

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное  
образовательное учреждение высшего образования  
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

**Кафедра инфокоммуникаций**

**Лабораторная работа 6**

Работа со списками в языке Python

Выполнил студент группы ИВТ-б-о-20-1

Симанский М.Ю « » \_\_\_\_\_ 20\_\_ г.

Подпись студента \_\_\_\_\_

Работа защищена « » \_\_\_\_\_ 20\_\_ г.

Проверил Воронкин Р.А. \_\_\_\_\_

(подпись)

**Цель работы:** приобретение навыков по работе со строками при написании программ с помощью языка программирования Python3.

Ссылка на репозиторий: <https://github.com/MaksimSimanskiy/lab4.git>

## Задание 1

12. Ввести список A из 10 элементов, найти сумму элементов, больших 2 и меньших 20 и кратных 8, их количество и вывести результаты на экран.

## Код

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
a = []
b = []
for i in range(10):
    a.append(int(input("Введите элемент списка " + str(i) + ' - ')))
b = list(filter(lambda x: 20 > x > 2 and x % 8 == 0, a))
print('Список A = {} \nВыбранные элементы = {} \nКоличество элементов = {} \nСумма = {}'.format(a, b, len(b), sum(b)))
```

## UML-диаграмма

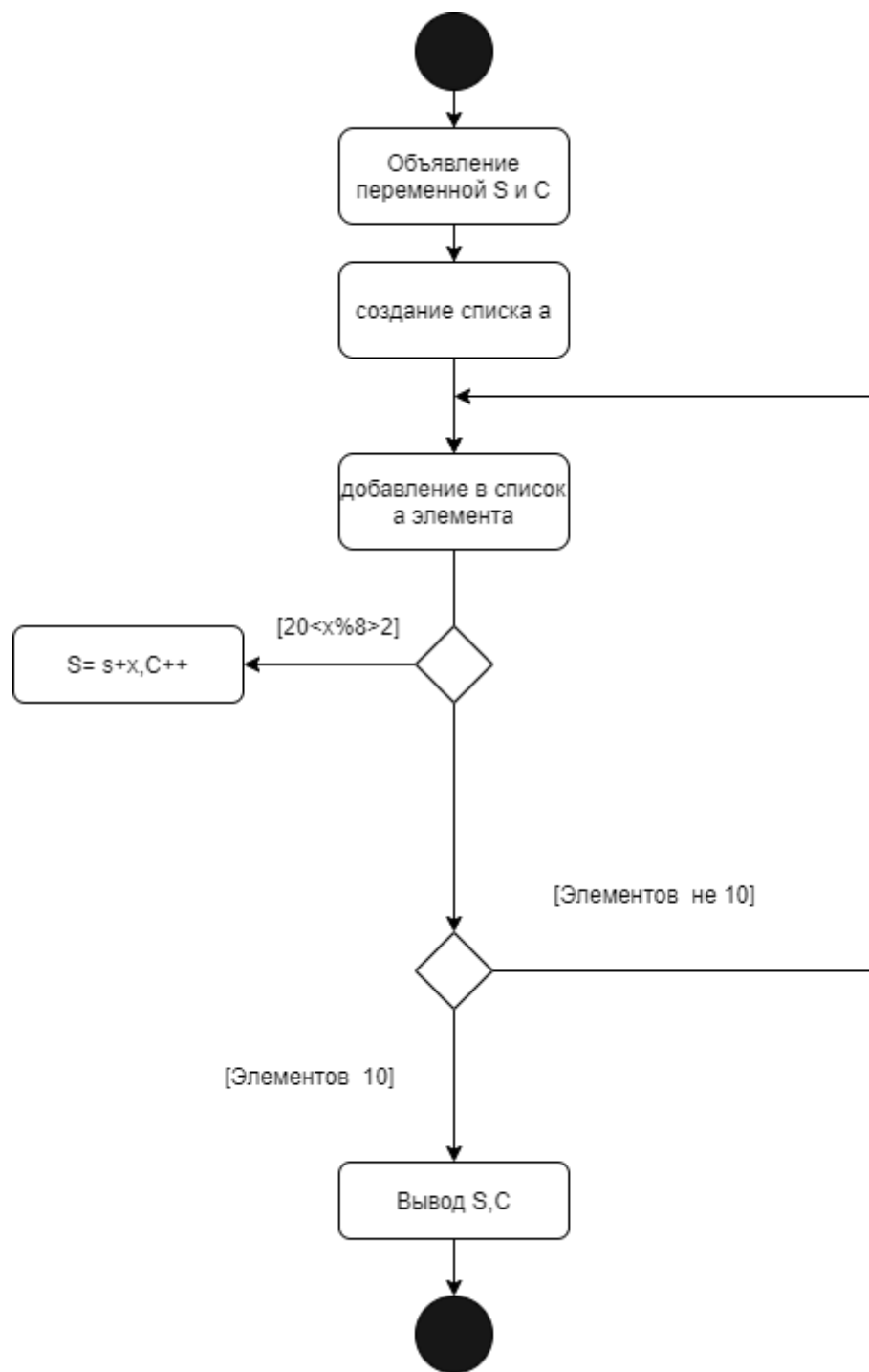


Рисунок 1 – UML-диаграмма

**Результат**

