

## ***Практическая работа. Абстрактный тип данных простая дробь***

### **Цель**

Сформировать практические навыки реализации абстрактных типов данных в соответствии с заданной спецификацией с помощью классов C++ и их модульного тестирования.

### **Задание**

1. Реализовать абстрактный тип данных «простая дробь», используя класс C++ в соответствии с приведенной ниже спецификацией.
2. Протестировать каждую операцию, определенную на типе данных, используя средства модульного тестирования.
3. Если необходимо, предусмотрите возбуждение исключительных ситуаций.

### **Спецификация типа данных «простые дроби».**

#### **ADT TFrac**

#### **Данные**

Простая дробь (тип TFrac) - это пара целых чисел: числитель и знаменатель (a/b). Простые дроби не изменяемые.

#### **Операции**

Операции могут вызываться только объектом простая дробь (тип **TFrac**), указатель на который в них передаётся по умолчанию. При описании операций этот объект называется «сама дробь» this.

<b><i>Конструктор</i></b>	
Вход:	Пара целых чисел a и b.
Предусловия:	b не равно 0. В противном случае возбуждается исключение.
Процесс:	Инициализирует поля простой дроби this: числитель

	<p>значением а, знаменатель - b. В случае необходимости дробь предварительно сокращается.</p> <p>Например:</p> <p><i>Конструктор</i>(6,3) = (2/1)</p> <p><i>Конструктор</i>(0,3) = (0/3).</p>
Выход:	Нет.
Постусловия:	Поля объекта проинициализированы начальными значениями.
<b><i>Конструктор</i></b>	
Вход:	Строковое представление простой дроби . Например: "7/9".
Предусловия:	b не равно 0. В противном случае возбуждается исключение.
Процесс:	<p>Инициализирует поля простой дроби this строкой f = "a/b". Числитель значением а, знаменатель - b. В случае необходимости дробь предварительно сокращается.</p> <p>Например:</p> <p><i>Конструктор</i>('6/3') = 2/1</p> <p><i>Конструктор</i>('0/3') = 0/3</p>
Выход:	Нет.
Постусловия:	Поля объекта проинициализированы начальными значениями.
<b>Копировать:</b>	
Вход:	Нет.
Предусловия:	Нет.
Процесс:	Создаёт копию самой дроби this с числителем, и

	знаменателем такими же, как у самой дроби.
Выход:	Простая дробь (тип TFrac). Например: $c = 2/1$ , Копировать( $c$ ) = $2/1$
Постусловия:	Нет.
<b>Сложить</b>	
Вход:	Простая дробь d (тип TFrac).
Предусловия:	Нет.
Процесс:	Создаёт и возвращает простую дробь (тип TFrac), полученную сложением самой дроби this = $a1/b1$ с $d = a2/b2$ : $((a1/b1)+(a2/b2)=(a1*b2 + a2*b1)/(b1*b2))$ . Например: $q = 1/2$ , $d = -3/4$ $q.Сложить(d) = -1/4$ .
Выход:	Простая дробь (тип TFrac).
Постусловия:	Нет.
<b>Умножить</b>	
Вход:	Простая дробь d (тип TFrac).
Предусловия:	Нет.
Процесс:	Создаёт простую дробь (тип TFrac), полученную умножением самой дроби this = $a1/b1$ на $d = a2/b2$ $((a1/b1)*(a2/b2)=(a1*a2)/(b1*b2))$ .
Выход:	Простая дробь (тип TFrac).
Постусловия:	Нет.
<b>Вычитать</b>	

