

**Московский государственный технический
университет им. Н.Э. Баумана**

Факультет Радиотехнический
Кафедра “Системы обработки информации и управления”

Курс «Парадигмы и конструкции языков программирования»

Отчет по лабораторной работе №1

Выполнил:
студент группы РТ5-31Б:
Стукановский Максим
Владимирович
Подпись и дата:

Проверил:
преподаватель каф. ИУ5
Гапанюк Юрий
Евгеньевич
Подпись и дата:

Москва, 2025 г.

1. Постановка задачи:

Разработать консольное приложение на языке Rust для решения биквадратного уравнения (вида $Ax^4 + Bx^2 + C = 0$), которое должно принимать коэффициенты **A, B, C** либо через параметры командной строки, либо интерактивно с клавиатуры (если параметры не предоставлены), при этом обеспечивать проверку корректности ввода: при некорректных значениях (не преобразуемых в float) запрашивать повторный ввод до получения валидных данных. Программа вычисляет дискриминант и определяет все действительные корни уравнения (если они существуют), учитывая особенности биквадратной подстановки.

2. Код программы:

```
#![allow(unused)]

use std::io; // библиотека ввода/вывода
use colored::*; // цвет вывода текста в консоли toml - colored = "2.1"

fn main() {
    loop {
        let mut coef = String::new();
        let mut result: Vec<f64> = Vec::new();

        println!("Введите коэффициенты биквадратного уравнения:");

        io::stdin().read_line(&mut coef).expect("Ошибка при чтении строки");

        let nums_result: Vec<Result<f64, _>> = coef
            .trim()
            .split_whitespace()
            .map(|x| x.parse())
            .collect();

        if nums_result.len() < 3 {
            println!("{}", "Необходимо ввести 3 коэффициента".red());
            continue;
        }

        if nums_result.iter().any(|x| x.is_err()) {
            println!("{}", "Один или несколько параметров не являются дробным числом - > f64".red());
            continue;
        }

        let nums: Vec<f64> = nums_result
            .into_iter()
            .map(|x| x.unwrap())
            .collect();

        if nums[0] == 0.0 {
            println!("{}", "Коэффициент a не может быть равен нулю".red());
            continue;
        }

        let a = nums[0];
        let b = nums[1];
        let c = nums[2];

        let discriminant = b * b - 4.0 * a * c;
        if discriminant >= 0.0 {
            let t1 = (-b + discriminant.sqrt()) / (2.0 * a);
            let t2 = (-b - discriminant.sqrt()) / (2.0 * a);

            if t1 >= 0.0 {
                result.push(t1.sqrt());
                result.push(-t1.sqrt());
            }

            if t2 >= 0.0 {
                result.push(t2.sqrt());
                result.push(-t2.sqrt());
            }

            if t1 != t2 && t2 >= 0.0 {
                result.push(t2.sqrt());
                result.push(-t2.sqrt());
            }
        }

        for i in result {
            if i == 0.0 || i == -0.0 {
```

```
        continue;
    }
    println!("{}", format!("Корень: {i}").green());
}
break;
} else {
    println!("{}", "Действительных корней нет".red());
}
}
}
```

3. Анализ результатов:

Ввод одного/нескольких параметров в формате строки:

Введите коэффициенты биквадратного уравнения:

a 12 b

Один или несколько параметров не являются дробным числом -> f64

Введите коэффициенты биквадратного уравнения:

Ввод «-»:

Введите коэффициенты биквадратного уравнения:

Уравнение не может быть пустым

Введите коэффициенты биквадратного уравнения:

Ввод менее трех аргументов:

Введите коэффициенты биквадратного уравнения:

2 3

Введите 3 коэффициента

Введите коэффициенты биквадратного уравнения:

Ввод аргументов «321 45 31»:

Введите коэффициенты биквадратного уравнения:

321 45 31

Действительных корней нет

Введите коэффициенты биквадратного уравнения:

Ввод аргументов «1 0 -1»:

Введите коэффициенты биквадратного уравнения:

1 0 -1

Корень: 1

Корень: -1

Ввод аргументов «1 -2 0»:

Введите коэффициенты биквадратного уравнения:

1 -2 0

Корень: 1.4142135623730951

Корень: -1.4142135623730951