

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙ-
СКОЙ ФЕДЕРАЦИИ**

**Федеральное государственное автономное
образовательное учреждение высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

Кафедра инфокоммуникаций

Основы кроссплатформенного программирования

Отчет по лабораторной работе №2.5

Тема: «Работа с кортежами в языке Python»

Выполнил студент группы

ИВТ-б-о-21-1

Артемьев А.В « » _____ 20__ г.

Подпись студента _____

Работа защищена « » _____ 20__ г.

Проверил доцент

Кафедры инфокоммуникаций, старший
преподаватель

Воронкин Р.А.

(подпись)

Ставрополь 2022

1. Создал репозиторий в GitHub, дополнил правила в .gitignore для работы с IDE PyCharm с ЯП Python, выбрал лицензию MIT, клонировал его на компьютер и организовал в соответствии с моделью ветвления git-flow.

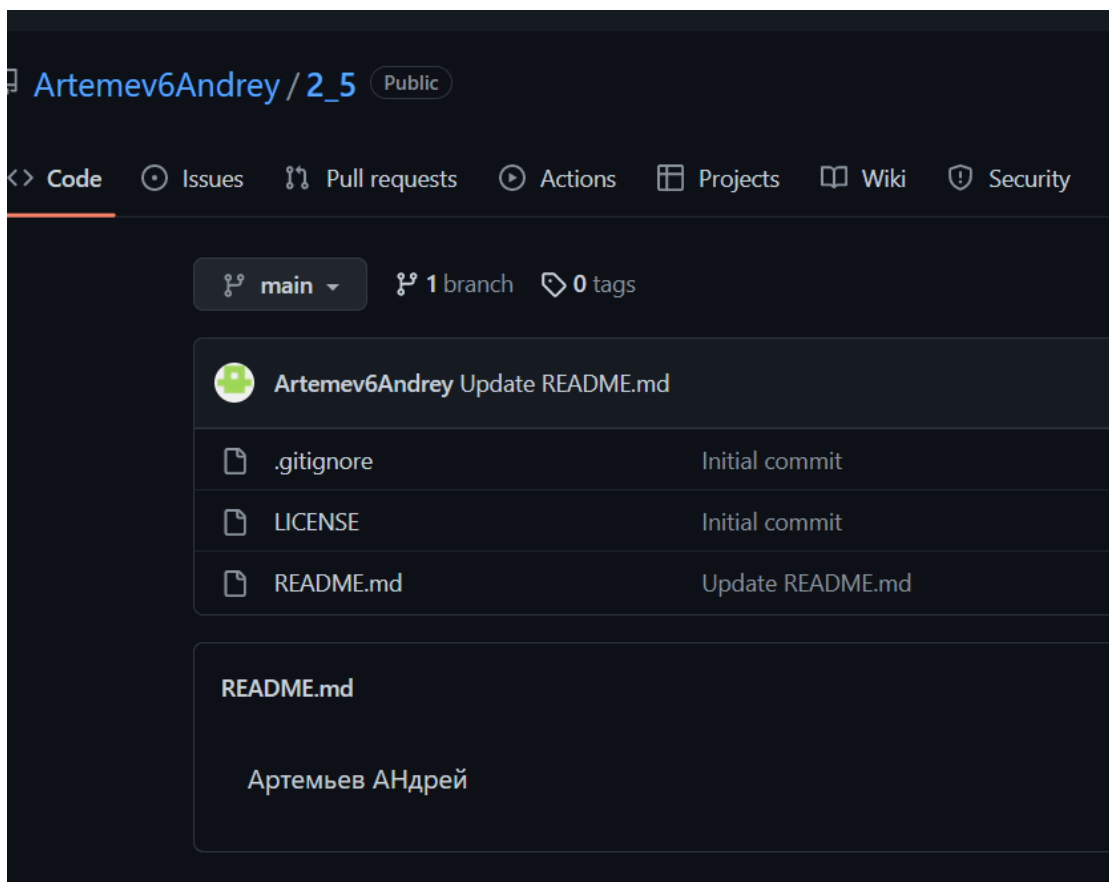


Рисунок 1.1 – Создал и настроил репозиторий

```

aa715@DESKTOP-70QJC9F MINGW64 ~/Desktop/2.5 (main)
$ git clone https://github.com/Artemev6Andrey/2_5.git
Cloning into '2_5'...
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (5/5), done.

aa715@DESKTOP-70QJC9F MINGW64 ~/Desktop/2.5 (main)
$ git flow init
Already initialized for gitflow.
To force reinitialization, use: git flow init -f

