

Основы программирования

Синтаксис языка C#

Арифметические операции,
приоритет операций

Группы операторов C#

- Арифметические операторы -- выполняют арифметические операции с числовыми операндами.
- Операторы сравнения -- сравнивают числовые операнды, в том числе проверяют равенство или неравенство операнды.
- Логические операторы -- выполняют логические операции с операндами `bool`.
- Битовые операторы и операторы сдвига -- выполняют битовые операции или операции сдвига с операндами целочисленных типов.
- Операторы присваивания
- и другие...

Выражения

- Выражение -- это правило, после выполнения которого получается результат, принадлежащий определенному типу.

Выражения состоят из операндов и операторов.
Операнды связываются операторами.

В выражении могут также участвовать вызовы функций и другие выражения.

Операторы определяют правило, которое выполняется с операндами.

Примеры выражений

```
int a, b, c;
```

```
a = 7;
```

```
b = a;
```

```
c = b++;
```

```
b = a + b * c;
```

```
c = a >= 100 ? b : c / 10;
```

```
a = (int)Math.Sqrt(b * b + c * c);
```

```
string s = "Строковый литерал";
```

```
char l = s[s.Length - 1];
```

Типы операций

- Унарные операции -- выполняются над одним операндом.
Например: $x++$
- Бинарные операции -- выполняются над двумя операндами.
Например: $x+y$
- Тернарные операции -- выполняются над несколькими операндами.
Например: $c = a \geq 100 ? b : c / 10;$

Запись операций

- с помощью спец символов
- буквенное представление
- знак операции это один или несколько символов
- внутри знака операции пробелы не допускаются, кроме (), [] и ? : .

Арифметические операторы

Поддерживаются всеми целочисленными типами и типами с плавающей запятой.

бинарные: * (умножение)
 / (деление)
 % (остаток от деления)
 + (сложение)
 - (вычитание)

унарные: ++ (приращение)
 -- (уменьшение)
 + (плюс)
 - (минус)

Особенности арифметических операций

```
int x = 10;
```

```
int z = x / 5;           // 2
```

если оба операнда целые числа, то результат также будет округляться до целого числа

```
double a = 10;
```

```
double b = 3;
```

```
double c = a/b;          // 3.33333333
```

```
double z = 10/4;          // 2
```

если оба операнда целые числа, то результат также будет округляться до целого числа

Особенности арифметических операций

(операция целочисленного деления)

```
int x = 10;
```

```
int z = x % 3; //результат равен 1
```

```
double x = 10.0;
```

```
double z = x % 3.0; //результат равен 1
```

Инкремент и декремент

Операторы инкремента (++) и декремента (--) увеличивают и уменьшают значение операнда на единицу, соответственно.

Поддерживается в двух формах:

- постфиксной, например, `x++` и `x--`
- префиксный, например, `++x` и `--x`.

`x = x + 1;` аналогично `x++`; или `++x`;

`x = x - 1;` аналогично `x--`; или `--x`;

Префиксная форма

Если такой оператор применен в префиксной форме, то сначала будет выполнена эта операция и далее будет использоваться новое значение операнда.

```
int x1 = 5;  
int z1 = ++x1;           // z1=6; x1=6
```

```
int x1 = 5;  
int z1 = --x1;           // z1=4; x1=4
```

Постфиксная форма

Если же оператор применен в постфиксной форме, то в выражении будет использовано его старое значение, и только затем будет выполнена операция по изменению значения операнда.

```
int x2 = 5;  
int z2 = x2++;           // z2=5; x2=6
```

```
int x2 = 5;  
int z2 = x2--;           // z2=5; x2=4
```

Приоритет. Пример

```
int a = 3;
```

```
int b = 5;
```

```
int c = 40;
```

```
int d = c---b*a;           // a=3   b=5   c=39   d=25
```

```
int d = (c--) - (b*a);     // a=3   b=5   c=39   d=25
```

```
int d = (c - (--b)) * a;   // a=3   b=4   c=40   d=108
```

Приоритет арифметических операций

от наивысшего к низшему:

1. Инкремент, декремент.
2. Умножение, деление, получение остатка.
3. Сложение, вычитание.

Для изменения порядка выполнения операций применяются скобки.

Ассоциативность

В зависимости от ассоциативности (порядка выполнения операций одинакового приоритета) есть два типа операторов:

- левоассоциативные операторы, исполняемые слева направо;
- правоассоциативные операторы, исполняемые справа налево.

Все арифметические операторы являются левоассоциативными.

Выражение $10/5*2$ необходимо трактовать как $(10/5)*2$, то есть результатом будет 4.