Университет ИТМО

**Лабораторная работа #4**

**Создание и использование методов**

Выполнил: Михайлов

Иван Юрьевич

Группа № К3121

Проверила: Филимонова И. А.

Санкт-Петербург

2020

**Цель работы:**

Рассмотреть способы работы с методами в языке программирования С#.

**Ход работы:**

**Упражнение 1.** Использование параметров в методах, возвращающих значения.

В проект добавлен класс Utils с методом Greater, который принимает по значению 2 параметра типа int и возвращает значение типа int, являющееся большим из двух передаваемых значений.

Внутри метода Main объявлены две целочисленные переменные x и y, значения которым передаются через консоль. Метод Greater вызывается на исполнение от имени класса и возвращённое им значение присваивается новой переменной типа int, которая выводится на консоль.

**Упражнение 2.** Использование в методах параметров, передаваемых по ссылке.

В класс Utils добавлен статический открытый метод Swap, который принимает два параметра a и b, передаваемых по ссылке. Внутри метода Swap описан код, меняющий местами значения a и b. Для этого создаётся дополнительная локальная переменная temp типа int, в которой в процессе перестановки значений a и b, будет временно храниться значение одной из переменных.

В методе Main класса Program вызывается метод Swap, передавая по ссылке значения переменных x и y в качестве параметров. На экран выводится значения переменных х и у до и после перестановки.

**Упражнение 3.** Использование возвращаемых параметров в методах.

В класс Utils добавлен статический открытый метод Factorial, возвращающий значение типа bool, отражающее успешность выполнения метода, который принимает по значению вещественную переменную n, а также имеет второй параметр типа out int, использующийся для возвращения результата. В нём объявлены три переменных: целочисленные f, имеющее начальное значение 1, и k, имеющее начальное значение 2, а также булевого типа ok. В цикле for реализовано вычисление факториала в переменной f, так как значение факториала растет достаточно быстро, цикл помещён в блок checked, реализующий проверку на арифметическое переполнение. Сам блок checked помещён в блок перехвата возможных исключений try, обработка которого в блоке catch заключается в обнулении результата и присвоении переменной ok значения false. Итоговое значение переменной f присваивается возвращаемому параметру answer, а сама функция возвращает значение переменной ok.

В методе Main класса Program объявлена переменная ok типа bool для хранения возвращаемого методом значения (true или false) и переменная f типа int для хранения факториала числа, рассчитанного в методе. В переменную х записывается считанное с консоли число. Вызван метод Factorial, которому передаются переменные x и f в качестве параметров, а возвращаемое методом значение присваивается переменной ok. Если переменная ok принимает значение true, на консоль выводятся значения x и f, в противном случае на экран выводится сообщение об ошибке.

