

Лабораторная работа №2

Разработка классов табулированных функций и обработка исключений

Пожидаев Богдан Витальевич 6204-010302D

1. Задание 1: Изучение классов исключений

Изучены стандартные классы исключений Java: **Exception**, **IndexOutOfBoundsException**, **ArrayIndexOutOfBoundsException**, **IllegalArgumentException**, **IllegalStateException**.

Получено понимание иерархии исключений и их применения.

2. Задание 2: Создание классов исключений

Созданы два пользовательских класса исключений:

FunctionPointIndexOutOfBoundsException- наследуется от **IndexOutOfBoundsException**, выбрасывается при выходе за границы набора точек.

InappropriateFunctionPointException- наследуется от **Exception**, выбрасывается при некорректном добавлении или изменении точки.

3. Задание 3: Модификация **TabulatedFunction**

Класс **TabulatedFunction** переименован в **ArrayTabulatedFunction** и модифицирован:

- Конструкторы выбрасывают **IllegalArgumentException** при неверных параметрах
- Методы работы с точками выбрасывают **FunctionPointIndexOutOfBoundsException** при неверных индексах
- Методы **setPoint()**, **setPointX()** и **addPoint()** выбрасывают **InappropriateFunctionPointException** при нарушении порядка точек
- Метод **deletePoint()** выбрасывает **IllegalStateException** при попытке удалить точку, если точек меньше трех

4. Задание 4-5: **LinkedListTabulatedFunction**

Реализован класс **LinkedListTabulatedFunction** с использованием двусвязного циклического списка:

- Внутренний класс **FunctionNode** для элементов списка
- Методы **getNodeByIndex()**, **addNodeToTail()**, **addNodeByIndex()**, **deleteNodeByIndex()** для работы со списком

- Оптимизированный доступ к элементам через кэширование последнего **accessed** узла
- Реализованы все методы интерфейса **TabulatedFunction**

5. Задание 6: Создание интерфейса

Создан интерфейс **TabulatedFunction**, содержащий объявления общих методов:

- Методы работы с точками (**getPoint**, **setPoint**, **addPoint**, **deletePoint**)
- Методы получения координат (**getPointX**, **getPointY**, **setPointX**, **setPointY**)
- Методы получения информации о функции (**getPointsCount**, **getLeftDomainBorder**, **getRightDomainBorder**, **getFunctionValue**)

Оба класса (**ArrayTabulatedFunction** и **LinkedListTabulatedFunction**) реализуют этот интерфейс.

6. Задание 7: Тестирование

Разработан класс **Main** для тестирования всех функций и обработки исключений. Тестирование включает:

- Создание функций через массивы и связные списки
- Проверку вычисления значений функции
- Тестирование добавления, изменения и удаления точек
- Проверку обработки всех типов исключений

7. Результаты тестирования

=== ТЕСТИРОВАНИЕ КЛАССОВ ТАБУЛИРОВАННОЙ ФУНКЦИИ ===

ТЕСТИРОВАНИЕ **ArrayTabulatedFunction**

Функция успешно создана

Область определения: [0,0, 5,0]

Количество точек: 6

ТЕСТИРОВАНИЕ **LinkedListTabulatedFunction**

Функция успешно создана

Область определения: [0,0, 5,0]

Количество точек: 6

ТЕСТИРОВАНИЕ ИСКЛЮЧЕНИЙ С ПРАВИЛЬНЫМИ СООБЩЕНИЯМИ

1. Тестирование конструктора:

IllegalArgumentException: Левая граница области определения \geq правой

2. Тестирование выхода за границы индексов:

FunctionPointIndexOutOfBoundsException: Индекс выходит за границы: 11

3. Тестирование несоответствующих точек:

InappropriateFunctionPointException: Новый x должен быть между соседними точками

4. Тестирование удаления точек:

IllegalStateException: Невозможно удалить точку, так как количество точек < 3

=== ТЕСТИРОВАНИЕ ЗАВЕРШЕНО ===