

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное автономное образовательное  
учреждение высшего образования  
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ  
УНИВЕРСИТЕТ»

**«Исследование основных возможностей Git и GitHub»**

**ОТЧЕТ**  
**по лабораторной работе**

«Лабораторная работа 2.4 Работа со списками в языке Python»

Выполнил:  
Гладкова Елена  
2 курс, группа ППЖ-б-о-21-1.  
Проверил:  
Воронкин Роман Александрович

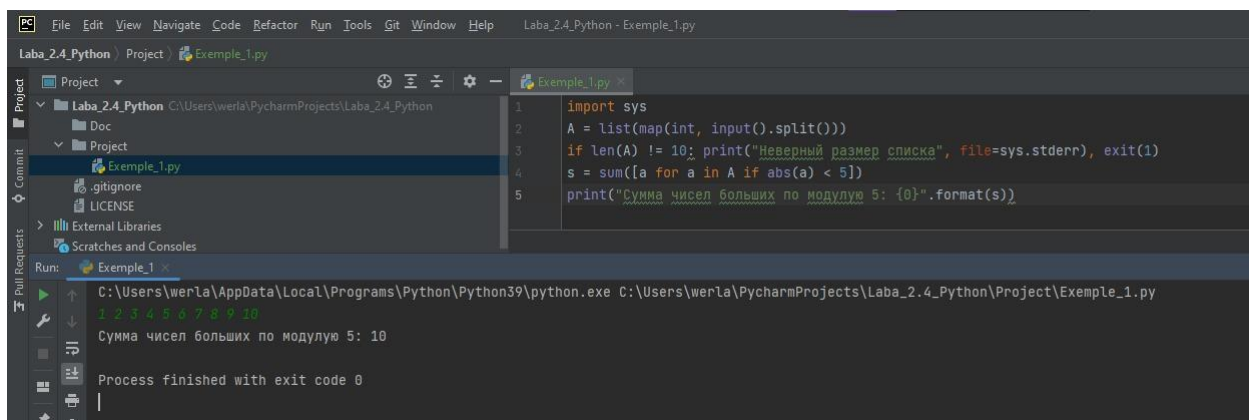
Ставрополь, 2022 г.

## Лабораторная работа 2.4 Работа со списками в языке Python

Цель работы: приобретение навыков по работе со списками при написании программ с помощью языка программирования Python версии 3.x.

7. Проработайте примеры лабораторной работы. Создайте для каждого примера отдельный модуль языка Python. Зафиксируйте изменения в репозитории.

