

Лабораторная работа №4

Лебедев Кирилл Дмитриевич 6204-010302D

Задание 1

Добавлены конструкторы в классы `ArrayTabulatedFunction` и `LinkedListTabulatedFunction`, принимающие массив объектов `FunctionPoint`. Реализована проверка на минимальное количество точек (≥ 2) и упорядоченность по оси X. Обеспечена корректная инкапсуляция через создание копий точек.

Результат: Конструкторы успешно создают объекты при корректных данных и выбрасывают `IllegalArgumentException` при нарушении условий.

Задание 2

Создан интерфейс `Function` с методами определения области значений и вычисления функции. Интерфейс `TabulatedFunction` теперь расширяет `Function`, исключив дублирующиеся методы.

```
package functions;
```

```
public interface Function {  
    public double getLeftDomainBorder();    // возвращает значение левой границы  
    // области определения функции  
    public double getRightDomainBorder();    // возвращает значение правой  
    // границы области определения функции  
    public double getFunctionValue(double x); // возвращает значение функции в  
    // заданной точке  
}
```

Результат: Построена четкая иерархия: табулированные функции стали частным случаем функций одной переменной.

Задание 3

В пакете `functions.basic` созданы классы:

- `Exp` - экспонента
- `Log` - логарифм с основанием в конструкторе
- `TrigonometricFunction` - абстрактный класс для тригонометрических функций
- `Sin`, `Cos`, `Tan` - конкретные реализации

Результат: Создана библиотека базовых математических функций с корректными областями определения.

Задание 4

В пакете `functions.meta` реализованы классы для комбинирования функций:

- `Sum`, `Mult` - сумма и произведение с пересечением областей определения
- `Power` - возведение в степень
- `Scale`, `Shift` - масштабирование и сдвиг
- `Composition` - композиция функций

Результат: Создан мощный механизм для построения сложных функций из простых компонентов.

Задание 5

Создан класс `Functions` с статическими методами-фабриками. Конструктор закрыт для обеспечения утилитарности класса.

```
package functions;

import functions.meta.*;

public final class Functions { 5 usages
    private Functions() { no usages
        // Запрет на создание объектов
    }

    public static Function shift(Function f, double shiftX, double shiftY) { return new Shift(f, shiftX, shiftY); }

    public static Function scale(Function f, double scaleX, double scaleY) { return new Scale(f, scaleX, scaleY); }

    public static Function power(Function f, double power) { return new Power(f, power); }

    public static Function sum(Function f1, Function f2) { return new Sum(f1, f2); }

    public static Function mult(Function f1, Function f2) { return new Mult(f1, f2); }

    public static Function composition(Function f1, Function f2) { return new Composition(f1, f2); }
}
```

Результат: Упрощенный API для создания мета-функций через статические методы.

Задание 6

Создан класс `TabulatedFunctions` с методом `tabulate()`, содержащий вспомогательные статические методы для работы с табулированными функциями. Реализована проверка границ области определения.

```
package functions;

import java.io.*;

public final class TabulatedFunctions { 10 usages
    private TabulatedFunctions() {} // Запрет на создание объектов no usages

    public static TabulatedFunction tabulate(Function function, double leftX, double rightX, int pointsCount) {...}

    public static void outputTabulatedFunction(TabulatedFunction function, OutputStream out) {...}

    public static TabulatedFunction inputTabulatedFunction(InputStream in) {...}

    public static void writeTabulatedFunction(TabulatedFunction function, Writer out) {...}

    // Чтение данных из потока
    public static TabulatedFunction readTabulatedFunction(Reader in) {...}
}
```

Результат: Возможность автоматического преобразования аналитических функций в табулированные представления.

Задание 7

Добавлены методы для работы с потоками:

- `outputTabulatedFunction()/inputTabulatedFunction()` - бинарный формат
- `writeTabulatedFunction()/readTabulatedFunction()` - текстовый формат

Результат: Исключения `IOException` преобразуются в `RuntimeException`. Потоки закрываются автоматически.

Задание 8

Проведено тестирование всех компонентов системы

Результат: всё работает. (результат в конце файла)

Задание 9

Реализована сериализация двумя способами:

- `Serializable` - автоматическая сериализация
- `Externalizable` - ручное управление процессом

Результат: `Serializable` проще в реализации, но больше размер файла. `Externalizable` эффективнее по размеру, но сложнее в реализации. Оба способа сохраняют целостность данных.

