

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 №8**  
**дисциплины «Программирование на Python»**

Выполнил:  
Середа Кирилл Витальевич  
1 курс, группа ИВТ-б-о-22-1,  
09.03.01 «Информатика и  
вычислительная техника», очная  
форма обучения

---

(подпись)

Руководитель практики:  
Воронкин Роман Александрович

---

(подпись)

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

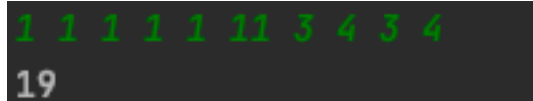
Ставрополь, 2023 г.

## Тема: Работа с кортежами в языке Python

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

### Ход выполнения:

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.



```
1 1 1 1 1 11 3 4 3 4
19
```

Рисунок 1- Результат выполнения программы

### Код программы:

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

import sys

if __name__ == '__main__':
    # Ввести кортеж одной строкой.
    A = tuple(map(int, input().split()))
    # Проверить количество 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)
```

### Код программы с использованием списковых включений:

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

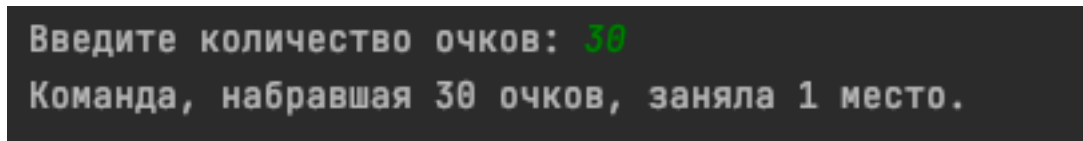
import sys

if __name__ == '__main__':
    # Ввести список одной строкой.
    A = list(map(int, input().split()))
    # Проверить количество 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)
```

2) Индивидуальное задание 1 – Вариант 1 - Известно количество очков, набранных каждой из 20 команд – участниц первенства по футболу. Перечень очков дан в порядке убывания (ни одна пара команд не набрала

одинаковое количество очков). Определить, какое место заняла команда, набравшая *n* очков (естественно, что значение *n* имеется в перечне). Условный оператор не использовать.



Введите количество очков: 30  
Команда, набравшая 30 очков, заняла 1 место.

Рисунок 2 – Результат выполнения программы

Код программы:

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

import sys

if __name__ == '__main__':

    teams_and_points = [
        ("Команда1", 30),
        ("Команда2", 25),
        ("Команда3", 20),
    ]

    n = int(input("Введите количество очков: "))

    sorted_teams = sorted(teams_and_points, key=lambda x: x[1], reverse=True)

    place = sorted_teams.index(next(filter(lambda x: x[1] == n,
sorted_teams))) + 1

    print(f"Команда, набравшая {n} очков, заняла {place} место.")
```

Ответы на вопросы:

1. Что такое списки в языке Python? Список в языке Python представляет собой структуру данных, предназначенную для хранения упорядоченной последовательности объектов различных типов. Список является изменяемым типом данных, позволяя добавлять, удалять и изменять элементы.
2. Каково назначение кортежей в языке Python? Кортеж в Python - это неизменяемая упорядоченная последовательность элементов. Кортежи часто используются для представления неизменяемых коллекций данных.
3. Как осуществляется создание кортежей? Кортеж создается перечислением элементов в круглых скобках: `my_tuple = (1, 2, 3)`. Пустой кортеж создается при помощи `my_empty_tuple = ()`. Один элемент кортежа задается как `single_element_tuple = (1,)`.

4. Как осуществляется доступ к элементам кортежа? Доступ к элементам кортежа осуществляется по индексу: `element = my_tuple[0]`.

5. Зачем нужна распаковка (деструктуризация) кортежа? Распаковка кортежа позволяет присваивать значения элементов кортежа переменным одновременно: `a, b, c = my_tuple`.

6. Какую роль играют кортежи в множественном присваивании? В множественном присваивании кортежи позволяют эффективно и удобно присваивать значения нескольким переменным одновременно.

7. Как выбрать элементы кортежа с помощью среза? Выбор элементов кортежа с помощью среза осуществляется также, как и в списках: `subset_tuple = my_tuple[1:3]`.

8. Как выполняется конкатенация и повторение кортежей? Кортежи могут быть объединены с использованием оператора `+` для конкатенации и оператора `*` для повторения: `concatenated_tuple = tuple1 + tuple2`, `repeated_tuple = my_tuple * 3`.

9. Как выполняется обход элементов кортежа? Обход элементов кортежа может быть выполнен с использованием цикла `for`: `for item in my_tuple:`.

10. Как проверить принадлежность элемента кортежу? Принадлежность элемента кортежу можно проверить с использованием операторов `in` и `not in`: `if element in my_tuple:`.

11. Какие методы работы с кортежами Вам известны? Некоторые методы кортежей включают `count()` для подсчета вхождений элемента и `index()` для нахождения индекса первого вхождения элемента.

12. Допустимо ли использование функций агрегации таких как `len()`, `sum()` и т. д. при работе с кортежами? Да, функции агрегации, такие как `len()`, `sum()`, могут быть использованы при работе с кортежами для получения их длины или суммы элементов.

13. Как создать кортеж с помощью спискового включения. Кортеж можно создать с помощью спискового включения и функции `tuple()`:  
`my_tuple = tuple(x for x in my_list)`

Вывод: в ходе выполнения лабораторной работы были изучены кортежи, а также методы работы с ними и проработаны примеры их использования. Приобретены навыки по работе с кортежами при написании программ с помощью языка программирования Python версии 3.x.