**МИНИCTEPCTBO НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ**

**ФЕДЕРАЦИИ**

**Федеральное государственное автономное образовательное учреждение высшего образования**

**«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

**Кафедра инфокоммуникаций**

**«Работа с кортежами в языке Python»**

**Отчет по лабораторной работе № 2.5**

**по дисциплине «Основы программной инженерии»**

|  |
| --- |
| Выполнил студент группы ПИЖ-б-о-21-1 |
| Кучеренко С. Ю. « » 2022г. |
| Подпись студента |
| Работа защищена « » 2022г. |
| Проверил Воронкин Р.А.  (подпись) |

Ставрополь 2022

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

**Выполнение работы:**

1. Изучить теоретический материал работы.
2. Создать общедоступный репозиторий на GitHub, в котором будет использована лицензия MIT и язык программирования Python.

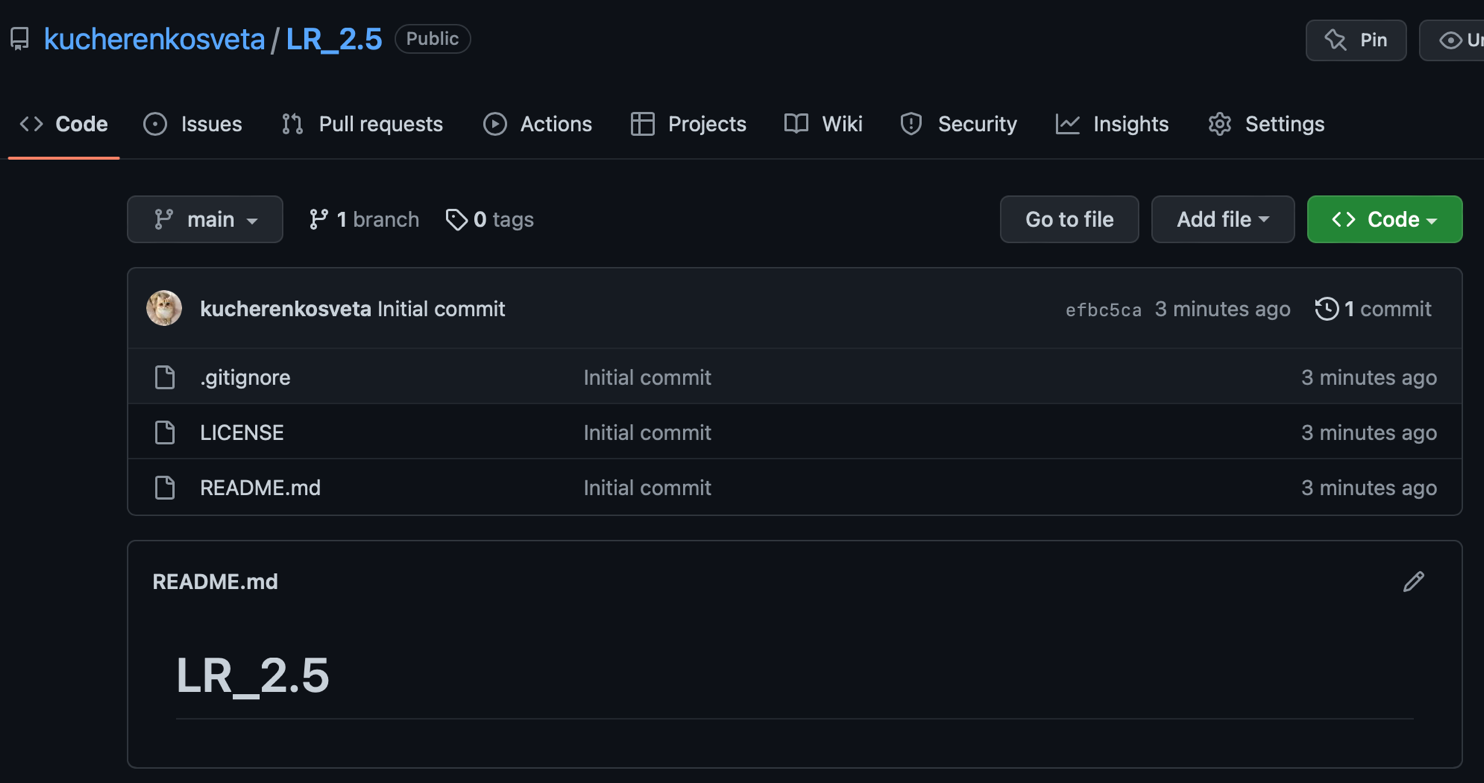


Рисунок 1 – Создание репозитория

1. Выполните клонирование созданного репозитория.

