

# Перебор последовательности целых чисел Работа с файлами (новая)

**Автор Стрельникова Л.В.**

**17** (повышенный уровень, время – 14 мин)

# Что проверяется?

Умение создавать собственные программы (20–40 строк) для обработки целочисленной информации и работы с файлами.

1. Основные конструкции языка программирования.
2. Система программирования.
3. Умение работать с файлами

## Что нужно знать:

- в известных задачах этого типа (не олимпиадных) нет ограничения на время выполнения, по крайней мере, оно несущественно для отрезков, заданных для перебора; поэтому можно использовать простой перебор без оптимизации;
- задачи этого типа предлагается решать с помощью электронных таблиц или собственной программы; как правило, написать правильную программу значительно проще
- пусть необходимо перебрать все целые числа на отрезке  $[a; b]$  и подсчитать, для скольких из них выполняется некоторое условие; общая структура цикла перебора записывается так (Python):

# Краткая теория по теме: «Файлы»

## Что нужно знать:

1. Открыть файл, содержимое его поместить в переменную - **f**

```
f = open ('имя файла. расширение файла')
```

```
f = open ('17-4. txt')
```

2. Текстовый файл чисел превратить в массив целых чисел

```
a = [int(i) for i in f]
```

3. Можно объединить две строки, получим:

```
a = [int(i) for i in open ('17-4. txt')]
```

# Общая структура цикла перебора

## Питон

```
count = 0
```

```
for n in range (a, b+1):
```

```
    if условие выполнено:
```

```
        count += 1
```

```
print( count )
```

## Паскаль

```
count = 0
```

```
for n:=a to b do
```

```
    if условие выполнено then
```

```
        count:= count + 1
```

```
writeln( count )
```

## Общая структура цикла перебора

Пусть необходимо перебрать все целые числа данного массива и подсчитать, для скольких из них выполняется некоторое условие; общая структура цикла перебора записывается так (Python):

```
Питон  
count = 0  
for n in range (len(a) - 1):  
    if условие выполнено:  
        count +=1  
print (count)
```

$f = [2, 5, 4, 8, 7, 1, 3, 4] ?$

**(len(a) - 1)**

## Общая структура вывода пар

4. Понимание пары чисел можно посмотреть на примере.

$f = [2, 5, 4, 8, 7, 1, 3, 4] ?$

```
f = open ('17-4.txt')
a = [int (i) for i in f]
for i in range (len (a) -1):
    print (a[i], a[i+1])
```

```
5437 3806
3806 7736
7736 5435
5435 7507
7507 7650
7650 5675
5675 9511
9511 2410
2410 8701
8701 8955
8955 9612
9612 6227
6227 3378
3378 1671
1671 3231
3231 2531
2531 1817
1817 6392
```

## Значение выражения: `float("inf")`

`float("inf")` создает специальное представление бесконечности (положительной), которое используется в числовых расчетах. Это значение больше любого другого числа, что позволяет удобно обнаруживать минимальное значение в наборе данных.

```
mn = float("inf")
```

Когда надо найти минимум из набора значений, можно использовать переменную `c float("inf")`, чтобы любое первое сравнение с ней гарантированно окажется меньше этого значения. Таким образом, это значение будет заменено реальным минимальным значением при первом сравнении.

## Пример использования

# Инициализация списка чисел

```
A = [34, 67, 23, 89, 12, 76]
```

# Инициализация переменной **min3** значением бесконечности

```
mn = float("inf")
```

# Проходим по всем числам в списке

```
for k in A:
```

# Если текущее число меньше значения **mn**

```
    if k < mn:
```

```
        mn = k
```

# Результат

```
print ("мин = ", mn)
```

**`mn = float("inf")`**

**служит удобным начальным  
значением, позволяющим корректно  
находить минимальные значения  
среди других чисел  
а выражение: `mx = float("-inf")` для  
максимальных значений**



# Часть 1

## Решение задач с использованием файла (средней сложности)

Автор Стрельникова Л.В.

## Задача №1 из решу ЕГЭ

- В файле содержится последовательность целых чисел. Элементы последовательности могут принимать целые значения от **-10 000** до **10 000** включительно.
- **Определите и запишите в ответе сначала**
  - 1. количество пар элементов последовательности, в которых хотя бы одно число делится на 3,**
  - 2. затем максимальную из сумм элементов таких пар.**

В данной задаче под парой подразумевается два идущих подряд элемента последовательности.

