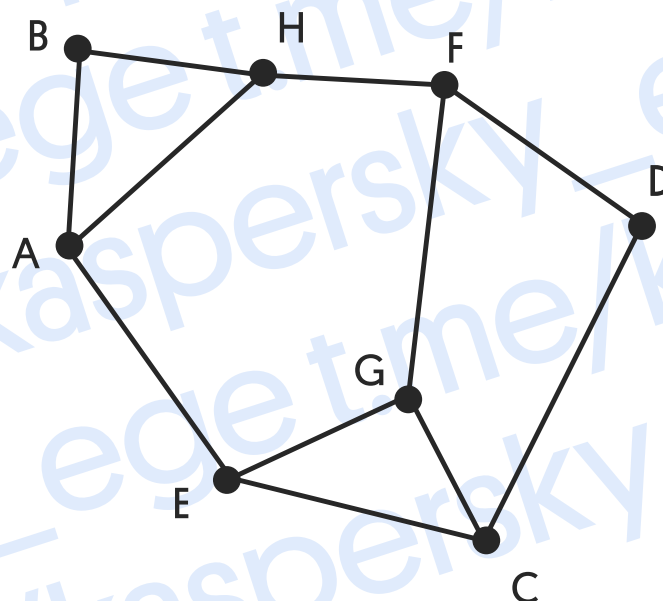


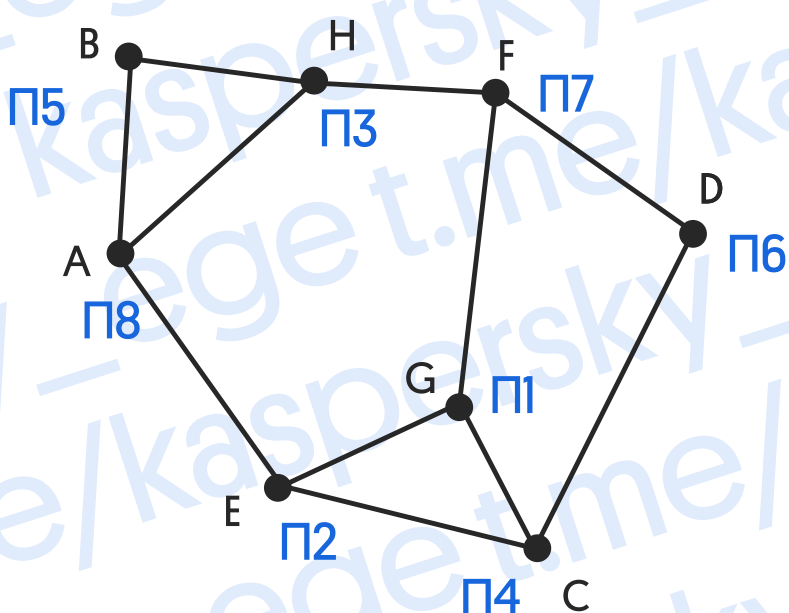
задача

На рисунке схема дорог N-ского района изображена в виде графа, в таблице содержатся сведения о протяжённости каждой из этих дорог (в километрах). Так как таблицу и схему рисовали независимо друг от друга, нумерация населённых пунктов в таблице никак не связана с буквенными обозначениями на графе. Определите, какова сумма протяжённости дорог из пункта В в пункт Н и из пункта А в пункт Е. В ответе запишите целое число.

	П1	П2	П3	П4	П5	П6	П7	П8
П1		53		1			2	
П2	53			13				8
П3					30		39	74
П4	1	13				5		
П5			30					3
П6				5			21	
П7	2		39			21		
П8		8	74		3			



решение



- 1) Нам нужно найти уникальную вершину второй степени, которая ведёт в две вершины степени 3, соединяющиеся между собой. Такой уникальной вершиной является В. Значит вершина В это №5.
- 2) Из неё ведут две дороги - №3 и №8. Выясним, какая из этих вершин ведёт в вершину степени 3, идущую в вершину степени 2. Вершина №3 идёт в вершину №7, идущую в вершину №6 второй степени.
- 3) Не составит труда найти другие вершины. Е - №2, С - №4, G - №1
- 4) Значит путь из А в Е равен 30, а путь из В в Н равен 8. Их сумма равна 38

ответ: 38

трансляция с разбором



задача

Миша заполнял таблицу истинности логической функции $F = [(x \rightarrow y) \rightarrow z] \vee \neg w$, но успел заполнить лишь фрагмент из трёх различных её строк, даже не указав, какому столбцу таблицы соответствует каждая из переменных w, x, y, z .

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

решение

python

```
print("x y z w")
for x in range(2):
    for y in range(2):
        for z in range(2):
            for w in range(2):
                f = ((x <= y) <= z) or (not(w))
                if f == 0:
                    print(x, y, z, w)
```

Получаем таблицу

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

Видим, что в заданной таблице только 3 переменная может не иметь нулей -> 3 переменная = w .

4 переменная может состоять только из нулей -> 4 переменная = z

По 3 строке из заданной таблицы видим, что оставшиеся x и y стоят в комбинации 0 и 1, значит 1 переменная с 0 - x , 2 переменная - y .

ответ: $xywz$



трансляция с разбором

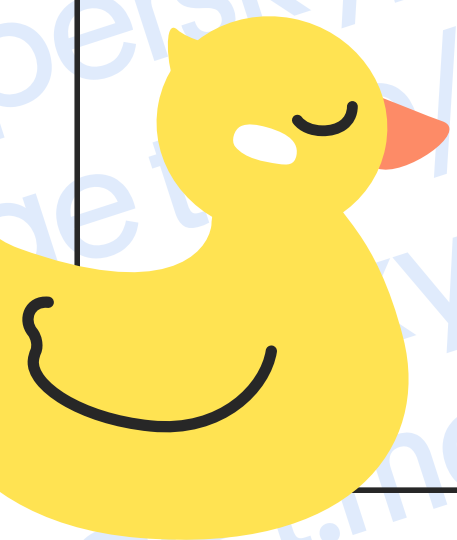


задача

Используя информацию из приведённой базы данных, определите общую выручку от продажи всех видов зефира, полученных магазинами на улице Metallургов за период с 4 по 13 июня включительно. В ответе запишите только число.

решение

1. Сначала в таблице “Движение товара” создадим столбец “Улица”, в который будем выводить магазин, в котором была произведена операция. Для этого используем функцию впр. Полная формула будет выглядеть вот так:
`=ВПР[С2;Магазин!А:С;3;ЛОЖЬ]`
2. Поскольку нас интересует общая выручка, то выведем цену за упаковку. Для этого создадим столбец “Цена” и в него будем выводить цену товаров из таблицы “Товар”. Формула будет выглядеть следующим образом:
`=ВПР[D2;Товар!А:F;6;ЛОЖЬ]`
3. Теперь добавим фильтры, чтобы получить интересующие нас значения. Нас интересует зефир, полученный магазинами на улице Metallургов за период с 4 по 13 июня включительно.
5. На отдельный лист копируем столбцы “Количество упаковок” и “Цена за упаковку”, чтобы посчитать интересующую нас выручку.
6. Чтобы посчитать общую выручку, перемножим количество упаковок на цену за упаковку. Формула будет выглядеть вот так: `=B2*A2`. Её надо растянуть вниз до конца.
7. Ответом будет получившаяся в столбце С сумма (через `СУММ()`).



трансляция с разбором



задача

По каналу связи передаются шифрованные сообщения, содержащие только десять букв: А, Б, Е, И, К, Л, Р, С, Т, У; для передачи используется неравномерный двоичный код. Для девяти букв используются кодовые слова.