```

Рисунок 1.3 – Скопировал репозиторий на ПК

2. Создал проект PyCharm в папке репозитория, проработал примеры ЛР

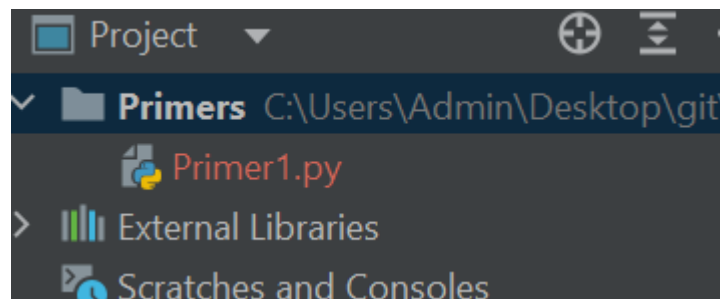


Рисунок 2.1 – Созданные проекты

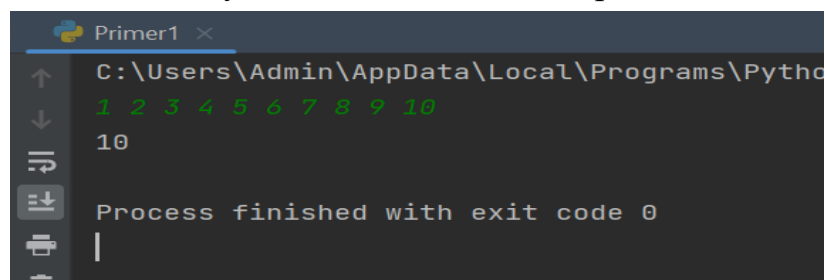
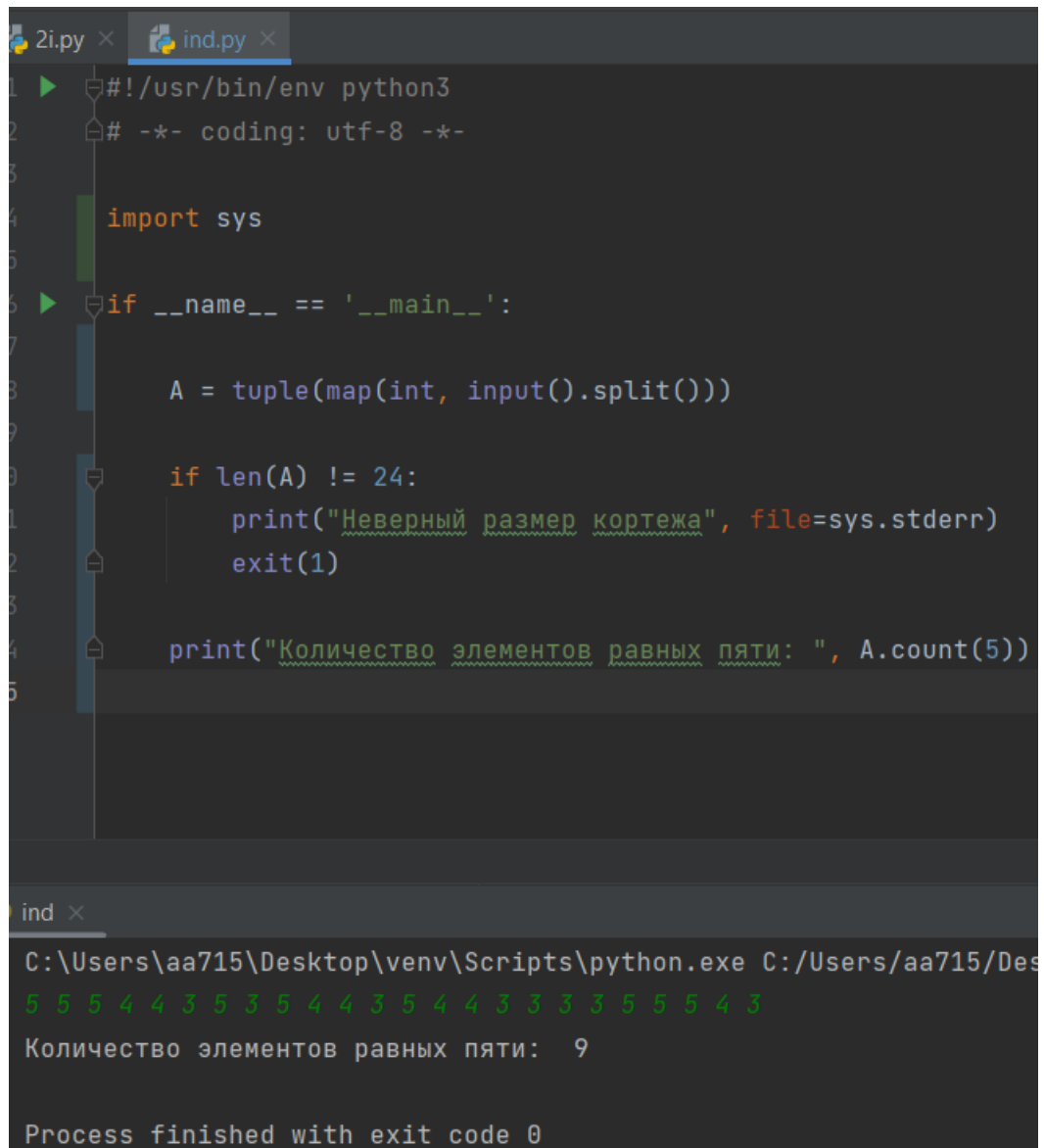