Вывод в main():

Тестирование всего и вся

1. Тест базовых функций

Экспонента:

Область определения: $[-\text{Infinity}, \text{Infinity}]$

$$\exp(-1,0) = 0,368$$

$$\exp(-0,5) = 0,607$$

$$\exp(0,0) = 1,000$$

$$\exp(0,5) = 1,649$$

$$\exp(1,0) = 2,718$$

Логарифм по основанию 10:

Область определения: $[0,0, \text{Infinity}]$

$$\log_{10}(0,1) = -1,000$$

$$\log_{10}(1,0) = 0,000$$

$$\log_{10}(10,0) = 1,000$$

Тригонометрические функции на $[0, \pi]$ с шагом 0.1:

$$x=0,0: \sin=0,000000, \cos=1,000000$$

$$x=0,1: \sin=0,099833, \cos=0,995004$$

$$x=0,2: \sin=0,198669, \cos=0,980067$$

$$x=0,3: \sin=0,295520, \cos=0,955336$$

$$x=0,4: \sin=0,389418, \cos=0,921061$$

$$x=0,5: \sin=0,479426, \cos=0,877583$$

$$x=0,6: \sin=0,564642, \cos=0,825336$$

$$x=0,7: \sin=0,644218, \cos=0,764842$$

$$x=0,8: \sin=0,717356, \cos=0,696707$$

$$x=0,9: \sin=0,783327, \cos=0,621610$$

x=1,0: sin=0,841471, cos=0,540302
x=1,1: sin=0,891207, cos=0,453596
x=1,2: sin=0,932039, cos=0,362358
x=1,3: sin=0,963558, cos=0,267499
x=1,4: sin=0,985450, cos=0,169967
x=1,5: sin=0,997495, cos=0,070737
x=1,6: sin=0,999574, cos=-0,029200
x=1,7: sin=0,991665, cos=-0,128844
x=1,8: sin=0,973848, cos=-0,227202
x=1,9: sin=0,946300, cos=-0,323290
x=2,0: sin=0,909297, cos=-0,416147
x=2,1: sin=0,863209, cos=-0,504846
x=2,2: sin=0,808496, cos=-0,588501
x=2,3: sin=0,745705, cos=-0,666276
x=2,4: sin=0,675463, cos=-0,737394
x=2,5: sin=0,598472, cos=-0,801144
x=2,6: sin=0,515501, cos=-0,856889
x=2,7: sin=0,427380, cos=-0,904072
x=2,8: sin=0,334988, cos=-0,942222
x=2,9: sin=0,239249, cos=-0,970958
x=3,0: sin=0,141120, cos=-0,989992
x=3,1: sin=0,041581, cos=-0,999135

2. Тест табулированной функции

ArrayTabulatedFunction (2x+1):

1 --- (0.0, 1.0)

2 --- (1.0, 3.0)

3 --- (2.0, 5.0)

4 --- (3.0, 7.0)

5 --- (4.0, 9.0)

6 --- (5.0, 11.0)

Интерполяция ArrayTabulatedFunction:

$f(0,5) = 2,000$

$f(1,5) = 4,000$

$f(2,5) = 6,000$

$f(3,5) = 8,000$

$f(4,5) = 10,000$

LinkedListTabulatedFunction (2x+1):

1 --- (0.0, 1.0)

2 --- (1.0, 3.0)

3 --- (2.0, 5.0)

4 --- (3.0, 7.0)

5 --- (4.0, 9.0)

6 --- (5.0, 11.0)

Модификация точек:

1 --- (0.0, 1.0)

2 --- (2.0, 10.0)

3 --- (2.5, 6.25)

4 --- (3.0, 7.0)

5 --- (4.0, 9.0)

6 --- (5.0, 11.0)

3. Тест мета-функций

Сравнение оригинальных и табулированных функций:

