Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное

учреждение высшего образования

ИРКУТСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ

ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Институт информационных Технологий

и Анализа Данных

**СВОЙ ТИП**

Отчет по лабораторной работе № 3

по дисциплине технологии программирования

Выполнил cтудент ИСТБ-22-3 К.Н. Толстиков

номер группы подпись И. О. Фамилия

дата

Принял З.А. Бахвалова

Должность подпись И. О. Фамилия

дата

Иркутск 2024

**Постановка задачи**

**Задача №12 из 1 лабораторной работы:**

Создать класс, реализующий операции в соответствии с заданием:

Правильная дробь, задаваемая числителем и знаменателем.

* сложение
* вычитание
* умножение
* деление двух дробей
* сокращение дроби
* сравнение двух дробей

Создать GUI интерфейс типа калькулятор для взаимодействия пользователя с программой. Пользователь вводит числитель и знаменатель дробей в текстовые поля. Операцию над дробями выбирает из выпадающего меню. Результат будет выведен программой в текстовое поле вывода. Пользователь должен учитывать, что результат будет вычислен и выведен только при правильном вводе дробей. (Правильным считается ввод, когда все числители и знаменатели являются целым числом в диапазоне (-2,147,483,648; 2,147,483,647)).

**Описание метода main:**

В контексте Windows Forms, который является фреймворком для создания графических интерфейсов пользователя в приложениях Windows, метод Main() используется для инициализации и запуска главного окна приложения. В методе Main() происходит вызов следующих методов:

* ApplicationConfiguration.Initialize(). Этот метод инициализирует конфигурацию приложения.
* Application.Run(new Form1()). Этот метод запускает главное окно приложения.

1. **Класс Form1**

Класс представляет собой окно (форму) в приложении Windows Forms. Класс наследуется от базового класса Form, что делает его полноценным графическим окном с поддержкой всех стандартных свойств, методов и событий, предоставляемых Windows Forms.

Таблица полей класса

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Имя | Модификатор доступа | Тип | Назначение |
| 1 | fraction\_1 | private | Объект класса Fraction | Дробь 1 |
| 2 | fraction\_2 | private | Объект класса Fraction | Дробь 2 |

* 1. **Метод Form1()**

Метод является конструктором класса. В нем происходит инициализация компонентов окна с помощью вызова метода InitializeComponent(), настройка расположения окна, восстановление значения в текстовом поле путем считывания значения из настроек.

* 1. **Метод create\_fraction1()**

Таблица спецификаций метода

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Имя | Назначение | Тип | Ед. Из. | ОДВ | Вход/выход |
| 1 | fraction\_1 | Дробь 1 | Fraction | - | - | Входные данные |

* 1. **Метод create\_fraction2()**

Таблица спецификаций метода

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Имя | Назначение | Тип | Ед. Из. | ОДВ | Вход/выход |
| 1 | fraction\_2 | Дробь 2 | Fraction | - | - | Входные данные |

* 1. **Метод update()**

Метод отвечает за вызов метода с аргументами истина или ложь в зависимости от условий существования дробей и их нормального ввода.

* 1. **isNullFraction(Fraction fr)**

Метод возвращает true, если дробь ничего не содержит; false, если дробь не пустая.

* 1. **Метод setActiveBtn(bool bl)**

Метод выставляет активность окна с выпадающими действиями согласно входному аргументу bl.

* 1. **Методы btnCleanNum1\_Click, btnCleanDen1\_Click, btnCleanNum2\_Click, btnCleanDen2\_Click**

Методы являются обработчиками нажатия кнопок «Очистить». Каждая из кнопок отвечает за очистку своего текстового поля на форме.

* 1. **Метод switchOper()**

Метод выполняет одно из действий над дробями в зависимости от выбранного действия в выпадающем меню.

Таблица спецификаций метода

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Имя | Назначение | Тип | Ед. Из. | ОДВ | Вход/выход |
| 1 | res1 | Результирующая дробь сложения дробей | Fraction | - | - | Выходные данные |
| 2 | res2 | Результирующая дробь вычитания первой дроби от второй | Fraction | - | - | Выходные данные |
| 3 | res3 | Результирующая дробь умножения дробей | Fraction | - | - | Выходные данные |
| 4 | res4 | Результирующая дробь деления дробей | Fraction | - | - | Выходные данные |

* 1. **Методы txtNum1\_TextChanged и txtDen1\_TextChanged**

Методы являются обработчиками события изменения данных в текстовых полях дроби 1.

* 1. **Методы txtNum2\_TextChanged и txtDen2\_TextChanged**

Методы являются обработчиками события изменения данных в текстовых полях дроби 2.

* 1. **Метод cmbOperations\_SelectedIndexChanged**

Метод является обработчиком выбора действия в выпадающем меню.

* 1. **Метод isNormalInput1()**

Метод возвращает true, если дробь 1 является целочисленной и входит в область допустимых значений int. Иначе, возвращает false.

