

## Бинарная арифметика

Бинарными называются операторы, которые имеют два аргумента.

В выражении  $-x$  оператор  $-$  унарный.

В выражении  $x - y$  оператор  $-$  бинарный.

Бинарные операции в порядке приоритета

Приоритет	Оператор	Значение
1	+ (унарный)	Реально ничего не изменяет
1	- (унарный)	Возвращает противоположное по знаку, равное по модулю значение
2	++ (унарный)	Оператор инкремента, увеличивает значение аргумента на 1
2	-- (унарный)	Оператор декремента, уменьшает значение аргумента на 1
3	* (бинарный)	Умножение
3	/ (бинарный)	Деление
3	% (бинарный)	Остаток (деление по модулю)
4	+ (бинарный)	Сложение

% (бинарный)	Остаток (деление по модулю)
--------------	-----------------------------

Этот оператор означает *получение остатка от деления*. Например, 5 входит в 14 два раза, и остаток при этом составляет 4. Т.е., 14, деленное по модулю на 5, равно 4.

14 % 5 равно

$14 - (14 / 5) * 5$

$14 - 2 * 5$

$14 - 10$

4

Для переменных действительного типа оператор деления по модулю не предназначен, поскольку он целиком основан на использовании округления до целых ( $14 / 5 = 2$ ).

$$145 \% 10 = 5$$

```
int x = 2;  
int y = 5;  
int c = y % x;  
c = 1
```

```
int x = 2;  
int y = 7;  
int c = y % x;  
c = 1
```

```
double x = 5;  
double y = 7;  
double c = y % x;  
c = error
```

## Выполнение унарных операций

Унарными называются те операторы, которые имеют только один аргумент, например  $-x$ .

```
int x = 5;  
int y = - 5;
```

```
int x = 5;  
x++;
```

Команды инкремента выполняются быстрее, чем команды сложения.

Операторы инкремента и декремента могут находиться либо перед аргументом (**префиксная форма**), либо после него (**постфиксная форма**).

Префиксная форма

```
int x, y;  
x = 5;  
y = ++x;
```

```
x = 6  
y = 6
```

Постфиксная форма

```
int a, b;  
a = 5;  
b = a++;
```

```
a = 6  
b = 5
```

Переменной **y** присваивается уже увеличенное префиксным оператором инкремента значение **x**, а переменной **b** передается еще не увеличенное постфиксным оператором значение **a**.