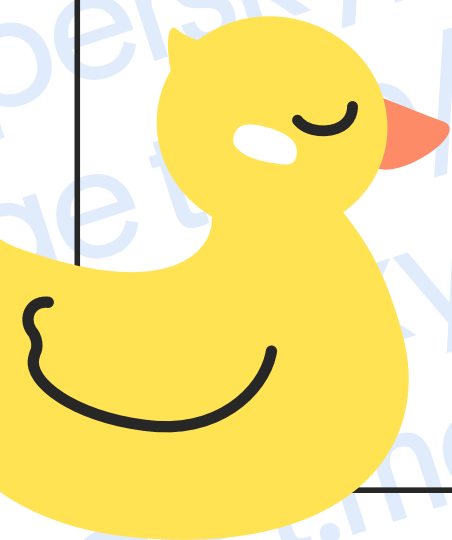
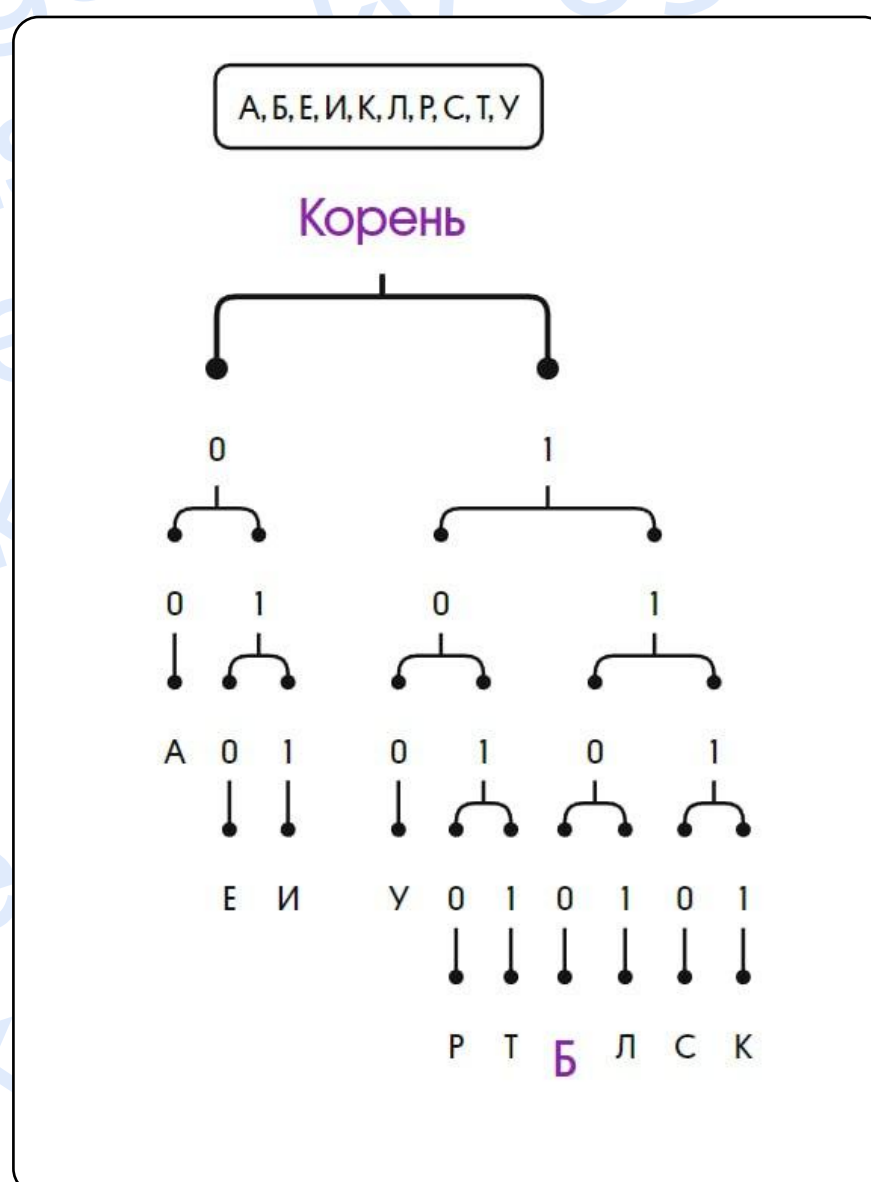
Укажите кратчайшее кодовое слово для буквы Б, при котором код будет удовлетворять условию Фано. Если таких кодов несколько, укажите код с наименьшим числовым значением.

Буква	Кодовое слово		Буква	Кодовое слово
А	00		Л	1101
Б			Р	1010
Е	010		С	1110
И	011		Т	1011
К	1111		У	100

решение

Расставим буквы на дерево. Видим, что осталась одна буква без кода и одна свободная ветка, повесим букву Б на эту ветку

ответ: 1100



трансляция с разбором



задача

На вход алгоритма подаётся натуральное число N . Алгоритм строит по нему новое число R следующим образом.

1. Строится двоичная запись числа N .
2. Далее эта запись обрабатывается по следующему правилу:
 - а) Если сумма цифр двоичной записи числа чётная, то к этой записи справа дописывается 0, и затем два левых разряда двоичной записи заменяются на 10.
 - б) Если сумма цифр двоичной записи числа нечётная, то к этой записи справа дописывается 1, а затем два левых разряда двоичной записи заменяются на 11. Полученная таким образом запись является двоичной записью искомого числа R .
3. Результат переводится в десятичную систему и выводится на экран.

Например, для исходного числа $6 = 110_2$ результатом является число $1000_2 = 8$, а для исходного числа $4 = 100_2$ результатом является число $1101_2 = 13$. Укажите максимальное число N , после обработки которого с помощью этого алгоритма получается число R , меньшее 20.

решение

python

```
def f(n):
    n = bin(n)[2:]
    if n.count('1') % 2 == 0:
        n = '10' + n[2:] + '0'
    else:
        n = '11' + n[2:] + '1'
    return int(n, 2)

res = set()
for n in range(2000):
    if f(n) < 20:
        res.add(n)
print(max(res))
```

ответ: 12

трансляция с разбором



задача

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 6 команд:

Поднять хвост, означающая переход к перемещению без рисования;

Опустить хвост, означающая переход в режим рисования;

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 направлению;

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.

Черепахе был дан для исполнения следующий алгоритм:

Повтори 9 [Вперёд 29 Направо 90 Вперёд 17 Направо 90]

Поднять хвост

Вперёд 5 Направо 90 Вперёд 1 Налево 90

Опустить хвост

Повтори 9 [Вперёд 64 Направо 90 Вперёд 48 Направо 90]

Определите периметр области пересечения фигур, ограниченных заданными алгоритмом линиями.

решение

```
from turtle import *
```

```
r = 10
```

```
tracer(0)
```

```
left(90)
```

```
pendown()
```

```
for i in range(9):
```

```
    forward(29*r)
```

```
    right(90)
```

```
    forward(17*r)
```

```
    right(90)
```

```
penup()
```

```
forward(5*r)
```

```
right(90)
```

```
forward(1*r)
```

```
left(90)
```

```
pendown()
```

```
for i in range(90):
```

