

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

Институт цифрового развития
Кафедра инфокоммуникаций

ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ №2.4
дисциплины «Основы кроссплатформенного программирования»

Выполнила:
Мурашко Анастасия Юрьевна
1 курс, группа ИТС-б-о-22-1,
11.03.02 «Инфокоммуникационные
технологии и системы связи»,
направленность (профиль)
«Инфокоммуникационные системы и
сети», очная форма обучения

(подпись)

Руководитель практики:
Воронкин Р. А., доцент кафедры
инфокоммуникаций

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2023 г.

Тема: Работа со списками в языке Python

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

Ход работы:

Вариант №10

Пример 1

Ввести список A из 10 элементов, найти сумму элементов, меньших по модулю 5, и вывести ее на экран.

```
1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3  import sys
4
5  if __name__ == '__main__':
6      # Ввести список одной строкой.
7      A = list(map(int, input().split()))
8      # Проверить количество элементов списка.
9      if len(A) != 10:
10         print("Неверный размер списка", file=sys.stderr)
11         exit(1)
12     # Найти искомую сумму.
13     s = sum([a for a in A if abs(a) < 5])
14     print(s)
```

Ln: 14, Col: 13

[Run](#) [Share](#) Command Line Arguments

1 23 54 22 11 22 43 0 2 -3

0



Рисунок 1. Задание 1

Пример 2

Написать программу, которая для целочисленного списка определяет, сколько положительных элементов располагается между его максимальным и минимальным элементами.

```
1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3  import sys
4
5  if __name__ == '__main__':
6      # Ввести список одной строкой.
7      a = list(map(int, input().split()))
8      # Если список пуст, завершить программу.
9      if not a:
10         print("Заданный список пуст", file=sys.stderr)
11         exit(1)
12     # Определить индексы минимального и максимального элементов.
13     a_min = a_max = a[0]
14     i_min = i_max = 0
15     for i, item in enumerate(a):
16         if item < a_min:
17             i_min, a_min = i, item
18         if item >= a_max:
19             i_max, a_max = i, item
20     #Проверить индексы и обменять их местами.
21     if i_min > i_max:
```

