

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 (01) ИЛИ нашлось (02) ИЛИ нашлось (03)

заменить (01, 2302)

заменить (02, 10)

заменить (03, 201)

КОНЕЦ ПОКА

КОНЕЦ

Известно, что исходная строка начиналась с нуля, а далее содержала только

единицы, двойки и тройки. После выполнения данной программы получилась

строка, содержащая 40 единиц, 10 двоек и 8 троек.

Сколько единиц было в исходной строке?

20	<b>6</b>	
----	----------	--

01-2302-2310

1)40

02-10

2)10

03-201-22302-22310

3)8

## полный перебор (неправильно понято задание)

```
"""mas = ['a', 'a', 'b', 'b']
print(''.join(mas)) """
```

```
def proverka (mas):
    str_mas = ''.join (map (str, mas))
    while '01' in str_mas or '02' in str_mas or '03' in str_mas:
        str_mas = str_mas.replace ('01', '2302', 1)
        str_mas = str_mas.replace ('02', '10', 1)
        str_mas = str_mas.replace ('03', '201', 1)
    c1 = str_mas.count ('1')
    c2 = str_mas.count ('2')
    c3 = str_mas.count ('3')
    if c1 > 30:
        # print(str_mas)
        print (c1, c2, c3)
    if c1 == 40 and c2 == 10 and c3 == 8:
        return 1
    else:
        return 0
```

```
mas = [0]
count = 0
while count < 100:
    for k in range (1, len (mas)):
        mas [k] = 1
    mas.append (1)
    # print (mas)
    if proverka (mas) == 1:
        print ('mas = ', mas)
        exit ()
    i = len (mas) - 1
    while i > 0:
        if mas [i] < 3:
            mas [i] += 1
            j = i + 1
            while j < len (mas):
                mas [j] = 1
                j += 1
            # print (mas)
            if proverka (mas) == 1:
                print ('mas = ', mas)
                exit ()
            i = len (mas) - 1
            continue
        i -= 1
    count += 1
```

## реалистичное решение

```
def proverka(mas):
    str_mas = ''.join(map(str, mas))
    while '01' in str_mas or '02' in str_mas or '03' in str_mas:
        str_mas = str_mas.replace('01', '2302', 1)
        str_mas = str_mas.replace('02', '10', 1)
        str_mas = str_mas.replace('03', '201', 1)
    c1 = str_mas.count('1')
    c2 = str_mas.count('2')
    c3 = str_mas.count('3')
    print(c1, c2, c3)
    if c1 == 40 and c2 == 10 and c3 == 8:
        return 1
    else:
        return 0
```

```
mas = [0]
countt = 0
```

```
for a in range(1,100):
    for b in range(1,100):
        for c in range(1,100):
            #print(a,b,c)
            mas = [0]
            for aa in range(a):
                mas.append(1)
            for bb in range(b):
                mas.append(2)
            for cc in range(c):
                mas.append(3)
            #print(mas)

            if proverka(mas) == 1:
                print('mas = ', mas)
                exit()
```

Дана программа для редактора:

НАЧАЛО

ПОКА нашлось (01) ИЛИ нашлось (02) ИЛИ нашлось (03)

заменить (01, 30)

заменить (02, 101)

заменить (03, 202)

КОНЕЦ ПОКА

КОНЕЦ

Известно, что исходная строка начиналась с нуля, а далее содержала только единицы, двойки и тройки. После выполнения данной программы получилась строка, содержащая 15 единиц, 10 двоек и 60 троек. Сколько единиц было в исходной строке?

**Решение.**

Заметим, что порядок цифр в исходной строке не важен. Из строки «01» получается строка «30». Строка «02» сначала преобразуется в строку «101», а затем получается строка «130». Строка «03» сначала преобразуется в строку «202», затем преобразуется в строку «2101», после чего получается строка «2130».

Заметим, что после преобразования исходной строки получилась строка, содержащая 10 двоек. Значит, в исходной строке должно содержаться 10 троек, поскольку только строка «03» после всех преобразований в получившейся подстроке содержит двойку. Таким образом, если исходная строка начинается с нуля, а далее содержит 10 троек, то получившаяся строка будет содержать 10 двоек, 10 единиц и 10 троек.

Поскольку строка после выполнения также должна содержать 15 единиц, в исходной строке, помимо 10 троек, должно содержаться ещё 5 двоек, поскольку строка «02» преобразуется в строку «130».

Тогда, чтобы получившаяся строка содержала 60 троек, она также должна содержать ещё 45 единиц, поскольку строка «01» преобразуется в строку «30».

Ответ: 45.

## официальное решение

for a in range(100):

for b in range(100):

for c in range(100):

s='0'+a\*1'+b\*2'+c\*3'

while '01' in s or '02' in s or '03' in s:

s=s.replace('01','2302',1)

s=s.replace('02','10',1)

s=s.replace('03','201',1)

if s.count('1')==40 and s.count('2')==10 and s.count('3')==8:

print(a)