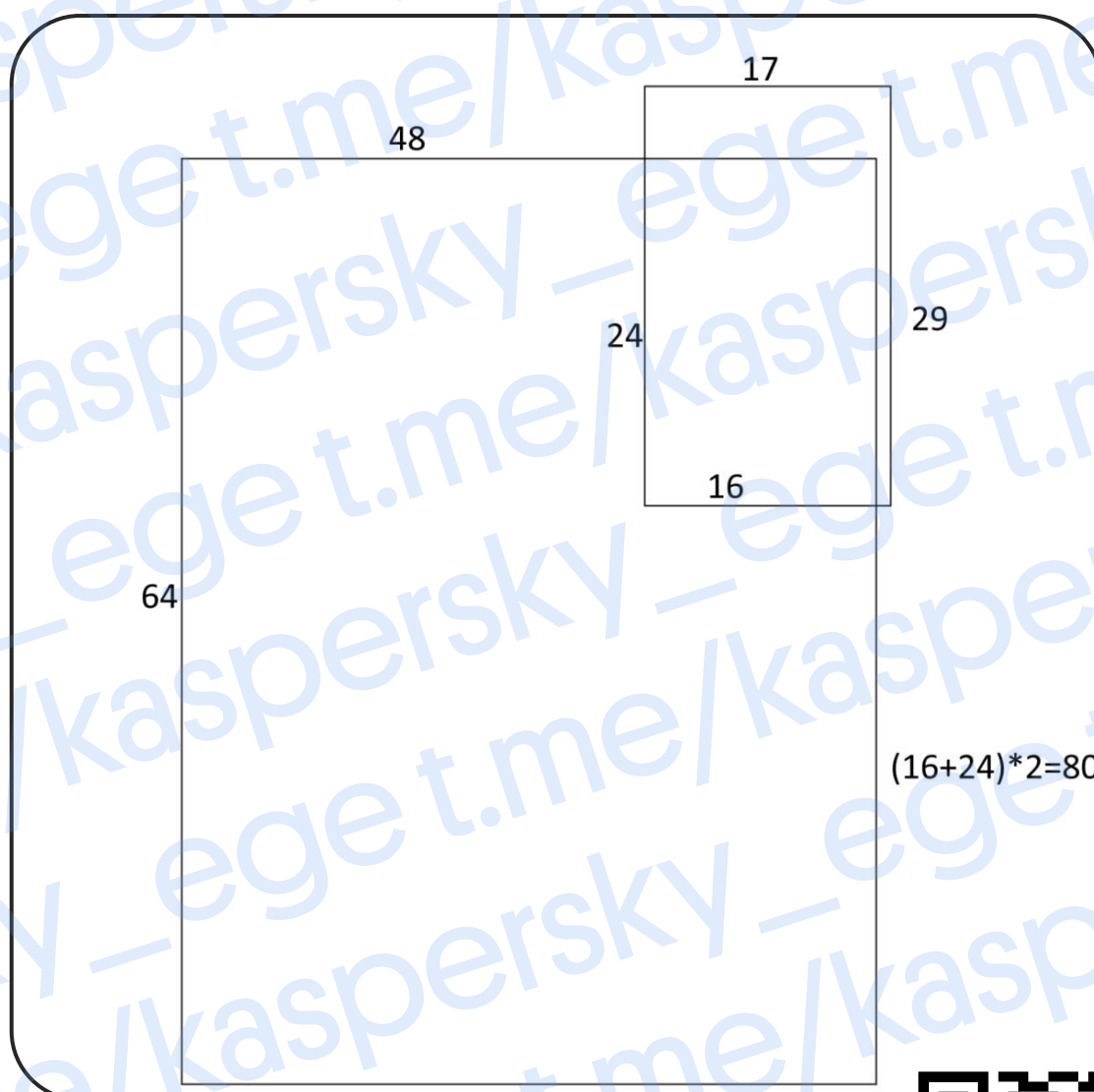
```
    forward(64*r)
```

```
    right(90)
```

```
    forward(48*r)
```

```
    right(90)
```

```
done()
```



ОТВЕТ: 80

трансляция с разбором

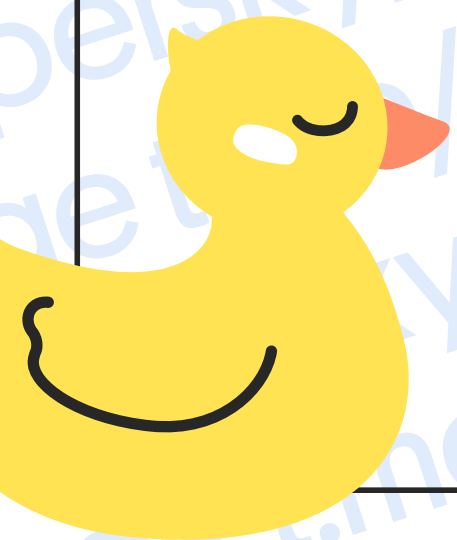
задача

Сколько секунд потребуется обычному модему, передающему сообщения со скоростью 28800 бит/с, чтобы передать цветное растровое изображение размером 1280x720 пикселей, при условии, что цвет каждого пикселя кодируется 4 байтами?

решение

$$[1280 * 720 * 4 * 8] / 28800 = 1024$$

ответ: 1024



трансляция с разбором



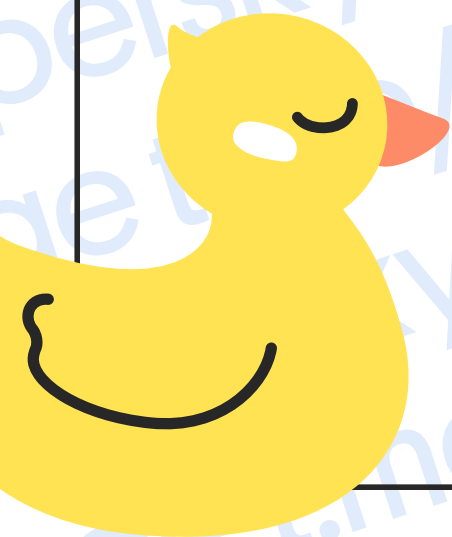
задача

Маша скачивает из Интернета альбом любимой группы, оцифрованный в формате стерео с частотой дискретизации 48000 Гц и разрешением 34 бит без использования сжатия. В альбоме 13 треков общей длительностью 42 минуты 20 секунд. Каждый трек содержит заголовок объемом 110 Кбайт. Сколько секунд будет скачиваться альбом по каналу связи со скоростью передачи данных 314 572 800 бит/с? В ответе запишите целую часть полученного числа.

решение

- 1) всего 2540 секунд
битрейт: $48000 \cdot 2 \cdot 34 = 3264000$ бит/сек
размер альбома: $3264000 \cdot 2540 = 8294960000$ бит
это 1036870000 байт
 - 2) размер одного заголовка 112640 байт
их всего: $112640 \cdot 13 = 1464320$ байт
 - 3) в итоге общий размер: $1036870000 + 1464320 = 1038334320$ байт
 - 4) скорость канала в байтах в секунду: $314572800 / 8 = 39321600$
 - 5) 1038334320 [байт] / 39321600 [байт/сек] = 26.42 сек
- Нужно взять только целую часть, берем 26.

ответ: 26



трансляция с разбором



задача

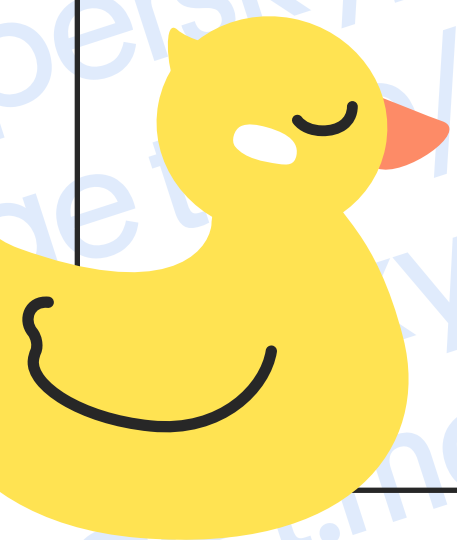
Определите количество 9-тиричных 6-тизначных чисел, которые не начинаются с нечетных цифр, не оканчиваются цифрами 2 или 3, содержат не менее двух цифр 1

решение

python

```
from itertools import *
k = 0
cif = "012345678"
for x in product(cif, repeat=6):
    a = "".join(x)
    if a[0] != "0" and int(a[0]) % 2 == 0 and a[-1] not in
        "23" and a.count("1") >= 2:
        k += 1
print(k)
```