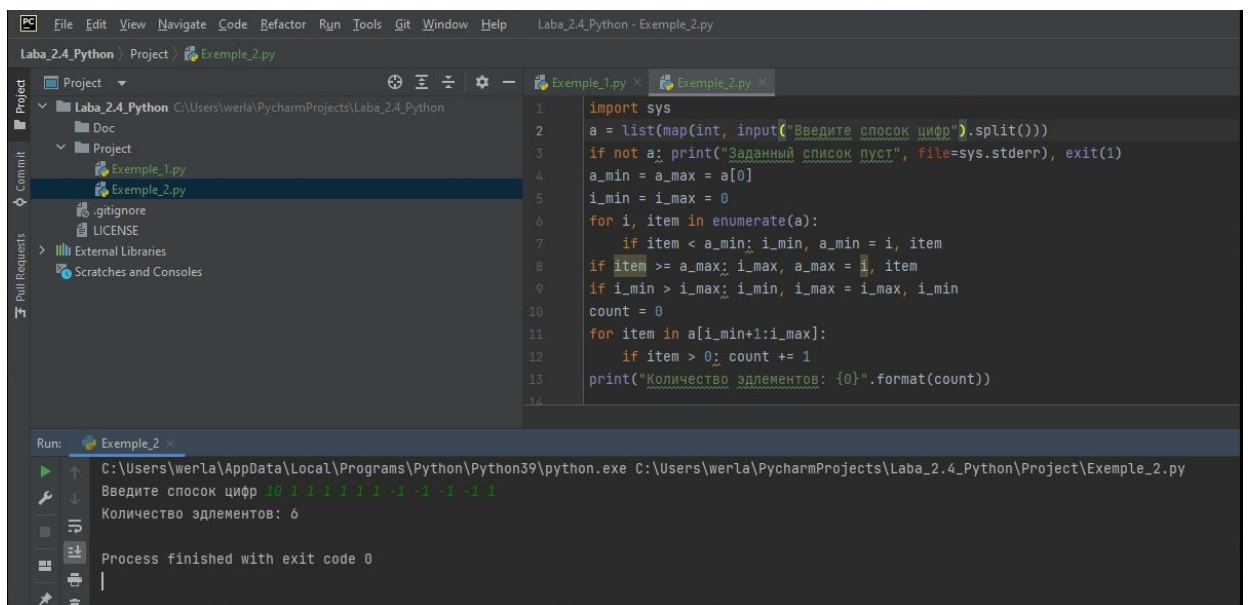
### 1 ПРИМЕР



```
File Edit View Navigate Code Refactor Run Tools Git Window Help Laba_2.4_Python - Exemple_1.py
Lab_2.4_Python Project Exemple_1.py
Project
  Laba_2.4_Python C:\Users\werla\PycharmProjects\Lab_2.4_Python
    Doc
    Project
    Exemple_1.py
    .gitignore
    LICENSE
  External Libraries
  Scratches and Consoles
Run: Exemple_1
C:\Users\werla\AppData\Local\Programs\Python\Python39\python.exe C:\Users\werla\PycharmProjects\Lab_2.4_Python\Project\Exemple_1.py
1 2 3 4 5 6 7 8 9 10
Сумма чисел больших по модулю 5: 10
Process finished with exit code 0
```

```
1 import sys
2 A = list(map(int, input().split()))
3 if len(A) != 10: print("Неверный размер списка", file=sys.stderr), exit(1)
4 s = sum([a for a in A if abs(a) < 5])
5 print("Сумма чисел больших по модулю 5: {}".format(s))
```