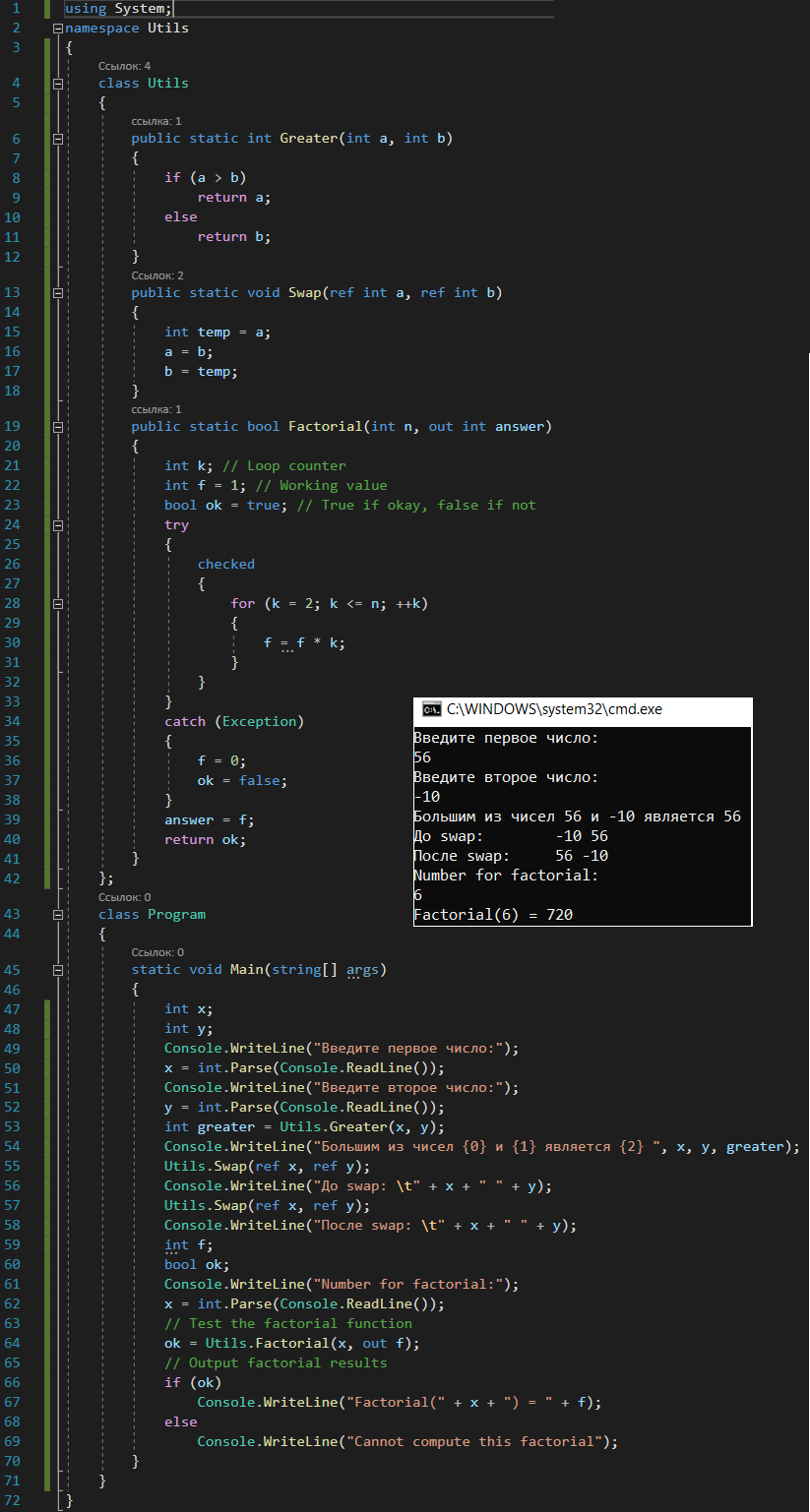


Рисунок 1 – Код упражнений 1, 2, 3

**Упражнение 4.** Расчет площади треугольника с помощью метода.

Создан класс Operation 3 методами:

1. Статический закрытый метод проверки наличия треугольника Exist. Метод принимает 3 параметра (стороны треугольника) типа float и возвращает булевую переменную exist. Если одна из сторон треугольника больше суммы двух других, то exist ложная.
2. Статический открытый метод расчета площади треугольника по формуле Герона Area. Метод принимает 3 параметра (стороны треугольника) типа float и возвращает значение площади типа double. Внутри происходит проверка существования треугольника, путём вызова метода Exist. Если треугольник существует, то возвращается значение площади, если нет, то выводится сообщение об ошибке и возвращается значение 0.
3. Перегруженный открытый статический метод Area. Метод принимает один параметр – сторону равностороннего треугольника и возвращает значение его площади по формуле Герона.

В методе Main() класса Program реализован запрос у пользователя типа треугольника (равносторонний или нет), а далее, с помощью конструкции switch-case, либо значение одной стороны, либо трех. После расчета площади на экран консоли выводится результат вычисления либо сообщение о несуществовании треугольника.

