

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 №2.5**  
**дисциплины «Основы кроссплатформенного программирования»**  
**Вариант \_\_\_\_**

Выполнил:

Быковская Стефания Станиславовна  
1 курс, группа ИТС-б-0-22-1,  
11.03.02 «Инфокоммуникационные  
технологии и системы связи», направ-  
ленность (профиль) «Инфокоммуника-  
ционные системы и сети», очная форма  
обучения

---

(подпись)

Руководитель практики:

Воронкин Р.А. доцент кафедры инфо-  
коммуникаций

---

(подпись)

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

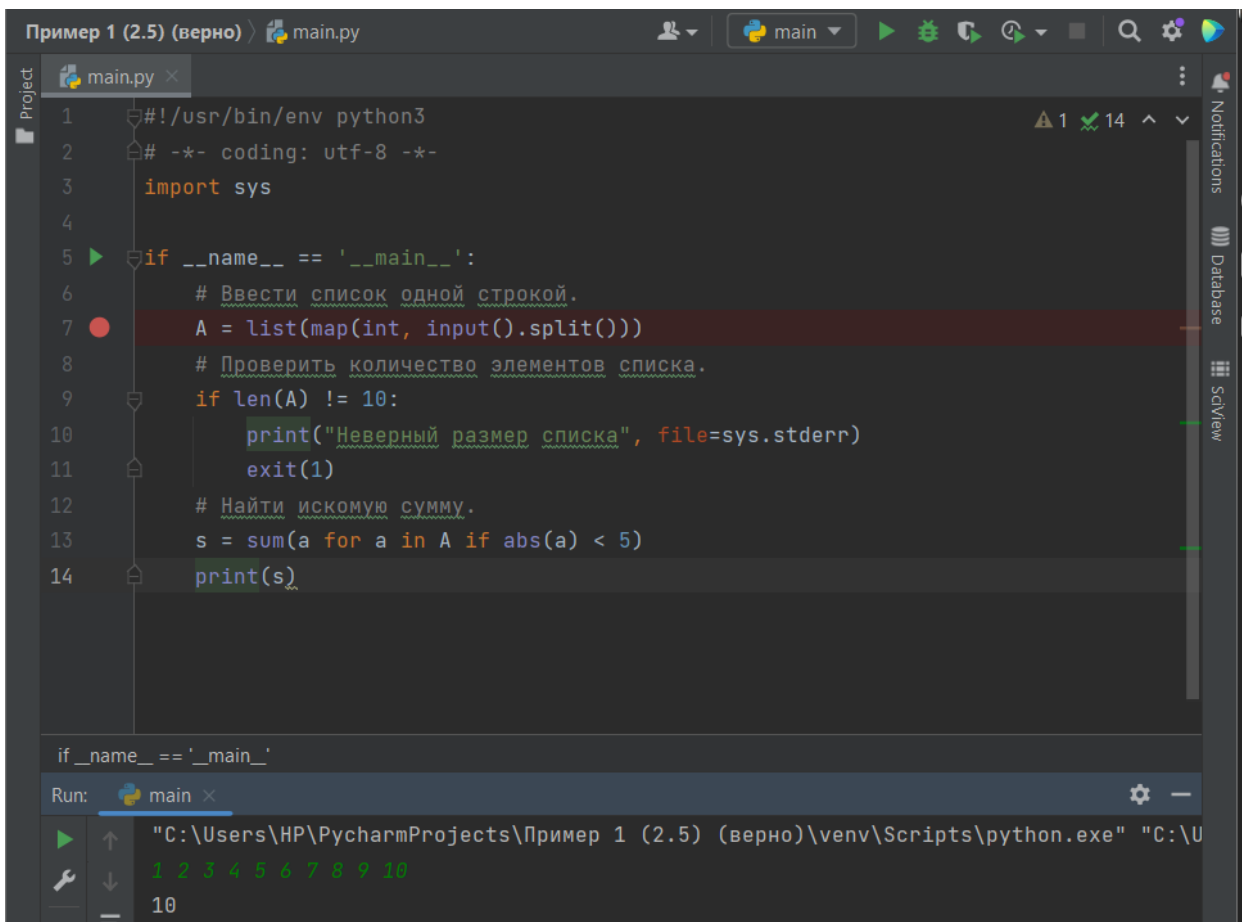
Ставрополь, 2023 г.