### 2 ПРИМЕР

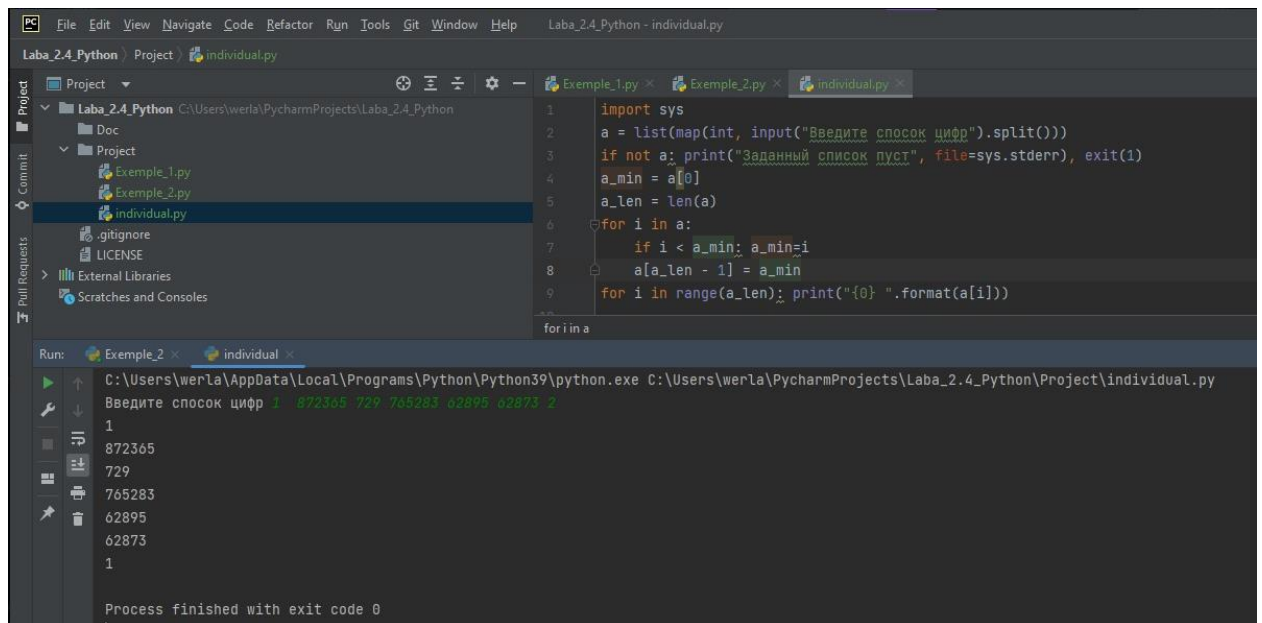


```
File Edit View Navigate Code Refactor Run Tools Git Window Help Laba_2.4_Python - Exemple_2.py
Lab_2.4_Python Project Exemple_2.py
Project
  Laba_2.4_Python C:\Users\werla\PycharmProjects\Lab_2.4_Python
    Doc
    Project
    Exemple_1.py
    Exemple_2.py
    .gitignore
    LICENSE
  External Libraries
  Scratches and Consoles
Run: Exemple_2
C:\Users\werla\AppData\Local\Programs\Python\Python39\python.exe C:\Users\werla\PycharmProjects\Lab_2.4_Python\Project\Exemple_2.py
Введите список цифр 10 1 1 1 1 1 1 -1 -1 -1 -1
Количество элементов: 6
Process finished with exit code 0
```

```
1 import sys
2 a = list(map(int, input("Введите список цифр").split()))
3 if not a: print("Заданный список пуст", file=sys.stderr), exit(1)
4 a_min = a_max = a[0]
5 i_min = i_max = 0
6 for i, item in enumerate(a):
7     if item < a_min: i_min, a_min = i, item
8     if item >= a_max: i_max, a_max = i, item
9     if i_min > i_max: i_min, i_max = i_max, i_min
10    count = 0
11    for item in a[i_min+1:i_max]:
12        if item > 0: count += 1
13    print("Количество элементов: {}".format(count))
14
```

### ЗАДАНИЕ 1

3. Ввести список А из 10 элементов, найти наименьший элемент и переставить его с последним элементом. Преобразованный массив вывести.



The screenshot shows the PyCharm IDE with a project named 'Laba\_2.4\_Python'. The file explorer on the left shows the project structure. The main editor displays a Python script named 'individual.py'. The script prompts the user to enter a list of numbers, finds the minimum element, and swaps it with the last element. The Run window at the bottom shows the execution output.

```
1 import sys
2 a = list(map(int, input("Введите список цифр").split()))
3 if not a: print("Заданный список пуст", file=sys.stderr), exit(1)
4 a_min = a[0]
5 a_len = len(a)
6 for i in a:
7     if i < a_min: a_min = i
8     a[a_len - 1] = a_min
9     for i in range(a_len): print("{} ".format(a[i]))
10    for i in a
```

Run: C:\Users\werla\AppData\Local\Programs\Python\Python39\python.exe C:\Users\werla\PycharmProjects\Laba\_2.4\_Python\Project\individual.py