Ln: 11, Col: 16

 Run  Share Command Line Arguments



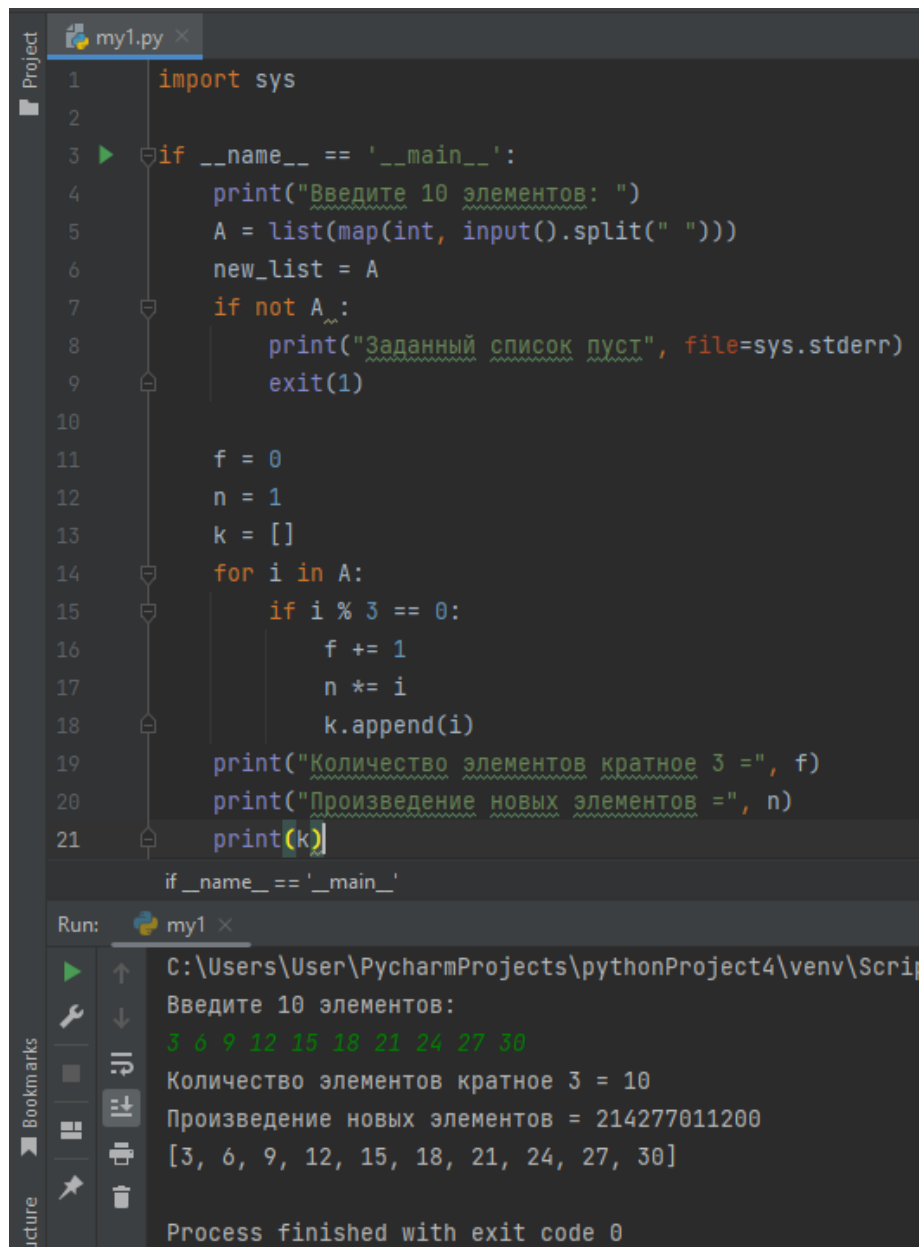
 12 1 5 2 11 -4
 4

Рисунок 2. Задание 2

Индивидуальные задания

Задание 1

Ввести список А из 10 элементов, найти произведение положительных элементов кратных 3, их количество и вывести результаты на экран.



The screenshot displays the PyCharm IDE interface. The top pane shows a Python file named `my1.py` with the following code:

```
1 import sys
2
3 if __name__ == '__main__':
4     print("Введите 10 элементов: ")
5     A = list(map(int, input().split(" ")))
6     new_list = A
7     if not A:
8         print("Заданный список пуст", file=sys.stderr)
9         exit(1)
10
11     f = 0
12     n = 1
13     k = []
14     for i in A:
15         if i % 3 == 0:
16             f += 1
17             n *= i
18             k.append(i)
19     print("Количество элементов кратное 3 =", f)
20     print("Произведение новых элементов =", n)
21     print(k)
```

