

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

Дубинин Олег Александрович

« » _____ 2023г.

Подпись студента _____

Работа защищена « » _____ 2023г.

Проверил доцент, кандидат технических
наук, доцент кафедры инфокоммуникаций

Воронкин Роман Александрович

(подпись)

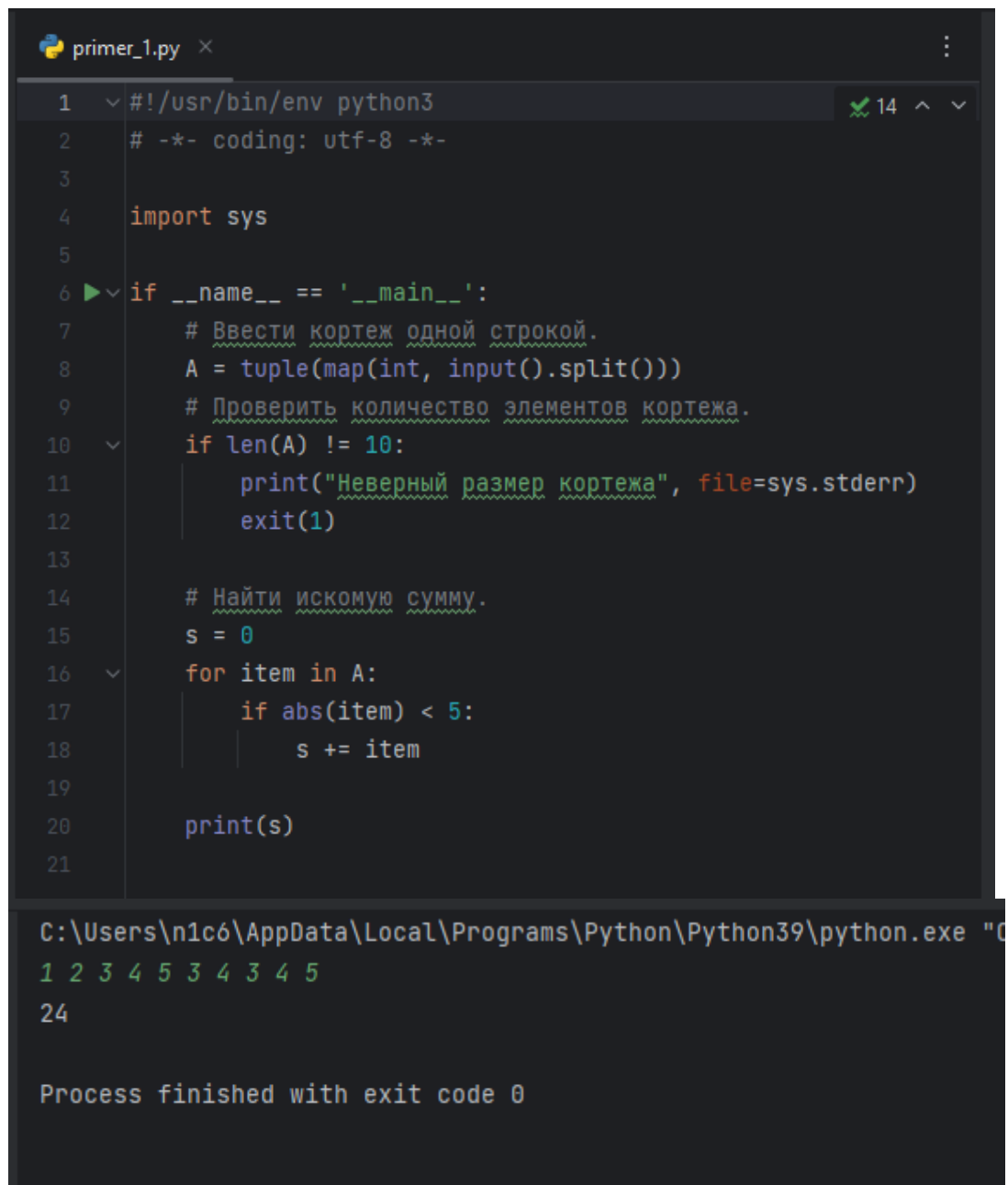
Ставрополь, 2023 г.

