

САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ
ФАКУЛЬТЕТ ИНФОКОММУНИКАЦИОННЫХ ТЕХНОЛОГИЙ

Отчет по лабораторной работе №4
по курсу «Объектно-ориентированное программирование»

Выполнил:

Влазнев Данила

К3223

Проверил:

Иванов С. Е.

Санкт-Петербург

2023 г.

Содержание

Содержание	2
Упражнение 1	3
Упражнение 2	3
Упражнение 3	4
Упражнение 4	6
Упражнение 5	7
Вывод	10

Упражнение 1

Создадим новый класс Utils со статическим методом Greater. Протестируем программу, сравнив два числа (См. Рисунок 1–3).

```
internal class Utils
{
    Ссылка: 1
    public static int Greater(int a, int b)
    {
        if (a > b)
            return a;
        else
            return b;
    }
}
```

Рисунок 1 – Код программы

```
static void Main(string[] args)
{
    int x;
    int y;
    try
    {
        Console.WriteLine("Введите первое число:");
        x = int.Parse(Console.ReadLine());
        Console.WriteLine("Введите второе число:");
        y = int.Parse(Console.ReadLine());
    }
    catch
    {
        Console.WriteLine("Некорректный ввод");
        return;
    }

    int greater = Utils.Greater(x, y);
    Console.WriteLine("Большим из чисел {0} и {1} является {2}", x, y, greater);
}
```

Рисунок 2 – Код программы

```
Введите первое число:
-2
Введите второе число:
5
Большим из чисел -2 и 5 является 5
```

Рисунок 3 – Пример вывода программы

Упражнение 2

Добавим в класс Utils метод Swar, который меняет местами значение переменных и протестируем его (См. рисунок 4–6).

```
public static void Swap(ref int a, ref int b)
{
    int temp = a;
    a = b;
    b = temp;
}
```

Рисунок 4 – Код программы

```
static void Main(string[] args)
{
    int x;
    int y;
    try
    {
        Console.WriteLine("Введите первое число:");
        x = int.Parse(Console.ReadLine());
        Console.WriteLine("Введите второе число:");
        y = int.Parse(Console.ReadLine());
    }
    catch
    {
        Console.WriteLine("Некорректный ввод");
        return;
    }

    Console.WriteLine("До swap: \t" + x + " " + y);
    Utils.Swap(ref x, ref y);
    Console.WriteLine("После swap: \t" + x + " " + y);
}
```

Рисунок 5 – Код программы

```
Введите первое число:
4
Введите второе число:
2
До swap:          4 2
После swap:       2 4
```

Рисунок 6 – Пример вывода программы

Упражнение 3

В классе Utils реализуем метод Factorial, который считает факториал числа и возвращает успешность операции (См. Рисунок 7–10).

```

public static bool Factorial(int n, out int answer)
{
    int k;
    int f = 1;
    bool ok = true;

    try
    {
        checked
        {
            for (k = 2; k <= n; k++)
            {
                f *= k;
            }
        }
    }
    catch
    {
        f = 0;
        ok = false;
    }
    answer = f;
    return ok;
}

```

Рисунок 7 – Код программы

```

static void Main(string[] args)
{
    int f;
    bool ok;
    int x;
    try
    {
        Console.WriteLine("Number for factorial::");
        x = int.Parse(Console.ReadLine());
    }
    catch
    {
        Console.WriteLine("Некорректный ввод");
        return;
    }
    ok = Utils.Factorial(x, out f);
    if (ok) Console.WriteLine($"Factorial {x} = {f}");
    else Console.WriteLine("Cannot compute this factorial");
}

```

Рисунок 8 – Код программы

```

Number for factorial::
12
Factorial 12 = 479001600

```

Рисунок 9 – Пример вывода программы

```

Number for factorial::
13
Cannot compute this factorial

```

Рисунок 10 – Пример вывода программы

Упражнение 4

Необходимо создать класс Operations, который рассчитывает площадь треугольника по 3 или 1 стороне, если он равносторонний (См. Рисунок 11).

```
private static bool CheckTriangleIfExists(double a, double b, double c)
{
    if (a <= 0 || b <= 0 || c <= 0 || a + b <= c || b + c <= a || c + a <= b)
    {
        return false;
    }
    return true;
}
Ссылка: 1
public static double Square(double a, double b, double c)
{
    if (!CheckTriangleIfExists(a, b, c)) return 0;
    double p = (a + b + c) / 2;
    double result = Math.Sqrt(p*(p-a)*(p-b)*(p-c));
    return result;
}
Ссылка: 1
public static double Square(double side)
{
    double result = Math.Pow(side, 2) * Math.Sqrt(3) / 4;
    return result;
}
```