ответ: 19868



трансляция с разбором



задача

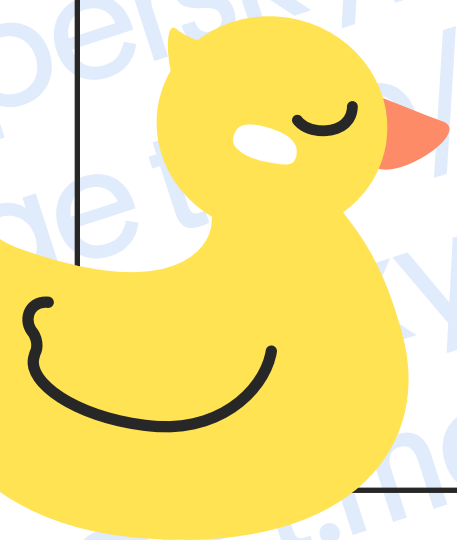
Файл электронной таблицы, в каждой строке 4 натуральных числа. Определите количество строк в таблице, содержащих числа, для которых выполнены оба условия:

- наибольшее из 4 чисел меньше суммы трёх других
- все четыре числа различны

решение

python

```
f = open('file.csv')
k = 0
for s in f:
    a = sorted([int(i) for i in s.split(';')])
    if len(set(a)) == len(a) and a[-1] < sum(a[:3]):
        k += 1
print(k)
```



трансляция с разбором



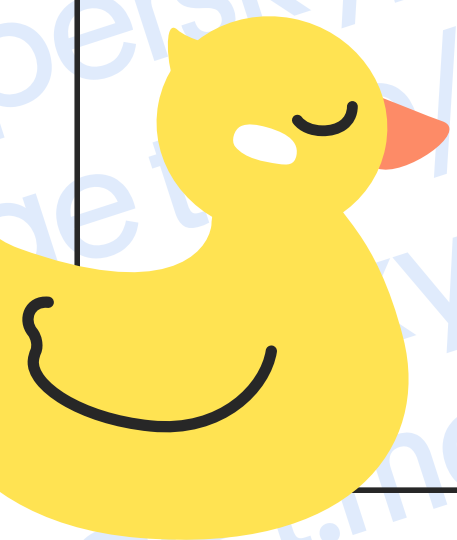
задача

С помощью текстового редактора определите, сколько раз встречается сочетание букв “рук” или “Рук” в тексте глав IV, V, VI, VII второй части тома 2 романа Л. Н. Толстого “Война и мир”. В ответе укажите только число.

решение

1. Выделяем вторую часть, из неё берем только 4-7 главы.
2. Нажимаем сочетание клавиш `ctrl + f` [windows] `cmd + f` [mac], чтобы открылась вкладка “Навигация”.
3. Вбиваем в поиск слово “рук” и получаем 12 слов.

Ответ: 12



трансляция с разбором



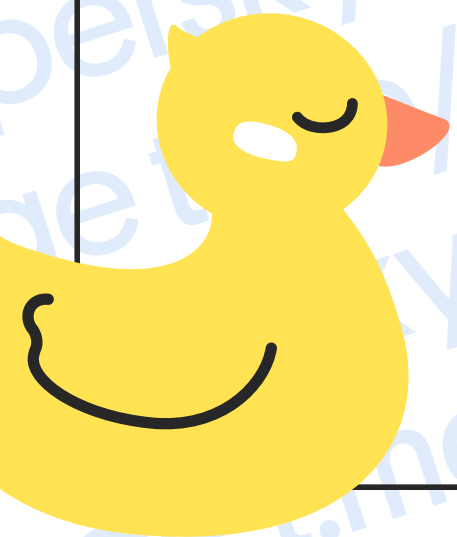
задача

В некоторой стране автомобильный номер состоит из 6 символов. В качестве символов используют 20 различных буквы и десятичные цифры в любом порядке. Каждый такой номер в компьютерной программе записывается минимально возможным и одинаковым целым количеством байтов, при этом используют посимвольное кодирование и все символы кодируются одинаковым и минимально возможным количеством битов. Определите объем памяти, отводимый этой программой для записи 60 номеров.

решение

1. мощность алфавита: $20 + 10 = 30$, значит вес одного символа = 5 бит
2. так как в номере 6 символов, объем номера $6 * 5 = 30$ бит = 4 байта.
3. значит объем памяти, отводимый для записи 60 номеров = $4 * 60 = 240$ байт

ответ: 240



трансляция с разбором



задача

На предприятии каждый изготовленной детали присваивают серийный номер, содержащий десятичные цифры, 20 латинских букв (без учёта регистра) и символы из 8164-символьного специального алфавита. В базе данных для хранения каждого серийного номера отведено одинаковое и минимально возможное число байт. При этом используется посимвольное кодирование серийных номеров, все символы кодируются одинаковым и минимально возможным числом бит. Известно, что для хранения 835 серийных номеров отведено более 156 Кбайт памяти. Определите минимально возможную длину серийного номера. В ответе запишите только целое число.

решение

Алфавит состоит из 10 (десятичные цифры) + 8164 (специальный альфавит) + 20 (латинские буквы) = 8194 символа, и так как $2^{13} < 8194 < 2^{14}$, то на один символ уйдёт 14 бит.

На один идентификатор уйдёт $\frac{14 * x}{8}$ байт.

Для хранения всех номеров необходимо $156 * 1024$ байт

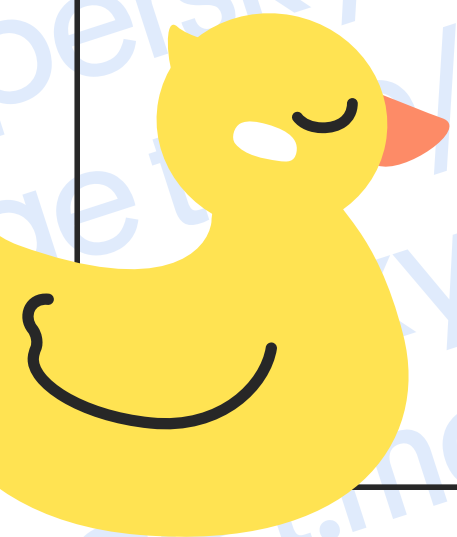
Составляем неравенство: $\frac{14 * x * 835}{8} \geq 156 * 1024$

$$x \geq \frac{156 * 1024 * 8}{14 * 835}$$

$$x \geq 109.3$$

Округляем в большую сторону - 110

Ответ: 110



трансляция с разбором



задача

Какая строка получится в результате применения приведённой ниже программы к строке состоящей из 108 идущих подряд цифр 7? В ответе запишите полученную строку

НАЧАЛО

ПОКА нашлось {33333} ИЛИ нашлось {777}

ЕСЛИ нашлось {33333}

ТО заменить {33333, 7}

ИНАЧЕ заменить {777, 3}

КОНЕЦ ЕСЛИ

КОНЕЦ ПОКА

КОНЕЦ

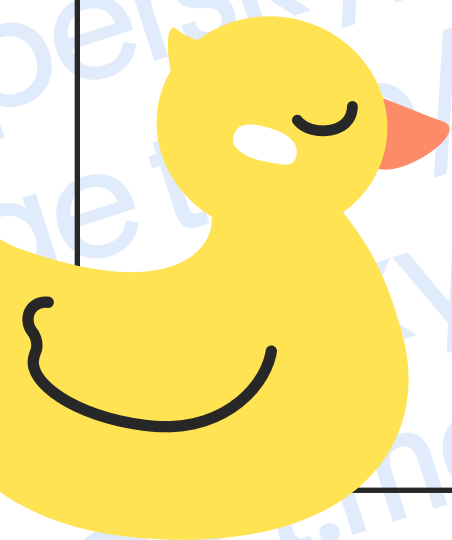
решение

```
python
s = '7' * 108

while '33333' in s or '777' in s:
    if '33333' in s:
        s = s.replace('33333', '7', 1)
    else:
        s = s.replace('777', '3', 1)

print(s)
```

ОТВЕТ: 3337



трансляция с разбором



задача

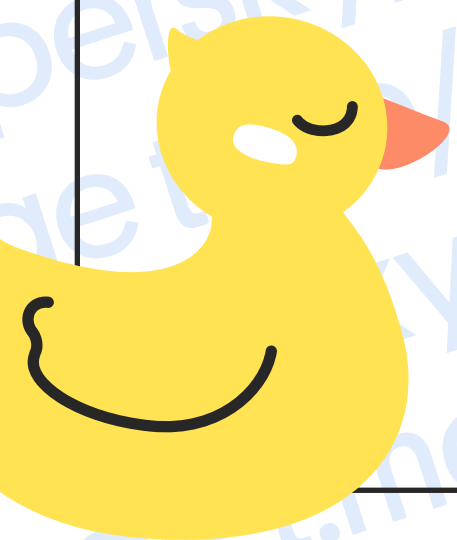
В терминологии сетей TCP/IP маской сети называют двоичное число, которое показывает, какая часть IP-адреса узла сети относится к адресу сети, а какая - к адресу узла в этой сети. Адрес сети получается в результате применения поразрядной конъюнкции к заданному адресу узла и его маске. Сеть задана IP-адресом 106.184.0.0 и сетевой маской 255.248.0.0. Сколько в этой сети IP-адресов, для которых количество единиц в двоичной записи IP-адреса не кратно 2?