**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.

Порядок выполнения работы:

**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.



```
1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 import sys
4
5 if __name__ == '__main__':
6     # Ввести список одной строкой.
7     A = list(map(int, input().split()))
8     # Проверить количество элементов списка.
9     if len(A) != 10:
10         print("Неверный размер списка", file=sys.stderr)
11         exit(1)
12     # Найти искомую сумму.
13     s = sum(a for a in A if abs(a) < 5)
14     print(s)
```

Run: main ×

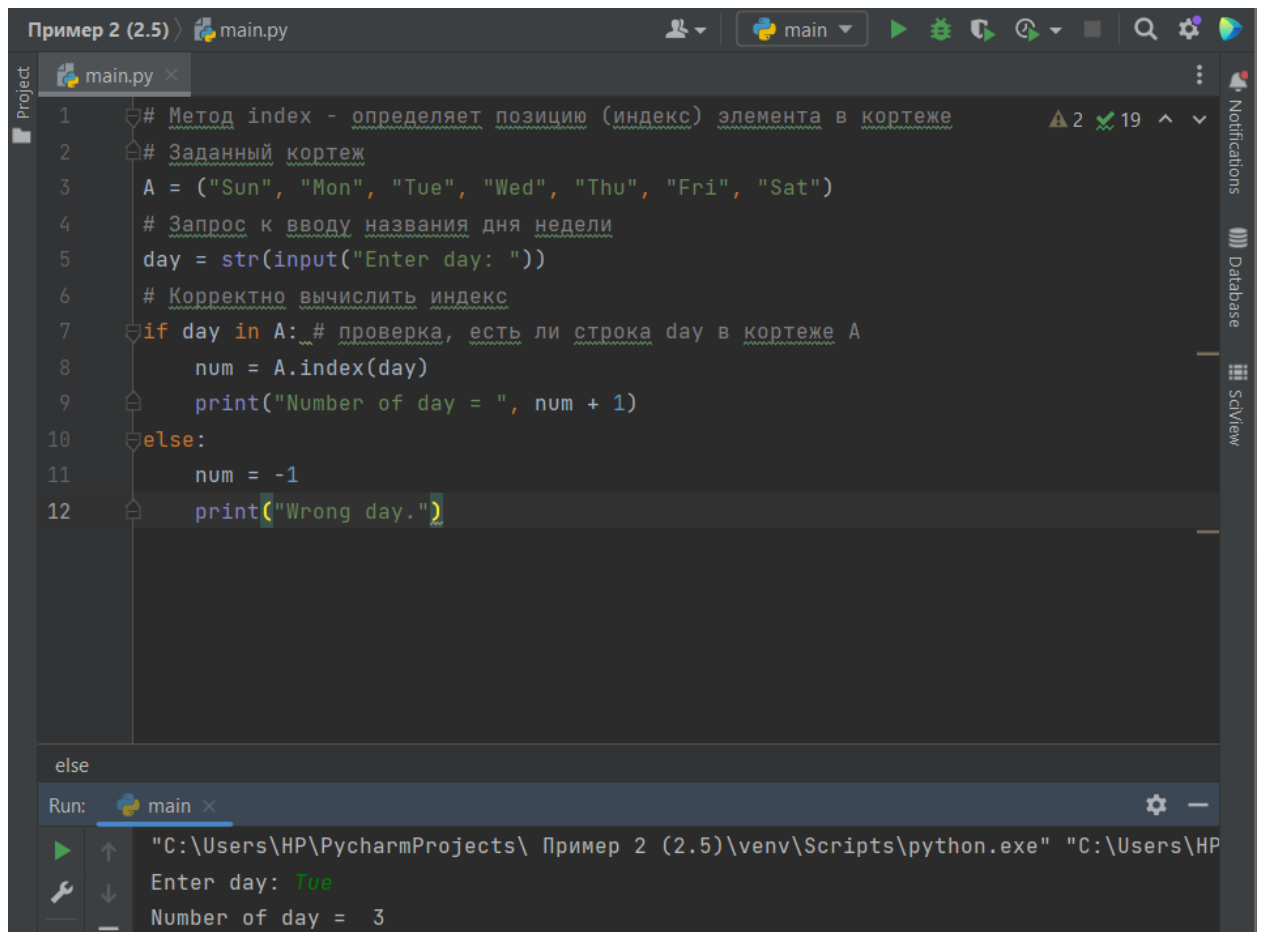
"C:\Users\HP\PycharmProjects\Пример 1 (2.5) (верно)\venv\Scripts\python.exe" "C:\U

