**Московский государственный технический**

**университет им. Н.Э. Баумана**

Факультет «Информатика и системы управления»

Кафедра ИУ5 «Системы обработки информации и управления»

Отчет по лабораторной работе №1

«Основные конструкции языка Python.»

|  |  |
| --- | --- |
| Выполнил: Сироткин Сергей |  |
| ИУ5-35Б |  |
|  |  |
|  |  |

**Задание:**

Разработать программу для решения [биквадратного уравнения.](https://ru.wikipedia.org/wiki/%D0%A3%D1%80%D0%B0%D0%B2%D0%BD%D0%B5%D0%BD%D0%B8%D0%B5_%D1%87%D0%B5%D1%82%D0%B2%D1%91%D1%80%D1%82%D0%BE%D0%B9_%D1%81%D1%82%D0%B5%D0%BF%D0%B5%D0%BD%D0%B8#%D0%91%D0%B8%D0%BA%D0%B2%D0%B0%D0%B4%D1%80%D0%B0%D1%82%D0%BD%D0%BE%D0%B5_%D1%83%D1%80%D0%B0%D0%B2%D0%BD%D0%B5%D0%BD%D0%B8%D0%B5)

1. Программа должна быть разработана в виде консольного приложения на языке Python.
2. Программа осуществляет ввод с клавиатуры коэффициентов А, В, С, вычисляет дискриминант и ДЕЙСТВИТЕЛЬНЫЕ корни уравнения (в зависимости от дискриминанта).
3. Коэффициенты А, В, С могут быть заданы в виде параметров командной строки ( [вариант задания параметров приведен в конце файла с примером кода](https://github.com/ugapanyuk/BKIT_2022/blob/main/code/lab1_code) ). Если они не заданы, то вводятся с клавиатуры в соответствии с пунктом 2. [Описание работы с параметрами командной строки.](https://realpython.com/python-command-line-arguments/#the-command-line-interface)
4. Если коэффициент А, В, С введен или задан в командной строке некорректно, то необходимо проигнорировать некорректное значение и вводить коэффициент повторно пока коэффициент не будет введен корректно. Корректно заданный коэффициент - это коэффициент, значение которого может быть без ошибок преобразовано в действительное число.
5. Дополнительное задание 1 (\*). Разработайте две программы на языке Python - одну с применением процедурной парадигмы, а другую с применением объектно-ориентированной парадигмы.
6. Дополнительное задание 2 (\*). Разработайте две программы - одну на языке Python, а другую на любом другом языке программирования (кроме С++).

**Текст программы:**

Объектно-ориентированная парадигма:

import sys

import math

class Roots:

def \_\_init\_\_(self):

self.a = 0.0

self.b = 0.0

self.c = 0.0

self.roots = []

self.num\_roots = 0

def read\_coef(self, index, prompt):

try:

coef = sys.argv[index]

except:

print(prompt)

coef = input()

coef = float(coef)

return coef

def get\_coefs(self):

self.a = self.read\_coef(1, 'Введите коэффициент А:')

self.b = self.read\_coef(2, 'Введите коэффициент B:')

self.c = self.read\_coef(3, 'Введите коэффициент C:')

def find\_roots(self):

a = self.a

b = self.b

c = self.c

d = b\*b - 4\*a\*c

if d > 0.0:

self.num\_roots = 2

root1 = (-b + math.sqrt(d)) / (2.0 \* a)

root2 = (-b - math.sqrt(d)) / (2.0 \* a)

self.roots.append(root1)

self.roots.append(root2)

elif d == 0.0:

self.num\_roots = 1

root = -b / (2.0 \* a)

self.roots.append(root)

def print\_roots(self):

if self.num\_roots != len(self.roots):

print("Ошибка")

else:

if self.num\_roots == 0:

print('Нет корней')

elif self.num\_roots == 1:

print('Один корень: ', self.roots[0])

elif self.num\_roots == 2:

print('Два корня:', self.roots[0], self.roots[1])

def main():

r = Roots()

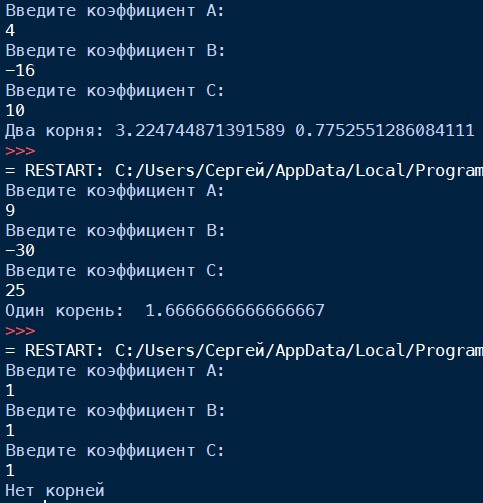
r.get\_coefs()

r.find\_roots()

r.print\_roots()

if \_\_name\_\_ == "\_\_main\_\_":

main()



Процедурная парадигма:

import sys

import math

def get\_coef(index, prompt):

try:

coef\_str = sys.argv[index]

except:

print(prompt)

coef\_str = input()

coef = float(coef\_str)

return coef

def get\_roots(a, b, c):

roots = []

d = b\*b - 4\*a\*c

if d > 0.0:

root1 = (-b + math.sqrt(d)) / (2.0 \* a)

root2 = (-b - math.sqrt(d)) / (2.0 \* a)

roots.append(root1)

roots.append(root2)

elif d == 0.0:

root = -b / (2.0 \* a)

roots.append(root)

return roots

def main():

a = get\_coef(1, 'Введите коэффициент А:')

b = get\_coef(2, 'Введите коэффициент B:')

c = get\_coef(3, 'Введите коэффициент C:')

roots = get\_roots(a,b,c)

if len(roots) == 0:

print('Нет корней')

elif len(roots) == 1:

print('Один корень:', roots[0])

elif len(roots) == 2:

print('Два корня:', roots[0], roots[1])

if \_\_name\_\_ == "\_\_main\_\_":

main()