решение

python

```
from ipaddress import *
k = 0
net = ip_network("106.184.0.0/255.248.0.0", 0)
for ip in net:
    if f'{ip:b}'.count("1") % 2 != 0:
        k += 1
print(k)
```

ответ: 262144



трансляция с разбором



задача

Дано арифметическое выражение

$$123xAB3_{31} + 3CEx321_{31}$$

Переменная x в каждом числе обозначает неизвестную цифру 31-ричной системы счисления. Необходимо найти наибольшее возможное значение x , при котором значение данной суммы кратно 17. В ответе необходимо указать частное от деления значения данной суммы на 17 в десятичной системе счисления. Основание системы счисления указывать не нужно.

решение

python

```
решение1
for x in range(30, -1, -1):
    first_part = int('1230AB3', 31)
    second_part = x * 31 ** 3
    a = first_part + second_part

    first_part = int('3CE0321', 31)
    second_part = x * 31 ** 3
    b = first_part + second_part

    if (a + b) % 17 == 0:
        print((a + b) // 17)
        break
```

```
решение2
for x in range(30, -1, -1):
    x1 = 3 * 31 ** 0 + 11 * 31 ** 1 + 10 * 31 ** 2 + x * 31 ** 3 +
        3 * 31 ** 4 + 2 * 31 ** 5 + 1 * 31 ** 6
    x2 = 1 * 31 ** 0 + 2 * 31 ** 1 + 3 * 31 ** 2 + x * 31 ** 3 +
        14 * 31 ** 4 + 12 * 31 ** 5 + 3 * 31 ** 6

    if (x1 + x2) % 17 == 0:
        print((x1 + x2) // 17)
        break
```

Ответ: 233409739

трансляция с разбором



задача

Значение арифметического выражения

$$3 * 289^{2024} + 81 * 49^{121} - 9 * 16^{81} - 6011$$

записали в системе счисления с основанием 31.

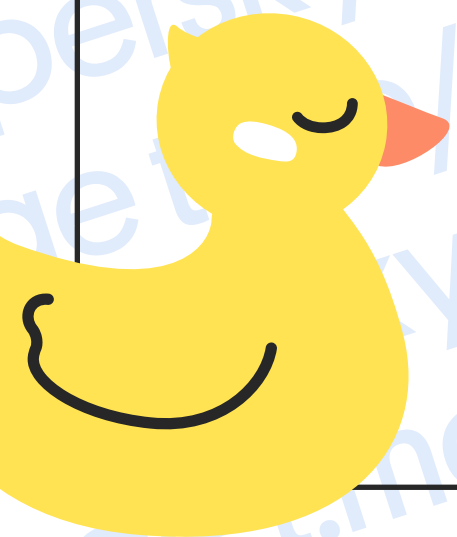
Определите сумму цифр с числовым значением, не превышающим 17, в записи этого числа.

решение

python

```
c = 3 * 289 ** 2024 + 81 * 49 ** 121 - 9 * 16 ** 81 - 6011
sumi = 0
while c > 0:
    if c % 31 <= 17:
        sumi += c % 31
    c //= 31
print(sumi)
```

ответ: 16750



трансляция с разбором



задача

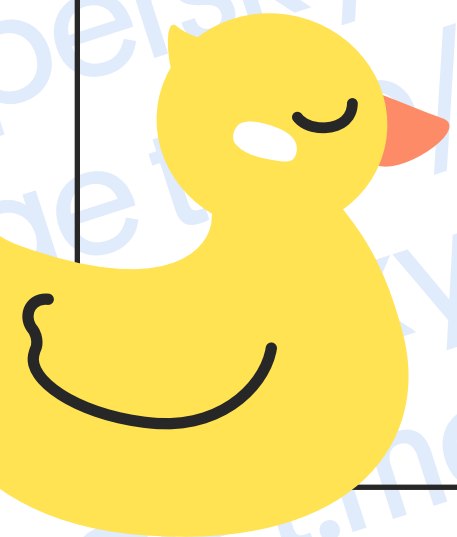
Значение арифметического выражения $3^{100} - x$, где x - целое положительное число, не превышающее 2030, записали в троичной системе счисления. Определите наименьшее значение x , при котором в троичной записи числа, являющегося значением данного арифметического выражения, содержится ровно два нуля. В ответе запишите число в десятичной системе счисления.

решение

python

```
for x in range(1, 2031):  
    c = 3 ** 100 - x  
    troich = ''  
    while c > 0:  
        troich = str(c % 3) + troich  
        c //= 3  
    if troich.count('0') == 2:  
        print(x)  
        break
```

ответ: 9



трансляция с разбором



задача

Для какого наименьшего натурального числа A логическое выражение истинно при любом целом положительном значении переменной x

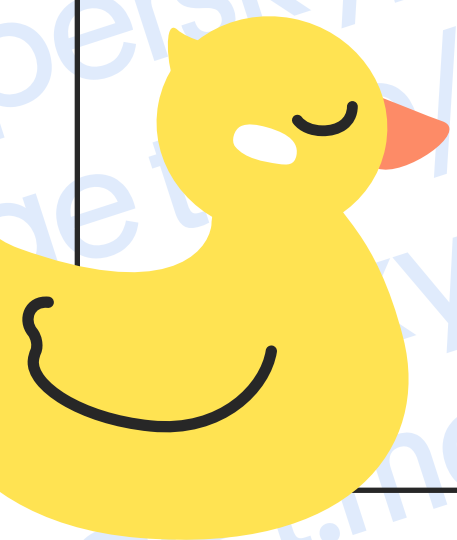
$$[\text{ДЕЛ}[x, 2] \rightarrow \neg \text{ДЕЛ}[x, 5]] \vee [x + A \geq 70]$$

решение

python

```
def f(x):  
    return ((x % 2 == 0) <= (x % 5 != 0)) or (x + a >= 70))  
  
for a in range(1, 1000):  
    if all(f(x) for x in range(1, 1000)):  
        print(a)  
        break
```

ответ: 60



трансляция с разбором



задача

Алгоритм вычисления значения функции $F[n]$, где n - натуральное число, задан следующими соотношениями:

$$F[n] = 1 \text{ при } n = 1$$

$$F[n] = 2 * n * F[n-1], \text{ если } n > 1$$

Чему равно значение выражения $(F[2024] - 4 * F[2023]) / F[2022]$?