Сравнение $\sin(x)$ и его табулированного аналога:

x=0,0: orig=0,000000, tab=0,000000

x=0,1: orig=0,099833, tab=0,097982

x=0,2: orig=0,198669, tab=0,195963

x=0,3: orig=0,295520, tab=0,293945

x=0,4: orig=0,389418, tab=0,385907

x=0,5: orig=0,479426, tab=0,472070

x=0,6: orig=0,564642, tab=0,558234

x=0,7: orig=0,644218, tab=0,643982

x=0,8: orig=0,717356, tab=0,707935

x=0,9: orig=0,783327, tab=0,771888

x=1,0: orig=0,841471, tab=0,835841

x=1,1: orig=0,891207, tab=0,883993

x=1,2: orig=0,932039, tab=0,918022

x=1,3: orig=0,963558, tab=0,952051

x=1,4: orig=0,985450, tab=0,984808

x=1,5: orig=0,997495, tab=0,984808

x=1,6: orig=0,999574, tab=0,984808

x=1,7: orig=0,991665, tab=0,984808

x=1,8: orig=0,973848, tab=0,966204

x=1,9: orig=0,946300, tab=0,932175

x=2,0: orig=0,909297, tab=0,898147

x=2,1: orig=0,863209, tab=0,862441

x=2,2: orig=0,808496, tab=0,798488

x=2,3: orig=0,745705, tab=0,734535

x=2,4: orig=0,675463, tab=0,670582

x=2,5: orig=0,598472, tab=0,594072

x=2,6: orig=0,515501, tab=0,507908

x=2,7: orig=0,427380, tab=0,421745

x=2,8: orig=0,334988, tab=0,334698
x=2,9: orig=0,239249, tab=0,236716
x=3,0: orig=0,141120, tab=0,138735
x=3,1: orig=0,041581, tab=0,040753

Сравнение $\cos(x)$ и его табулированного аналога:

x=0,0: orig=1,000000, tab=1,000000
x=0,1: orig=0,995004, tab=0,982723
x=0,2: orig=0,980067, tab=0,965446
x=0,3: orig=0,955336, tab=0,948170
x=0,4: orig=0,921061, tab=0,914355
x=0,5: orig=0,877583, tab=0,864608
x=0,6: orig=0,825336, tab=0,814862
x=0,7: orig=0,764842, tab=0,764620
x=0,8: orig=0,696707, tab=0,688404
x=0,9: orig=0,621610, tab=0,612188
x=1,0: orig=0,540302, tab=0,535972
x=1,1: orig=0,453596, tab=0,450633
x=1,2: orig=0,362358, tab=0,357141
x=1,3: orig=0,267499, tab=0,263648
x=1,4: orig=0,169967, tab=0,169931
x=1,5: orig=0,070737, tab=0,070437
x=1,6: orig=-0,029200, tab=-0,029056
x=1,7: orig=-0,128844, tab=-0,128549
x=1,8: orig=-0,227202, tab=-0,224761
x=1,9: orig=-0,323290, tab=-0,318254
x=2,0: orig=-0,416147, tab=-0,411747
x=2,1: orig=-0,504846, tab=-0,504272
x=2,2: orig=-0,588501, tab=-0,580488

x=2,3: orig=-0,666276, tab=-0,656704
x=2,4: orig=-0,737394, tab=-0,732920
x=2,5: orig=-0,801144, tab=-0,794171
x=2,6: orig=-0,856889, tab=-0,843917
x=2,7: orig=-0,904072, tab=-0,893664
x=2,8: orig=-0,942222, tab=-0,940984
x=2,9: orig=-0,970958, tab=-0,958261
x=3,0: orig=-0,989992, tab=-0,975537
x=3,1: orig=-0,999135, tab=-0,992814

Сумма квадратов табулированных син и кос:

$\sin^2(x) + \cos^2(x)$ (табулированные, 10 точек):

x=0,0: result=1,0000000000
x=0,1: result=0,9753452893
x=0,2: result=0,9704883234
x=0,3: result=0,9854291023
x=0,4: result=0,9849684785
x=0,5: result=0,9703975808
x=0,6: result=0,9756244278
x=0,7: result=0,9993578909
x=0,8: result=0,9750730614
x=0,9: result=0,9705859766
x=1,0: result=0,9858966365
x=1,1: result=0,9845147652
x=1,2: result=0,9703137486
x=1,3: result=0,9759104767
x=1,4: result=0,9987226923
x=1,5: result=0,9748077439
x=1,6: result=0,9706905402

x=1,7: result=0,9863710813
x=1,8: result=0,9840679624
x=1,9: result=0,9702368269
x=2,0: result=0,9762034362
x=2,1: result=0,9980944042
x=2,2: result=0,9745493369
x=2,3: result=0,9708020143
x=2,4: result=0,9868524365
x=2,5: result=0,9836280701
x=2,6: result=0,9701668157
x=2,7: result=0,9765033061
x=2,8: result=0,9974730266
x=2,9: result=0,9742978404
x=3,0: result=0,9709203989
x=3,1: result=0,9873407022

