

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ФГАОУ высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

Кафедра инфокоммуникаций

Основы кроссплатформенного программирования

**Отчет по лабораторной работе №6
Работа с кортежами в языке Python**

Выполнил студент группы

ИТС-б-з-22-1

Потеев А.С. « » _____ 2023г.

Подпись студента _____

Работа защищена « » _____ 2023г.

Проверил доцент, кандидат технических
наук, доцент кафедры инфокоммуникаций
Воронкин Роман Александрович

(подпись)

Ставрополь, 2023 г.

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

Ход работы:

Создал общедоступный репозиторий на GitHub (<https://github.com/poteevprog/Lab6>)

Пример 1. Ввести кортеж A из 10 элементов, найти сумму элементов, меньших по модулю 5, и вывести ее на экран. Использовать в программе вместо списков кортежи.

```
individ1.py  primer1.py x
1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3
4  import sys
5
6  if __name__ == '__main__':
7      # Ввести кортеж одной строкой.
8      A = tuple(map(int, input().split()))
9      # Проверить количество элементов кортежа.
10     if len(A) != 10:
11         print("Неверный размер кортежа", file=sys.stderr)
12         exit(1)
13
14     # Найти искомую сумму.
15     s = 0
16     for item in A:
17         if abs(item) < 5:
18             s += item
19
20     print(s)
21
```

```
"C:\Users\user\Desktop\учеба\1 курс\2 семестр\Основы
5 6 7 88 8 5 67 5 1 11
1

Process finished with exit code 0
|
```

Рисунок 1 – Окно программы примера 1

Индивидуальное задание 1.

12. В начале кортежа записано несколько равных между собой элементов. Определить количество таких элементов и вывести все элементы, следующие за последним из них. Рассмотреть возможность того, что весь массив заполнен одинаковыми элементами. Условный оператор не использовать.

```
individual1.py × individ1.py
1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3
4  ► if __name__ == '__main__':
5      a = tuple(map(int, input("Введите кортеж: ").split()))
6
7      k = a.count(a[0])
8      index = a[0]
9      a1 = list(a) # переделываем кортеж в список
10
11     i = 0
12     while a1[i] == a[0]:
13         a1.remove(a[0])
14
15     if k == len(a):
16         print("Кортеж состоит из одинаковых элементов.")
17     else:
18         print("Количество первых элементов равных между собой: ", k)
19         print("Элементы после первых равных элементов: ", a1)
20
```

```
"C:\Users\user\Desktop\учеба\1 курс\2 семестр\Основы кросспла
Введите кортеж: 1 1 1 2 3 4
Количество первых элементов равных между собой: 3
Элементы после первых равных элементов: [2, 3, 4]

Process finished with exit code 0
```

Рисунок 2 – Окно программы для первой задачи и проверка кода на работоспособность.

Ответы на контрольные вопросы:

1. Что такое кортежи в языке Python?

Кортеж (tuple) – это неизменяемая структура данных, которая по своему подобию очень похожа на список.

2. Каково назначение кортежей в языке Python?

Чтобы обезопасить данные от случайного изменения. Если мы получили откуда-то массив данных, и у нас есть желание поработать с ним, но при этом непосредственно менять данные мы не собираемся, тогда, это как раз тот случай, когда кортежи придутся как нельзя кстати. Используя их в данной задаче, мы дополнительно получаем сразу несколько бонусов – во первых, это экономия места. Дело в том, что кортежи в памяти занимают меньший объем по сравнению со списками.

3. Как осуществляется создание кортежей?

функция tuple() лежит в основе создания кортежей

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?

кортежи часто содержат значения разных типов, и помнить, по какому индексу что лежит — очень непросто. Для упрощения этой задачи нужна деструктуризация

6. Какую роль играют кортежи в множественном присваивании?

Важную, т. к. с помощью него можно присвоить одной переменной множество значений

7. Как выбрать элементы кортежа с помощью среза?

Операция `T[i:j]` выбирает элементы от `i` до `j`

8. Как выполняется конкатенация и повторение кортежей?

Для кортежей можно выполнять операцию конкатенации, которая обозначается символом `+`. Так же кортеж может быть образован путем операции повторения, обозначаемой символом `*`.

9. Как выполняется обход элементов кортежа?

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

10. Как проверить принадлежность элемента кортежу?

С помощью Операции `in`

11. Какие методы работы с кортежами Вам известны?

Метод `index()` и метод `count()`.

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?

Синтаксис генератора списков устроен следующим образом:

```
new_list = [выражение for элемент in последовательность if условие]
```

Вывод: Я на лабораторной работе приобрел навыки по работе с кортежами при написании программ с помощью языка программирования Python версии 3.x.