**Например**, для последовательности из пяти элементов: 6; 2; 9; -3; 6 — **4** - число пар, **11** максимальная сумма пары

## Задача №1 из решу ЕГЭ

```
n = 0
mx = -10001
f = open('17.txt')
a = [int(i) for i in f]
for i in range(len(a) - 1):
    if (a[i]%3 == 0) or (a[i + 1]%3 == 0):
        n += 1
        mx = max(mx, a[i] + a[i + 1])
print(n, mx)
```

## Задача №1 из решу ЕГЭ

```
n = 0; mx = float('-inf')
a = [int(i) for i in open('17.txt')]
for i in range(len(a)-1):
    if (a[i]%3 == 0) or (a[i+1]%3 == 0):
        n+= 1
        mx = max(mx, (a[i]+a[i+1]))
print(n, mx)
```

## Задача №2 с файлом 17-1

В файле 17-1.txt содержится последовательность целых чисел. Элементы последовательности могут принимать целые значения от -10 000 до 10 000 включительно.

**Определите и запишите в ответе сначала**

- 1. количество пар элементов последовательности, в которых хотя бы одно число делится на 7, а другое при этом не делится на 17.**
- 2. Затем - минимальную из сумм элементов таких пар.**

В данной задаче под парой подразумевается два идущих подряд элемента последовательности. Например, для последовательности **-45; 14; 22; -21; 34** ответом будет пара чисел: **3** и **-31**.

## Решение задачи №2 с файлом 17-1

```
a = [int(i) for i in open('17-1.txt')]
mn = float('inf'); k = 0
for i in range(len(a)-1):
    if ((a[i]%7==0) and (a[i+1]%17!=0))\
        or ((a[i+1]%7==0) and (a[i]%17!=0)):
        k+=1
        mn = min(mn, (a[i]+ a[i + 1]))
print(k, mn)
```

2510 -19677

## Задача №3 с файлом 17

- ❖ В файле содержится последовательность целых чисел. Элементы последовательности могут принимать целые значения от  $-10\,000$  до  $10\,000$  включительно.
- ❖ **Определите и запишите в ответе сначала количество пар элементов последовательности, в которых хотя бы одно число делится на 5, затем максимальную из сумм элементов таких пар.**
- ❖ В данной задаче под парой подразумевается два идущих подряд элемента последовательности. Например, для последовательности из пяти элементов: 6; 2; 9; -3; 6 — ответ 4 11.

## Решение задача №3

```
n = 0;  mx = float("-inf")
a = [int(i) for i in open('17.txt')]
for i in range(len(a) - 1):
    if (a[i]%5==0) or (a[i+1]%5==0):
        n+=1
        mx= max(mx, a[i]+a[i+1])
print(n, mx)                                1765 1990
```

## Задача №4 с файлом 17-4

В файле 17-4.txt содержится последовательность целых чисел. Элементы последовательности могут принимать целые значения от **0 до 10 000** включительно.

**Определите количество пар, в которых**

- оба элемента меньше, чем среднее арифметическое всех чисел в файле,**
- и хотя бы один из двух элементов оканчивается на 9.**

В ответе запишите два числа:

- сначала количество найденных пар,**
- а затем – максимальную сумму элементов таких пар.**

В данной задаче под парой подразумевается два идущих подряд элемента последовательности.

```
# меньше среднего арифметического, и оканчивается на 9
f=open('17-4.txt')
# содержимое файла переводим в числовой массив
a = [int(i) for i in f]
# находим среднее арифметическое элементов массива
sr=0; sr=sum(a)/len(a)
print('sr=', sr)
# Ищем max сумму пары
k = mx = 0
for i in range(len(a)-1):
    if ((a[i]<sr) and (a[i+1]<sr)) and ((a[i] % 10 == 9) or (a[i+1] % 10 == 9)):
        k+=1
        if a[i]+a[i+1] > mx:
            mx = a[i] + a[i+1]
print(k, mx)
```

```
sr= 5442.9905
86 10184
```

## Задача №5 с файлом 17-4

В файле 17-4.txt содержится последовательность целых чисел.

Элементы последовательности могут принимать целые значения от 0 до 10 000 включительно.

Определите количество пар, в которых:

1. хотя бы один из двух элементов меньше, чем среднее арифметическое всех чисел в файле,
2. и десятичная запись хотя бы одного из двух элементов не содержит цифру 5.

В ответе запишите **два числа**: сначала

1. количество найденных пар,
2. затем – минимальную сумму элементов таких пар.

В данной задаче под парой подразумевается два идущих подряд элемента последовательности.