```
Введите элемент списка 0 - 4
Введите элемент списка 1 - 5
Введите элемент списка 2 - 6
Введите элемент списка 3 - 0
Введите элемент списка 4 - 8
Введите элемент списка 5 - 16
Введите элемент списка 6 - 45
Введите элемент списка 7 - 6
Введите элемент списка 8 - 44
Введите элемент списка 9 - 24
Список A = [4, 5, 6, 0, 8, 16, 45, 6, 44, 24]
Выбранные элементы = [8, 16]
Количество элементов = 2
Сумма = 24

Process finished with exit code 0
```

Рисунок 2 – Результат работы

## Задание 2

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

- 1) номер максимального по модулю элемента списка;
- 2) сумму элементов списка, расположенных после первого положительного элемента. Преобразовать список таким образом, чтобы сначала располагались все элементы, целая часть которых лежит в интервале  $[a, b]$ , а потом - все остальные.

## Код

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
c = int(input("Введите начало интервала "))
d = int(input("Введите конец интервала "))
a = []

for i in range(5):
    a.append(float(input("Введите элемент списка " + str(i) + " - ")))

print('Максимальный элемент ', max(a, key=abs))
```

```

print('Индекс максимального элемента ' , a.index(max(a, key=abs)))
for i in range(len(a)):
    if (a[i] > 0.):
        k = i
        break
print('Сумма чисел после первого положительного числа',sum(a[k:]))
print('Отсортированный массив',sorted(a,key = lambda x: d > x > c,reverse =
True))