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

Ход работы:

<https://github.com/OLEG1232155/6> - репозиторий

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
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\n1c6\AppData\Local\Programs\Python\Python39\python.exe "C:\Users\n1c6\AppData\Local\Programs\Python\Python39\primer_1.py"

1 2 3 4 5 3 4 3 4 5

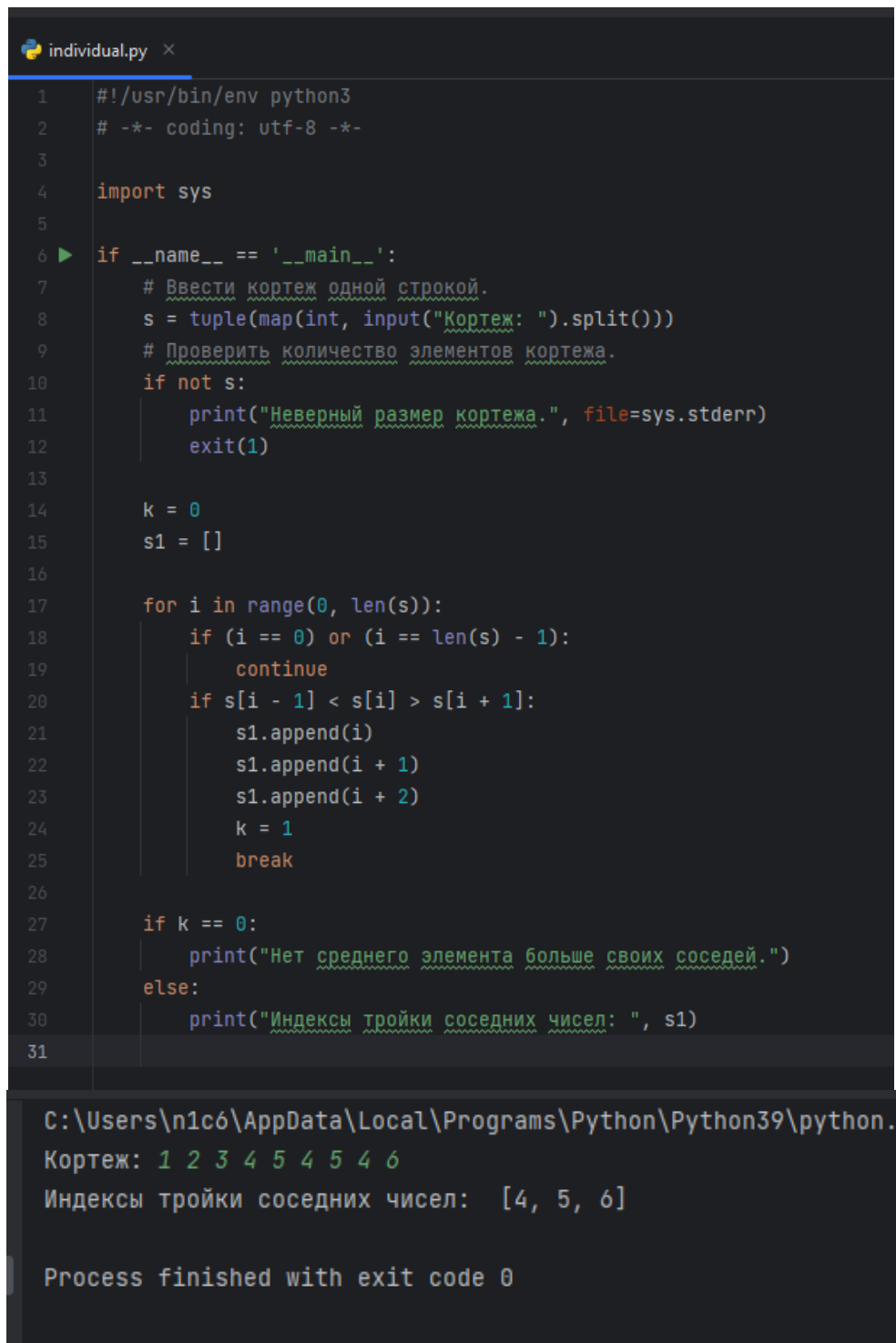
24

Process finished with exit code 0

Рисунок 1 – Окно программы первого примера

Индивидуальное задание 1.

8. Определить, есть ли в кортеже хотя бы одна тройка соседних чисел, в которой средний элемент больше своих «соседей», т. е. предшествующего и последующего. В случае положительного ответа определить номера элементов первой из таких троек.



```
individual.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      s = tuple(map(int, input("Кортеж: ").split()))
9      # Проверить количество элементов кортежа.
10     if not s:
11         print("Неверный размер кортежа.", file=sys.stderr)
12         exit(1)
13
14     k = 0
15     s1 = []
16
17     for i in range(0, len(s)):
18         if (i == 0) or (i == len(s) - 1):
19             continue
20         if s[i - 1] < s[i] > s[i + 1]:
21             s1.append(i)
22             s1.append(i + 1)
23             s1.append(i + 2)
24             k = 1
25             break
26
27     if k == 0:
28         print("Нет среднего элемента больше своих соседей.")
29     else:
30         print("Индексы тройки соседних чисел: ", s1)
31
```

C:\Users\n1c6\AppData\Local\Programs\Python\Python39\python.
Кортеж: 1 2 3 4 5 4 5 4 6
Индексы тройки соседних чисел: [4, 5, 6]

Process finished with exit code 0

Рисунок 2 – Окно программы для индивидуального задания 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 – вопервых, 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: приобрел навыки по работе с кортежами при написании программ с помощью языка программирования Python версии 3.x.