## Содержит цифру 5

```
f=open('17-4.txt')
a= [int(i) for i in f]
sr=sum(a)/len(a)
c = 0
mn = 10000000000
for i in range(len(a)-1):
    if (a[i]<sr) and (a[i+1]<sr) and (('5' in str (a[i]))
    or ('5' in str(a[i+1] ))):
        c+=1
        if a[i]+a[i+1] < mn:
            mn=a[i]+a[i+1]
print(c, mn)
```

424 2378

## Не содержит цифру 5

```
a = [int(i) for i in open('17-4.txt')]
sr = sum(a)/len(a)
k = 0; mn = float('inf')
for i in range(len(a)-1):
    if ((a[i]<sr) and (a[i+1]<sr)) and ((not('5'
in str(a[i]))) or (not('5' in str(a[i+1])))):
        k+=1
        if a[i]+a[i+1] < mn: mn=a[i]+a[i+1]
print(k, mn)
```

424 2378

## Задача №6 с файлом 17-4

В файле 17-4.txt содержится последовательность целых чисел.

Элементы последовательности могут принимать целые значения от 0 до 10 000 включительно.

Определите количество пар, в которых

1. хотя бы один из двух элементов больше, чем среднее арифметическое всех чисел в файле,
2. а их сумма оканчивается на 9.

В ответе запишите **два числа**:

1. сначала количество найденных пар, а
2. затем – минимальную сумму элементов таких пар.

В данной задаче под парой подразумевается два идущих подряд элемента последовательности.

# Решение задача №6 файлом 17-4

```
a = [int(i) for i in open('17-4.txt')]
sr=sum(a)/len(a)
# ищем max число в массиве
k = 0; mn = float('inf')
for i in range(len(a)-1):
    if ((a[i]>sr) or (a[i+1]>sr)) and ((a[i] + a[i+1]) % 10 == 9):
        k+=1
        if a[i]+a[i+1] < mn: mn=a[i]+a[i+1]
print(k, mn)
```

**1065 6491**

# Решение задача №6 файлом 17-2

```
a = [int(i) for i in open('17-2.txt')]
sr=0 ; sr=sum(a)/len(a)
print('sr=',sr)
# ищем max число в массиве
k = 0; mn = float('inf')
for i in range(len(a)-1):
    if ((a[i]>sr) or (a[i+1]>sr)) and ((a[i] + a[i+1]) % 10 == 9):
        k+=1
        if a[i]+a[i+1] < mn: mn=a[i]+a[i+1]
print(k, mn)
```

**3737 6569**

## Задача №7 с файлом 17-4

(А. Кабанов) В файле [17-4.txt](#) содержится последовательность целых чисел. Элементы последовательности могут принимать целые значения от 0 до 10 000 включительно. Рассматривается множество элементов последовательности, которые удовлетворяют следующим условиям:

- кратны 3, но не кратны 9;
- последняя цифра не менее 4.

Найдите количество таких чисел и целую часть их среднего арифметического.



```
a = [int(i) for i in open('17-4.txt')]
res=[]
for k in a:
    if k%3==0 and k%9!=0 and k%10>=4:
        res.append(k)
print(len(res), sum(res)//len(res))
```

```
s = [int(i) for i in open('17-4.txt')]
res = []
for c in s:
    if c % 3 == 0 and c % 9 != 0 and c % 10 >= 4:
        res.append(c)
print(len(res), sum(res)//len(res) )
```

247 5706



## Часть 2

# Решение задач с ИСПОЛЬЗОВАНИЕМ файла (повышенной сложности)

**Автор Стрельникова Л.В.**

## Задача №2 из решу ЕГЭ

В файле содержится последовательность из 10 000 натуральных чисел. Каждое число не превышает 10 000.

Определите и запишите в ответе:

1. сначала количество пар элементов последовательности, у которых различные остатки от деления на  $d = 160$  и хотя бы одно из чисел делится на  $p = 7$ ,
2. затем максимальную из сумм элементов таких пар.

В данной задаче под парой подразумевается два различных элемента последовательности. Порядок элементов в паре не важен.

Пример входных данных: 168, 7, 320, 328

Пример выходных данных : 4 488

Пояснение: Из 4 чисел можно составить 6 пар. В данном случае условиям удовлетворяют пары: 168 и 320, 168 и 7, 320 и 7, 328 и 7.

Максимальную сумму дает пара 168 и 320 — 488.

## Задача №2 из решу ЕГЭ

```
a = [int(i) for i in open('17_0.txt')]
k = mx = 0

for i in range(len(a) - 1):
    for j in range(i + 1, len(a)):
        if (a[i] % 160 != a[j] % 160) and ((a[i] % 7 == 0) or (a[j] % 7 == 0)):
            k += 1
            mx = max(mx, (a[i] + a[j]))
print(k, mx)
```

Ответ: 12749665 19989

```
f = open('17.txt')
a = [int(i) for i in f]
s = 0
mx = 0
for i in range(len(a) - 1):
    for j in range(i + 1, len(a)):
        if (a[i] % 160 != a[j] % 160) and ((a[i] % 7 == 0) ↵
↵ or (a[j] % 7 == 0)):
            s += 1
            mx = max(mx, a[i] + a[j])
print(s, mx)
```

12749665 19989

## Задача №46 из решу ЕГЭ

В файле содержится последовательность целых чисел. Задание 17  
Элементы последовательности могут принимать целые значения от 1 до 100 000 включительно.