```

## UML-диаграмма

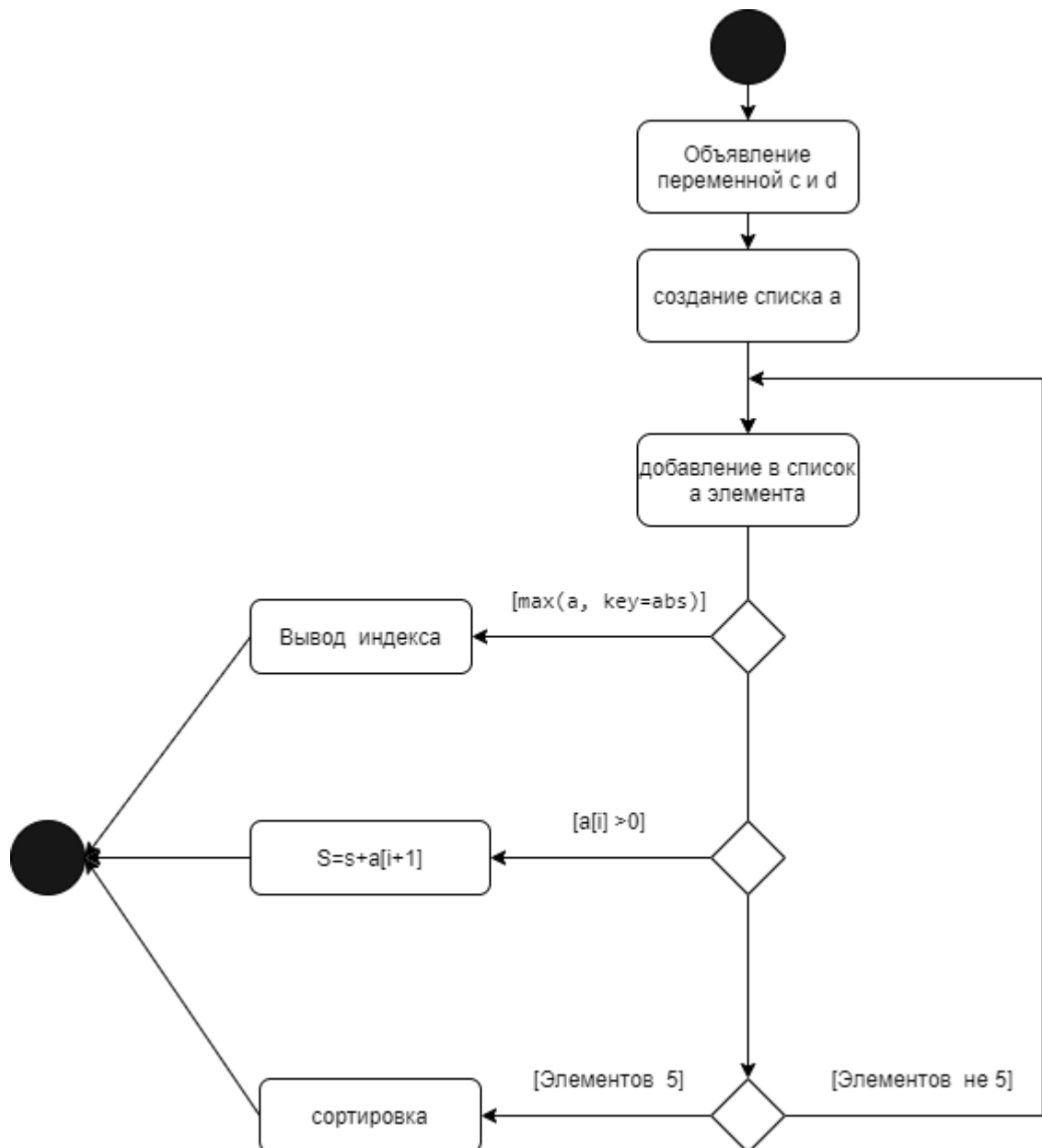


Рисунок 3 – UML-диаграмма

## Результат

```
Введите начало интервала 4
Введите конец интервала 8
Введите элемент списка 0 - -5
Введите элемент списка 1 - -5
Введите элемент списка 2 - 45
Введите элемент списка 3 - 8
Введите элемент списка 4 - 5
Максимальный элемент 45.0
Индекс максимального элемента 2
Сумма чисел после первого положительного числа 58.0
Отсортированный массив [5.0, -5.0, -5.0, 45.0, 8.0]

Process finished with exit code 0
```

Рисунок 4 – Результат работы

## Ответы на контрольные вопросы

1. Список (list) – это структура данных для хранения объектов различных типов.
2. Для создания списка нужно заключить элементы в квадратные скобки
3. Список является изменяемым типом данных. При его создании 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. `my_list = ['один', 'два', 'три', 'четыре', 'пять']` `for elem in my_list:`  
`print(elem)`
5. Существуют + и \*.

6. Для того, чтобы проверить, есть ли заданный элемент в списке Python необходимо использовать оператор `in`
7. Метод `count` можно использовать для определения числа сколько раз данный элемент встречается в списке
8. Метод `insert` можно использовать, чтобы вставить элемент в список
9. Для сортировки списка нужно использовать метод `sort`
10. Удалить элемент можно, написав его индекс в методе `pop`, `remove` и `del`
11. List Comprehensions чаще всего на русский язык переводят как абстракция списков или списковое включение, является частью синтаксиса языка, которая предоставляет простой способ построения списков. Для самой обработки существуют функции `map` и `filter`
12. Слайс задается тройкой чисел, разделенных запятой: `start:stop:step`. `Start` – позиция с которой нужно начать выборку, `stop` – конечная позиция, `step` – шаг. При этом необходимо помнить, что выборка не включает элемент определяемый `stop`.
13. Для работы со списками Python предоставляет следующие функции:  
`len(L)` - получить число элементов в списке `L` . `min(L)` - получить минимальный элемент списка `L` . `max(L)` - получить максимальный элемент списка `L` . `sum(L)` - получить сумму элементов списка `L` , если список `L` содержит только числовые значения.
14. При помощи метода `copy`
15. Функция `sorted()` создает новый отсортированный список, в то время как метод `list.sort()` сортирует список на месте. Если вы хотите сохранить, несортированный список использовать функцию `sorted()`. Другое отличие состоит в том, что функция `sorted()` работает с любым повторяемым объектом.