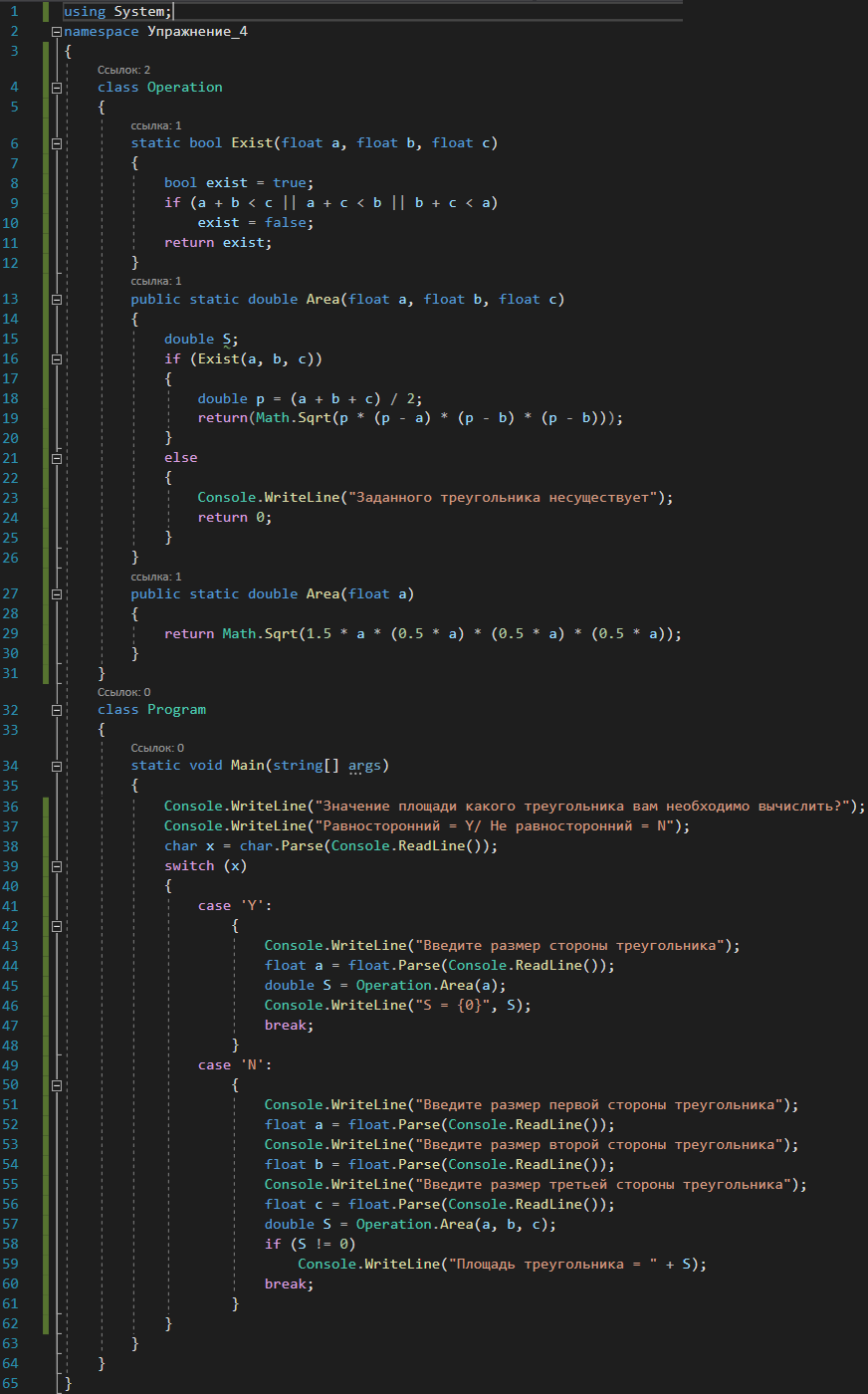


Рисунок 2 – Площадь треугольника

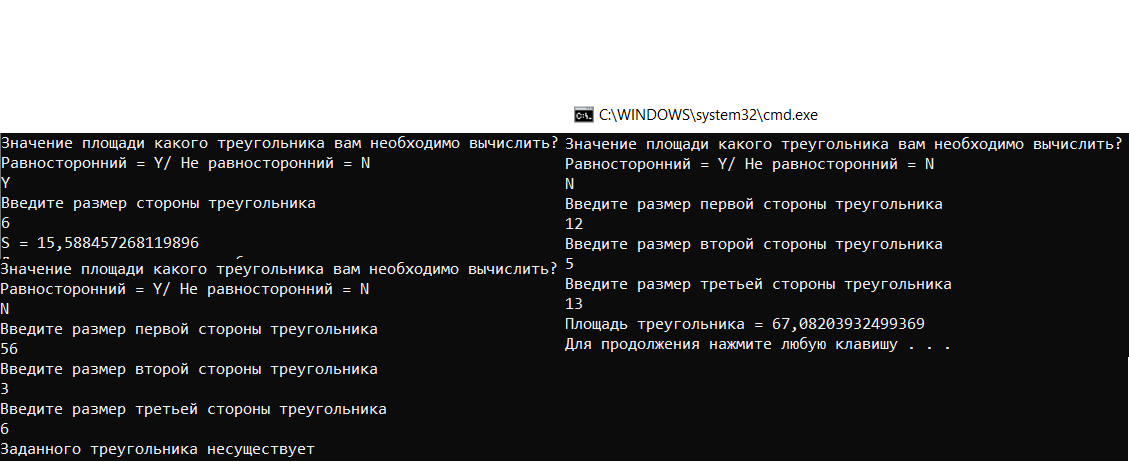


Рисунок 3 – Результат выполнения программы

**Упражнение 5.** Вычисление корней квадратного уравнения.

Добавлен класс Myroot с методом Roots, вычисляющим значение дискриминанта. Метод принимает 3 переменных (коэффициенты уравнения) типа float по значению и 2 корня уравнения типа double по ссылке. Поставлено условие: если дискриминант больше нуля, то вычисляются 2 корня, и функция возвращает значение 1; если равен нулю, то вычисляется единственный корень, и функция возвращает значение 0; если меньше нуля, то функция возвращает значение -1.

