

# Операции над числовыми переменными

**№ урока:** 11 **Курс:** Процедурное программирование на языке C#

**Средства обучения:** Visual Studio 2019 Community Edition

## Обзор, цель и назначение урока

На уроке будут рассмотрены способы проведения арифметических действий над теми числовыми данными, которые мы помещаем в различные переменные.

## Изучив материал данного занятия, учащийся сможет:

- Понимать порядок и ход выполнения арифметических действий в выражениях.
- Уметь выполнять простейшие арифметические операции над числовыми переменными.
- Понимать ход выполнения и выполнять простейшую отладку своего приложения.

## Содержание урока

1. Простейшие операции над числовыми переменными
2. Приведение числовых типов данных
3. Порядок действий в выражениях

## Резюме

- **Математическое выражение** – это одна или несколько величин (переменных или констант), соединенных между собой знаками арифметических действий (+, -, \*, /, ...) и знаками последовательности действий – () – круглыми скобками.
- **Арифметическая операция** - определенное арифметическое действие над числовыми значениями: сложение, вычитание, умножение, деление. В операциях принимают участие два участника (такие операции еще называют - Бинарными арифметическими операциями)
- Участники арифметических операций называются – **операндами**.
- Для того чтобы преобразовать результат выражения к другому типу, следует всё выражение взять в круглые скобки и перед выражением поставить оператор преобразования типа.
- Оператор присвоения (=) сохраняет значение своего правого операнда в месте хранения – переменной, обозначенной в левом операнде. Операнды должны быть одного типа (или правый операнд должен допускать явное преобразование в тип левого операнда).
- Если после знака присвоения идет выражение с вычислением или передачей каких-либо значений, то данная операция выполняется справа налево. Для повышения приоритета операции можно использовать круглые скобки ( ).
- Язык C# предоставляет большой набор операторов, которые представляют собой символы, определяющие операции, которые необходимо выполнить с выражением. К операторам, которые выполняют арифметические операции можно отнести операторы:
  - + (сложения);
  - - (вычитания);
  - \* (умножения);
  - / (деления);
  - % (получения остатка от деления).

- Порядок выполнения арифметических действий, в выражениях на языке C#, совпадает с порядком выполнения арифметических действий в математике.
- Операции умножения, деления, получения остатка от деления имеют больший приоритет, чем сложения и вычитания, поэтому выполняются в первую очередь.
- Все арифметические операции, производимые над двумя значениями типа ([byte](#), [sbyte](#), [short](#), [ushort](#)) в качестве результата, возвращают значение типа [int](#).
- Для типов [int](#), [uint](#), [long](#) и [ulong](#), не происходит преобразования типа результата арифметических операций.
- **CIL** (Common Intermediate Language), или просто IL - **промежуточный язык** Microsoft, по сути является низкоуровневым языком платформы .NET.
- Непонятные числа, появляющиеся в выражениях непонятно откуда, без того, чтобы под используемую величину заранее создавать переменную, программисты называют - **«магическими числами»**. Магическими, потому что программа непонятна для анализа, но при этом работает. И такая программа оказывается понятной, только тому программисту, который ее написал.
- Код, который написан, с минимумом комментариев, и при этом не потерявший своей понятности, называют – **«само-документируемым кодом»**.
- Ошибка в логике программы, из-за которой программа выдает неожиданное поведение и, как следствие, неожиданный результат, называется - **«Баг»**.
- Инструмент поиска багов называется «Дебаггер» или по-русски – **Отладчик**. Анализ программы с использованием отладчика (или Дебаггера) называется – **Отладка** (или Debug).
- **Step Over** (переводится, как - переступить) – это команда, которая пошагово выполняет нашу программу в режиме отладки по одной строке, переступая через строку и при этом выполняет программный код на той строке, которую переступила. В Visual Studio Step Over выполняется нажатием клавиши F10.
- **Точка останова** (англ. breakpoint) — это преднамеренное прерывание выполнения программы, при котором происходит вызов отладчика. После перехода к отладчику программист может исследовать состояние программы (логи, состояние памяти, регистров процессора, стека и т. п.) для того, чтобы определить, правильно ли ведёт себя программа. В отличие от полной остановки, с помощью останова, после работы в отладчике программа может быть завершена либо продолжена с того же места, где произошёл останов.

### Закрепление материала

- Что такое магические числа?
- В какой тип преобразуются результаты арифметических действий с типом byte?
- Что нужно сделать, чтобы получить результат вычисления действия с двумя переменными типа byte в виде новой переменной типа byte?
- Как можно повысить приоритет математической операции в выражении?
- Что такое отладка приложения?
- Что такое точка останова? Как использовать точки останова во время отладки?

### Самостоятельная деятельность учащегося

- Ознакомьтесь с дополнительными материалами к уроку.
- Создайте переменные известных вам числовых типов и присвойте им значения. Затем создайте переменную типа sbyte и поместите в нее результат последовательного сложения созданных ранее переменных. Вычисления выполняйте в одном выражении. Затем создайте переменную типа short и поместите в нее результат последовательного

вычитания созданных ранее переменных в одном выражении. Полученные результаты выведите на экран.

- Откройте созданное вами ранее приложение в программе dotPeek и ознакомьтесь с кодом в декомпилированном виде. Ознакомьтесь с IL кодом созданного вами приложения.

### Рекомендуемые ресурсы

dotPeek — бесплатный инструмент для декомпиляции

<https://www.jetbrains.com/ru-ru/decompiler/>

[https://ru.wikipedia.org/wiki/Common\\_Intermediate\\_Language](https://ru.wikipedia.org/wiki/Common_Intermediate_Language)

Standard ECMA-335 Common Language Infrastructure (CLI)

<https://www.ecma-international.org/publications/standards/Ecma-335.htm>

IDA Pro Disassembler — интерактивный дизассемблер

<https://www.hex-rays.com/products/ida/>

[https://ru.wikipedia.org/wiki/Программная\\_ошибка](https://ru.wikipedia.org/wiki/Программная_ошибка)

Знакомство с отладчиком Visual Studio

<https://docs.microsoft.com/ru-ru/visualstudio/debugger/debugger-feature-tour?view=vs-2019>

Пройдите через код с отладчиком Visual Studio

<https://docs.microsoft.com/ru-ru/visualstudio/debugger/navigating-through-code-with-the-debugger?view=vs-2019>

[https://ru.wikipedia.org/wiki/Точка\\_останова](https://ru.wikipedia.org/wiki/Точка_останова)