решение

```
python

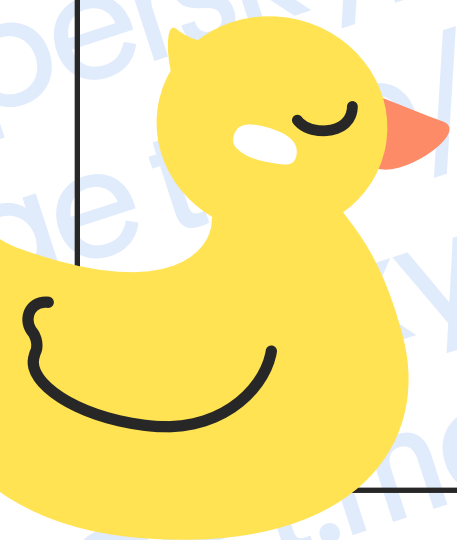
from functools import lru_cache

@lru_cache(None)
def f(n):
    if n == 1:
        return 1
    if n > 1:
        return 2 * n * f(n - 1)

for i in range(2025):
    f(i)

print((f(2024) - 4 * f(2023)) // f(2022))
```

ответ: 16362024



трансляция с разбором



задача

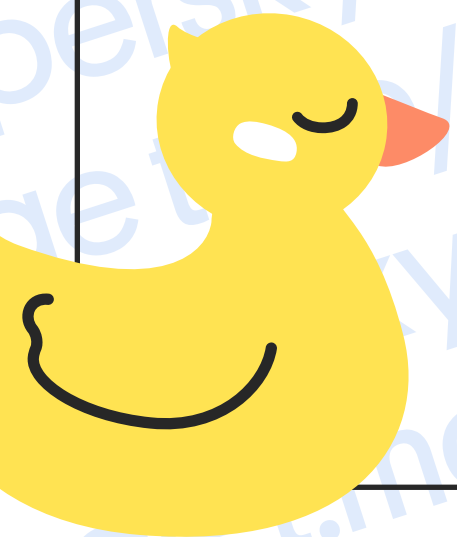
В файле содержится последовательность натуральных чисел. Все элементы могут принимать целые значения от 1 до 100 000 включительно. Определите количество пар последовательности, в которых сумма остатков от деления обоих элементов на 18 равна минимальному элементу последовательности. В ответе запишите количество найденных пар, затем максимальную из сумм элементов таких пар. В данной задаче под парой подразумевается два идущих подряд элемента последовательности.

решение

```
python

file = [int(x) for x in open('17.txt')]
min = min([int(x) for x in file])
count = 0
maxsumma = 0
for i in range(len(file)-1):
    if file[i] % 18 + file[i + 1] % 18 == min:
        count += 1
        maxsumma = max(maxsumma, file[i] + file[i + 1])
print(count, maxsumma)
```

ответ: для каждого файла разный ответ!



трансляция с разбором



задача

Одна куча камней. Первый ход делает Петя. За один ход можно добавить в кучу один камень или увеличить количество камней в куче в два раза. У каждого неограниченное количество камней. Игра завершается, когда количество камней в куче не менее 69. Победитель тот, кто сделал последний ход, т.е. получил кучу из 69 и более камней. В начальный момент в куче было S камней. $1 \leq S \leq 68$

Укажите минимальное значение S , при котором Петя не может выиграть за один ход, но при любом ходе Пети Ваня может выиграть своим первым ходом

решение

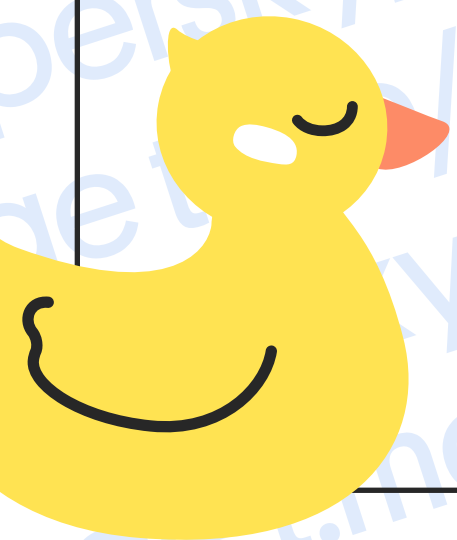
python

```
from functools import lru_cache

def step(p):
    return p + 1, p * 2

@lru_cache(None)
def game(p):
    if p >= 69: return 0
    if any(game(p1) == 0 for p1 in step(p)): return 1
    if all(game(p2) == 1 for p2 in step(p)): return 2

for p in range(1, 68):
    if game(p) == 2:
        print(p)
        break
```



трансляция с разбором



задача

Найдите два наименьших значения S , при которых у Пети есть выигрышная стратегия, причём одновременно выполняются два условия:

- Петя не может выиграть за один ход
 - Петя может выиграть своим вторым ходом независимо от того, как будет ходить Ваня
- Значения в порядке возрастания

решение

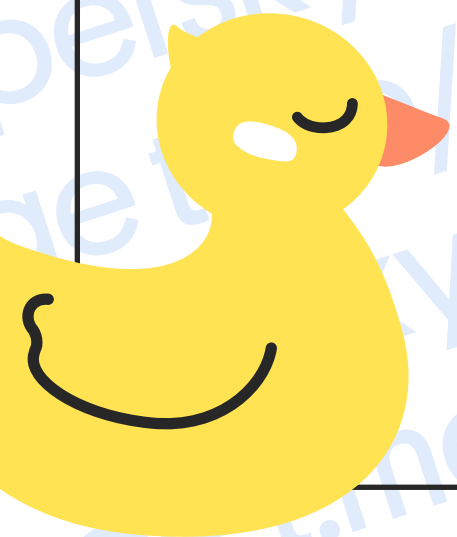
python

```
from functools import lru_cache

def step(p):
    return p + 1, p * 2

@lru_cache(None)
def game(p):
    if p >= 69: return 0
    if any(game(p1) == 0 for p1 in step(p)): return 1
    if all(game(p2) == 1 for p2 in step(p)): return 2
    if any(game(p3) == 2 for p3 in step(p)): return 3

for p in range(1, 68):
    if game(p) == 3:
        print(p)
```



трансляция с разбором



задача

Найдите 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 есть стратегия, которая позволяет ему гарантированно выиграть первым ходом

Если найдено несколько значений S , в ответе запишите наименьшее из них

решение

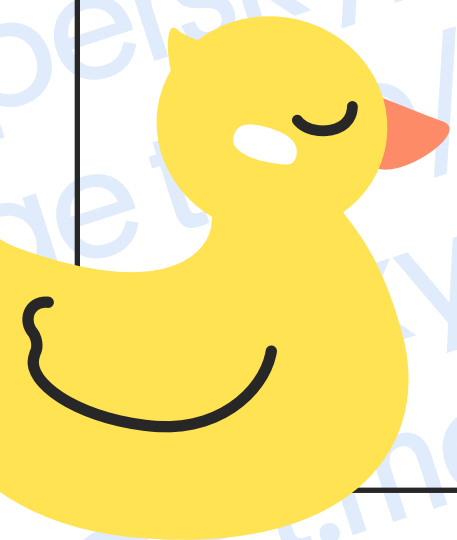
python

```
from functools import lru_cache

def step(p):
    return p + 1, p * 2

@lru_cache(None)
def game(p):
    if p >= 69: return 0
    if any(game(p1) == 0 for p1 in step(p)): return 1
    if all(game(p2) == 1 for p2 in step(p)): return 2
    if any(game(p3) == 2 for p3 in step(p)): return 3
    if all(game(p4) in (1, 3) for p4 in step(p)): return 4

for p in range(1, 68):
    if game(p) == 4:
        print(p)
```



трансляция с разбором



задача

В файле содержится информация о совокупности N вычислительных процессов, которые могут выполняться параллельно или последовательно.

