

# Программирование на языке C++

## § 56. Вычисления

# Типы данных

- `int` // целое
- `long int` // длинное целое
- `float` // вещественное
- `double` // веществ. двойной точности
- `bool` // логические значения
- `char` // СИМВОЛ

# Арифметическое выражения

$$a = (c + b \cdot 5 \cdot 3 - 1) / 2 \cdot d;$$

**Приоритет** (*старшинство*):

- 1) скобки
- 2) умножение и деление
- 3) сложение и вычитание

$$a = \frac{c + b \cdot 5 \cdot 3 - 1}{2} \cdot d$$

# Деление

Результат деления целого на целое – **целое** число (остаток отбрасывается):

```
int a = 3, b = 4;
```

```
float x;
```

```
x = 3 / 4;
```

```
x = 3. / 4;
```

```
x = 3 / 4.;
```

```
x = a / 4;
```

```
x = a / 4.;
```

```
x = a / b;
```

```
x = float(a) / 4;
```

```
x = a / float(b);
```



Что запишется в **x**?

# Остаток от деления

`%` – остаток от деления

```
int a, b, d;  
d = 85;  
b = d / 10;  
a = d % 10;  
d = a % b;  
d = b % a;
```

Для отрицательных чисел:

```
int a = -7;  
b = a / 2;  
d = a % 2;
```



В математике не так!

остаток  $\geq 0$

$$-7 = (-4) * 2 + 1$$

# Сокращенная запись операций

```
int a, b;
```

```
...
```

```
a ++;      // a = a + 1;
```

```
a --;      // a = a - 1;
```

```
a += b;    // a = a + b;
```

```
a -= b;    // a = a - b;
```

```
a *= b;    // a = a * b;
```

```
a /= b;    // a = a / b;
```

```
a %= b;    // a = a % b;
```

# Вещественные числа



Целая и дробная части числа разделяются точкой!

## Форматы вывода:

```
float x = 123.456;
cout.width(10);
cout.precision(5);
cout << x << endl;
```

5 значащих цифр

123.46

```
cout.width(10);
cout.precision(2);
cout << x << endl;
```

всего 10 знаков

1.2e+002

$1,2 \cdot 10^2$

# Вещественные числа

## Экспоненциальный (научный) формат:

```
float x;
x = 1. / 30000;
cout << x;
x = 12345678.;
cout << x;
```

$3,33333 \cdot 10^{-5}$

3.33333e-005

1.23457e+007

```
float x = 123.456;
cout.width(10);
cout.precision(2);
cout << scientific << x;
```

$1,23457 \cdot 10^7$

в дробной части

1.23e+002

научный



# Стандартные функции

```
#include <cmath>
```

подключить  
математическую  
библиотеку

- abs** (x) — модуль целого числа
- fabs** (x) — модуль вещественного числа
- sqrt** (x) — квадратный корень
- sin** (x) — синус угла, заданного **в радианах**
- cos** (x) — косинус угла, заданного **в радианах**
- exp** (x) — экспонента  $e^x$
- ln** (x) — натуральный логарифм
- pow** (x, y) —  $x^y$ : возведение числа  $x$  в степень  $y$
- floor** (x) — округление «вниз»
- ceil** (x) — округление «вверх»

```
float x;  
x = floor(1.6); // 1  
x = ceil(1.6); // 2
```

```
x = floor(-1.6); //-2  
x = ceil(-1.6); // -1
```

# Случайные числа

## Случайно...

- встретить друга на улице
- разбить тарелку
- найти 10 рублей
- выиграть в лотерею

## Случайный выбор:

- жеребьевка на соревнованиях
- выигравшие номера в лотерее

## Как получить случайность?



# Случайные числа на компьютере

## Электронный генератор



- нужно специальное устройство
- нельзя воспроизвести результаты

**Псевдослучайные числа** – обладают свойствами случайных чисел, но каждое следующее число вычисляется по заданной формуле.

## Метод середины квадрата (Дж. фон Нейман)

зерно

564321

в квадрате

• малый период  
(последовательность повторяется через  $10^6$  чисел)

318458191041

209938992481

# Генератор случайных чисел

```
#include <cstdlib>
```

Генератор на отрезке  $[0, \text{RAND\_MAX}]$ :

```
int X, Y;
X = rand(); // псевдослучайное число
Y = rand(); // это уже другое число!
```

англ. *random* – случайный

Целые числа на отрезке  $[a, b]$ :

```
int X, Y;
X = a + rand() % (b - a + 1);
Y = a + rand() % (b - a + 1);
```



Почему так?

$[0, b-a]$

# Задачи

**«С»:** Получить случайное трехзначное число и вывести через запятую его отдельные цифры.

**Пример:**

Получено число 123.

Его цифры 1, 2, 3.