

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное  
образовательное учреждение высшего  
образования

«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Кафедра инфокоммуникаций

Основы кроссплатформенного программирования

Отчет по лабораторной работе №5

**Работа со  
списками в языке Python**

Выполнил студент группы ИТС-б-о-21-1

Тимофеева Марина Сергеевна

«    » \_\_\_\_\_ 20\_\_ г.

Подпись студента \_\_\_\_\_

Проверил: Доцент, к.т.н, доцент

кафедры инфокоммуникаций

Воронкин А. В.

Работа защищена с оценкой: \_\_\_\_\_

\_\_\_\_\_  
(подпись)

Ставрополь, 2022

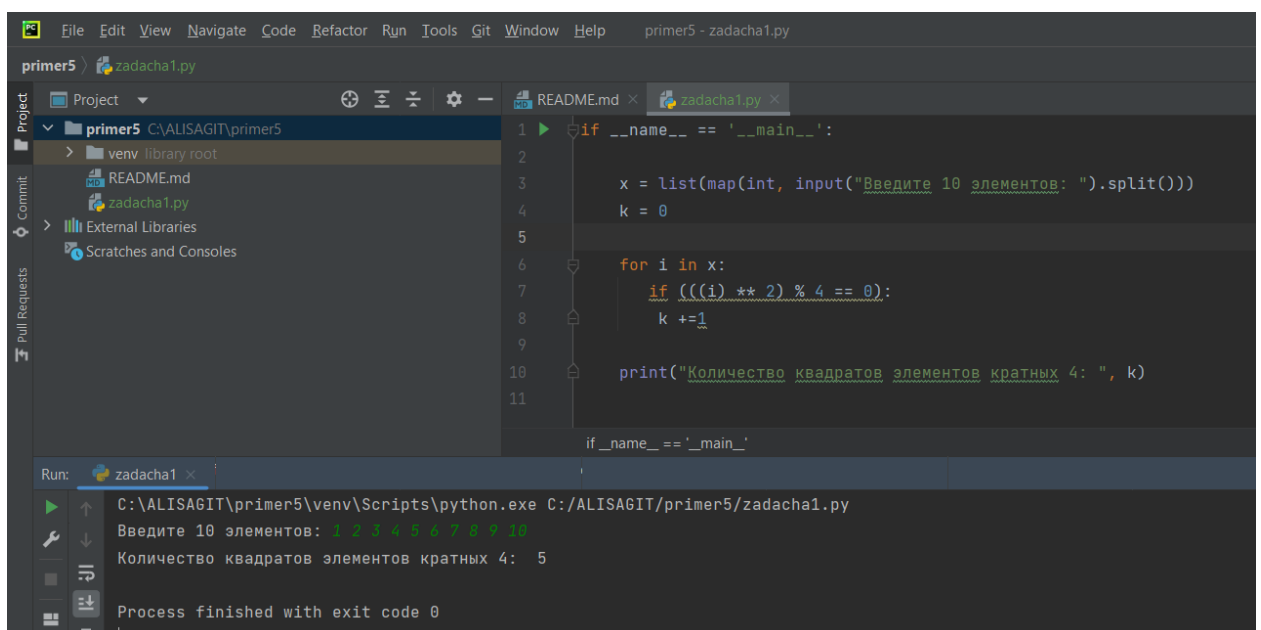
**Цель работы:** приобретение навыков по работе со списками при написании программ с помощью языка программирования Python.

Порядок выполнения работы:

Создала общедоступный репозиторий на GitHub (<https://github.com/AlisaL1sa/primer5>)

## Задача 1.

17. Ввести список *A* из 10 элементов, найти квадраты элементов кратных 4 и их количество. Преобразованный массив вывести.



