

Министерство науки и высшего образования Российской Федерации

федеральное государственное автономное
образовательное учреждение высшего образования
«Самарский национальный исследовательский университет
имени академика С.П. Королева»

Институт информатики и кибернетики

Кафедра технической кибернетики

Отчет по лабораторной работе №3

Дисциплина: «ООП»

Тема «Исключения и интерфейсы»

Выполнил: Чечеткин Д.А.

Группа: 6201-120303D

Самара, 2025

Отчет по лабораторной работе 3

Задание 1.

`java.lang.Exception` — базовый класс для всех проверяемых исключений.

`java.lang.IndexOutOfBoundsException` — ошибка при выходе за пределы индекса.

`java.lang.ArrayIndexOutOfBoundsException` — частный случай предыдущего, когда индекс выходит за границы массива.

`java.lang.IllegalArgumentException` — используется, если метод получил некорректный аргумент.

`java.lang.IllegalStateException` — сигнализирует о некорректном состоянии объекта при вызове метода.

Задание 2.

`FunctionPointIndexOutOfBoundsException` — выбрасывается при обращении к несуществующей точке функции.

`InappropriateFunctionPointException` — выбрасывается при попытке добавить или изменить точку функции так, что нарушается упорядоченность точек по оси X.

Задание 3.

Добавлены проверки и выброс исключений в конструкторах и методах.

Задание 4.

Создан новый класс `LinkedListTabulatedFunction`, который описывает табулированную функцию, но хранит данные не в массиве, а в двусвязном циклическом списке.

Вспомогательный класс `FunctionNode` — хранит данные о точке функции и ссылки на соседние узлы (`next`, `prev`). Имеет уровень доступа `private`, чтобы избежать нарушения инкапсуляции.

Задание 5.

В классе `LinkedListTabulatedFunction` реализованы те же конструкторы, методы и исключения, что и в `ArrayTabulatedFunction`.

Задание 6.

Класс TabulatedFunction был переименован в ArrayTabulatedFunction.

Создан интерфейс TabulatedFuncion, объявляющий все методы, реализуемые в классах ArrayTabulatedFunction и LinkedListTabulatedFunction.

```
package functions;

public interface TabulatedFunction { 5 usages 2 implementations
    double getLeftDomainBorder(); 2 usages 2 implementations
    double getRightDomainBorder(); 2 usages 2 implementations
    double getFunctionValue(double x); 2 usages 2 implementations
    int getPointsCount(); 1 usage 2 implementations
    FunctionPoint getPoint(int index); 2 usages 2 implementations
    void setPoint(int index, FunctionPoint point) throws InappropriateFunctionPointException;
    double getPointX(int index); 1 usage 2 implementations
    void setPointX(int index, double x) throws InappropriateFunctionPointException; no usages 2
    double getPointY(int index); 1 usage 2 implementations
    void setPointY(int index, double y); 1 usage 2 implementations
    void deletePoint(int index); 1 usage 2 implementations
    void addPoint(FunctionPoint point) throws InappropriateFunctionPointException; 1 usage 2 imp
}
```

Задание 7.

Проверка работы классов

ArrayTabulatedFunction:

Исходная функция:

```
Point 0: (0.0; 0.0)
Point 1: (1.0; 1.0)
Point 2: (2.0; 128.0)
Point 3: (3.0; 2187.0)
Point 4: (4.0; 16384.0)
```

Копия точки с индексом 3: (3.0; 2187.0)

Изменение координаты Y точки с индексом 3:

```
Point 0: (0.0; 0.0)
Point 1: (1.0; 1.0)
Point 2: (2.0; 128.0)
Point 3: (3.0; 7.0)
Point 4: (4.0; 16384.0)
```

Добавление точки (0.7; 11):

```
Point 0: (0.0; 0.0)
Point 1: (0.7; 11.0)
Point 2: (1.0; 1.0)
Point 3: (2.0; 128.0)
Point 4: (3.0; 7.0)
Point 5: (4.0; 16384.0)
```

Удаление точки с индексом 2:

```
Point 0: (0.0; 0.0)
Point 1: (0.7; 11.0)
Point 2: (2.0; 128.0)
Point 3: (3.0; 7.0)
Point 4: (4.0; 16384.0)
```

Значение функции в точке x = -1:

y = NaN

Значение функции в точке x = 3.4:

y = 6557.799999999998

Проверка исключения:

Исключение: Индекс 100 выходит за границу

LinkedListTabulatedFunction:

Исходная функция:

```
Point 0: (0.0; 0.0)
Point 1: (1.0; 1.0)
Point 2: (2.0; 128.0)
Point 3: (3.0; 2187.0)
Point 4: (4.0; 16384.0)
```

Копия точки с индексом 3: (3.0; 2187.0)

Изменение координаты Y точки с индексом 3:

```
Point 0: (0.0; 0.0)
Point 1: (1.0; 1.0)
Point 2: (2.0; 128.0)
Point 3: (3.0; 7.0)
Point 4: (4.0; 16384.0)
```

Добавление точки (0.7; 11):

```
Point 0: (0.0; 0.0)
Point 1: (0.7; 11.0)
Point 2: (1.0; 1.0)
Point 3: (2.0; 128.0)
Point 4: (3.0; 7.0)
Point 5: (4.0; 16384.0)
```

Удаление точки с индексом 2:

```
Point 0: (0.0; 0.0)
Point 1: (0.7; 11.0)
Point 2: (2.0; 128.0)
Point 3: (3.0; 7.0)
Point 4: (4.0; 16384.0)
```

Значение функции в точке x = -1:

Исключение при вычислении функции: x вне области определения

Значение функции в точке x = 3.4:

y = 6557.79999999998

Проверка исключения:

Исключение: Индекс 100 выходит за границу