В методе Main() класса Program объявлены переменные a, b, c типа float и х1, х2 типа double. В переменную result записывается результат выполнения функции Roots. В конструкции if-else-if реализован вывод 2 корней, если результат выполнения функции 1, одного корня если результат выполнения функции 0, и сообщения «корней нет» если результат выполнения функции -1.

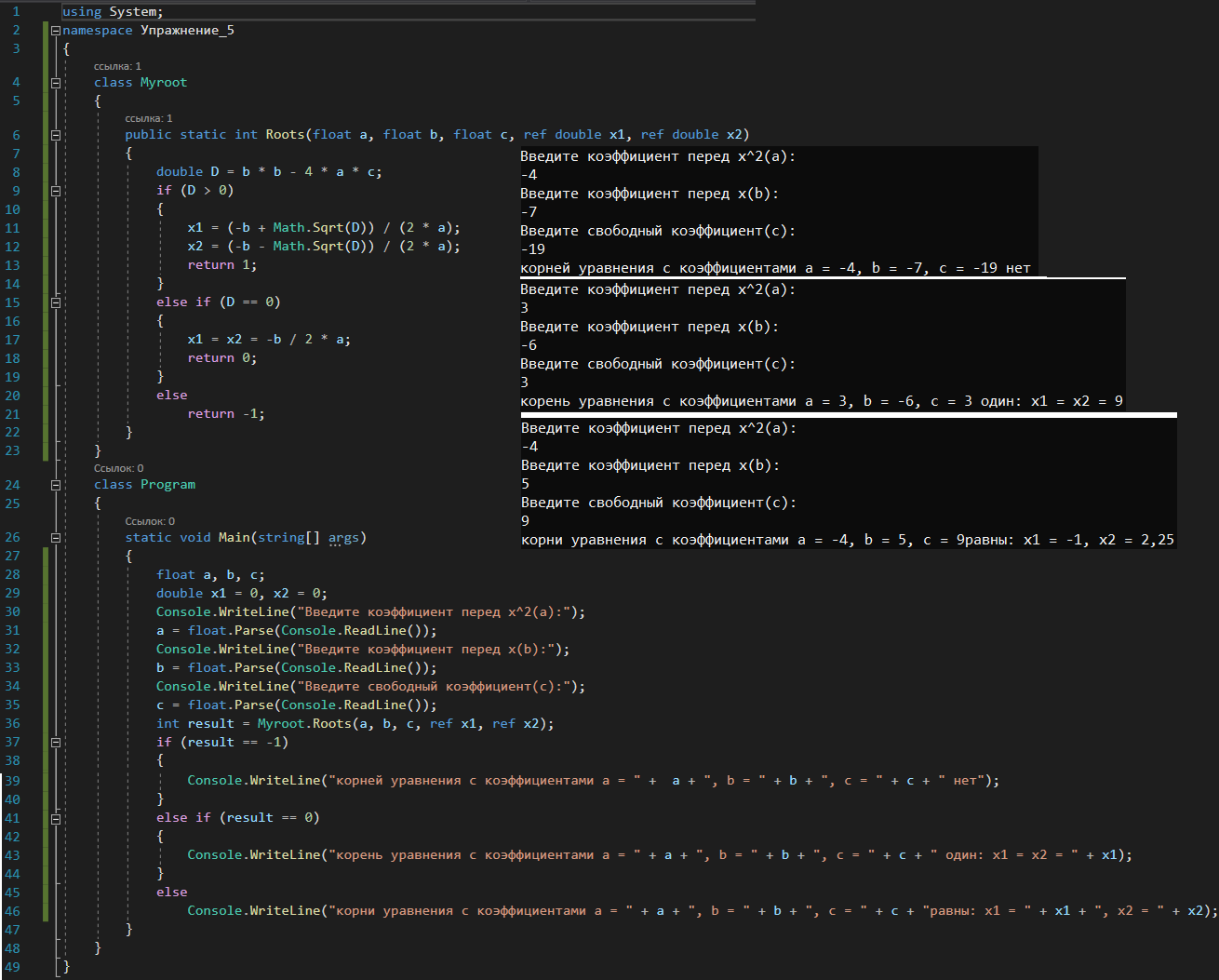


Рисунок 4 – Расчёт корней квадратного уравнения

**Вывод:**

В ходе выполнения лабораторной работы было рассмотрено: способ создания методов в отдельном классе, передача параметров в методы по значению и по ссылке c помощью ключевого слова ref, объявление и вызов методов с возвращаемыми значениями и без (void), использование возвращаемых параметров в методах с помощью ключевого слова out.