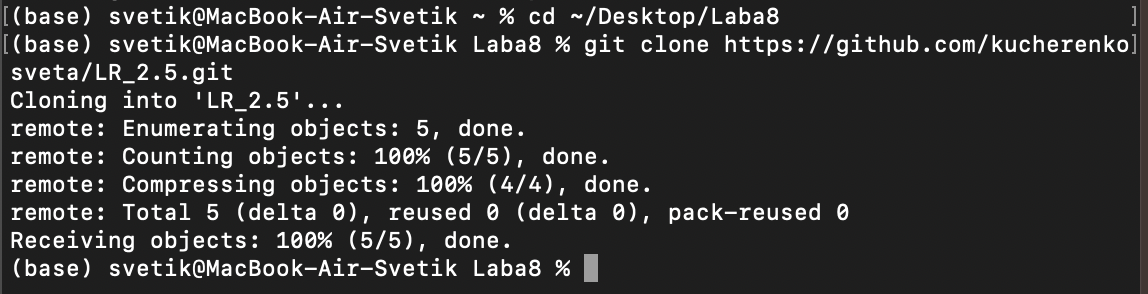
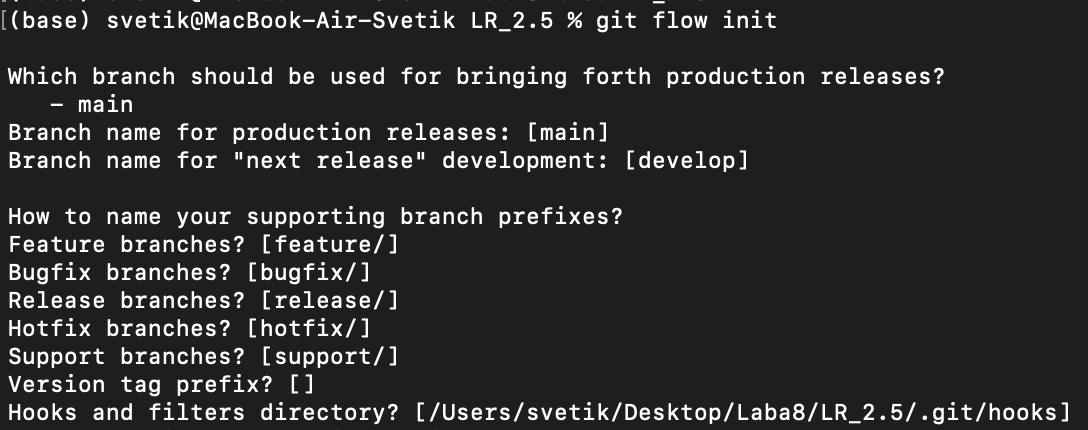


Рисунок 2 – Клонирование репозитория

1. Дополните файл .gitignore необходимыми правилами для работы с IDE PyCharm.
2. Организуйте свой репозиторий в соответствие с моделью ветвления git-flow.

