

**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:

студент группы ИУ5-32Б
Канаева Д.Ч.

Подпись и дата:

Проверил:

преподаватель каф. ИУ5
Нардид А.Н.

Подпись и дата:

Цель лабораторной работы: изучение основных конструкций языка Python.

Задание: 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 C++).

Текст программы:

оор.py

```
import sys
import math

class EquationSolver:
    def __init__(self):
        self.a = 0.0
        self.b = 0.0
        self.c = 0.0

    def get_coef(self, index):
        coef_1 = sys.argv[index]
        flag = 1
        while(flag):
            try:
                coef = float(coef_1)
                flag = 0
            except:
                print("Коэффициент ", index, " введен некорректно! Введите его снова.")
                coef_1 = input()
```

```

        return coef

    def solve(self):
        dis = self.b**2 - 4*self.a*self.c
        if dis >= 0.0:
            x1_1 = (-self.b + math.sqrt(dis))/(2*self.a)
            flag1 = 0
            flag2 = 0

            if x1_1 >= 0:
                x1 = math.sqrt(x1_1)
                x2 = x1 * (-1)
                flag1 = 1

            x3_3 = (-self.b - math.sqrt(dis))/(2*self.a)
            if x3_3 >= 0:
                x3 = math.sqrt(x3_3)
                x4 = x3 * (-1)
                flag2 = 1

            if flag1 and flag2:
                return x1, x2, x3, x4
            elif flag1:
                return x1, x2
            elif flag2:
                return x3, x4

        return 0

def main():
    r = EquationSolver()
    r.a = r.get_coef(1)
    r.b = r.get_coef(2)
    r.c = r.get_coef(3)

    roots = r.solve()

    if roots:
        print("Корни уравнения:", roots)
    else:
        print("Нет действительных корней уравнения.")

if __name__ == "__main__":
    main()

```

proc.py

```

import sys
import math

def get_coef(index):
    coef_1 = sys.argv[index]
    flag = 1
    while(flag):
        try:
            coef = float(coef_1)
            flag = 0
        except:
            print("Коэффициент ", index, " введен некорректно! Введите его снова.")
            coef_1 = input()
    return coef

def solve(a, b, c):

```

```

dis = b**2 - 4*a*c
if dis >= 0.0:
    x1_1 = (-b + math.sqrt(dis))/(2*a)
    flag1 = 0
    flag2 = 0

    if x1_1 >= 0:
        x1 = math.sqrt(x1_1)
        x2 = x1 * (-1)
        flag1 = 1

    x3_3 = (-b - math.sqrt(dis))/(2*a)
    if x3_3 >= 0:
        x3 = math.sqrt(x3_3)
        x4 = x3 * (-1)
        flag2 = 1

    if flag1 and flag2:
        return x1, x2, x3, x4
    elif flag1:
        return x1, x2
    elif flag2:
        return x3, x4

    return 0

def main():
    a = get_coef(1)
    b = get_coef(2)
    c = get_coef(3)

    roots = solve(a, b, c)

    if roots:
        print("Корни уравнения:", roots)
    else:
        print("Нет действительных корней уравнения.")

if __name__ == "__main__":
    main()

```

Java.java

```

import java.util.Scanner;

public class Java {

    public static double get_coef(int index) {
        Scanner input = new Scanner(System.in);
        String coef_1 = input.nextLine();
        int flag = 1;
        double coef = 0;

        while(flag == 1) {
            try {
                coef = Double.parseDouble(coef_1);
                flag = 0;
            }
            catch(NumberFormatException e) {
                System.out.println("Коэффициент " + index + " введен некорректно! Введите его снова.");
                coef_1 = input.nextLine();
            }
        }
    }
}

```

```

        return coef;
    }

    public static double[] solve(double a, double b, double c) {
        double[] roots = new double[4];
        double dis = b*b - 4*a*c;

        if (dis >= 0.0) {
            double x1_1 = (-b + Math.sqrt(dis))/(2*a);
            int flag1 = 1;
            int flag2 = 1;

            if (x1_1 >= 0) {
                double x1 = Math.sqrt(x1_1);
                double x2 = x1 * (-1);
                flag1 = 0;
                roots[0] = x1;
                roots[1] = x2;
            }

            double x3_3 = (-b - Math.sqrt(dis))/(2*a);
            if (x3_3 >= 0) {
                double x3 = Math.sqrt(x3_3);
                double x4 = x3 * (-1);
                flag2 = 0;
                roots[2] = x3;
                roots[3] = x4;
            }

            if (flag1 == 0 && flag2 == 0) return roots;
            else if (flag1 == 0) {
                double[] roots1 = {roots[0], roots[1]};
                return roots1;
            }
            else if (flag2 == 0) {
                double[] roots2 = {roots[2], roots[3]};
                return roots2;
            }
        }
        return null;
    }

    public static void main(String[] args) {
        double a = get_coef(1);
        double b = get_coef(2);
        double c = get_coef(3);

        double[] roots = solve(a, b, c);

        if (roots != null) {
            System.out.println("Корни уравнения: " + roots[0] + ", " +
roots[1]);
            if (roots.length == 4) {
                System.out.println("Корни уравнения: " + roots[2] + ", " +
roots[3]);
            }
        }
        else System.out.println("Нет действительных корней уравнения.");
    }
}

```

Экранные формы с примерами выполнения программы:

```
C:\Users\123\Documents\PCPL_3sem\lab1>python proc.py 4 -8 g
Коэффициент 3 введен некорректно! Введите его снова.
3
Корни уравнения: (1.224744871391589, -1.224744871391589, 0.7071067811865476, -0.7071067811865476)

C:\Users\123\Documents\PCPL_3sem\lab1>python oop.py gjdks 8 fcdx
Коэффициент 1 введен некорректно! Введите его снова.
bdvdx
Коэффициент 1 введен некорректно! Введите его снова.
4
Коэффициент 3 введен некорректно! Введите его снова.
3
Нет действительных корней уравнения.

C:\Users\123\Documents\PCPL_3sem\lab1>java Java.java
1
-2
4
Нет действительных корней уравнения.

C:\Users\123\Documents\PCPL_3sem\lab1>java Java.java
1
-2
-8
Корни уравнения: 2.0, -2.0
```