**Московский государственный технический**

**университет им. Н.Э. Баумана**

Факультет ИУ

Кафедра ИУ5

Курс «Основы информатики»

Отчет по лабораторной работе №\_1\_

«Основные конструкции языка Python»

|  |  |  |
| --- | --- | --- |
| Выполнил: |  | Проверил: |
| студент группы ИУ5-23Б: |  | преподаватель каф. \_\_\_\_ |
| Селиханович Д.С. |  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| Подпись и дата: |  | Подпись и дата: |

Москва, 2024 г.

**Постановка задачи**

Разработать программу для решения биквадратного уравнения.

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 ). Если они не заданы, то вводятся с клавиатуры в соответствии с пунктом 2. Описание работы с параметрами командной строки.
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**

**\lab\_01\lab\_01\_PP.py**

import sys

import math

def get\_coefs():

    coefs = []

    try:

        help = sys.argv[1:]

        coefs = [float(i) for i in help]

        if (len(help)!=3):

            1/0

    except:

        coefs=[]

        for i in range(3):

            while(1):

                print(f"\nВведите {i+1} коэффициент")

                try:

                    coefs.append(float(input()))

                    break

                except:

                    print("\nНеправильный тип коэффициента.")

    return coefs

def get\_roots(coefs):

    result = []

    discr = pow(coefs[1],2)-4\*coefs[0]\*coefs[2]

    if (discr < 0):

        return result

    elif (discr == 0):

        result.append(-coefs[1]/(2\*coefs[0]))

    else:

        result.append((-coefs[1]+math.sqrt(discr))/(2\*coefs[0]))

        result.append((-coefs[1]-math.sqrt(discr))/(2\*coefs[0]))

    for i in range(len(result)):

        try:

            result[i]=math.sqrt(result[i])

            if (result[i]!=0):

                result.append(-result[i])

        except:

            result.pop(i)

    return result

def main():

    answer = get\_roots(get\_coefs())

    if (len(answer) == 0):

        print("Нет корней.")

    else:

        print("Корни уравнения: ",\*answer)

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

    main()

**\lab\_01\lab\_01\_OOP.py**

import sys

import math

class Equation:

    coefs=[]

    def \_\_init\_\_(self):

        try:

            help = sys.argv[1:]

            Equation.coefs = [float(i) for i in help]

            if (len(help)!=3):

                1/0

        except:

            coefs=[]

            for i in range(3):

                while(1):

                    print(f"\nВведите {i+1} коэффициент")

                    try:

                        Equation.coefs.append(float(input()))

                        break

                    except:

                        print("\nНеправильный тип коэффициента.")

    def get\_roots(coefs):

        result = []

        discr = pow(Equation.coefs[1],2)-4\*Equation.coefs[0]\*Equation.coefs[2]

        if (discr < 0):

            return result

        elif (discr == 0):

            result.append(-Equation.coefs[1]/(2\*Equation.coefs[0]))

        else:

            result.append((-Equation.coefs[1]+math.sqrt(discr))/(2\*Equation.coefs[0]))

            result.append((-Equation.coefs[1]-math.sqrt(discr))/(2\*Equation.coefs[0]))

        for i in range(len(result)):

            try:

                result[i]=math.sqrt(result[i])

                if (result[i]!=0):

                    result.append(-result[i])

            except:

                result.pop(i)

        return result

def main():

    answer = Equation().get\_roots()

    if (len(answer) == 0):

        print("Нет корней.")

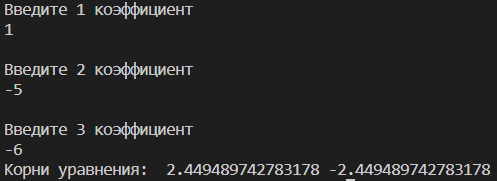
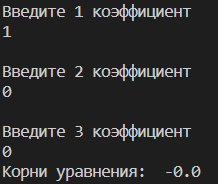
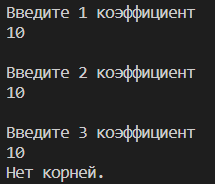
    else:

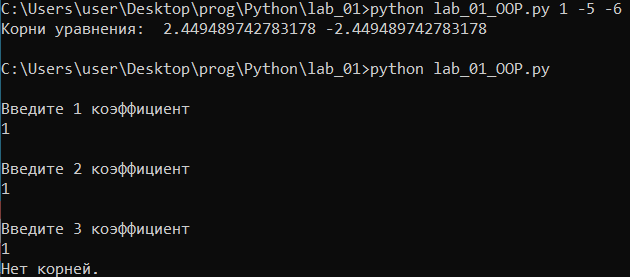
        print("Корни уравнения: ",\*answer)

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

    main()

**Анализ результатов**

**  **

****