Определите количество пар последовательности, в которых

- 1. только один из элементов является двузначным числом,**
- 2. а сумма элементов пары кратна максимальному двузначному элементу последовательности.**

В ответе запишите:

- 1. количество найденных пар,**
- 2. затем максимальную из сумм элементов таких пар.**

В данной задаче под парой подразумевается два идущих подряд элемента последовательности.

## Решение задачи №46 из решу ЕГЭ

```
a = [int(i) for i in open('1_17.txt')]
mx = n = 0; rez = 0
for i in range(len(a)):
    if 9 < a[i] < 100:
        mx = max(mx, a[i])
for i in range(len(a) - 1):
    if (9 < a[i] < 100) != (9 < a[i+1] < 100) and
((a[i]+a[i+1])% mx==0):
        n += 1
        rez = max(rez, (a[i] + a[i+1]))
print(n, rez)
```

16 9702



## Часть 2

# Решение задач с использованием файла (высокой сложности)

От Полякова

## Задача №1 с файлом 17-278

В файле [17-278.txt](#) содержится последовательность целых чисел. Элементы последовательности могут принимать целые значения от 0 до 1000.

**Определите сначала количество пар, в которых оба элемента больше, чем сумма всех цифр «4» в пятеричной записи всех чисел в файле, кратных 12, а затем максимальную из сумм таких пар.**

Под парой подразумевается два идущих подряд элемента последовательности.

```

f = [ int(x) for x in open('17-278.txt') ]
def toBase( n, m ):
    s = ''
    n = abs(n)
    while n:
        s = str(n % m) + s
        n //= m
    return s

s = 0
for x in f:
    if x % 32 == 0:
        s += 3*toBase(x, 5).count('3')
count, ma = 0, 0
for i in range(len(f)-2):
    if f[i] > s or f[i+1] > s or f[i+2] > s:
        count += 1
        ma = max( ma, f[i]+f[i+1]+f[i+2] )
print( count, ma )

```

# Самостоятельная работа

## Задача №1 демо-2021

Рассматривается множество целых чисел, принадлежащих числовому отрезку  $[1016; 7937]$ , которые делятся на 3 и не делятся на 7, 17, 19, 27.

**Найдите количество таких чисел и максимальное из них. В ответе запишите два целых числа: сначала количество, затем максимальное число.**

Для выполнения этого задания можно написать программу или воспользоваться редактором электронных таблиц.

```
count = 0
```

```
maxGood = 0
```

```
for n in range(1017, 7937+1, 3):
```

```
    if (n % 3 == 0) and (n % 7 != 0) and \
```

```
       (n % 17 != 0) and (n % 19 != 0) and (n % 27 != 0):
```

```
        maxGood = n
```

```
        count += 1
```

```
print(count, maxGood)
```

**1568, 7935**

```
count = 0; maxGood = 0
for n in range(1017, 7937+1, 3):
    if (n % 7 != 0) and (n % 17 != 0) and (n % 19 != 0) and (n % 27 != 0):
        maxGood = n
        count += 1
print(count, maxGood)
```

**1568, 7935**

```
A=[n for n in range(1016, 7937+1) if (n%3==0 and
n%7!=0 and n%17!=0 and n%19!=0 and n%27!=0)]
print(len(A), max(A))
```

# Домашняя работа

## **ФИПИ**

- В файле содержится последовательность натуральных чисел, каждое из которых не превышает 100 000.
    1. **Определите количество троек элементов последовательности, в которых ровно два из трёх элементов являются трёхзначными числами,**
    2. **а сумма элементов тройки не больше максимального элемента последовательности, оканчивающегося на 13.**
  - Гарантируется, что в последовательности есть хотя бы одно число, оканчивающееся на 13.
    1. **В ответе запишите количество найденных троек чисел,**
    2. **И максимальную из сумм элементов таких троек.**
- В данной задаче под тройкой подразумевается три идущих подряд элемента последовательности.

# Проверка домашней задачи

```
k = mx = t = 0
for i in range(len(a)-2):
    if len(str(a[i]))==3:t+=1
    if len(str(a[i+1]))==3:t+=1
    if len(str(a[i+2]))==3:t+=1
    if t==2 and a[i]+a[i+1]+a[i+2]<=mx13:
        k+=1
        mx=max(mx, a[i]+a[i+1]+a[i+2])
print(k, mx)
```