Министерство образования и науки Российской Федерации  
  
Федеральное государственное автономное образовательноеучреждение высшего образования«Самарский государственный аэрокосмический университет имени академика С.П. Королева(национальный исследовательский университет)»

Факультет информатики  
Кафедра технической кибернетики

**Лабораторная работа № 3  
по курсу «Объектно-ориентированное программирование»**

Студент: Борисов Д.С.   
группа 6210  
Преподаватель:

Самара 2018

Задание

**1 Задание:** Ознакомиться (изучить документацию) со следующими классами исключений, входящих в API Java:

* java.lang.Exception;
* java.lang.IndexOutOfBoundsException;
* java.lang.ArrayIndexOutOfBoundsException;
* java.lang.IllegalArgumentException;
* java.lang.IllegalStateException.

1. **Задание:** В пакете functions создать два класса исключений:

* FunctionPointIndexOutOfBoundsException – исключение выхода за границы набора точек при обращении к ним по номеру, наследует от класса IndexOutOfBoundsException;
* InappropriateFunctionPointException – исключение, выбрасываемое при попытке добавления или изменения точки функции несоответствующим образом, наследует от класса Exception.

**3 Задание:** В разработанный ранее класс TabulatedFunction внести изменения, обеспечивающие выбрасывание исключений методами класса.

**4 Задание:** В пакете functions создать класс LinkedListTabulatedFunction,

объект которого также должен описывать табулированную функцию. Отличие этого класса должно заключаться в том, что для хранения набора точек в нем должен использоваться не массив, а динамическая структура – связный список.

* Описать класс элементов списка FunctionNode, содержащий информационное

поле для хранения данных типа FunctionPoint, а также поля для хранения ссылок на предыдущий и следующий элемент. Выберите и обоснуйте место описания класса и его видимость. Также выберите и обоснуйте реализацию инкапсуляции в этом классе.

* Описать класс LinkedListTabulatedFunction объектов списка, содержащий поле ссылки на объект головы, а также иные вспомогательные поля.
* В классе LinkedListTabulatedFunction реализовать метод FunctionNode getNodeByIndex(int index), возвращающий ссылку на объект элемента списка по его номеру. Нумерация значащих элементов (голова списка в данном случае к ним не относится) должна начинаться с 0. Метод должен обеспечивать оптимизацию доступа к элементам списка.
* В классе LinkedListTabulatedFunction реализовать метод FunctionNode addNodeToTail(), добавляющий новый элемент в конец списка и возвращающий ссылку на объект этого элемента.
* В классе LinkedListTabulatedFunction реализовать метод FunctionNode addNodeByIndex(int index), добавляющий новый элемент в указанную позицию списка и возвращающий ссылку на объект этого элемента.
* В классе LinkedListTabulatedFunction реализовать метод FunctionNode deleteNodeByIndex(int index), удаляющий элемент списка по номеру и возвращающий ссылку на объект удаленного элемента.

**5 Задание:** Для обеспечения второй функции класса LinkedListTabulatedFunction реализовать в классе конструкторы и методы, аналогичные конструкторам и методам класса TabulatedFunction. Конструкторы должны иметь те же параметры, методы должны иметь те же сигнатуры. Также должны выбрасываться те же виды исключений в тех же случаях.

**6 Задание:** Класс TabulatedFunction переименовать в класс ArrayTabulatedFunction.

Создать интерфейс TabulatedFunction, содержащий объявления общих методов классов ArrayTabulatedFunction и LinkedListTabulatedFunction.

Сделать так, чтобы оба класса функций реализовывали созданный интерфейс.

**7 Задание:** Проверить работу написанных классов.

Реферат

В процессе написания тестовых заданий необходимо ознакомиться с особенностями вызова исключений, изучить их документацию, создать два новых исключения и расставить исключения в определённых местах, а также реализовать двусвязный список.

Цель работы – дополнить пакет для работы с функциями одной переменной, заданными в табличной форме, добавив классы исключений, новый класс функций и базовый интерфейс.

В ходе выполнения работы запрещено использовать классы из пакета java.util.

Содержание

[Введение 6](#_Toc448163917)

[Выполнение заданий 7](#_Toc448163918)

[Заключение 8](#_Toc448163921)

Введение

Лабораторная работа №3 представляет собой ознакомление с основами работы с исключениями и обработкой ошибочных ситуаций, написание нового класса – двусвязного списка, а также его методов.

**ВЫПОЛНЕНИЕ ЗАДАНИЙ**

**1 Задание:** Ознакомиться с документацией классов исключений можно на сайте <https://docs.oracle.com>

**2 Задание:** В пакете functions создаётся два класса исключений, наследующиеся от соответствующих классов.

**3 Задание:** В определённых методах класса TabulatedFunction добавить обработку исключительных ситуаций написав throw <Имя соответствующего исключения>.

**4 Задание:** Для написания класса LinkedListTabulatedFunction необходимо сначала написать внутренний класс FunctionNode, имеющий поле FunctionPoint и поля ссылок на следующую и предыдущую точку соответственно также имеющих тип FunctionPoint. Класс LinkedListTabulatedFunction имеет поле головы списка (имеющий значение Null) и ссылки на первый и последний элемент в списке. Добавить для этого класса методы: получение элемента списка по его номеру, добавление элемента списка в его конец, а также в определённое место списка по индексу, удаление элемента по номеру.

**5 Задание:** Для написания конструкторов и методов для класса LinkedListTabulatedFunction, аналогичные конструкторам и методам класса TabulatedFunction, достаточно следовать тем же правилам, что в классе TabulatedFunction. Однако стоит не забывать о ссылках на след. и пред. элементы, которые нужно также менять, а также о обработках исключений.

**6 Задание:** Создав интерфейс TabulatedFunctionImpl и занесся туда общие методы классов ArrayTabulatedFunction и LinkedListTabulatedFunction, мы объединили эти два класса, и теперь они реализуют один интерфейс, тем самым суть работы с табулированными функциями заключена в типе интерфейса.

**7 Задание:** В методе main() достаточно просто проверить все написанные методы, конструкторы и работу классов ArrayTabulatedFunction и LinkedListTabulatedFunction.

Заключение

В процессе работы были созданы два новых исключения , расставленные в определённых местах написанного кода, новый класс – двусвязный список LinkedListTabulatedFunction, его конструктор и методы, аналогичные классу ArrayTabulatedFunction, а также интерфейс, который эти два класса будут реализовать.