

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

Институт цифрового развития  
Кафедра инфокоммуникаций

**«Работа с кортежами в языке Python»**

**ОТЧЕТ**  
**по лабораторной работе №8**  
**дисциплины**  
**«Основы программной инженерии»**

Выполнил:  
Зиёдуллаев Жавохир Эркин угли  
2 курс, группа ПИЖ-б-о-21-1,  
09.03.04 «Программная инженерия»,  
направленность (профиль) «Разработка  
и сопровождение программного  
обеспечения», очная форма обучения

---

(подпись)

Проверил:

---

(подпись)

Отчет защищен с оценкой \_\_\_\_\_ Дата защиты \_\_\_\_\_

Ставрополь, 2022 г.

```
#!/usr/bin/env python 3
# -*- coding: utf-8 -*-

import sys

if __name__ == '__main__':
    # Ввести кортеж одной строкой.
    A = tuple(map(int, input().split()))
    # проверить кол-во элементов кортежа.
    if len(A) != 10:
        print("Неверный размер кортежа", file=sys.stderr)
        exit(1)

    # Найти искомую сумму.
    s = 0
    for item in A:
        if abs(item) < 5:
            s += item
    print(s)
```

individ2 x 1task x

C:\Users\work\PycharmProjects\pythonProject2\venv\Scripts\python.exe C

4 5 3 2 9 8 10 1 2 7

12

Пример 2

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

import sys

if __name__ == '__main__':
    # ввести список одной строкой.
    A = list(map(int, input().split()))
    # Проверить код-во элементов списка.
    if len(A) != 10:
        print("Неверный размер списка", file=sys.stderr)
        exit(1)

    #Найти искомую сумму.
    s = sum(a for a in A if abs(a) < 5)
    print(s)
```

individ2 x 2task x

C:\Users\work\PycharmProjects\pythonProject2\venv\Scripts\python.exe C:\Users\work\PycharmProjects\pythonProject2\venv\Scripts\python.exe

2 5 4 3 7 8 9 10 1 6

10

## Индивидуальное задание

5. Если в кортеже есть хотя бы одна пара одинаковых соседних элементов, то напечатать все элементы, следующие за элементами первой из таких пар.

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

a = [int(i) for i in input().split()]
res = []
for i in range(len(a) - 1):
    if (a[i] > 0 and a[i + 1] > 0) or (a[i] < 0 and a[i + 1] < 0):
        res.append(a[i + 1])
        res.append(a[i + 1])
print(res[0], res[1])
```

individ2 x indiv1 x

C:\Users\work\PycharmProjects\pythonProject2\venv\Scripts\python.exe C:\Users\work\PycharmProjects\pythonProject2\venv\Scripts\python.exe

3 4 5 6

4 4

Ссылка: <https://github.com/javoxir21/laba-8.gits>

### Вопросы для защиты работы:

1. Что такое кортеж в языке Python? Кортеж – неизменяемая структура данных, которая по своему подобию очень похожа на список
2. Каково назначение кортежей в языке Python? В целях экономии места, безопасности данных от случайного изменения
3. Как осуществляется создание кортежей?

```
>>> a = ()
>>> print(type(a))
<class 'tuple'>
>>> b = tuple()
>>> print(type(b))
<class 'tuple'>
```

4. Как осуществляется доступ к элементам кортежа? Доступ к элементам кортежа осуществляется как к элементам списка – через указание индекса
5. Зачем нужна распаковка (деструктуризация) кортежа? Обращение по индексу не самый удобный способ работы с кортежами. Кортежи часто содержат значения разных типов, и помнить, по какому индексу что лежит – непросто. Можно разобрать кортеж

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

(name, age) = name_and_age
name # 'Bob'
age  # 42
```

Именно таким способом принято получать и сразу разбирать значения, которые возвращает функция (если таковая возвращает несколько значений, конечно):

```
(quotient, modulo) = div_mod(13, 4)
```

Соответственно кортеж из одного элемента нужно разбирать так:

```
(a,) = (42,)
a # 42
```

6. Какую роль играют кортежи в множественном присваивании? Используя множественное присваивание можно совершить обмен между двумя переменными

```

a = 100
b = 'foo'

(a, b) = (b, a)
a # 'foo'
b # 100

```

Строку `(a, b) = (b, a)` нужно понимать как "присвоить в `a` и `b` значения из кортежа, состоящего из значений переменных `b` и `a`".

## 7. Как выбрать элементы кортежа с помощью среза?

С помощью операции взятия среза можно получить другой кортеж. Общая форма операции взятия среза для кортежа следующая

```
T2 = T1[i:j]
```

здесь

- `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, ..., j-1`. Значение `j` определяет позицию за последним элементом среза.

Операция взятия среза для кортежа может иметь модификации такие же как и для списков.

Например

```

# Операция [i:j] - взятие среза
# 1. кортеж, содержащий целые числа
A = ( 0, 1, 2, 3)
item = A[0:2] # item = (0, 1)

```

```

# 2. кортеж, содержащий список
A = ( 2.5, ['abcd', True, 3.1415], 8, False, 'z')
item = A[1:3] # item = (['abcd', True, 3.1415], 8)

# 3. кортеж, содержащий вложенный кортеж
A = (3, 8, -11, "program")
B = ("Python", A, True)
item = B[3:] # item = ('Python', (3, 8, -11, 'program'), True)
item = B[1:] # item = ((3, 8, -11, 'program'), True)

```

## 8. Как выполняется конкатенация и повторение кортежей?

```
T3 = T1 + T2
```

где

- $T1, T2$  – кортежи, для которых нужно выполнить операцию конкатенации. Операнды  $T1, T2$  обязательно должны быть кортежами. При выполнении операции конкатенации для кортежей, использовать в качестве операндов любые другие типы (строки, списки) запрещено;
- $T3$  – кортеж, который есть результатом.

## 9. Как выполняется обход элементов кортежа? Элементы кортежа можно последовательно просмотреть с помощью операторов цикла while или for

## 10. Как проверить принадлежность элемента кортежу?

Проверить принадлежность элемента можно с помощью операции in

## 11. Как методы работы с кортежами вам известны? Для поиска позиции элемента в кортеже – метод index(). Количество вхождений элемента в кортеж – метод count()

## 12. Допустимо ли использование функций агрегации таких как len() sum() при работе с кортежами? Да, допустимо

## 13. Как создать кортеж с помощью спискового включения? $x = 10$

```
a = tuple([i for i in range(x)])
```

```
a = tuple(int(i) for i in input().split())
```