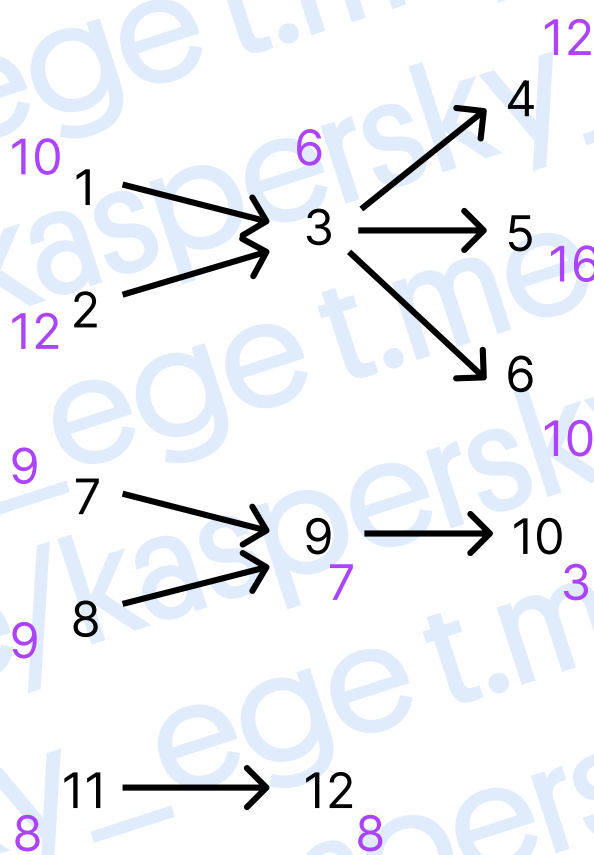
Будем говорить, что процесс B зависит от процесса A , если для выполнения процесса B необходимы результаты выполнения процесса A . В этом случае процессы A и B могут выполняться только последовательно.

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 независим, то в таблице указано значение 0.

Типовой пример организации данных в файле

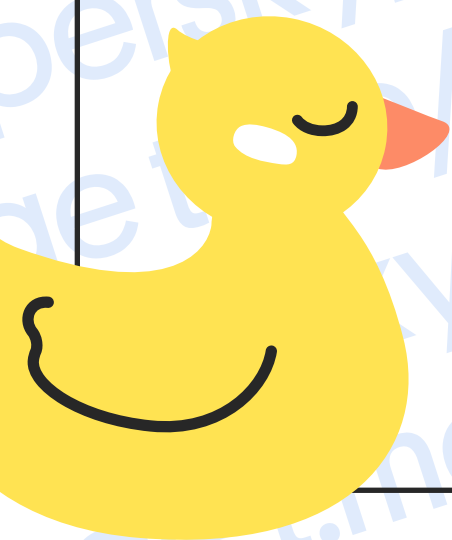
Определите максимальное количество миллисекунд, когда одновременно может выполняться максимальное количество процессов.

решение



1. Нарисуем граф с взаимосвязями всех процессов и подпишем время выполнения каждого
2. Процессы 11 и 12 объединяем как последовательные, их общее время выполнения $8 + 8 = 16$
3. Из первой ветки берем процессы 4-6, из второй 7-8, из третьей 11-12, чтобы получить максимальное число процессов
4. Берем минимальное время выполнения из всех процессов, т.е. их общее время работы - 9

ответ: 9



трансляция с разбором



задача

Исполнитель преобразует число на экране.

У исполнителя есть две команды, которые обозначены латинскими буквами:

A. Вычти 2

B. Найди целую часть от деления на 2

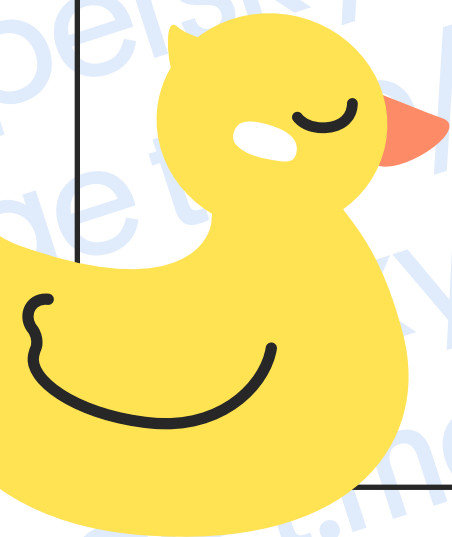
Сколько существует программ, для которых при исходном числе 30 результатом является число 1, и при этом траектория вычислений содержит число 14?

решение

python

```
def f(st,en):  
    if st == en:  
        return 1  
    elif st < en:  
        return 0  
    else:  
        return f(st-2, en) + f(st//2, en)  
  
print(f(30,14)*f(14,1))
```

ответ: 36



трансляция с разбором



задача

Исполнитель преобразует число на экране.

У исполнителя есть три команды, которым присвоены номера:

А. Вычти 1

В. Вычти 2

С. Найди целую часть от деления на 3

Программа для исполнителя – это последовательность команд.

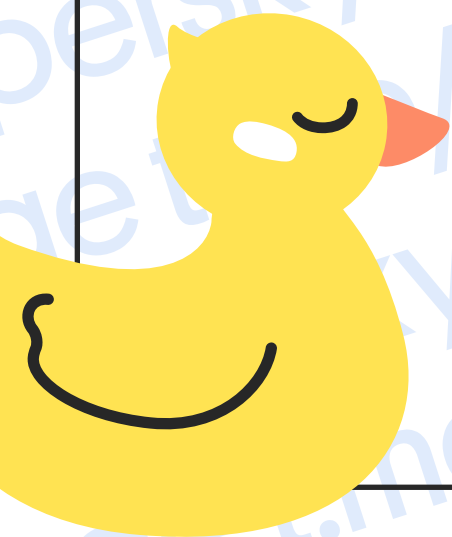
Сколько существует программ, для которых при исходном числе 16 результатом является число 6, при этом траектория вычислений содержит число 11?

решение

python

```
def f(a, b):  
    if a == b:  
        return 1  
    elif a < b:  
        return 0  
    else:  
        return f(a - 1, b) + f(a - 2, b) + f(a // 3, b)  
  
print(f(16, 11) * f(11, 6))
```

ответ: 64



трансляция с разбором



задача

Текстовый файл состоит из заглавных букв латинского алфавита А, В, С, D, Е и F. Определите максимальное количество идущих подряд символов в прилагаемом файле, среди которых пара символов CD (в указанном порядке) встречается не более 140 раз.

Для выполнения этого задания следует написать программу.

решение

```
python

f = open('24.txt').readline()
f = f.replace('CD', '1')
res = 0
l = 0
am = 0
for r in range(len(f)):
    if f[r] == '1':
        am += 1
    while am > 140:
        if f[l] == '1':
            am -= 1
        l += 1
    if am == 140:
        if r - l + 1 > res:
            res = max(res, r - l + 1)
            resr = r
            resl = l
            print(l, r)
res += 140

if resl > 0:
    res += 1
if resr < len(f) - 1:
    res += 1
print(res)
```

ответ: для разных файлов разный ответ!

трансляция с разбором



задача

Напишите программу, которая перебирает целые числа, большие 600000 в порядке возрастания и ищет среди них такие, у которых есть натуральный делитель оканчивающийся на цифру 7 и не равный ни самому числу, ни числу 7. В ответе запишите в первом столбце таблицы первые пять найденных чисел в порядке возрастания, а во втором - наименьший делитель для каждого из них

решение

python

```
def f(x):  
    a = set()  
    for i in range(2, int(x**0.5) + 1):  
        if x % i == 0:  
            if (i != 7 and i % 10 == 7):  
                a.add(i)  
            if ((x // i) != 7 and (x // i) % 10 == 7):  
                a.add(x // i)  
    return sorted(list(a))  
  
counter = 0  
for i in range(600001, 10**20):  
    if len(f(i)) != 0:  
        print(i, f(i)[0])  
        counter += 1  
    if counter == 5:  
        break
```

ОТВЕТ:

