

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «Белгородский государственный  
национальный исследовательский университет»**

Лабораторная работа №2  
дисциплина «Современные методы, среды и языки программирования»  
по теме «Создание и применение лямбда-выражений как параметров и результатов методов в программах на  
языке Java»

Выполнил: студент группы 12002135

Проверил:

Макаров Д.С.

## Лабораторная работа №2

### «Создание и применение лямбда-выражений как параметров и результатов методов в программах на языке Java»

**Цель работы:** Научиться писать код лямбда-выражений на языке Java в соответствии с основными принципами объектно-ориентированного программирования, а также выполнять реализацию выражений для решения типовых задач.

Необходимо выполнить следующие задания на языке Java:

1. Написать функциональный интерфейс с методом, который принимает число и возвращает булево значение. Написать реализацию такого интерфейса в виде лямбда-выражения, которое возвращает true если переданное число делится без остатка на 13.
2. Написать функциональный интерфейс с методом, который принимает две строки и возвращает тоже строку. Написать реализацию такого интерфейса в виде лямбды, которая возвращает ту строку, которая длиннее.
3. Написать функциональный интерфейс с методом, который принимает три дробных числа: a, b, c и возвращает тоже дробное число. Написать реализацию такого интерфейса в виде лямбда-выражения, которое возвращает дискриминант. Кто забыл,  $D = b^2 - 4 * a * c$ .
4. Используя функциональный интерфейс из задачи 3 написать лямбда выражение, которое возвращает результат операции  $a * b^c$ .

### Ход работы

# Приложение

## Содержимое файла main.go

```
package main

import "math"

type task1 struct {
    foo func(a int) bool
}

type task2 struct {
    foo func(a, b string) string
}

type task3 struct {
    foo func(a, b, c float64) float64
}

func main() {
    t1 := task1{}
    t1.foo = func(a int) bool {
        return a%13 == 0
    }

    t2 := task2{}
    t2.foo = func(a, b string) string {
        if len(a) >= len(b) {
            return a
        }
        return b
    }

    t3 := task3{}
    t3.foo = func(a, b, c float64) float64 {
        return b*b - 4*a*c
    }

    t4 := task3{}
    t4.foo = func(a, b, c float64) float64 {
        return a * math.Pow(b, c)
    }
}
```