The screenshot shows a code editor with a file named `zadacha1.py`. The code is as follows:

```
1 if __name__ == '__main__':
2
3     x = list(map(int, input("Введите 10 элементов: ").split()))
4     k = 0
5
6     for i in x:
7         if ((i ** 2) % 4 == 0):
8             k += 1
9
10    print("Количество квадратов элементов кратных 4: ", k)
11
12 if __name__ == '__main__':
```

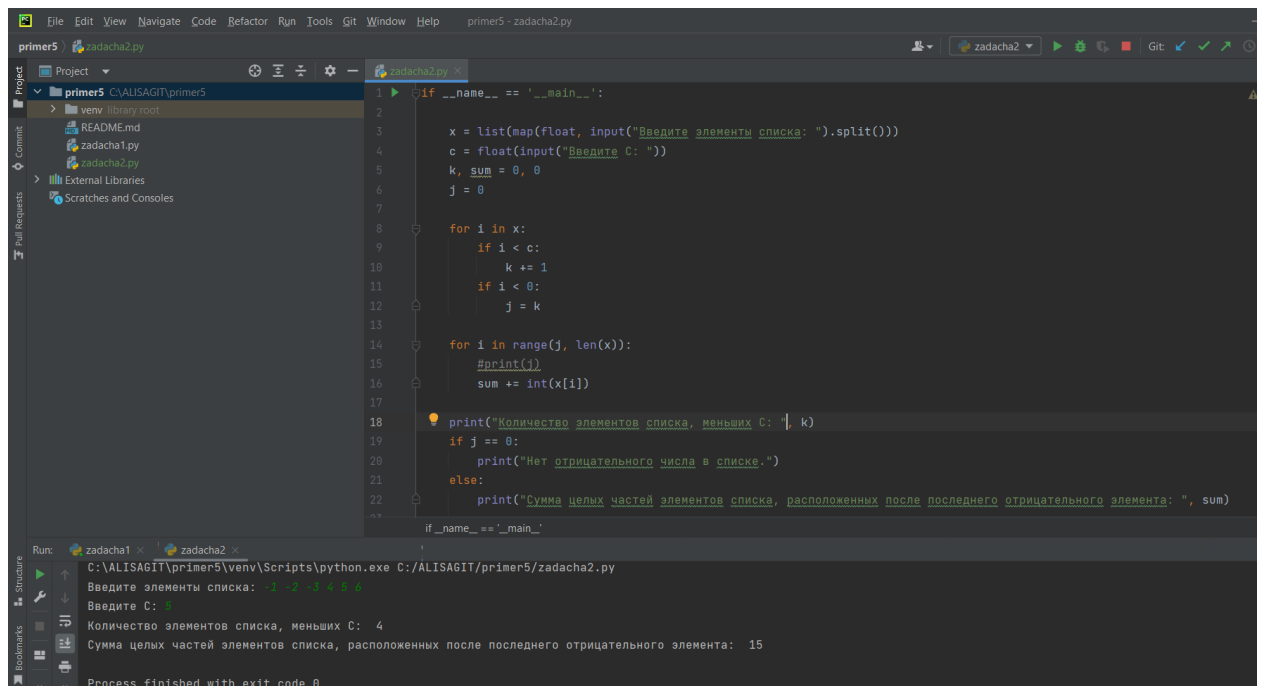
The Run console at the bottom shows the execution of the script. The command executed is `C:\ALISAGIT\primer5\venv\Scripts\python.exe C:/ALISAGIT/primer5/zadacha1.py`. The input provided is `1 2 3 4 5 6 7 8 9 10`. The output is `Количество квадратов элементов кратных 4: 5`. The process finished with exit code 0.

Рисунок 1. Код для первой задачи и проверка его работоспособности.

## Задача 2.

17. В списке, состоящем из вещественных элементов, вычислить:

1. количество элементов списка, меньших *C*;
2. сумму целых частей элементов списка, расположенных после последнего отрицательного элемента.



