Федеральное государственное бюджетное образовательное учреждение высшего образования

«Сибирский государственный университет телекоммуникаций и информатики»

(СибГУТИ)

Кафедра прикладной математики и кибернетики

Лабораторная работа № 8

по дисциплине «Современные технологии программирования»

Выполнил:

студент группы ИП-712

Алексеев Степан Владимирович

ФИО студента

Работу проверил:

ассистент кафедры Агалаков А.А.

ФИО преподавателя

Новосибирск 2020 г.

Оглавление

[ЗАДАНИЕ 2](#_Toc53933610)

[ТЕСТОВЫЕ НАБОРЫ ДАННЫХ 3](#_Toc53933611)

[ДЕМОНСТРАЦИЯ РАБОТЫ ПРОГРАММЫ 4](#_Toc53933612)

[ВЫВОД 5](#_Toc53933613)

[ПРИЛОЖЕНИЕ 6](#_Toc53933614)

[Листинг 1. TEditorForCommonFraction.cs 6](#_Toc53933615)

[Листинг 2. TEditorTests.cs 7](#_Toc53933616)

# ЗАДАНИЕ

1. Разработать и реализовать класс TEditor «Ввод и редактирование простых дробей», используя класс С++.

2. Протестировать каждую операцию, определенную на типе данных, используя средства модульного тестирования.

3. Если необходимо, предусмотрите возбуждение исключительных ситуаций.

# ТЕСТОВЫЕ НАБОРЫ ДАННЫХ

TFrac tf2 = new TFrac(36, 7);

TEditorForCommonFraction te = new TEditorForCommonFraction();

te.writeNewFToFraction(ref tf2, "27/4");

Assert.AreEqual(tf2.f, "27/4");

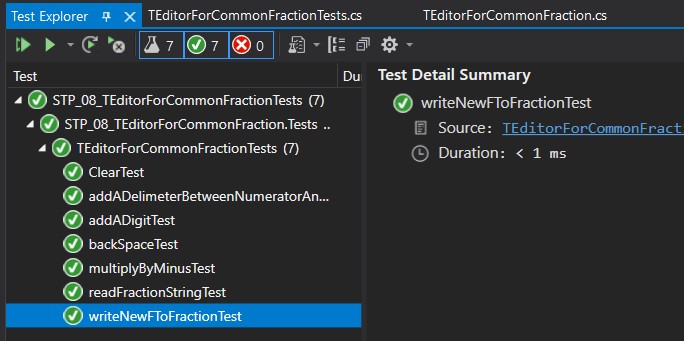
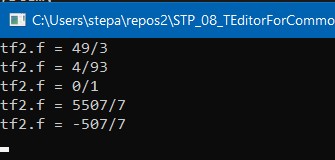
TFrac tf2 = new TFrac(36, 7);

TEditorForCommonFraction te = new TEditorForCommonFraction();

string str = te.readFractionString(tf2);

Assert.AreEqual(str, "36/7");

# ДЕМОНСТРАЦИЯ РАБОТЫ ПРОГРАММЫ



# ВЫВОД

Подлатал класс TFrac для большей совместимости с новыми требованиями методов класса TEditor. Познакомился с новыми способами передачи объектов по ссылке в языке сишарп.

# ПРИЛОЖЕНИЕ

## Листинг 1. TEditorForCommonFraction.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using STP\_04\_ADT\_TFrac;

namespace STP\_08\_TEditorForCommonFraction

{

public class TEditorForCommonFraction

{

static void Main(string[] args)

{

TEditorForCommonFraction te = new TEditorForCommonFraction();

TFrac tf2 = new TFrac(49, 3);

Console.WriteLine("tf2.f = " + tf2.f);

te.addADelimeterBetweenNumeratorAndDenominator(tf2, 1);

Console.WriteLine("tf2.f = " + tf2.f);

te.Clear(ref tf2);

Console.WriteLine("tf2.f = " + tf2.f);

tf2 = new TFrac(507, 7);

te.addADigit(ref tf2, 5, 0);

