

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 №4

«Работа со списками в языке Python»

Выполнил студент группы ИТС-б-о-21-1

Крамаренко Илья Витальевич

« » _____ 20__ г.

Подпись студента _____

Проверил: Доцент, к.т.н, доцент кафедры
инфокоммуникаций

Воронкин А. В.

Работа защищена с оценкой: _____

(подпись)

Ставрополь, 2022

Лабораторная работа 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.

Ход работы:

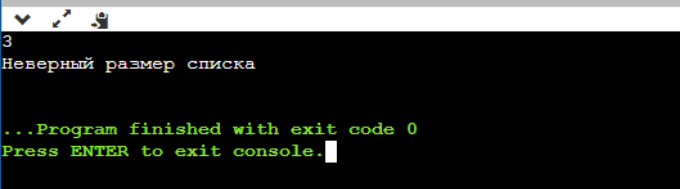
Создадим общедоступный репозиторий -

https://github.com/IliyaKr/Lab_rab_4.git

Работа с примерами:

Пример 1:

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



```
3
Неверный размер списка

...Program finished with exit code 0
Press ENTER to exit console.
```

Рис 1. Окно вывода для Примера 1.

Пример 2:

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

Рис 2. Окно вывода для Примера 2.

Индивидуальные задания:

Вариант 7

Задание 1.

```
3
Количество элементов списка кратные 3 = 1
Новый список элементов кратные трем: [3]
Последний элемент такого списка = 3
```

Рис 3. Окно вывода Задания 1.

Задание 2.

В списке, состоящем из вещественных элементов, вычислить:

1. номер минимального элемента списка;
2. сумму элементов списка, расположенных между первым и вторым отрицательными элементами.

Преобразовать список таким образом, чтобы сначала располагались все элементы, модуль которых не превышает 1, а потом - все остальные.

```
5
Индекс минимального элемента: 0
12 - сумма.
```

Рис 4. Окно вывода Задания 2.

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 ссылки 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().

Вывод: приобрёл навыки по работе со списками при написании программ с помощью языка программирования Python версии 3.x.