**Московский государственный технический**

**университет им. Н.Э. Баумана**

Факультет «Информатика и системы управления»

Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Парадигмы и конструкции языков программирования»

Отчет по лабораторной работе №1

«Основные конструкции языка Python»

|  |  |  |
| --- | --- | --- |
| Выполнил: |  | Проверил: |
| студент группы ИУ5-34Б |  | преподаватель каф. ИУ5 |
| Шараев Вячеслав Андреевич |  | Нардид Анатолий Николаевич |
|  |  |  |
|  |  |  |

Москва, 2024 г.

Постановка задачи:

Разработать программу для решения биквадратного уравнения.

Программа должна быть разработана в виде консольного приложения на языке Python.

Программа осуществляет ввод с клавиатуры коэффициентов А, В, С, вычисляет дискриминант и ДЕЙСТВИТЕЛЬНЫЕ корни уравнения (в зависимости от дискриминанта).

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.

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.

Дополнительное задание 2 (\*). Разработайте две программы - одну на языке Python, а другую на любом другом языке программирования (кроме С++).

В качестве второго языка программирования был выбран C#

Текст программы:

Lab1.py

import sys

import math

def is\_float(value):

try:

float(value)

return False

except ValueError:

return True

def get\_coef(index, text):

try:

coef\_s = sys.argv[index]

except:

print(text)

coef\_s = input()

while is\_float(coef\_s):

coef\_s = input()

coef = float(coef\_s)

return coef

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

roots = []

if a == 0:

if b == 0:

return roots

else:

if (-c/b) < 0:

roots.append(math.sqrt(math.sqrt((-1)\*(-c/b))))

roots.append((-1)\*roots[0])

else:

roots.append(math.sqrt(math.sqrt(-c/b)))

roots.append((-1)\*roots[0])

return roots

else:

D = b\*\*2 - 4\*a\*c

if D == 0.0:

if (-b / (2.0\*a)) < 0:

roots.append(math.sqrt((-1)\*(-b / (2.0\*a))))

roots.append((-1)\*roots[0])

else:

roots.append(math.sqrt(-b / (2.0\*a)))

roots.append((-1)\*roots[0])

elif D > 0:

if ((-b + math.sqrt(D)) / (2.0\*a)) < 0:

roots.append(math.sqrt((-1)\*(-b + math.sqrt(D)) / (2.0\*a)))

roots.append((-1)\*roots[0])

else:

roots.append(math.sqrt((-b + math.sqrt(D)) / (2.0\*a)))

roots.append((-1)\*roots[0])

if ((-b - math.sqrt(D)) / 2.0\*a) < 0:

roots.append(math.sqrt((-1)\*(-b - math.sqrt(D)) / (2.0\*a)))

roots.append((-1)\*roots[2])

else:

roots.append(math.sqrt((-b - math.sqrt(D)) / (2.0\*a)))

roots.append((-1)\*roots[2])

return roots

def main():

a = get\_coef(1, "Введите коэффициент A: ")

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("Действительных корней нет")

elif len(roots) == 2:

print("x1 =", roots[0])

print("x2 =", roots[1])

else:

print("x1 =", roots[0])

print("x2 =", roots[1])

print("x3 =", roots[2])

print("x4 =", roots[3])

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

main()

Program.cs

﻿using System;

double get\_coef(string text){

Console.WriteLine(text);

double coef = 0;

bool result = false;

while(!result){

string? input = Console.ReadLine();

result = double.TryParse(input, out coef);

}

return coef;

}

static double [] get\_roots(double a, double b, double c, ref int i){

double[] roots = { 0, 0, 0, 0 };

i = 0;

if (a == 0){

if (b == 0){

return roots;

}

else{

if ((-c/b) < 0){

roots[i] = Math.Sqrt(Math.Sqrt((-1) \* (-c / b)));

roots[++i] = (-1)\*roots[i - 1];

}

else{

roots[i] = Math.Sqrt(Math.Sqrt((-c / b)));

roots[++i] = (-1) \* roots[i - 1];

}

return roots;

}

}

else{

i = 0;

double D = Math.Pow(b, 2) - 4\*a\*c;

if (D == 0.0){

if ((-b / (2.0\*a)) < 0){

roots[i] = Math.Sqrt((-1) \* (-b / (2.0 \* a)));

roots[++i] = (-1)\*roots[i - 1];

}

else{

roots[i] = Math.Sqrt(-b / (2.0 \* a));

roots[++i] = (-1)\*roots[i - 1];

}

}

else if (D > 0){

i = 0;

if (((-b + Math.Sqrt(D)) / (2.0 \* a)) < 0){

roots[i] = Math.Sqrt((-1) \* (-b + Math.Sqrt(D)) / (2.0 \* a));

roots[++i] = (-1)\*roots[i - 1];

}

else{

roots[i] = Math.Sqrt((-b + Math.Sqrt(D)) / (2.0 \* a));

roots[++i] = (-1)\*roots[i - 1];

}

if (((-b - Math.Sqrt(D)) / (2.0 \* a)) < 0){

roots[++i] = Math.Sqrt((-1) \* (-b - Math.Sqrt(D)) / (2.0 \* a));

roots[++i] = (-1)\*roots[i - 1];

}

else{

roots[++i] = Math.Sqrt((-b - Math.Sqrt(D)) / (2.0 \* a));

roots[++i] = (-1)\*roots[i - 1];

}

}

}

return roots;

}

void main(){

double a = get\_coef("Введите коэффициент A: ");

double b = get\_coef("Введите коэффициент B: ");

double c = get\_coef("Введите коэффициент C: ");

int i = 0;

double [] ans\_roots = get\_roots(a, b, c, ref i);

if (i == 0){

Console.WriteLine("Действительных корней нет");

}

else if (i == 1){

Console.WriteLine("x1 = " + ans\_roots[0]);

Console.WriteLine("x2 = " + ans\_roots[1]);

}

else{

Console.WriteLine("x1 = " + ans\_roots[0]);

Console.WriteLine("x2 = " + ans\_roots[1]);

Console.WriteLine("x3 = " + ans\_roots[2]);

Console.WriteLine("x4 = " + ans\_roots[3]);

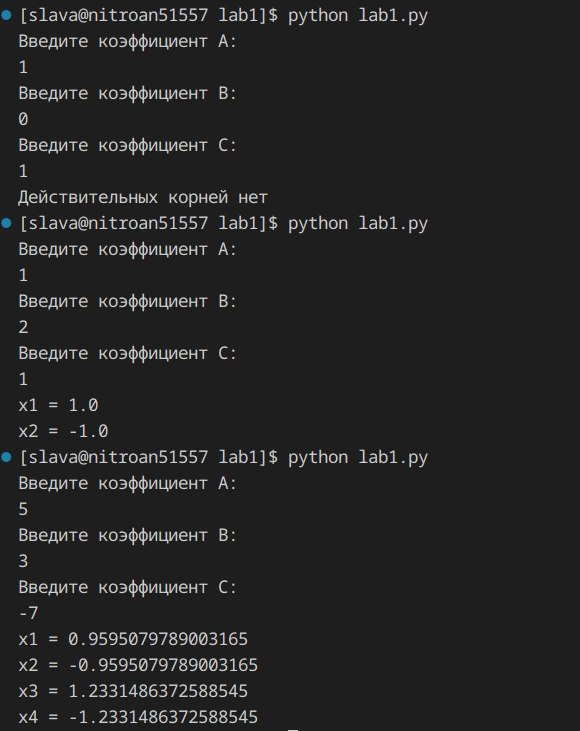
}

}

main();

Результат выполнения программы:

Python:



C#

