Государственный Университет Молдовы

Факультет Математики и Информатики

Департамент Информатики

Лабораторная работа №2

“Tehnici Avansate De Programare”

Проверил: Георгий Латул

Выполнил: Чобану Артём

Группа: i1902

Кишинев 2021

**Задание:**

Создать собственное исключение (class).  
Создать метод (throw), который может возбуждать это исключение(throws).  
Написать метод, перехватывающий и обрабатывающий исключение (try-catch), возбуждаемое другим методом.  
Создать класс, который ведет себя как float, но не допускающий значений >+35767 и < -35768, а также значений POSITIVE\_INFINITY, NEGATIVE\_INFINITY, NaN вместо них записывать 33333.  
Выход за пределы значений >+35767 и < -35768, а также значение POSITIVE\_INFINITY рассматривать как исключительную ситуацию.  
Использовать класс для выполнения арифметических операций (x/y x%y).

Класс-исключение “Исключение-запрещённое число”

public class NotAllowedNumberException extends Exception  
{  
 public NotAllowedNumberException(String message)  
 {  
 super(message);  
 }  
}

Класс, который ведёт себя как float:

public class FakeFloat  
{  
 private double \_value;  
 public double Get\_Value() { return \_value; }  
 public void Set\_Value(double value) throws NotAllowedNumberException  
 {  
 if (value == Double.NEGATIVE\_INFINITY || Double.isNaN(value))  
 {  
 \_value = 33333;  
 }  
 if (value > 35767 || value < -35768 ||  
 value == Double.POSITIVE\_INFINITY)  
 {  
 \_value = 3333;  
 throw new NotAllowedNumberException(

"Значение должно быть больше чем 35767 и меньше -35768");  
 }  
 \_value = value;  
 }  
 public FakeFloat(double value)  
 {  
 \_value = value;  
 }  
 public void AddFakeFloat(FakeFloat value) throws NotAllowedNumberException  
 {  
 Set\_Value(\_value + value.Get\_Value());  
 }  
 public void SubstractFakeFloat(FakeFloat value) throws NotAllowedNumberException  
 {  
 Set\_Value(\_value - value.Get\_Value());  
 }  
 public void DivideByFakeFloat(FakeFloat value) throws NotAllowedNumberException  
 {  
 Set\_Value(\_value / value.Get\_Value());  
 }  
 public void MultiplyByFakeFloat(FakeFloat value) throws NotAllowedNumberException  
 {  
 Set\_Value(\_value \* value.Get\_Value());  
 }  
 //Остаток от деления  
public void Remainder(FakeFloat value) throws NotAllowedNumberException  
 {  
 Set\_Value(\_value % value.Get\_Value());  
 }  
}

Main:

public class Main  
{  
 public static void main(String[] args)  
 {  
 FakeFloat fakeFloat1 = new FakeFloat(5);  
 FakeFloat fakeFloat2 = new FakeFloat(6);  
  
 System.out.printf("Значение 1: %s %n", fakeFloat1.Get\_Value());  
 System.out.printf("Значение 2: %s %n", fakeFloat2.Get\_Value());  
  
 System.out.println();  
 try  
 {  
 fakeFloat1.AddFakeFloat(fakeFloat2);  
 System.out.printf("Значение 1 + значение 2: %s %n", fakeFloat1.Get\_Value());  
  
 fakeFloat2.SubstractFakeFloat(new FakeFloat(50));  
 System.out.printf("Значение 2 - 50: %s %n", fakeFloat2.Get\_Value());  
  
 fakeFloat2.DivideByFakeFloat(fakeFloat1);  
 System.out.printf("Значение 2 поделённое на значение 1: %s %n", fakeFloat2.Get\_Value());  
  
 fakeFloat2.MultiplyByFakeFloat(fakeFloat1);  
 System.out.printf("Значение 2 умноженное на значение 1: %s %n", fakeFloat2.Get\_Value());  
  
 fakeFloat1.Remainder(new FakeFloat(5));  
 System.out.printf("Остаток от деления значения 1 на 5: %s %n", fakeFloat1.Get\_Value());  
 } catch (NotAllowedNumberException exception)  
 {  
 System.out.printf("%n Ошибка! %s %n", exception.getMessage());  
 }  
  
 System.out.println();  
 System.out.println("Тестирование обработки ошибок!");  
 try  
 {  
 System.out.println("Передача значения POSITIVE\_INFINITY");  
 fakeFloat1.Set\_Value(Double.POSITIVE\_INFINITY);  
 } catch (NotAllowedNumberException exception)  
 {  
 System.out.printf("Ошибка! %s. Значение fakeFloat: %s %n", exception.getMessage(), fakeFloat1.Get\_Value());  
 }  
  
 try  
 {  
 System.out.println("Передача значения -50000");  
 fakeFloat1.Set\_Value(-50000);  
 } catch (NotAllowedNumberException exception)  
 {  
 System.out.printf("Ошибка! %s. Значение fakeFloat: %s %n", exception.getMessage(), fakeFloat1.Get\_Value());  
 }  
  
 try  
 {  
 System.out.println("Передача значения 50000");  
 fakeFloat1.Set\_Value(50000);  
 } catch (NotAllowedNumberException exception)  
 {  
 System.out.printf("Ошибка! %s. Значение fakeFloat: %s %n", exception.getMessage(), fakeFloat1.Get\_Value());  
 }  
 }  
}