Рисунок 11 – Код программы

Протестируем данный созданный класс (См. Рисунок 12–16).

```
static void Main(string[] args)
{
    Console.WriteLine("Треугольник равносторонний? (Y/N)");
    string command = Console.ReadLine();

    if (command == "Y")
    {
        Console.WriteLine("Введите сторону треугольника");
        double side;
        try
        {
            side = double.Parse(Console.ReadLine());
            if (side <= 0) throw new ArgumentException();
        }
        catch
        {
            Console.WriteLine("Некорректный ввод");
            return;
        }
        double square = Operations.Square(side);
        Console.WriteLine($"Площадь треугольника со стороной {side} = {square}");
    }
}
```

Рисунок 12 – Код программы

```

else if (command == "N")
{
    double a, b, c;
    Console.WriteLine("Введите 3 стороны треугольника (через enter)");
    try
    {
        a = double.Parse(Console.ReadLine());
        b = double.Parse(Console.ReadLine());
        c = double.Parse(Console.ReadLine());
        if (a <= 0 || b <= 0 || c <= 0) throw new ArgumentException();
    }
    catch
    {
        Console.WriteLine("Некорректный ввод");
        return;
    }
    double square = Operations.Square(a, b, c);
    Console.WriteLine($"Площадь треугольника со сторонами {a}, {b}, {c} = {square}");
}
}

```

Рисунок 13 – Код программы

```

else
{
    Console.WriteLine("Некорректный ввод");
    return;
}

```

Рисунок 14 – Код программы

```

Треугольник равносторонний? (Y/N)
Y
Введите сторону треугольника
6
Площадь треугольника со стороной 6 = 15,588457268119894

```

Рисунок 15 – Пример вывода программы

```

Треугольник равносторонний? (Y/N)
N
Введите 3 стороны треугольника (через enter)
3
4
5
Площадь треугольника со сторонами 3, 4, 5 = 6

```

Рисунок 16 – Пример вывода программы

Упражнение 5

Необходимо написать программу, которая будет находить корни квадратных уравнений и протестировать ее (См. Рисунок 17–23).

```

public static int Solve(double a, double b, double c, out double x1, out double x2)
{
    if (a == 0)
    {
        x1 = 0;
        x2 = 0;
        return -1;
    }

    double discriminant = b * b - 4 * a * c;

    if (discriminant < 0)
    {
        x1 = 0;
        x2 = 0;
        return -1;
    }
    else if (discriminant == 0)
    {
        x1 = -b / (2 * a);
        x2 = x1;
        return 0;
    }
}

```

Рисунок 17 – Код программы

```

    else
    {
        x1 = (-b - Math.Sqrt(discriminant)) / (2 * a);
        x2 = (-b + Math.Sqrt(discriminant)) / (2 * a);
        return 1;
    }
}

```

Рисунок 18 – Код программы

```

static void Main(string[] args)
{
    double a, b, c;
    Console.WriteLine("Введите коэффициенты a, b, c (через enter)");
    try
    {
        a = double.Parse(Console.ReadLine());
        b = double.Parse(Console.ReadLine());
        c = double.Parse(Console.ReadLine());
    }
    catch
    {
        Console.WriteLine("Некорректный ввод");
        return;
    }

    double x1, x2;
    int result = EquatationSolver.Solve(a, b, c, out x1, out x2);
    if (result == -1)
    {
        Console.WriteLine($"a = {a}, b = {b}, c = {c}, корней нет");
    }
}

```

Рисунок 19 – Пример вывода программы


```

else if (result == 0)
{
    Console.WriteLine($"a = {a}, b = {b}, c = {c}, x1 = x2 = {x1:N2}");
}
else
{
    Console.WriteLine($"a = {a}, b = {b}, c = {c}, x1 = {x1:N2}, x2 = {x2:N2}");
}

```

Рисунок 20 – Пример вывода программы

```

Введите коэффициенты a, b, c (через enter)
1
2
1
a = 1, b = 2, c = 1, x1 = x2 = -1,00

```

Рисунок 21 – Код программы

```

Введите коэффициенты a, b, c (через enter)
10
1
2
a = 10, b = 1, c = 2, корней нет

```

Рисунок 22 – Пример вывода программы

```

Введите коэффициенты a, b, c (через enter)
8
6
1
a = 8, b = 6, c = 1, x1 = -0,50, x2 = -0,25

```

Рисунок 23 – Пример вывода программы

Вывод

Я выполнил все задания и научился работать с методами класса.