1 2 3 4 5 6 7 8 9 10

10

*Рисунок 1 – Выполнение примера 1.*

**Пример 2.** В примере, в перечне названий дней недели вычисляется порядковый номер дня.



The screenshot shows the PyCharm IDE interface. The main editor window displays a Python script named `main.py` with the following code:

```
1 # Метод index - определяет позицию (индекс) элемента в кортеже
2 # Заданный кортеж
3 A = ("Sun", "Mon", "Tue", "Wed", "Thu", "Fri", "Sat")
4 # Запрос к вводу названия дня недели
5 day = str(input("Enter day: "))
6 # Корректно вычислить индекс
7 if day in A: # проверка, есть ли строка day в кортеже A
8     num = A.index(day)
9     print("Number of day = ", num + 1)
10 else:
11     num = -1
12     print("Wrong day.")
```

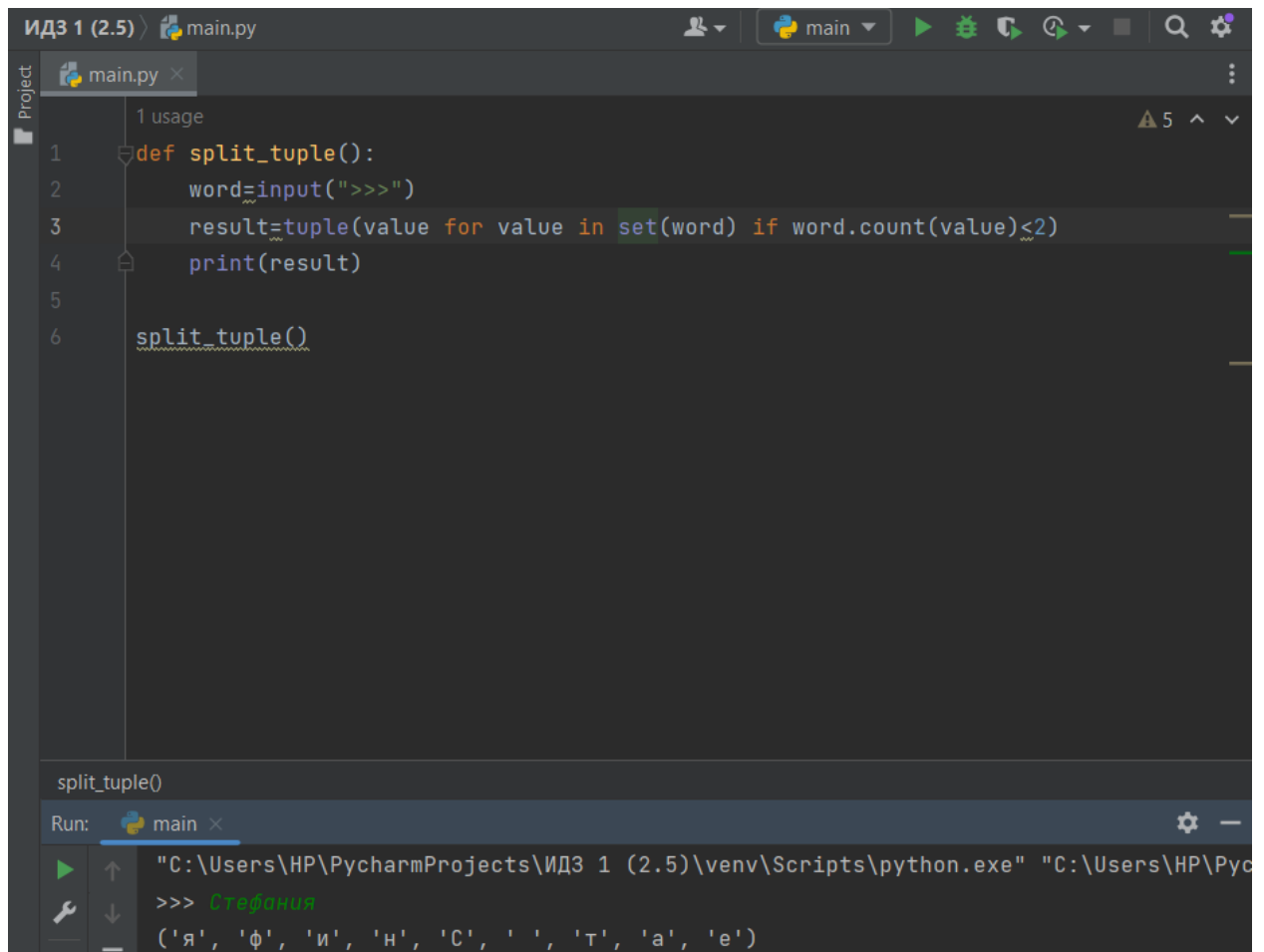
Below the editor, the Run console shows the execution of the script. The command prompt is `"C:\Users\HP\PycharmProjects\ Пример 2 (2.5)\venv\Scripts\python.exe" "C:\Users\HP\`. The input is `Enter day: Tue`, and the output is `Number of day = 3`.