Рисунок 2.2 – Результат выполнения Примера 1

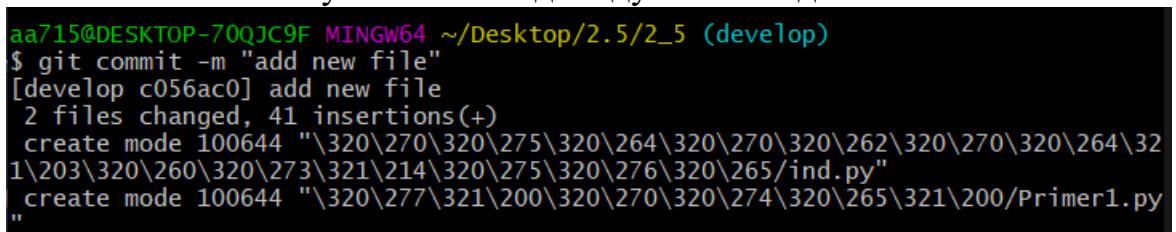
3. Выполнил индивидуальное задание.

Известны оценки по геометрии каждого из 24 учеников класса. В начале списка перечислены все пятерки, затем все остальные оценки. Сколько учеников имеет по геометрии оценку «5»? Условный оператор не использовать.



```
2i.py x ind.py x
1  > #!/usr/bin/env python3
2  > # -*- coding: utf-8 -*-
3
4  > import sys
5
6  > if __name__ == '__main__':
7
8      > A = tuple(map(int, input().split()))
9
10     > if len(A) != 24:
11         > print("Неверный размер кортежа", file=sys.stderr)
12         > exit(1)
13
14     > print("Количество элементов равных пяти: ", A.count(5))
15
ind x
C:\Users\aa715\Desktop\venv\Scripts\python.exe C:/Users/aa715/Des
5 5 5 4 4 3 5 3 5 4 4 3 5 4 4 3 3 3 3 5 5 5 4 3
Количество элементов равных пяти: 9
Process finished with exit code 0
```

Рисунок 3.1 – Индивидуальное задание



```
aa715@DESKTOP-70QJC9F MINGW64 ~/Desktop/2.5/2_5 (develop)
$ git commit -m "add new file"
[develop c056ac0] add new file
2 files changed, 41 insertions(+)
create mode 100644 "\320\270\320\275\320\264\320\270\320\262\320\270\320\264\32
1\203\320\260\320\273\321\214\320\275\320\276\320\265\ind.py"
create mode 100644 "\320\277\321\200\320\270\320\274\320\265\321\200\Primer1.py"
"
```

Рисунок 4.1 – Сделал коммит всех изменений

```

aa715@DESKTOP-70QJC9F MINGW64 ~/Desktop/2.5/2_5 (main)
$ git merge develop
Updating 143869a..c056ac0
Fast-forward
.../ind.py" | 20 ++++++++++++++++++++++
.../Primer1.py" | 21 ++++++++++++++++++++++
2 files changed, 41 insertions(+)
create mode 100644 "\\320\\270\\320\\275\\320\\264\\320\\270\\320\\262\\320\\270\\320\\264\\320\\270\\320\\260\\320\\273\\321\\214\\320\\275\\320\\276\\320\\265\\ind.py"
create mode 100644 "\\320\\277\\321\\200\\320\\270\\320\\274\\320\\265\\321\\200\\Primer1.py"

```

Рисунок 4.2 – Переход на ветку main и последующее её слияние с develop

```

aa715@DESKTOP-70QJC9F MINGW64 ~/Desktop/2.5/2_5 (main)
$ git push --force
Enumerating objects: 11, done.
Counting objects: 100% (11/11), done.
Delta compression using up to 8 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (11/11), 3.21 KiB | 3.21 MiB/s, done.
Total 11 (delta 1), reused 4 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/Artemev6Andrey/2_5.git
+ 01b5063...c056ac0 main -> main (forced update)

```

Рисунок 4.3 – Отправка изменений на удалённый репозиторий

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?

Существует несколько причин, по которым стоит использовать кортежи вместо списков. Одна из них — это 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. Кортежи в памяти занимают меньший объем по сравнению со списками. Кортежи работают быстрее, чем списки

3. Как осуществляется создание кортежей?

```
a = ()
```

```
b = tuple()
```

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?

Обращение по индексу, это не самый удобный способ работы с кортежами. Дело в том, что кортежи часто содержат значения разных типов, и помнить, по какому индексу что лежит — очень непросто.

6. Какую роль играют кортежи в множественном присваивании?

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

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.

8. Как выполняется конкатенация и повторение кортежей?

Для кортежей можно выполнять операцию конкатенации, которая обозначается символом +. $T3 = T1 + T2$

9. Как выполняется обход элементов кортежа?

Элементы кортежа можно последовательно просмотреть с помощью операторов цикла while или for.

10. Как проверить принадлежность элемента кортежу?

Проверка вхождения элемента в кортеж - оператор in.

11. Какие методы работы с кортежами Вам известны?

index(), 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.

Так же, как и список.

Вывод: научился работать с кортежами в языке программирования Python.