

МИНЕСТЕРСТВО НАУКИ ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ФГАОУ ВО «СЕВЕРО - КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Институт цифрового развития
Кафедра инфокоммуникаций

ОТЧЁТ
по лабораторной работе №3
Дисциплина: «Языки программирования»

Выполнил: студентка 2 курса
группы ИТС-б-о-20-1
Игнатова Елизавета Сергеевна

Проверил:
к.ф.-м.н., доцент
кафедры инфокоммуникаций
Воронкин Роман Александрович

Работа защищена с оценкой: _____

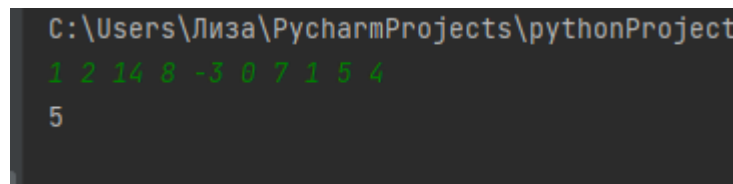
Ставрополь, 2021

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

Порядок выполнения работы:

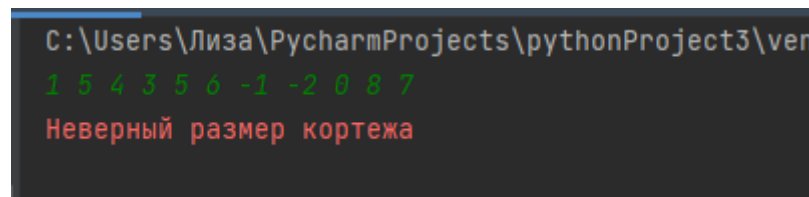
Ссылка на репозиторий: <https://github.com/Nebula139/Sky2.3>

Индивидуальное 1. Ввести кортеж из 10 элементов, найти сумму элементов, меньших по модулю 5, и вывести её на экран.



```
C:\Users\Лиза\PycharmProjects\pythonProject
1 2 14 8 -3 0 7 1 5 4
5
```

Рисунок 1 – Результат выполнения задания (при вводе кортежа из 10 элементов)

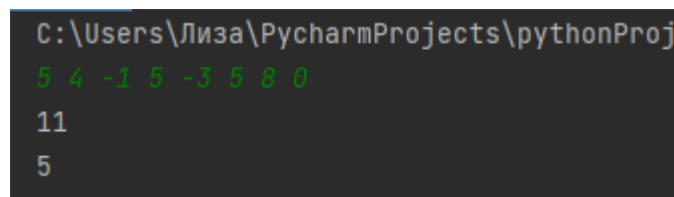


```
C:\Users\Лиза\PycharmProjects\pythonProject3\ven
1 5 4 3 5 6 -1 -2 0 8 7
Неверный размер кортежа
```

Рисунок 2 – Результат выполнения задания (при вводе кортежа из любого другого количества элементов)

Индивидуальное 2. В кортеже, состоящем из вещественных элементов, вычислить:

- 1) сумму элементов списка с нечетными номерами;
- 2) сумму элементов списка, расположенных между первым и последним отрицательными элементами.



```
C:\Users\Лиза\PycharmProjects\pythonProj
5 4 -1 5 -3 5 8 0
11
5
```

Рисунок 3 – результат работы кода

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. Каково назначение кортежей в языке Python?

Во-первых, это экономия места. Дело в том, что кортежи в памяти занимают меньший объем по сравнению со списками. Во-вторых – прирост производительности, который связан с тем, что кортежи работают быстрее, чем списки (т. е. на операции перебора элементов и т. п. будет тратиться меньше времени). Во-третьих – это безопасность данных от случайного изменения.

3. Как осуществляется создание кортежей?

Для создания пустого кортежа можно воспользоваться одной из следующих команд.

```
>>> a = ()
```

```
>>> b = tuple()
```

Кортеж с заданным содержанием создается также как список, только вместо квадратных скобок используются круглые.

4. Как осуществляется доступ к элементам кортежа?

Доступ к элементам кортежа осуществляется также как к элементам списка – через указание индекса.

5. Зачем нужна распаковка (деструктуризация) кортежа?

Обращение по индексу, это не самый удобный способ работы с кортежами. Дело в том, что кортежи часто содержат значения разных типов, и помнить, по какому индексу что лежит — очень непросто. Но есть способ лучше! Как мы кортеж собираем, так его можно и разобрать:

```
name_and_age = ('Bob', 42)
```

```
(name, age) = name_and_age
```

```
name # 'Bob'
```

```
age # 42
```

6. Какую роль играют кортежи во множественном присваивании?

Благодаря тому, что кортежи легко собирать и разбирать, в Python удобно делать такие вещи, как множественное присваивание:

$$(a, b, c) = (1, 2, 3)$$

a # 1

b # 2

c # 3

7. Как выбрать элементы кортежа с помощью среза?

С помощью операции взятия среза можно получить другой кортеж.

Общая форма операции взятия среза для кортежа: $T2 = T1[i:j]$, где $T2$ – новый кортеж, который получается из кортежа $T1$; $T1$ – исходный кортеж, для которого происходит срез; i, j – соответственно нижняя и верхняя границы среза. Фактически берутся ко вниманию элементы, лежащие на позициях $i, i+1, \dots, j-1$. Значение j определяет позицию за последним элементом среза.

8. Как выполняется конкатенация и повторение кортежей?

Для кортежей можно выполнять операцию конкатенации, которая обозначается символом $+$. В простейшем случае для конкатенации двух кортежей общая форма операции следующая:

$$T3 = T1 + T2$$

Кортеж может быть образован путем операции повторения, обозначаемой символом $*$. При использовании в выражении общая форма операции следующая:

$$T2 = T1 * n$$

9. Как выполняется обход элементов кортежа?

Элементы кортежа можно последовательно просмотреть с помощью операторов цикла `while` или `for`.

10. Как проверить принадлежность элемента кортежу?

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

11. Какие методы работы с кортежами Вам известны?

Метод `index()` – поиск позиции элемента в кортеже.

Метод `count()` – количество вхождений элемента в кортеж.

12. Допустимо ли использование функций агрегации, таких как `len()` , `sum()` и т. д. при работе с кортежами?

Допустимо.

13. Как создать кортеж с помощью спискового включения.

В отличие от выражения `[a for a in A ...]`, которое на выходе дает нам список, выражение `(a for a in A ...)` дает на выходе специальный объект генератора, а не кортеж. Для преобразования генератора в кортеж необходимо воспользоваться вызовом `tuple()`.

Вывод: были приобретены навыки по работе с кортежами при написании программ с помощью языка программирования Python3.