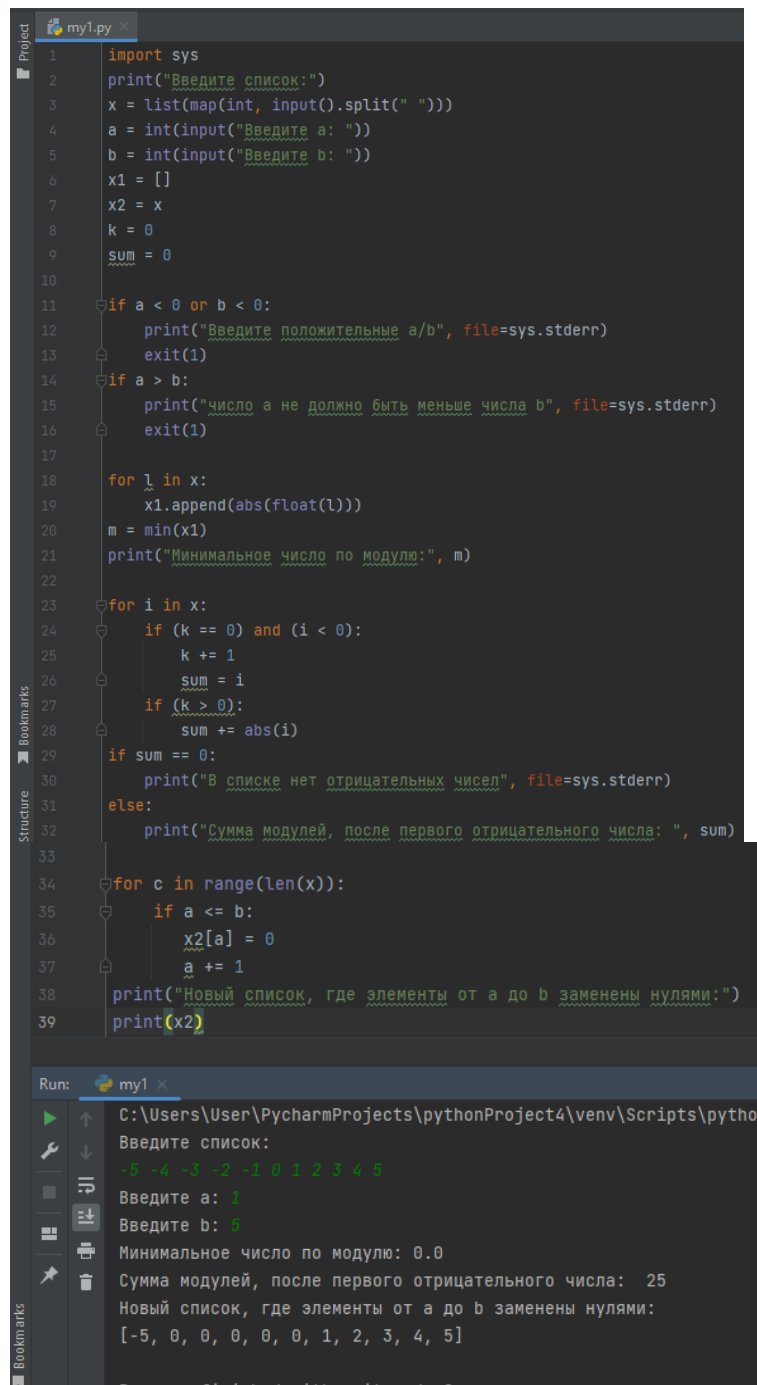
The bottom pane shows the execution output for the `my1` script:

```
Run: my1
C:\Users\User\PycharmProjects\pythonProject4\venv\Scripts\python.exe
Введите 10 элементов:
3 6 9 12 15 18 21 24 27 30
Количество элементов кратное 3 = 10
Произведение новых элементов = 214277011200
[3, 6, 9, 12, 15, 18, 21, 24, 27, 30]
Process finished with exit code 0
```

Рисунок 3. Задание 1

Задание 2

В списке, состоящем из вещественных элементов, вычислить: 1. номер минимального по модулю элемента списка; 2. сумму модулей элементов списка, расположенных после первого отрицательного элемента. Сжать список, удалив из него все элементы, величина которых находится в интервале $[a, b]$. Освободившиеся в конце списка элементы заполнить нулями.



The screenshot displays the PyCharm IDE interface. The top pane shows a Python file named `my1.py` with the following code:

```
1 import sys
2 print("Введите список:")
3 x = list(map(int, input().split(" ")))
4 a = int(input("Введите a: "))
5 b = int(input("Введите b: "))
6 x1 = []
7 x2 = x
8 k = 0
9 sum = 0
10
11 if a < 0 or b < 0:
12     print("Введите положительные a/b", file=sys.stderr)
13     exit(1)
14 if a > b:
15     print("число a не должно быть меньше числа b", file=sys.stderr)
16     exit(1)
17
18 for l in x:
19     x1.append(abs(float(l)))
20 m = min(x1)
21 print("Минимальное число по модулю:", m)
22
23 for i in x:
24     if (k == 0) and (i < 0):
25         k += 1
26         sum = i
27     if (k > 0):
28         sum += abs(i)
29 if sum == 0:
30     print("В списке нет отрицательных чисел", file=sys.stderr)
31 else:
32     print("Сумма модулей, после первого отрицательного числа: ", sum)
33
34 for c in range(len(x)):
35     if a <= b:
36         x2[a] = 0
37         a += 1
38 print("Новый список, где элементы от a до b заменены нулями:")
39 print(x2)
```

The bottom pane shows the execution output for the script:

```
Run: my1 x
C:\Users\User\PycharmProjects\pythonProject4\venv\Scripts\python.exe
Введите список:
-5 -4 -3 -2 -1 0 1 2 3 4 5
Введите a: 1
Введите b: 6
Минимальное число по модулю: 0.0
Сумма модулей, после первого отрицательного числа: 25
Новый список, где элементы от a до b заменены нулями:
[-5, 0, 0, 0, 0, 0, 1, 2, 3, 4, 5]
```

Рисунок 4. Задание 2

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 другие элементы данных в памяти. В отличие от таких типов данных 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: В ходе выполнения лабораторной работы приобретены навыки по работе со списками при написании программ с помощью языка программирования Python версии 3.x.