

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 «Основы программной инженерии»

Выполнила:
Ламская Ксения Вячеславовна
2 курс, группа ПИЖ-б-о-22-1,
09.03.04 «Программная инженерия»,
направленность (профиль) «Разработка и
сопровождение программного
обеспечения», очная форма обучения

(подпись)

Доцент кафедры инфокоммуникаций
Воронкин Роман Александрович

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2023 г.

Тема: работа со кортежами в языке Python.

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

Порядок выполнения работы

1. Создание репозитория GitHub.

Create a new repository



A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk ().*

Owner *	Repository name *
<div>ksenia-lamskaya ▾</div>	<div>8laba</div>
✔ 8laba is available.	

Great repository names are short and memorable. Need inspiration? How about [glowing-goggles](#) ?

Description (optional)

- ☒  **Public**
Anyone on the internet can see this repository. You choose who can commit.
- ☐  **Private**
You choose who can see and commit to this repository.

Initialize this repository with:

- ☒ **Add a README file**
This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore

.gitignore template: Python ▾

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license

License: MIT License ▾

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

This will set  **main** as the default branch. Change the default name in your [settings](#).

 You are creating a public repository in your personal account.

Create repository

Рисунок 1 – Создания репозитория

2. Проработала примеры из лабораторной работы.

```
#!/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)
```

Рисунок 2.1 – Код из примера 1

```
1 5 -3 6 9 9 -3 6 14 50
-5
```

Рисунок 2.2 – Вывод программы из примера 1

3. Если в кортеже есть хотя бы одна тройка соседних чисел, в которой средний элемент больше своих «соседей», т. е. предшествующего и последующего, то напечатать все элементы, предшествующие элементам последней из таких троек.

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

import sys

if __name__ == '__main__':
    num = tuple(map(int, input('Введите числа: ').split()))
    result = []
    for i, char in enumerate(num):
        if i + 2 > len(num):
            break
        elif num[i-1] < num[i] > num[i+1]:
            result = num[:i-1]

    if result:
        print(result)
    else:
        print("Тройки соседних чисел с условием не найдено.")
```

Рисунок 3.1 – Код программы

```
Введите числа: 1 1 1 1 3 4 2
(1, 1, 1, 1, 3)
PS C:\Ksen\8laba\task> & C:/Use
k/first.py
Введите числа: 1 1 1 1 3 4 2
(1, 1, 1, 1)
PS C:\Ksen\8laba\task> & C:/Use
k/first.py
Введите числа: 1 1 3 4 2 5 6 4
(1, 1, 3, 4, 2)
```

Рисунок 3.2 – Вывод программы

Ответы на контрольные вопросы

1. Что такое списки в языке Python?

Список – это структура данных, которая хранит различные объекты.

2. Каково назначение кортежей в языке Python?

Кортежи используются для хранения неизменяемых последовательностей объектов разных типов.

3. Как осуществляется создание кортежей?

Кортежи создаются с использованием круглых скобок и элементы разделяются запятыми. Например: `my_tuple = (1, 2, 3)`. Или с помощью функции `tuple`. Например: `my_tuple = tuple()`.

4. Как осуществляется доступ к элементам кортежа?

Доступ к элементам кортежа осуществляется по индексу.

5. Зачем нужна распаковка (деструктуризация) кортежа?

Распаковка (деструктуризация) кортежа позволяет присвоить его элементы отдельным переменным.

6. Какую роль играют кортежи в множественном присваивании?

Используя множественное присваивание, можно проверить интересный трюк: обмен значениями между двумя переменными.

7. Как выбрать элементы кортежа с помощью среза?

Для выбора элементов кортежа с помощью среза используется синтаксис `[start:end]`.

8. Как выполняется конкатенация и повторение кортежей?

Конкатенация кортежей выполняется с помощью оператора `+`. Повторение кортежа выполняется с помощью оператора `*`.

9. Как выполняется обход элементов кортежа?

Обход элементов кортежа может осуществляться с помощью циклов.

10. Как проверить принадлежность элемента кортежу?

Принадлежность элемента кортежу можно проверить с помощью оператора `in`.

11. Какие методы работы с кортежами Вам известны?

count(), index().

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

Да, функции агрегации такие как len(), sum() и т.д. могут быть использованы при работе с кортежами.

13. Как создать кортеж с помощью спискового включения.

Кортежи могут быть созданы с помощью спискового включения.
Например: my_tuple = tuple(i for i in range(10)).