

# Лабораторная работа № 4

## Использование методов

### В лабораторной работе используется Microsoft Visual Studio 2008

#### Упражнение 1 – Использование параметров в методах, возвращающих значение

В этом упражнении необходимо определить входные параметры для метода, возвращающего значение. Программа должна читать с консоли два целочисленных числа, сравнивать их, и выводить то из них, которое больше.

1. Откройте Visual Studio 2008.
2. В меню File выберите New, затем выберите Project.
3. Из списка шаблонов выберите Console Application. Впишите название проекта, например, Utils.
4. Укажите путь к проекту C:\Labs\Lab4
5. В классе Utils создайте метод public static Greater с двумя целочисленными параметрами *a* и *b*. Метод должен возвращать целочисленное значение большего из этих 2-х чисел.
6. Создайте класс *Test*. В методе *Main* определите 3 целочисленных переменных: *x*, *y* и *greater*.
7. С помощью методов Console.WriteLine, Console.ReadLine and int.Parse обеспечьте ввод и преобразование исходных данных.
8. Вызовите метод Greater с параметрами *x* и *y* и результат присвойте переменной *greater*.
9. Выведите на консоль полученное число.
10. Выполните компиляцию программы, откорректируйте ошибки и сохраните результат. Покажите результат преподавателю. Код сохраните для отчета.

#### Упражнение 2 – Использование методов со ссылочными параметрами

В этом упражнении необходимо внести изменения в программу, написанную в предыдущем упражнении. Нужно создать метод, который будет изменять значения параметров. Параметры должны передавать по ссылке.

1. Откройте проект, который Вы написали в предыдущем упражнении.
2. В класс Utils добавьте метод public static void Swap с двумя целочисленными параметрами *a* и *b*, которые передаются по ссылке.
3. В теле метода поменяйте значения переменных *a* и *b*, введя вспомогательную переменную *temp*.
4. Отредактируйте метод Main. В классе *Test*. После определения 3 целочисленных переменных: *x*, *y* и *greater* и ввода исходных данных (п.7 упр.1) добавьте вывод исходных значений переменных *x* и *y* на консоль, вызов метода Swap и вывод данных после того, как был произведен обмен значениями переменных *x* и *y*.
5. Выполните компиляцию программы, откорректируйте ошибки и сохраните результат.

### Упражнение 3 – Использование методов с выходными параметрами

В этом упражнении необходимо внести изменения в программу, написанную в предыдущем упражнении. Нужно создать новый метод *Factorial*, который вычисляет факториал. У метода надо определить входной целочисленный параметр.

1. Откройте проект, который Вы написали в предыдущем упражнении.
2. В класс *Utils* добавьте новый *public static* метод и назовите его *Factorial*.
3. Метод должен иметь 2 параметра: входной целочисленный параметр *n* и выходной целочисленный параметр *answer*.
4. Метод *Factorial* должен возвращать значение типа *bool*, если метод выполняется успешно. Если метод вызывает ошибку, то надо запустить исключение.
5. Используйте цикл *for* для вычисления факториала.
6. Выполните компиляцию программы, откорректируйте ошибки и сохраните результат.
7. Протестируйте результат работы метода *Factorial*. Напишите в методе *Main* вызов метода *Factorial*.
8. Выполните компиляцию программы, откорректируйте ошибки и сохраните результат.

### Упражнение 4 – Использование

В этом упражнении необходимо написать альтернативный вариант вычисления факториала с применением рекурсии. Метод назовите *RecursiveFactorial*.