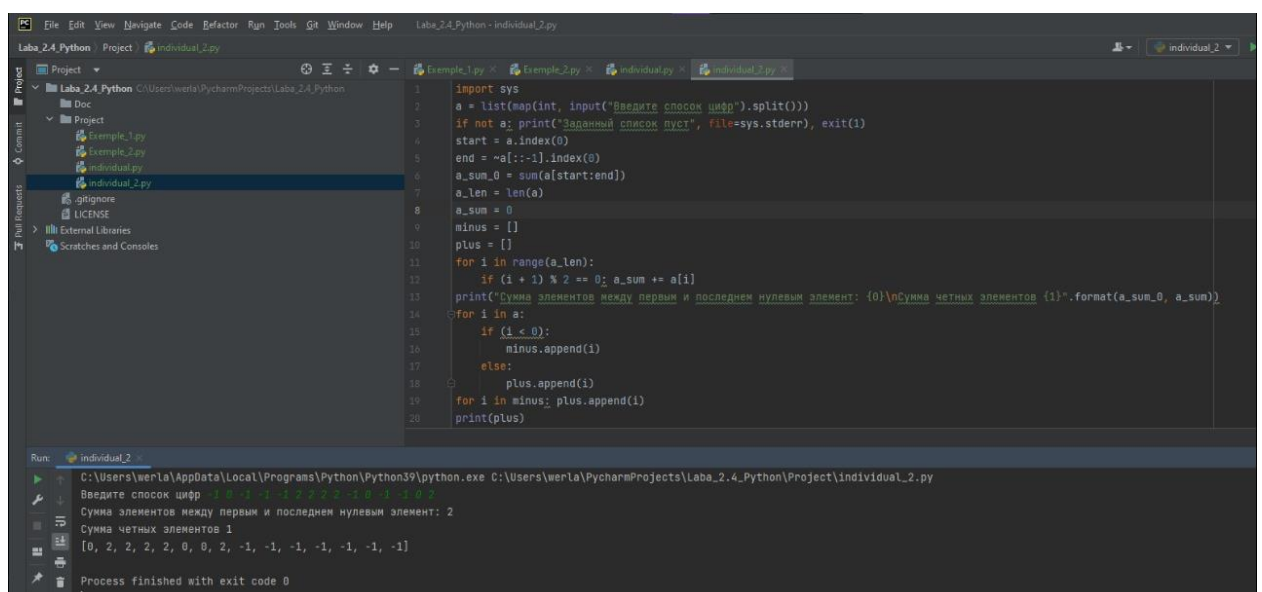
Введите список цифр 1 872365 729 765283 62895 62873 2

1  
872365  
729  
765283  
62895  
62873  
1

Process finished with exit code 0

## ЗАДАНИЕ 2

3. В списке, состоящем из целых элементов, вычислить: 1. произведение элементов списка с четными номерами; 2. сумму элементов списка, расположенных между первым и последним нулевыми элементами. Преобразовать список таким образом, чтобы сначала располагались все положительные элементы, а потом - все отрицательные (элементы, равные 0, считать положительными).



The screenshot shows the PyCharm IDE with a project named 'Laba\_2.4\_Python'. The file explorer on the left shows the project structure. The main editor displays a Python script named 'individual\_2.py'. The script prompts the user to enter a list of numbers, calculates the product of elements at even indices and the sum of elements between the first and last zero elements, and then sorts the list by sign. The Run window at the bottom shows the execution output.

```
1 import sys
2 a = list(map(int, input("Введите список цифр").split()))
3 if not a: print("Заданный список пуст", file=sys.stderr), exit(1)
4 start = a.index(0)
5 end = a[::-1].index(0)
6 a_sum_0 = sum(a[start:end])
7 a_len = len(a)
8 a_sum = 0
9 minus = []
10 plus = []
11 for i in range(a_len):
12     if (i + 1) % 2 == 0: a_sum += a[i]
13 print("Сумма элементов между первым и последним нулевым элементом: {} \n Сумма четных элементов {}".format(a_sum_0, a_sum))
14 for i in a:
15     if (i < 0):
16         minus.append(i)
17     else:
18         plus.append(i)
19 for i in minus: plus.append(i)
20 print(plus)
```

Run: C:\Users\werla\AppData\Local\Programs\Python\Python39\python.exe C:\Users\werla\PycharmProjects\Laba\_2.4\_Python\Project\individual\_2.py

Введите список цифр 1 0 1 -1 -1 2 2 2 2 -1 0 1 1 0 2

Сумма элементов между первым и последним нулевым элементом: 2

Сумма четных элементов 1

[0, 2, 2, 2, 2, 0, 0, 2, -1, -1, -1, -1, -1, -1, -1]

Process finished with exit code 0