Console.WriteLine("tf2.f = " + tf2.f);

tf2 = te.multiplyByMinus(tf2);

Console.WriteLine("tf2.f = " + tf2.f);

Console.ReadLine();

}

//в дробь добавляю цифру digit.

public string addADigit(ref TFrac tf, int digit, int index)

{

tf.f = tf.f.Insert(index, digit.ToString());

return tf.f;

}

public TFrac multiplyByMinus(TFrac tf)

{

return new TFrac(tf.numerator \* (-1), tf.denominator);

}

public void addADelimeterBetweenNumeratorAndDenominator(TFrac tf, int index)

{//В общем перемещает слеш по новому индексу, а если его не было, то просто устанавливает

if (tf.f.Contains("/"))

{

int ind = tf.f.IndexOf("/");

tf.f = tf.f.Remove(ind, 1);

}

tf.f = tf.f.Insert(index, "/");

}

public void backSpace(TFrac tf)

{

tf.f = tf.f.Remove(tf.f.Length - 1, 1);

}

public void Clear(ref TFrac tf)

{

if(tf == null)

{

throw new WrongInputException();

}

/\* tf.numerator = 0;

tf.denominator = 1;

tf.f = "0/1";\*/

tf = new TFrac(0, 1);

}

public string readFractionString(TFrac tf)

{

return tf.f;

}

public void writeNewFToFraction(ref TFrac tf, string newFR)

{

tf = new TFrac(newFR);

}

}

public class WrongInputException : Exception

{ public WrongInputException()

{

Console.WriteLine("wrong input");

}

}

}}

## Листинг 2. TEditorTests.cs

using Microsoft.VisualStudio.TestTools.UnitTesting;

using STP\_04\_ADT\_TFrac;

using STP\_08\_TEditorForCommonFraction;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace STP\_08\_TEditorForCommonFraction.Tests

{

[TestClass()]

public class TEditorForCommonFractionTests

{

TFrac tf = new TFrac(0, 1);

TEditorForCommonFraction te = new TEditorForCommonFraction();

[TestMethod()]

public void addADigitTest()

{

te.addADigit(ref tf, 3, 0);

Assert.AreEqual(tf.f, "30/1");

}

[TestMethod()]

public void multiplyByMinusTest()

{

TFrac tf2 = new TFrac(35, 1);

TEditorForCommonFraction te = new TEditorForCommonFraction();

TFrac tfMinus = te.multiplyByMinus( tf2);

Assert.AreEqual(tfMinus.f, "-35/1");

}

[TestMethod()]

public void addADelimeterBetweenNumeratorAndDenominatorTest()

{

TFrac tf2 = new TFrac(36, 1);

TEditorForCommonFraction te = new TEditorForCommonFraction();

te.addADelimeterBetweenNumeratorAndDenominator(tf2, 1);

Assert.AreEqual(tf2.f, "3/61");

}

[TestMethod()]

public void backSpaceTest()

{

TFrac tf2 = new TFrac(36, 1);

TEditorForCommonFraction te = new TEditorForCommonFraction();

te.addADelimeterBetweenNumeratorAndDenominator(tf2, 1);

Assert.AreEqual(tf2.f, "3/61");

}

[TestMethod()]

public void ClearTest()

{

TFrac tf2 = new TFrac(36, 1);

TEditorForCommonFraction te = new TEditorForCommonFraction();

te.Clear(ref tf2);

Assert.AreEqual(tf2.f, "0/1");

}

[TestMethod()]

public void readFractionStringTest()

{

TFrac tf2 = new TFrac(36, 7);

TEditorForCommonFraction te = new TEditorForCommonFraction();

string str = te.readFractionString(tf2);

Assert.AreEqual(str, "36/7");

}

[TestMethod()]

public void writeNewFToFractionTest()

{

TFrac tf2 = new TFrac(36, 7);

TEditorForCommonFraction te = new TEditorForCommonFraction();

te.writeNewFToFraction(ref tf2, "27/4");

Assert.AreEqual(tf2.f, "27/4");

}

}

}