

## Лабораторная работа №2

Выполнил: Латахин Егор Дмитриевич,

студент группы: 6201-120303D, Фотоника и Оптикоинформатика

### Задание 1 Создание пакета functions

Создан пакет 'functions' для организации классов. Все основные классы размещены в этом пакете.

### Задание 2 Класс FunctionPoint

Реализован класс 'FunctionPoint', представляющий точку функции с координатами (x, y).

Конструкторы:

- 'FunctionPoint(double x, double y)' - создает точку с заданными координатами
- 'FunctionPoint(FunctionPoint point)' - копирует точку
- 'FunctionPoint()' - создает точку (0, 0)

Методы доступа:

- 'getX()', 'getY()' - получение координат
- 'setX()', 'setY()' - установка координат

### Задание 3 Класс TabulatedFunction

Создан класс для представления табулированной функции, где точки хранятся в массиве 'FunctionPoint[]', упорядоченном по возрастанию x.

Конструкторы:

- 'TabulatedFunction(double leftX, double rightX, int pointsCount)' - создает функцию с равномерным распределением точек, значения  $y = 0$
- 'TabulatedFunction(double leftX, double rightX, double[] values)' - создает функцию с заданными в массиве values значениями y

Оба проверяют корректность и генерируют исключения.

### Задание 4 Методы работы с функцией

- 'getLeftDomainBorder()' - левая граница Области Определения
- 'getRightDomainBorder()' - правая граница Области Определения
- 'getFunctionValue(double x)' - значение функции в точке x

Метод 'getFunctionValue()' использует линейную интерполяцию для вычисления значений между точками табуляции (для точек вне Области Определения возвращает Double.NaN).

### Задание 5 Методы работы с точками

Реализован полный набор методов для работы с отдельными точками функции:

- 'getPointsCount()' - количество точек
- 'getPoint(int index)' - получение копии точки (инкапсуляция)
- 'setPoint(int index, FunctionPoint point)' - замена точки с проверкой порядка
- 'getPointX()', 'getPointY()' - получение координат и 'setPointX()', 'setPointY()' - установка координат

Методы 'setPoint()' и 'setPointX()' проверяют, нарушается ли упорядоченность точек по x.

### Задание 6 Изменение количества точек

Реализованы методы для изменения структуры:

- ‘deletePoint(int index)’ - удаление точки (минимум 2 точки должно остаться)
- ‘addPoint(FunctionPoint point)’ - добавление новой точки с сохранением порядка

Метод ‘addPoint()’ автоматически расширяет массив при необходимости и вставляет точку в правильную позицию для сохранения упорядоченности.

#### Задание 7 Тестирование

Создан класс “Main” для тестирования функциональности.

Результаты вывода в консоль:

test getFunctionValue

f(-4.0) = NaN

f(-3.0) = 9.0

f(-2.5) = 6.5

f(-1.2) = 1.5999999999999996

f(0.0) = 0.0

f(1.7) = 3.0999999999999996

f(3.0) = 9.0

f(4.0) = NaN

after modification

0: (-2.0; 5.0)

1: (-1.0; 1.0)

2: (0.0; 0.0)

3: (1.0; 1.0)

4: (2.0; 4.0)

5: (2.2; 5.0)

6: (3.0; 9.0)

```
Windows PowerShell
PS D:\Lab\OOP\Lab-2-2025> JAVA MAIN.JAVA
test getFunctionValue
f(-4.0) = NaN
f(-3.0) = 9.0
f(-2.5) = 6.5
f(-1.2) = 1.5999999999999996
f(0.0) = 0.0
f(1.7) = 3.0999999999999996
f(3.0) = 9.0
f(4.0) = NaN

after modification
0: (-2.0; 5.0)
1: (-1.0; 1.0)
2: (0.0; 0.0)
3: (1.0; 1.0)
4: (2.0; 4.0)
5: (2.2; 5.0)
6: (3.0; 9.0)
PS D:\Lab\OOP\Lab-2-2025>
```