных в файле:

ID процесса B	Время выполнения процесса B (мс)	ID процесса(-ов) A
1	4	0
2	3	0
3	1	1;2
4	7	3

Скажем, что счётчик времени длится до тех пор, пока количество выполняемых процессов не изменится. При этом если выполнение одного процесса закончилось и сразу началось выполнение другого (то есть количество одновременно выполняемых процессов не поменялось), то счётчик не останавливается.

Все независимые друг от друга процессы стартовали одновременно, а зависимые процессы стартовали одновременно с завершением всех процессов, от которых они зависят.

Определите количество одновременно выполняемых процессов, которые длились 9 мс подряд.

В ответе запишите только число.

**Ответ.**

4

**Решение.**

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ	AK	AL	AM									
1	ID процесса B	Время выполнения процесса B (мс)	ID процессов A	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36									
2	1	9	0	1	1	1	1	1	1	1	1	1																																				
3	2	14	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1																															
4	3	10	1; 2															1	1	1	1	1	1	1	1	1																						
5	4	4	3															1	1	1	1	1	1	1	1		1	1	1	1																		
6	5	7	3																								1	1	1	1	1	1	1															
7	6	5	2															1	1	1	1	1	1	1																								
8	7	6	4; 6																																													
9	8	7	6																																													
10	9	8	0	1	1	1	1	1	1	1	1	1																																				
11	10	21	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1																						
12	11	10	9										1	1	1	1	1	1	1	1	1	1	1	1																								
13	12	4	10																							1	1	1	1																			
14	13	8	4; 6																																													
15	14	11	12																																													
16				4	4	4	4	4	4	4	4	4	4	3	3	3	3	3	4	4	4	4	4	3	3	3	3	3	4	4	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		
17													9					5					4					6	4	2																		
18																																																

Для независимых процессов, начиная со столбца D, будем проставлять единицы вправо. Количество единиц равно количеству мс выполнения каждого процесса. Для удобства выберем ширину столбцов 2 – 3.

Затем переходим к зависимым процессам и начинаем также проставлять единицы, но уже после выполнения независимых процессов.

В 16 строке столбца D прописываем формулу СУММ(D2:D15) и растягиваем её вправо. Полученные значения – количество выполняемых одновременно процессов.

Также снизу подпишем сколько мс длилось количество процессов без изменений. Получаем, что 4 процесса длились 9 мс.

### Задача 23.

У исполнителя Калькулятор имеются три команды:

1. Прибавить 1
2. Прибавить 3
3. Умножить на 2

Найдите количество существующих программ, для которых при исходном числе 10 результатом является число 50, и при этом траектория вычислений не содержит двух подряд повторяющихся команд.

**Ответ.**

25

**Решение.**

```
def f(a, b, last_move):
    if a > b:
        return 0
    if a == b:
        return 1
    if last_move == "":
        return f(a + 1, b, "+1") + f(a + 3, b, "+3") + f(a * 2, b, "*2")
    elif last_move == "+1":
        return f(a + 3, b, "+3") + f(a * 2, b, "*2")
    elif last_move == "+3":
        return f(a + 1, b, "+1") + f(a * 2, b, "*2")
    elif last_move == "*2":
        return f(a + 1, b, "+1") + f(a + 3, b, "+3")

print(f(10, 50, ""))
```

### Задача 24.

Задание выполняется с использованием прилагаемых файлов.

Текстовый файл состоит не более чем из  $10^6$  символов и содержит только латинские заглавные буквы. Определите длину самой короткой цепочки символов, которая начинается и заканчивается буквой *F*, и содержит не менее 3 букв *A* в цепочке.

**Ответ.**

7

**Решение.**

```
f = open("24.txt")
n = f.readline()
n = n.split("F")
mn = 100000000
for i in range(1, len(n) - 1):
    if n[i].count("A") >= 3 and len(n[i]) < mn:
        mn = len(n[i])
print(mn + 2) # 2 буквы - F по бокам
```

### Задача 25.

Назовём маской числа последовательность цифр, в которой также могут встречаться следующие символы:

- символ «?» означает ровно одну произвольную цифру;
- символ «\*» означает любую последовательность цифр произвольной длины; в том числе «\*» может задавать и пустую последовательность.

Среди натуральных чисел, не превышающих  $10^8$ , найдите все числа, соответствующие маске  $4*56*8$  и делящиеся **ровно на одно** из чисел из набора [7; 17; 23] без остатка. В ответе запишите количество найденных чисел.

**Ответ.**

11367

**Решение.**

```
from fnmatch import *

ans = 0
for start in [7, 17, 23]:
    for i in range(start, 10 ** 8 + 1, start):
        if fnmatch(str(i), '4*56*8'):
            c = 0
            if i % 7 == 0:
                c += 1
            if i % 17 == 0:
                c += 1
            if i % 23 == 0:
                c += 1
            if c == 1:
                ans += 1
print(ans)
```

## Задача 26.

Задание выполняется с использованием прилагаемых файлов.

Предприятие производит оптовую закупку некоторых изделий  $A$  и  $B$ , на которую выделена определённая сумма денег. У поставщика есть в наличии партии этих изделий различных модификаций по различной цене. На выделенные деньги необходимо приобрести как можно больше изделий  $A$  независимо от модификации. Если у поставщика закончатся изделия  $A$ , то на оставшиеся деньги необходимо приобрести как можно больше изделий  $B$ . Известны выделенная для закупки сумма, а также количество и цена различных модификаций данных изделий у поставщика. Необходимо определить, сколько будет закуплено изделий  $A$  и  $B$  и какая сумма останется неиспользованной.