* 1. **Метод isNormalInput2()**

Метод возвращает true, если дробь 2 является целочисленной и входит в область допустимых значений int. Иначе, возвращает false.

1. **Класс Fraction**

Класс реализует свой созданный тип правильной дроби.

Таблица полей класса

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Имя | Модификатор доступа | Тип | Назначение |
| 1 | numerator | private | int | Числитель |
| 2 | denominator | private | int | Знаменатель |

* 1. **Метод getNumerator**

Метод возвращает числитель дроби.

* 1. **Метод getDenominator**

Метод возвращает знаменатель дроби.

* 1. **Метод Fraction(int numerator, int denominator)**

Метод является конструктором класса Fraction.

* 1. **Оператор operator +(Fraction a, Fraction b)**

Реализует обработку сложения двух дробей и возвращает результирующую дробь.

Таблица спецификаций

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Имя | Назначение | Тип | Ед. Из. | ОДВ | Вход/выход |
| 1 | newNumerator | Результирующий числитель | int | - | [-2147483648 ,2147483647] | Выходные данные |
| 2 | newDenominator | Результирующий знаменатель | int | - | [-2147483648 ,2147483647] | Выходные данные |

* 1. **Оператор operator -(Fraction a, Fraction b)**

Реализует обработку вычитания от первой дроби вторую и возвращает результирующую дробь.

Таблица спецификаций

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Имя | Назначение | Тип | Ед. Из. | ОДВ | Вход/выход |
| 1 | newNumerator | Результирующий числитель | int | - | [-2147483648 ,2147483647] | Выходные данные |
| 2 | newDenominator | Результирующий знаменатель | int | - | [-2147483648 ,2147483647] | Выходные данные |

* 1. **Оператор operator \*(Fraction a, Fraction b)**

Реализует обработку умножения двух дробей и возвращает результирующую дробь.

Таблица спецификаций

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Имя | Назначение | Тип | Ед. Из. | ОДВ | Вход/выход |
| 1 | newNumerator | Результирующий числитель | int | - | [-2147483648 ,2147483647] | Выходные данные |
| 2 | newDenominator | Результирующий знаменатель | int | - | [-2147483648 ,2147483647] | Выходные данные |

* 1. **Оператор operator /(Fraction a, Fraction b)**

Реализует обработку деления первой дроби на вторую и возвращает результирующую дробь.

Таблица спецификаций

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Имя | Назначение | Тип | Ед. Из. | ОДВ | Вход/выход |
| 1 | newNumerator | Результирующий числитель | int | - | [-2147483648 ,2147483647] | Выходные данные |
| 2 | newDenominator | Результирующий знаменатель | int | - | [-2147483648 ,2147483647] | Выходные данные |

* 1. **Метод ToString**

Метод перезаписывает метод ToString для String. Возвращает строку со значениями числителя и знаменателя.

* 1. **Метод normilize**

Метод нормализует дробь (Приводит к максимально краткому описанию, работает со знаками отрицания).

Таблица спецификаций

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Имя | Назначение | Тип | Ед. Из. | ОДВ | Вход/выход |
| 1 | gcd | Наибольший общий делитель | int | - | [1 ,2147483647] | Выходные данные |

* 1. **Метод findMax(int a, int b)**

Метод реализует алгоритм Евклида, который заключается в поиске наибольшего общего делителя для двух чисел.

Таблица спецификаций

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Имя | Назначение | Тип | Ед. Из. | ОДВ | Вход/выход |
| 1 | a | Число 1 | int | - | [-2147483648 ,2147483647] | Входные и выходные данные |
| 2 | b | Число 2 | int | - | [-2147483648 ,2147483647] | Входные и выходные данные |

* 1. **Метод sravn(Fraction fr2)**

Метод реализует сравнение двух дробей, возвращает знак сравнения.

Таблица спецификаций

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Имя | Назначение | Тип | Ед. Из. | ОДВ | Вход/выход |
| 1 | fr2 | Сравниваемая дробь 2 | Fraction | - | - | Входные данные |
| 2 | Num1 | Числитель 1 дроби | int | - | [-2147483648 ,2147483647] | Входные данные |
| 3 | Num2 | Числитель 2 дроби | int | - | [-2147483648 ,2147483647] | Входные данные |
| 4 | Den1 | Знаменатель 1 дроби | int | - | [-2147483648 ,2147483647] | Входные данные |
| 5 | Den2 | Знаменатель 2 дроби | int | - | [-2147483648 ,2147483647] | Входные данные |

