**Московский государственный технический**

**университет им. Н.Э. Баумана**

**Факультет ИУ**

**Кафедра ИУ5**

**Курс «Основы информатики»**

**Отчет лабораторной работе №1**

**lab\_python\_intro**

Выполнил студент группы ИУ5-33Б:

Османов З. Ш.

Подпись и дата:

Проверил преподаватель каф.:

Гапанюк Ю. Е.

Подпись и дата:

Москва, 2024 г

**Описание задания**

**Код программы:**

**1 вариант-процедурный lab1.py**

import sys

import math

def get\_coeff(prompt):

    while True:

        try:

            return float(input(prompt))

        except ValueError:

            print("Некорректный ввод.")

def get\_coeff\_sys(arg\_index, prompt):

    try:

        return float(sys.argv[arg\_index])

    except (IndexError, ValueError):

        return get\_coeff(prompt)

def solve\_quad(a, b, c):

    dis = b\*\*2 - 4 \* a \* c

    if dis > 0:

        temp1 = (-b + dis\*\*0.5) / (2 \* a)

        temp2 = (-b - dis\*\*0.5) / (2 \* a)

        return temp1, temp2

    elif dis == 0:

        temp = -b / (2 \* a)

        return temp, None

    else:

        return None, None

def solve\_biq(a, b, c):

    if a == 0:

        print("A = 0.")

        return

    temp1, temp2 = solve\_quad(a, b, c)

    roots = []

    if temp1 is not None and temp1 >= 0:

        x1 = temp1\*\*0.5

        x2 = - temp1\*\*0.5

        roots += [x1, x2]

    if temp2 is not None and temp2 >= 0:

        x3 = temp2\*\*0.5

        x4 = - temp2\*\*0.5

        roots += [x3, x4]

    print(roots)

    if len(roots):

        print("корни уравнения: ", sorted(set(roots)))

    else:

        print("корней нет.")

a = get\_coeff\_sys(1, "Коэффициент A: ")

b = get\_coeff\_sys(2, "Коэффициент B: ")

c = get\_coeff\_sys(3, "Коэффициент C: ")

solve\_biq(a, b, c)

**2 вариант-ООП lab1\_oop.py**

import sys

import math

class Biquadratic:

    def \_\_init\_\_(self, a, b, c):

        self.a = a

        self.b = b

        self.c = c

    def solve\_quad(self):

        dis = self.b\*\*2 - 4 \* self.a \* self.c

        if dis > 0:

            temp1 = (-self.b + dis\*\*0.5) / (2 \* self.a)

            temp2 = (-self.b - dis\*\*0.5) / (2 \* self.a)

            return temp1, temp2

        elif dis == 0:

            t = -self.b / (2 \* self.a)

            return t, None

        else:

            return None, None

    def solve\_biq(self):

        if self.a == 0:

            print("A = 0.")

            return

        temp1, temp2 = self.solve\_quad()

        roots = []

        if temp1 is not None and temp1 >= 0:

            x1 = temp1\*\*0.5

            x2 = - temp1\*\*0.5

            roots += [x1, x2]

        if temp2 is not None and temp2 >= 0:

            x3 = temp2\*\*0.5

            x4 = - temp2\*\*0.5

            roots += [x3, x4]

        if roots:

            print("корни уравнения: ", sorted(set(roots)))

        else:

            print("корней нет.")

class Input:

    def get\_coeff(prompt):

        while True:

            try:

                return float(input(prompt))

            except ValueError:

                print("Некорректный ввод.")

    def get\_coeff\_sys(arg\_index, prompt):

        try:

            return float(sys.argv[arg\_index])

        except (IndexError, ValueError):

            return Input.get\_coeff(prompt)

a = Input.get\_coeff\_sys(1, "коэффициент A: ")

b = Input.get\_coeff\_sys(2, "коэффициент B: ")

c = Input.get\_coeff\_sys(3, "коэффициент C: ")

equation = Biquadratic(a, b, c)

equation.solve\_biq()

**Выполнение программы**