*Входные данные.*

Первая строка входного файла содержит два целых числа:  $N$  – общее количество партий изделий у поставщика и  $M$  – сумма выделенных на закупку денег (в рублях). Каждая из следующих  $N$  строк описывает одну партию и содержит два целых числа (цена одного изделия в рублях и количество изделий в партии) и один символ (латинская буква  $A$  или  $B$ ), определяющий тип изделия. Все данные в строках входного файла отделены одним пробелом.

В ответе запишите два целых числа без пробела: сначала количество закупленных изделий обоих типов, затем оставшуюся неиспользованной сумму денег.

*Пример входного файла:*

```
4 1000
30 8 A
50 12 B
40 14 A
30 60 B
```

В данном случае сначала нужно купить изделия  $A$ : 8 изделий по 30 рублей и 14 изделий по 40 рублей. На это будет потрачено 800 рублей. На оставшиеся 200 рублей можно купить 6 изделий  $B$  по 30 рублей. Таким образом, всего будет куплено 28 изделий типа  $A$  и останется 20 рублей. В ответе надо записать числа 28 и 20.

**Ответ.**

9223 534

**Решение.**

Открываем Excel, заходим в «Открыть» и выбираем текстовый документ из задачи. В вышедшем окне в первом пункте выбираем формат данных с разделителями. Во втором пункте помимо знака табуляции выбираем пробел. Формат данных столбца общий.

Перетаскиваем числа из первой строки в бок, чтобы не мешались. Пустую первую строку убираем (ПКМ на номер строки – удалить). Выделяем всю таблицу со значениями (столбцы  $A$ ,  $B$ ,  $C$ ), заходим в «Сортировка и фильтр» – «Настраиваемая сортировка», в первую очередь сортируем по столбцу  $C$  в порядке от  $A$  до  $Я$ , затем добавляем уровень для сортировки по столбцу  $A$  по возрастанию.

В соседнем свободном столбце вводим формулу  $=A1*B1$  и продлеваем до конца, тем самым считая стоимость каждой партии. Выделяем ячейки столбца до тех пор, пока значение суммы не превысит число  $M$ . Если у вас не отображается сумма на нижней панели, то её можно включить через контекстное меню, нажав правой кнопкой мыши на нижнюю панель и поставив галочку рядом с соответствующим пунктом. Сумма превышает число

М при последнем числе в 180-й ячейке, значит, сумма у нас в отрезке D1:D179, она равна **2980678**. Остаток равен  $3000000 - 2980678 = 19322$ .

Смотрим, сколько изделий можно купить из следующей партии. Делим **19322** на стоимость одного изделия (**671**), нам нужно целое число от результата – **28**. Стоимость **28** изделий равна  $671 \cdot 28 = 18788$ , остаток равен  $19322 - 18788 = 534$ .

Выясняем, сколько у нас всего куплено изделий:  $=\text{СУММ}(B1:B179)+28$ , выходит **9223**. Остаток, как мы ранее вычислили, равен **534**.

### **Задача 27.**

*Задание выполняется с использованием прилагаемых файлов.*

Геодезист измеряет высоту над уровнем моря (в миллиметрах) относительно уровня начала дороги, для каждой из  $N$  её метровых отметок. Нумерация отметок начинается с единицы.

Проектировщикам необходимо выбрать участок дороги длиной не менее  $K$  метров, на котором значение суммы всех высот, выраженное в миллиметрах, максимально. Это значение называется оценкой участка дороги. Начало и конец искомого участка совпадают с метровыми отметками на дороге. Началом участка считается метровая отметка дороги с меньшим номером.

Определите две метровые отметки дороги так, чтобы расстояние между ними было не менее  $K$  метров, а оценка соответствующего участка дороги – максимально возможной. Укажите в ответе найденное числовое значение максимальной оценки, выраженное в миллиметрах.

*Входные данные.*

Даны два входных файла (файл  $A$  и файл  $B$ ), каждый из которых в первой строке содержит натуральное число  $N$  ( $1 < N \leq 10\,000\,000$ ) – количество метровых отметок, и натуральное число  $K$  ( $1 < K < N$ ) – минимально допустимое расстояние (в метрах) между двумя отметками дороги. В каждой из следующих  $N$  строк находится одно целое число, не превышающее по модулю  $10\,000\,000$ : высота относительно уровня начального участка дороги (в миллиметрах) на соответствующей метровой отметке дороги.

*Пример входного файла:*

```
5 2
-200
-50
500
100
-100
```

При таких исходных данных искомая величина равна  $-50 + 500 + 100 = 550$  для участка дороги длиной 2 от 2-й до 4-й отметки. Ответ: 550. В ответе укажите два числа: сначала искомое значение для файла  $A$ , затем для файла  $B$ .

**Ответ.**

217 522021740

**Решение.**

На следующей странице.

```
f = open("27.txt")
n, k = map(int, f.readline().split())
a = [int(i) for i in f]
s = 0
for i in range(k + 1):
    s += a[i]
ps = 0
smMax = s - ps
psMin = 10 ** 10
for i in range(k + 1, n):
    s += a[i]
    ps += a[i - k - 1]
    psMin = min(psMin, ps)
    smMax = max(smMax, s - psMin, s)
print(smMax)
```

**Подписывайся на наши социальные сети:**