Вход:	Простая дробь d (тип TFrac).
Предусловия:	Нет.
Процесс:	Создаёт и возвращает простую дробь (тип TFrac), полученную вычитанием $d = a_2/b_2$ из самой дроби $this = a_1/b_1$ : $((a_1/b_1)-(a_2/b_2)=(a_1 * b_2 - a_2 * b_1)/(b_1 * b_2))$ .  Например: $q = (1/2), d = (1/2)$ $q.Вычесть(d) = (0/1)$ .
Выход:	Простая дробь (тип TFrac).
Постусловия:	Нет
<b>Делить</b>	
Вход:	Простая дробь d (тип TFrac).
Предусловия:	Числитель числа d не равно 0.
Процесс:	Создаёт и возвращает простую дробь (тип TFrac), полученное делением самой дроби $this = a_1/b_1$ на дробь $d = a_2/b_2$ : $((a_1/b_1)/(a_2/b_2)=(a_1 * b_2)/(a_2 * b_1))$ .
Выход:	Простая дробь (тип TFrac).
Постусловия:	Нет.
<b>Квадрат</b>	
Вход:	Нет.
Предусловия:	Нет.
Процесс:	Создаёт и возвращает простую дробь (тип TFrac), полученную умножением самой дроби $this$ на себя: $((a/b)*(a/b)=(a * a)/(b * b))$ .
Выход:	Простая дробь (тип TFrac).
Постусловия:	Нет.

<b>Обратное</b>	
Вход:	Нет.
Предусловия:	Нет.
Процесс:	Создаёт и возвращает простую дробь (тип TFrac), полученное делением единицы на саму дробь this: $1/((a/b) = b/a$ .
Выход:	Простая дробь (тип TFrac)
Постусловия:	Нет.
<b>Минус</b>	
Вход:	Нет.
Предусловия:	Нет.
Процесс:	Создаёт простую дробь, являющуюся разностью простых дробей z и this, где z - простая дробь (0/1.
Выход:	Простая дробь (тип TFrac).
Постусловия:	Нет.
<b>Равно</b>	
Вход:	Простая дробь d (тип TFrac).
Предусловия:	Нет
Процесс:	Сравнивает саму простую дробь this и d. Возвращает значение True, если this и d - тождественные простые дроби, и значение False - в противном случае.
Выход:	Булевское значение.
Постусловия:	Нет.
<b>Больше</b>	
Вход:	Простая дробь d (тип TFrac).
Предусловия:	Нет.

Процесс:	Сравнивает самую простую дробь <code>this</code> и <code>d</code> . Возвращает значение <code>True</code> , если <code>this &gt; d</code> , - значение <code>False</code> - в противном случае.
Выход:	Булевское значение.
Постусловия:	Нет.
<b><i>ВзятьЧислительЧисло</i></b>	
Вход:	
Предусловия:	Нет.
Процесс:	Возвращает значение числителя дроби <code>this</code> в числовом формате.
Выход:	Вещественное значение.
Постусловия:	Нет.
<b><i>ВзятьЗнаменательЧисло</i></b>	
Вход:	Нет.
Предусловия:	Нет.
Процесс:	Возвращает значение знаменателя дроби <code>this</code> в числовом формате.
Выход:	Вещественное значение.
Постусловия:	Нет.
<b><i>ВзятьЧислительСтрока</i></b>	
Вход:	Нет.
Предусловия:	Нет.
Процесс:	Возвращает значение числителя дроби <code>this</code> в строковом формате.
Выход:	Строка.
Постусловия:	Нет.

<b><i>ВзятьЗнаменательСтрока</i></b>	
Вход:	Нет.
Предусловия:	Нет.
Процесс:	Возвращает значение знаменателя дроби this в строковом формате.
Выход:	Строка.
Постусловия:	Нет.
<b><i>ВзятьДробьСтрока</i></b>	
Вход:	Нет.
Предусловия:	Нет.
Процесс:	Возвращает значение простой дроби this, в строковом формате.
Выход:	Строка.
Постусловия:	Нет.

***end TFrac***

## Рекомендации к выполнению

1. Тип данных реализовать, используя класс C++ в составе проекта Консольное приложение.
2. Для возбуждения исключений опишите класс исключительных ситуаций, как наследника библиотечного класса.
3. Для тестирования используйте проект модульного теста.
4. Тип данных реализовать в отдельном модуле UFrac.

## Содержание отчета

1. Задание.
2. Текст программы.

3. Тестовые наборы данных для тестирования типа данных.
4. Результаты тестирования.

### **Контрольные вопросы**

1. Особенности описания методов класса?
2. Особенности описания и назначение конструктора класса?
3. Видимость идентификаторов в описании класса?
4. Особенности вызова методов применительно к объектам класса?
5. Что такое абстрактный тип данных?