Рисунок 3 – Организация репозитория в соответствии с моделью git-flow

1. Создайте проект PyCharm в папке репозитория

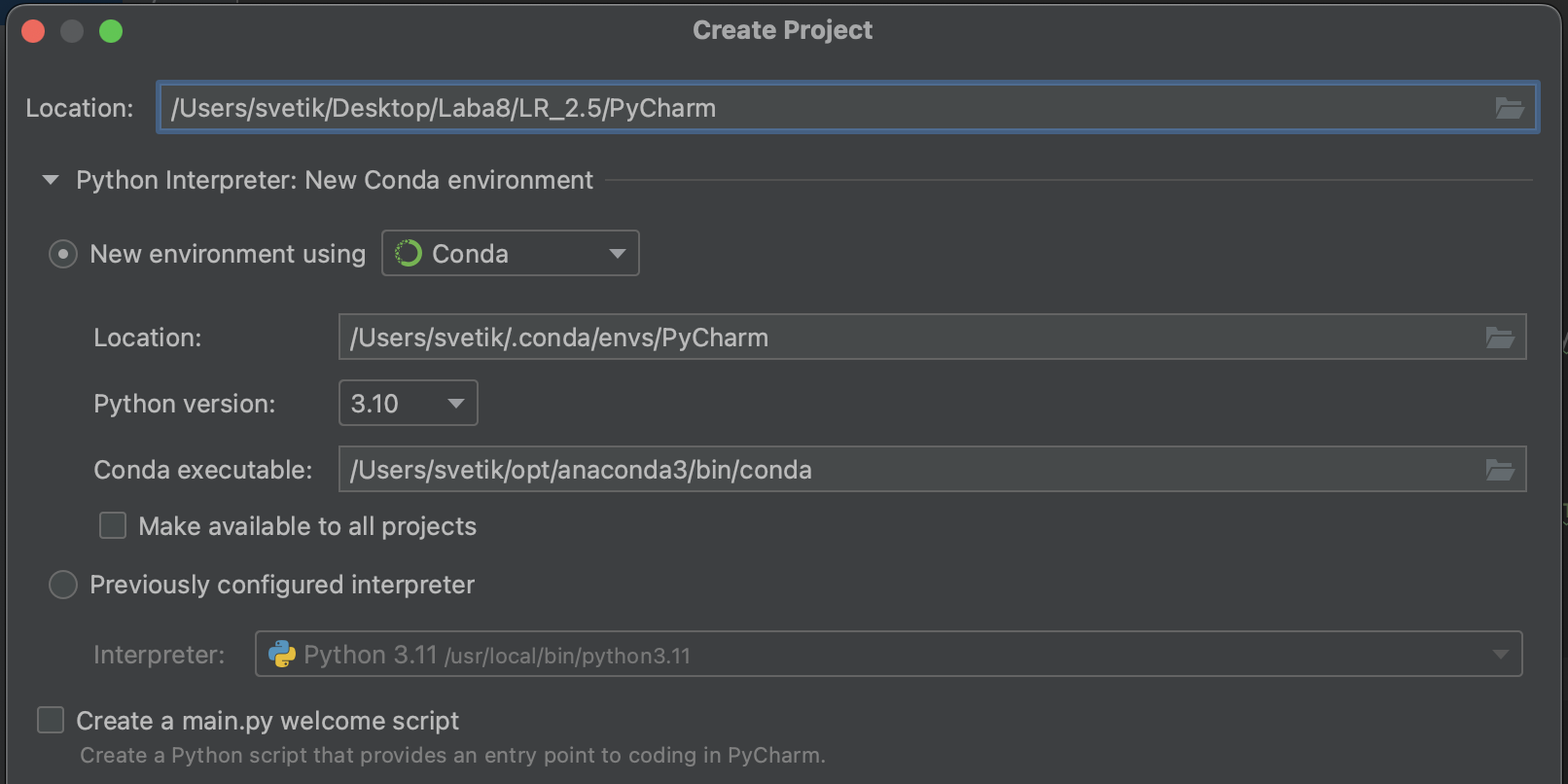


Рисунок 4 – Создание проекта PyCharm в папке репозитория

1. Проработайте примеры лабораторной работы. Создайте для каждого примера отдельный модуль языка Python. Зафиксируйте изменения в репозитории.

**Пример 1.** Ввести кортеж А из 10 элементов, найти сумму элементов, меньших по модулю 5, и вывести ее на экран. Использовать в программе вместо списков кортежи.

#!/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)

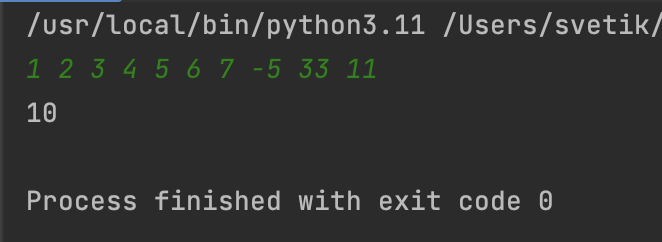


Рисунок 5 – Результат работы программы

Решение через списковые включения:

#!/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)

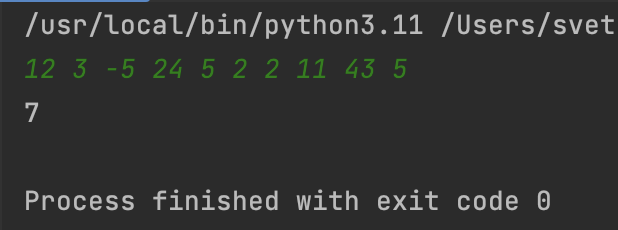


Рисунок 8 – Результат работы программы

1. Выполните индивидуальные задания, согласно своему варианту.