*Рисунок 2 – Выполнение примера 2.*

### *Индивидуальное задание*

#### *Вариант 5*

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



The screenshot shows the PyCharm IDE interface. The top toolbar includes icons for running, debugging, and other development tools. The main editor window displays a Python file named `main.py` with the following code:

```
1 usage
2
3 def split_tuple():
4     word=input(">>>")
5     result=tuple(value for value in set(word) if word.count(value)<2)
6     print(result)
7
8 split_tuple()
```

Below the editor, the 'Run' console shows the execution of the `split_tuple()` function. The output is a tuple of unique characters from the input word 'Стефания', where each character appears only once:

```
>>> Стефания
('я', 'ф', 'и', 'н', 'с', ' ', 'т', 'а', 'е')
```

*Рисунок 3 – Выполнение ИДЗ.*

### Ответы на 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. В Python не обязательно, чтобы все элементы списка были одного типа.

2) Каково назначение кортежей в языке Python?

Кортеж в Python - это более быстрый и неизменяемый аналог списка. Он очень часто используется для защиты хранимых данных приложения от незапланированных или непреднамеренных изменений.

3) Как осуществляется создание кортежей?

Кортеж создается путем помещения всех элементов (элементов) в круглые скобки (), разделенных запятыми.

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?

Деструктуризация кортежа позволяет извлекать отдельные элементы кортежа и присваивать их значения переменным. Часто кортежи содержат значения разных типов. Сложно запомнить каким индексом обозначается каждое значение. Чтобы упростить работу, можно разобрать кортеж.

6) Какую роль играют кортежи в множественном присваивании?

Кортежи играют важную роль в множественном присваивании, потому что они могут быть использованы для одновременного присваивания нескольких переменных из кортежа.

7) Как выбрать элементы кортежа с помощью среза?

Элементы кортежа можно выбрать с помощью среза, указав индексы начала и конца нужного диапазона через двоеточие в квадратных скобках.

8) Как выполняется конкатенация и повторение кортежей?

Конкатенация (объединение) кортежей в Python выполняется с помощью оператора "+". Для повторения кортежа используется оператор "\*".

9) Как выполняется обход элементов кортежа?

Обход элементов кортежа может быть выполнен с помощью цикла for.

10) Как проверить принадлежность элемента кортежу?

Для проверки нахождения элемента в кортеже можно использовать оператор in.

11) Какие методы работы с кортежами Вам известны?

— count(x) - возвращает количество вхождений элемента x в кортеж.

— index(x) - возвращает индекс первого вхождения элемента x в кортеже.

— len(t) - возвращает длину кортежа t.

— `sorted()` - возвращает новый отсортированный список из элементов кортежа.

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

Да, использование функций агрегации, таких как `len()`, `sum()`, `min()`, `max()` и т.д. является допустимым при работе с кортежами в Python, так как они позволяют получать информацию о элементах кортежа и выполнять различные математические операции над ними.

13) Как создать кортеж с помощью спискового включения.

Для создания кортежа с помощью спискового включения можно использовать функцию `tuple()`, которая преобразует список в кортеж.

**Выводы:** приобрела навыки по работе с кортежами при написании программ с помощью языка программирования Python 3.x.