**НЕГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ ЧАСТНОЕ** **УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ** **«МОСКОВСКИЙ ФИНАНСОВО-ПРОМЫШЛЕННЫЙ УНИВЕРСИТЕТ** **«СИНЕРГИЯ»**

|  |  |  |
| --- | --- | --- |
| **Факультет/Институт** |  | Информационных технологий |
|  |  | (наименование факультета/ Института) |
| **Направление/специальность** |  | 09.02.07 Информационные системы и программирование |
| **подготовки:** |  | (код и наименование направления /специальности подготовки) |
| **Форма обучения:** |  | очная |
|  |  | (очная, очно-заочная, заочная) |
|  |  |  |

.

**Отчет по лабораторной работе № 5**

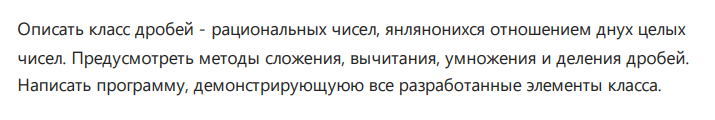
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **на тему** | |  | Использование операторов цикла | |
|  | | |  | (наименование темы) |
|  |  | |  | |
| **по дисциплине** | | | Разработка программных модулей | |
|  | | |  | (наименование дисциплины) |

|  |  |  |  |
| --- | --- | --- | --- |
| **Обучающийся** |  | Касмынин Артем Егорович |  |
|  |  | (ФИО) |  |
| **Группа** |  | ДКИП 111 прог |  |
|  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Преподаватель** |  |  |  |  |
|  |  | (ФИО) |  | (подпись) |

**Москва. 2025**

**Задание**



Код using System;

class Fraction {

public int Numerator { get; private set; }  
public int Denominator { get; private set; }  
  
  
public Fraction(int numerator, int denominator)  
{  
 if (denominator == 0)  
 throw new ArgumentException("Знаменатель не может быть равен нулю.");  
  
   
 int gcd = GCD(Math.Abs(numerator), Math.Abs(denominator));  
 Numerator = numerator / gcd;  
 Denominator = denominator / gcd;  
  
   
 if (Denominator < 0)  
 {  
 Numerator \*= -1;  
 Denominator \*= -1;  
 }  
}  
  
  
private static int GCD(int a, int b)  
{  
 while (b != 0)  
 {  
 int temp = b;  
 b = a % b;  
 a = temp;  
 }  
 return a;  
}  
  
  
public static Fraction operator +(Fraction a, Fraction b)  
{  
 int newNumerator = a.Numerator \* b.Denominator + b.Numerator \* a.Denominator;  
 int newDenominator = a.Denominator \* b.Denominator;  
 return new Fraction(newNumerator, newDenominator);  
}  
  
  
public static Fraction operator -(Fraction a, Fraction b)  
{  
 int newNumerator = a.Numerator \* b.Denominator - b.Numerator \* a.Denominator;  
 int newDenominator = a.Denominator \* b.Denominator;  
 return new Fraction(newNumerator, newDenominator);  
}  
  
  
public static Fraction operator \*(Fraction a, Fraction b)  
{  
 return new Fraction(a.Numerator \* b.Numerator, a.Denominator \* b.Denominator);  
}  
  
  
public static Fraction operator /(Fraction a, Fraction b)  
{  
 if (b.Numerator == 0)  
 throw new DivideByZeroException("Нельзя делить на нулевую дробь.");  
 return new Fraction(a.Numerator \* b.Denominator, a.Denominator \* b.Numerator);  
}  
  
  
public override string ToString()  
{  
 if (Denominator == 1)  
 return Numerator.ToString();  
 return $"{Numerator}/{Denominator}";  
}

}

class Program { static void Main() { Console.WriteLine("Демонстрация работы с дробями:");

Fraction a = new Fraction(3, 4);  
 Fraction b = new Fraction(2, 5);  
 Fraction c = new Fraction(6, 8);   
  
 Console.WriteLine($"Дробь a: {a}");  
 Console.WriteLine($"Дробь b: {b}");  
 Console.WriteLine($"Дробь c: {c} (создана как 6/8, автоматически сокращена)");  
  
   
 Console.WriteLine($"\nСложение: {a} + {b} = {a + b}");  
 Console.WriteLine($"Вычитание: {a} - {b} = {a - b}");  
 Console.WriteLine($"Умножение: {a} \* {b} = {a \* b}");  
 Console.WriteLine($"Деление: {a} / {b} = {a / b}");  
  
   
 Fraction d = new Fraction(1, 2);  
 Fraction e = new Fraction(1, 3);  
 Console.WriteLine($"\nДополнительные примеры:");  
 Console.WriteLine($"{d} + {e} = {d + e}");  
 Console.WriteLine($"{d} - {e} = {d - e}");  
 Console.WriteLine($"{d} \* {e} = {d \* e}");  
 Console.WriteLine($"{d} / {e} = {d / e}");  
  
   
 Fraction f = new Fraction(5, 1);  
 Console.WriteLine($"\nПример с целым числом: {f}");  
}

}

Скрин

