



**Типы данных.
Организация диалога с
пользователем.
Программирование
линейных алгоритмов**



Типы данных в C++

- Арифметические типы:

`int` — целый тип со знаком

min: -2147483648, max: 2147483647

`double` — действительный тип
удвоенной точности

min: 2.225073858507201e-308

max: 1.797693134862316e+308



Библиотека `cmath`

- Включается директивой:

```
#include <cmath>
```

- Основные функции:

`sqrt(x)` — квадратный корень из x

`pow(a, b)` — a^b

`hypot(a, b)` — гипотенуза

`round(x)` — округление по правилам арифметики



Библиотека <cmath>

`trunc(x)` – отбрасывание дробной части


`abs(x)` и `fabs(x)` модуль целого и вещественного числа

`sin(x)`, `cos(x)`, `tan(x)`, `asin(x)`,

`acos(x)`, `atan(x)` –

тригонометрические

`M_PI` – число π (3.14...)



Арифметические операции (по уменьшению приоритета)

- $++$, $--$
- $*$, $/$, $\%$
- $+$, $-$
- $+=$, $-=$
- $*=$, $/=$, $\%=$



Арифметические операции

- Унарные. Инкремент и декремент
форма префиксная постфиксная

инкремент

++a

a++

декремент

--a

a--



Арифметические операции

- Сокращенная форма присваивания

$a = a + 1$ $a += 1$

$a = a * b$ $a *= b$

$a = a / 2$ $a /= 2$

Арифметические операции

- Уменьшение или увеличение на единицу

`a = a + 1` `a += 1` `a++` `++a`

`a = a - 1` `a -= 1` `a--` `--a`



Арифметические операции

- **Приоритет** (precedence) определяет старшинство операции.
- Несколько операций могут иметь равный приоритет. В этом случае включается порядок вычисления - **ассоциативность** (associativity). Ассоциативность может быть либо слева-направо, либо справа-налево.



Арифметические операции

- Приоритет и ассоциативность определяют направление вычислительного процесса. Например:

$2 + 4 * 6 / 3$ // результат 10, а не 12!

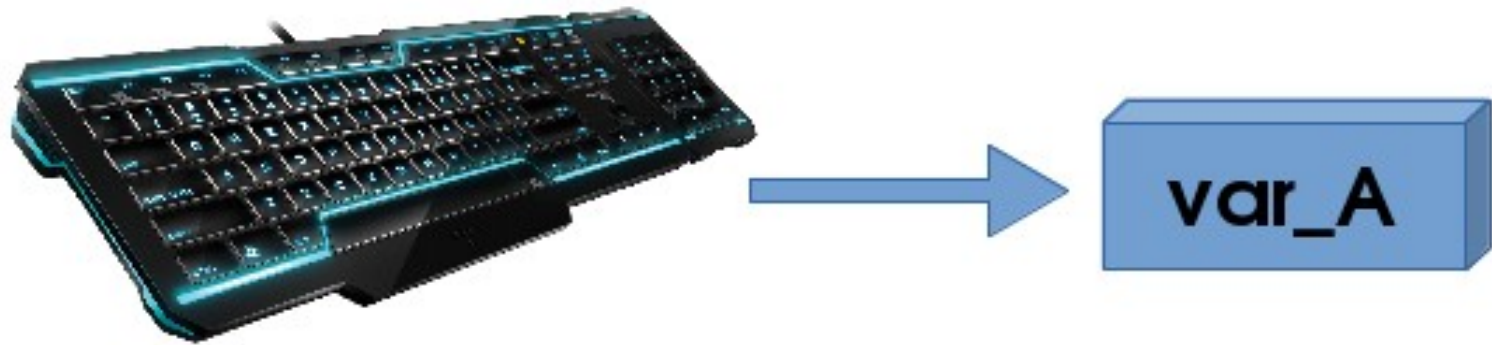
$10 - 5 - 2$ // результат 3, а не 7!



Поток ввода. Поточковый объект cin

- Для ввода данных используется поток ввода. Поток ввода начинается с определения **поточкового объекта ввода – cin** (читается "си-ин").
- Поток ввода использует операцию **извлечения >>** (или **ввода**) для чтения данных в поток со стандартного устройства (клавиатуры)

Поток ввода. Поточковый объект cin



```
cin >> var_A;
```



Поток ввода. Поточковый объект cin

```
int a;  
cin >> a;
```

```
int b, c, d;  
cin >> a >> b >> c;
```



Диалог с пользователем

- Данные не должны вводиться “вслепую”, т. е. для ввода тех или иных данных необходимо приглашение. Такие приглашения называются **диалогами с пользователем** или **дружественными интерфейсами**. Для создания диалогов потоки ввода/вывода используются **совместно**.



Программа «Приветствие»

```
int main() {  
    int Y;  
    string S;  
    cout << "Как тебя зовут?\n";  
    cin >> S;  
    cout << "Сколько тебе лет?\n";  
    cin >> Y;  
    cout << "Здравствуй, " << S  
        << "! Тебе всего " << Y  
        << " лет." << endl;  
    return 0;  
}
```



Диалог с пользователем

```
cout << "a = ";
```


```
cin >> a;
```

```
cout << "b = ";
```

```
cin >> b;
```


```
cout << "a = "; cin >> a;
```

```
cout << "b = "; cin >> b;
```


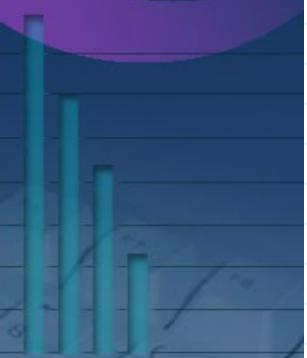



Расчет прямоугольного треугольника

- По двум катетам определить:
гипотенузу, периметр, площадь
- $c^2 = a^2 + b^2$
- $S = \frac{1}{2} ab$
- $P = a + b + c$



```
#include <iostream>
#include <iomanip>
#include <cmath>
using namespace std;
```



```
int main() {
    double a, b;
    cout << "a = "; cin >> a;
    cout << "b = "; cin >> b;
    double c = hypot(a, b);
    double S = 0.5 * a * b;
    double P = a + b + c;
    cout << setprecision(3)
         << fixed
         << "c = " << setw(7) << c << "\n"
         << "S = " << setw(7) << S << "\n"
         << "P = " << setw(7) << P << endl;
    return 0;
}
```