**Таблица тестов**

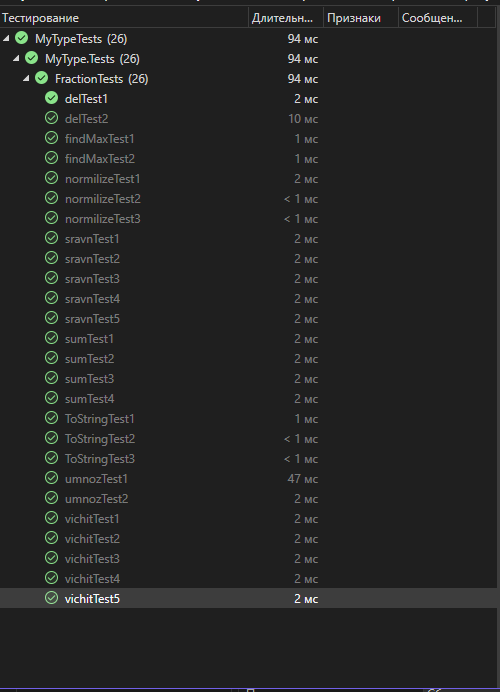
В тестах проверяем корректность выполнения методов класса Fraction

| Номер теста | Что проверяем | Входные данные | Выходные данные |
| --- | --- | --- | --- |
| 1 | Проверка на преобразования типа Дроби в строку 1/10 | fraction(1, 10) | “1/10” |
| 2 | Проверка на преобразования типа Дроби в строку 0/0 -> 0 | fraction(0, 0) | “0” |
| 3 | Проверка на преобразования типа Дроби в строку 8/1 -> 8 | fraction(8, 1) | “8” |
| 4 | Проверка на сокращение дроби 5/10 -> 1/2 | fraction(5, 10) | “1/2” |
| 5 | Проверка на нормализацию дроби 1/-10 - > -1/10 | fraction(1, -10) | “-1/10” |
| 6 | Проверка на нормализацию дроби -1/-10 - > 1/10 | fraction(-1, -10) | “1/10” |
| 7 | Проверка на поиск наибольшего общего делителя для двух чисел алгоритмом Евклида | A = 10  B = 50 | 10 |
| 8 | Проверка на поиск наибольшего общего делителя для двух чисел алгоритмом Евклида | A = 50  B = 10 | 10 |
| 9 | Проверка на сравнение одной дроби + сокращение дроби. Одинаковые знаменатели | fraction1(1, 10)  fraction2(2, 10) | res = "1/10 < 1/5" |
| 10 | Проверка на сравнение одной дроби с другой. Одинаковые знаменатели | fraction1(1, 10)  fraction2(-1, 10) | res = "1/10 > -1/10" |
| 11 | Проверка на сравнение одной дроби с другой. Одинаковые знаменатели | fraction1(-1, 10)  fraction2(-1, 10) | res = "-1/10 = -1/10" |
| 12 | Проверка на сравнение одной дроби с другой + сокращение дроби. Одинаковые знаменатели | fraction1(-1, 10)  fraction2(-5, 10) | res = "-1/10 > -1/2" |
| 13 | Проверка на сравнение одной дроби с другой. Разные знаменатели | fraction1(3, 22)  fraction2(5, 64) | res = "3/22 > 5/64" |
| 14 | Сложение одинаковых дробей | fraction1(1, 4)  fraction2(1, 4) | "3/8" |
| 15 | Сложение дробей с разными знаменателями | fraction1(1, 4)  fraction2(1, 8) | "3/8" |
| 16 | Сложение отрицательной дроби с положительной | fraction1(-1, 4)  fraction2(1, 4) | "0" |
| 17 | Сложение двух отрицательных дробей | fraction1(-1, 4)  fraction2(-1, 4) | "-1/2" |
| 18 | Вычитание одинаковых дробей | fraction1(1, 4)  fraction2(1, 4) | "0" |
| 19 | Вычитание дробей с разными знаменателями | fraction1(1, 4)  fraction2(1, 8) | "1/8" |
| 20 | Вычитание от отрицательной дроби положительную | fraction1(-1, 4)  fraction2(1, 4) | "-1/2" |
| 21 | Вычитание двух отрицательных дробей | fraction1(-1, 4)  fraction2(-1, 4) | "0" |
| 22 | Вычитание от положительной дроби отрицательной | fraction1(1, 4)  fraction2(-1, 4) | "1/2" |
| 23 | Умножение положительной дроби на отрицательную | fraction1(1, 4)  fraction2(-1, 4) | "-1/16" |
| 24 | Умножение отрицательной дроби на отрицательную | fraction1(1, 4)  fraction2(-1, 4) | "1/16" |
| 25 | Деление положительной дроби на отрицательную | fraction1(1, 4)  fraction2(-1, 4) | "-1" |
| 26 | Деление отрицательной дроби на отрицательную | fraction1(-1, 4)  fraction2(-1, 4) | "1" |

**Код программы и автотестов доступен по ссылке:**

[**https://github.com/AshHvostovsky/MyType**](https://github.com/AshHvostovsky/MyType)

**Проверка тестов**

****