**Рисунок 2.** Код для второй задачи и проверка его работоспособности.

### **Ответы на контрольные вопросы:**

#### **1. Что такое списки в языке Python?**

Список (list) – это структура данных для хранения объектов различных типов. В нем можно хранить объекты различных типов. Размер списка не статичен, его можно изменять. Список по своей природе является изменяемым типом данных. Переменная, определяемая как список, содержит ссылку на структуру в памяти, которая в свою очередь хранит на какие-либо другие объекты или структуры.

#### **2. Как осуществляется создание списка в Python?**

Для создания списка нужно заключить элементы в квадратные скобки.

#### **3. Как организовано хранение списков в оперативной памяти?**

При создании списка в памяти резервируется область, которую можно условно назвать некоторым “контейнером”, в котором хранятся ссылки на другие элементы данных в памяти. В отличие от таких типов данных, как число или строка, содержимое “контейнера” списка можно менять.

4. Каким образом можно перебрать все элементы списка? Читать элементы списка можно с помощью следующего цикла: `my_list = ['один', 'два',`

'три', 'четыре', 'пять']

```
for elem in my_list:  
    print(elem)
```

5. Какие существуют арифметические операции со списками?

Для объединения списков можно использовать оператор сложения ( + ).

Список можно повторить с помощью оператора умножения ( \* ).

6. Как проверить есть ли элемент в списке?

Для того, чтобы проверить, есть ли заданный элемент в списке Python необходимо использовать оператор in.

7. Как определить число вхождений заданного элемента в списке?

Метод count можно использовать для определения числа сколько раз данный элемент встречается в списке.

8. Как осуществляется добавление (вставка) элемента в список?

Метод append можно использовать для добавления элемента в список.

Метод insert можно использовать, чтобы вставить элемент в список.

9. Как выполнить сортировку списка?

Для сортировки списка нужно использовать метод sort. Для сортировки списка в порядке убывания необходимо вызвать метод sort с аргументом reverse=True.

10. Как удалить один или несколько элементов из списка?

Удалить элемент можно, написав его индекс в методе pop. Если не указывать индекс, то функция удалит последний элемент. Элемент можно удалить с помощью метода remove. Оператор del можно использовать для тех же целей.

Можно удалить несколько элементов с помощью оператора среза.

Можно удалить все элементы из списка с помощью метода clear.

11. Что такое списковое включение и как с его помощью осуществлять обработку списков?

List Comprehensions чаще всего на русский язык переводят как абстракция списков или списковое включение, является частью синтаксиса

языка, которая предоставляет простой способ построения списков.

В языке Python есть две очень мощные функции для работы с коллекциями: `map` и `filter`. Они позволяют использовать функциональный стиль программирования, не прибегая к помощи циклов, для работы с такими типами как `list`, `tuple`, `set`, `dict` и т.п. Списковое включение позволяет обойтись без этих функций.

12. Какие существуют функции агрегации для работы со списками?

Для работы со списками Python предоставляет следующие функции:

1. `len(L)` - получить число элементов в списке `L`
2. `min(L)` - получить минимальный элемент списка `L`
3. `max(L)` - получить максимальный элемент списка `L`
4. `sum(L)` - получить сумму элементов списка `L`, если список `L`

содержит только числовые значения.

13. Как создать копию списка? `copy.copy(x)`

14. Самостоятельно изучите функцию `sorted` языка Python. В чем ее отличие от метода `sort` списков?

Функция `sorted()` в Python возвращает отсортированный список из элементов в итерируемом объекте. `list.sort()` на 13% быстрее, чем `sorted()`.

15. Самостоятельно изучите функцию `sorted` языка Python. В чем ее отличие от метода `sort` списков?

Функция `sort()` очень похожа на `sorted()`, но в отличие от `sorted` она ничего не возвращает и не вносит изменений в исходную последовательность. Более того, `sort()` является методом класса `list` и может использоваться только со списками. Синтаксис: `List_name.sort(key, reverse=False)` Параметры: ключ: Функция, которая служит ключом для сравнения сортировки. реверс: Если `true`, то список сортируется в порядке убывания.

**Вывод:** приобрела навык по работе со списками при написании программ с помощью языка программирования Python.