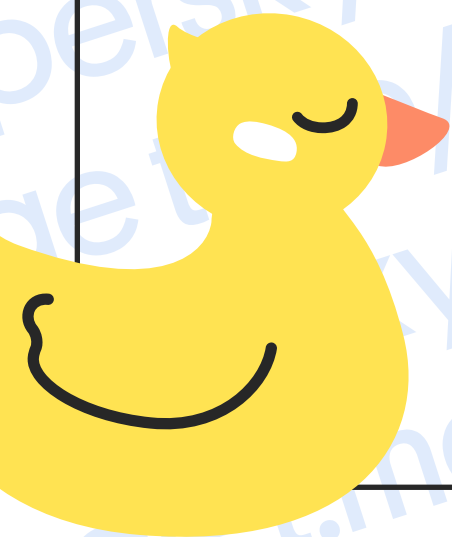
600001437

60000247

6000031227

600005217

60001216667



трансляция с разбором



задача

Известно какие места в зале заняты. Необходимо купить билет на такое место в ряду, чтобы перед ним как можно больше идущих подряд кресел с таким же номером было свободно. Если места удовлетворяющие этому условию есть в нескольких рядах, то нужно выбрать ряд, расположенный как можно ближе к сцене. В ответе запишите два целых числа, номер ряда и количество свободных кресел перед выбранным местом. Нумерация рядов и мест ведётся с 1. Гарантируется, что хотя бы одно такое место в зале есть

В первой строке три числа: N - количество занятых мест в зале (целое положительное не больше 10000), M - количество рядов (целое положительное не больше 100000), K - количество мест в каждом ряду (целое положительное не больше 100000). В следующих N строках находятся пары натуральных чисел: номер ряда и номер места занятого кресла соответственно (первое число не превышает M, второе - K)

решение

python

```
f = open('26.txt')
ryad_mesto = {}

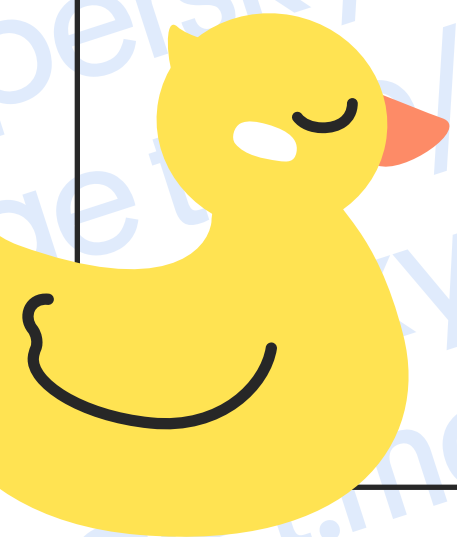
n = int(f.readline()) # Считываем N - количество занятых мест
# Затем заполняем словарь
for i in range(n):
    r, m = map(int, f.readline().split()) # считываем ряд и место в две переменные
    if m in ryad_mesto: # Если ключ с этим местом уже существует
        ryad_mesto[m].append(r) # Добавляем в список новый ряд
    else:
        ryad_mesto[m] = [r] # Если нет, создаем новый ключ с новым местом

nom_zan = 0
count_svob = 0

# Затем проходим по ключам (местам) в порядке возрастания
# И будем считать максимальное количество свободных
for m in sorted(ryad_mesto.keys()):
    ryad_mesto[m].sort() #
    mesto = ryad_mesto[m] #

    for i in range(len(mesto) - 1):
        if mesto[i+1] - mesto[i] - 1 > count_svob:
            nom_zan = mesto[i] - 1
            count_svob = mesto[i+1] - mesto[i] - 1

print(nom_zan, count_svob)
```



трансляция с разбором



задача

Пусть S - последовательность из N целых чисел, пронумерованных подряд, начиная с 1. Обозначим S_i, S_j, S_k как три элемента последовательности S , где $i < j < k$. Определите в последовательности S три таких члена S_i, S_j, S_k , что $S_i > S_j, S_k > S_j$, и значение выражения $[S_i - S_j] + [S_k - S_j]$ максимально.

Гарантируется, что в последовательности есть три числа S_i, S_j, S_k , удовлетворяющие условию задачи.

Входные данные

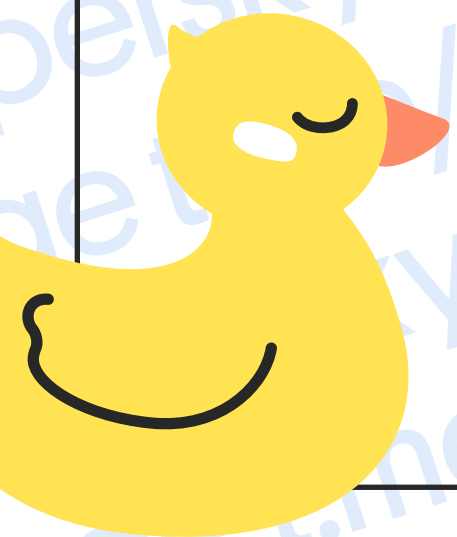
Дано два входных файла (файл А и файл В), каждый из которых в первой строке содержит число N ($5 < N < 10\,000\,000$) - количество целых чисел. Каждая из следующих N строк содержит одно целое число, не превышающее по модулю 1000.

В ответ укажите два числа: сначала значение искомой величины для файла А, затем - для файла В.

решение для файла А

python

```
f = open('27a.txt')
n = int(f.readline())
a = [int(x) for x in f]
res = 0
for i in range(n):
    for j in range(i + 1, n):
        for k in range(j + 1, n):
            s1 = a[i]
            s2 = a[j]
            s3 = a[k]
            if s1 > s2 < s3:
                res = max(res, (s1 - s2) + (s3 - s2))
print(res)
```



трансляция с разбором



задача

Пусть S - последовательность из N целых чисел, пронумерованных подряд, начиная с 1. Обозначим S_i, S_j, S_k как три элемента последовательности S , где $i < j < k$. Определите в последовательности S три таких члена S_i, S_j, S_k , что $S_i > S_j, S_k > S_j$, и значение выражения $(S_i - S_j) + (S_k - S_j)$ максимально. Гарантируется, что в последовательности есть три числа S_i, S_j, S_k , удовлетворяющие условию задачи.

Входные данные

Дано два входных файла (файл А и файл В), каждый из которых в первой строке содержит число N ($5 < N < 10\,000\,000$) - количество целых чисел. Каждая из следующих N строк содержит одно целое число, не превышающее по модулю 1000.

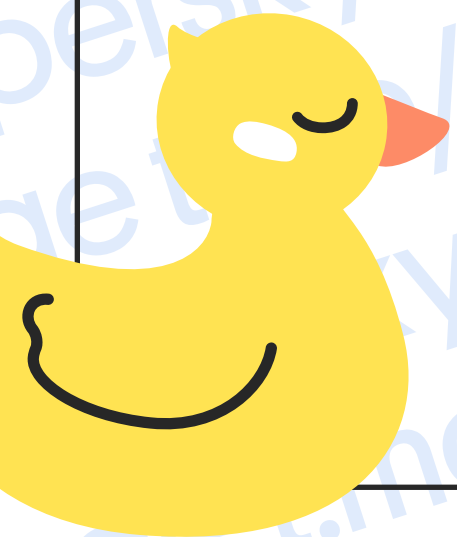
В ответ укажите два числа: сначала значение искомой величины для файла А, затем - для файла В.

решение для файлов А и В

```
python

f = open('27.txt')
n = int(f.readline())
a = [int(x) for x in f]
res = 0
for i in range(n):
    for j in range(i + 1, n):
        for k in range(j + 1, n):
            s1 = a[i]
            s2 = a[j]
            s3 = a[k]
            if s1 > s2 < s3:
                res = max(res, (s1 - s2) + (s3 - s2))

print(res)
res = 0
m1 = m2 = m3 = r1 = 0
for i in range(n - 2):
    m1 = max(a[i], m1)
    if m1 - a[i + 1] > r1:
        r1 = m1 - a[i + 1]
        m2 = a[i + 1]
    m3 = a[i + 2]
    res = max(res, m3 - m2 + r1)
print(res)
```



трансляция с разбором