Вариант 9. 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 -\*-  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 # Ввести кортеж одной строкой.  
 A = tuple(map(int, input().split()))  
  
 for item in range(len(A)-2, 1, -1):  
 if A[item-1] < A[item] and A[item] > A[item+1]:  
 i = A[item]  
 print(i)  
 break  
 print(A[:item-1])

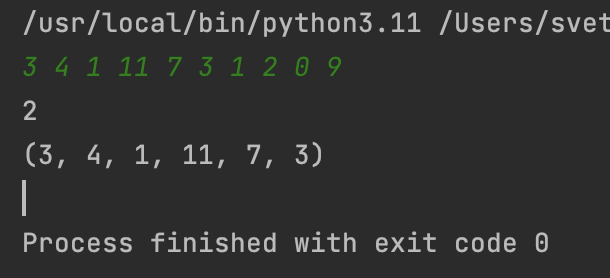


Рисунок 9 – Результат работы программы

**Вопросы для защиты работы**

1. **Что такое списки в языке Python?**

Кортеж (tuple) – это неизменяемая структура данных, которая по своему подобию очень похожа на список. Как вы наверное знаете, список – это изменяемый тип данных. Т.

1. **Каково назначение кортежей в языке Python?**

Т.к. кортеж – неизменяемый тип данных, это защищает данные от случайного изменения. К тому же, из-за отсутствия операций изменения делают работу кортежей быстрее и место они занимают меньше, чем списки.

1. **Как осуществляется создание кортежей?**

Создание кортежей осуществляется следующим образом: a = ()

b = tuple()

В скобках через запятую перечисляются элементы кортежа

Если кортеж состоит из одного элемента, то после него нужно поставить «,»:

tuple = (42,)

1. **Как осуществляется доступ к элементам кортежа?**

Доступ к элементам кортежа осуществляется аналогично доступу к элементам списка – через индекс.

>>> a = (1, 2, 3, 4, 5)

>>> print(a[0]) 1

1. **Зачем нужна распаковка (деструктуризация) кортежа?**

Это облегчает доступ к элементам, осуществляя его не по индексу, а с помощью переменных

name\_and\_age = ('Bob', 42) (name, age) = name\_and\_age name # 'Bob'

age # 42

1. **Какую роль играют кортежи в множественном присваивании?**

Благодаря тому, что кортежи легко собирать и разбирать, в Python удобно делать такие вещи, как множественное присваивание. Смотрите:

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

a = 100

b = 'foo'

(a, b) = (b, a) a # 'foo'

b # 100

Строку (a, b) = (b, a) нужно понимать как "присвоить в а и b значения из кортежа, состоящего из значений переменных b и a ".

1. **Как выбрать элементы кортежа с помощью среза?** Аналогично со списком:

>>> a = (1, 2, 3. 4. 5)

>>> print(a[1:3]) (2, 3)

1. **Как выполняется конкатенация и повторение кортежей?**

Конкатенация обозначается знаком «+», по сути, эта операция объединяет несколько кортежей в один новый. форма: T3 = T1 + T2, где T1, T2 – кортежи, над которыми выполняется операция, а T3 – новый кортеж.

1. **Как выполняется обход элементов кортежа?**

Обход элементов кортежа можно осуществить c помощью циклов for или while:

for i in A:

while i < len(A):

i = i + 1

1. **Как проверить принадлежность элемента кортежу?**

Чтобы проверить принадлежность элемента кортежу необходимо воспользоваться операцией in, форма: a = i in A, где a =true/false, i – искомый элемент, А – кортеж, в котором осуществляется поиск.

1. **Какие методы работы с кортежами Вам известны?**

Метод index() осуществляет поиск позиции элемента в кортеже, форма: pos = T.index(i), где роs – переменная, в которую будет записан индекс, Т – кортеж в котором осуществляется поиск, i – искомый элемент.

Метод count() осуществляет подсчет количества вхождений элемента в кортеж, форма: k = T.count(i), где k – искомое количество, Т – кортеж в котором осуществляется поиск, i – искомый элемент.

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

Да

1. **Как создать кортеж с помощью спискового включения.**

Аналогично списку, только скобки не квадратные, а круглые, и выражение дает на выходе специальный объект генератора, а не кортеж, для преобразования которого необходимо воспользоваться вызовом tuple().