$3 * \sin(2x)$:

x=0,000: result=0,000
x=0,785: result=1,148
x=1,571: result=2,121
x=2,356: result=2,772
x=3,142: result=3,000

$\cos(x - \pi/2) + 1$:

x=0,000: result=1,000
x=0,785: result=1,707
x=1,571: result=2,000
x=2,356: result=1,707
x=3,142: result=1,000

$\sin(x) * \cos(x)$:

x=0,000: result=0,000

x=0,785: result=0,500

x=1,571: result=0,000

x=2,356: result=-0,500

x=3,142: result=-0,000

4. Тест файловых операций

Экспонента - текстовый формат:

Экспонента (0 до 10, шаг 1):

Оригинал vs Прочитанный (текстовый формат):

x=0,0: orig=1,000000, read=1,000000

x=1,0: orig=2,718282, read=2,718282

x=2,0: orig=7,389056, read=7,389056

x=3,0: orig=20,085537, read=20,085537

x=4,0: orig=54,598150, read=54,598150

x=5,0: orig=148,413159, read=148,413159

x=6,0: orig=403,428793, read=403,428793

x=7,0: orig=1096,633158, read=1096,633158

x=8,0: orig=2980,957987, read=2980,957987

x=9,0: orig=8103,083928, read=8103,083928

x=10,0: orig=22026,465795, read=22026,465795

Логарифм - бинарный формат:

Логарифм (0.1 до 10, шаг 1):

Оригинал vs Прочитанный (бинарный формат):

x=0,1: orig=-2,302585, read=-2,302585

x=1,1: orig=0,092705, read=0,092705

x=2,1: orig=0,740233, read=0,740233

x=3,1: orig=1,130147, read=1,130147

x=4,1: orig=1,409999, read=1,409999

x=5,1: orig=1,628429, read=1,628429

x=6,1: orig=1,807603, read=1,807603

x=7,1: orig=1,959502, read=1,959502

x=8,1: orig=2,091344, read=2,091344

x=9,1: orig=2,207812, read=2,207812

5. Тест сериализации (Serializable)

Оригинальная функция $\ln(\exp(x)) = x$ (0 до 10, 11 точек):

$f(0,0) = 0,000000$

$f(1,0) = 1,000000$

$f(2,0) = 2,000000$

$f(3,0) = 3,000000$

$f(4,0) = 4,000000$

$f(5,0) = 5,000000$

$f(6,0) = 6,000000$

$f(7,0) = 7,000000$

$f(8,0) = 8,000000$

$f(9,0) = 9,000000$

$f(10,0) = 10,000000$

Десериализованная функция:

$f(0,0) = 0,000000$

$f(1,0) = 1,000000$

$f(2,0) = 2,000000$

$f(3,0) = 3,000000$

$f(4,0) = 4,000000$

$f(5,0) = 5,000000$

$$f(6,0) = 6,000000$$

$$f(7,0) = 7,000000$$

$$f(8,0) = 8,000000$$

$$f(9,0) = 9,000000$$

$$f(10,0) = 10,000000$$

6. Тест сериализации (Externalizable)

Оригинальная функция (Externalizable):

$$f(0,0) = 0,000000$$

$$f(1,0) = 1,000000$$

$$f(2,0) = 2,000000$$

$$f(3,0) = 3,000000$$

$$f(4,0) = 4,000000$$

$$f(5,0) = 5,000000$$

$$f(6,0) = 6,000000$$

$$f(7,0) = 7,000000$$

$$f(8,0) = 8,000000$$

$$f(9,0) = 9,000000$$

$$f(10,0) = 10,000000$$

Десериализованная функция (Externalizable):

$$f(0,0) = 0,000000$$

$$f(1,0) = 1,000000$$

$$f(2,0) = 2,000000$$

$$f(3,0) = 3,000000$$

$$f(4,0) = 4,000000$$

$$f(5,0) = 5,000000$$

$$f(6,0) = 6,000000$$

$$f(7,0) = 7,000000$$

$$f(8,0) = 8,000000$$

$$f(9,0) = 9,000000$$

$$f(10,0) = 10,000000$$