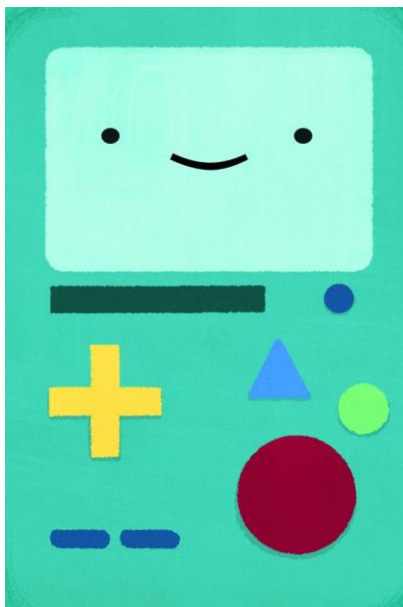




C++ для дітей

Андрій Янковський

2020

C++ (Сі-плюс-плюс) – мова програмування, що була створена ще в 1979 році на базі мови **C**. На відміну від попередника **C++** отримав можливість писати величезні програми (десятки тисяч рядків коду) і при цьому не плутатися в ньому.

Писати код C++ можна в програмах:

Visual Studio, Dev C++, Code::Blocks, Eclipse, CodeLite, NetBeans, Qt Creator



Ось приклад програми:

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

int main()
{
    cout << "Hello, World!";

    return 0;
}
```

Натиснемо кнопку



І на екрані з'явиться рядок

Hello, World!

в чорному вікні. Це вікно і є
консольною програмою.



Але може трапитися так, що після виводу консольного повідомлення програма не побачить інших команд і **закриється**.

Тоді в **Visual Studio** після команди вивести на екран можна вставити команду затримки `system("pause")`, програма припинить роботу лише після натискання на будь-яку клавішу.

В інших середовищах для розробки програм можна спробувати підключити вгорі бібліотеку `conio.h` і визвати функцію `getch()`

```
#include <iostream>
```

```
#include <conio.h>
```

```
using namespace std;
```

```
int main()
```

```
{
```

```
    cout << "Hello, World!";
```

```
    getch();
```

```
    return 0;
```

```
}
```

Ця функція ще може називатися `_getch()`, або можна просто поросити користувача, щось ввести з клавіатури:

```
int e;
```

```
cin >> e;
```

Наступна команда **endl** – вона переміщує курсор на наступний рядок:

```
cout << "123" << endl;
```

А тепер намалюємо **ялинку**:

```
cout << "    *    " << endl;
cout << "   ***   " << endl;
cout << "  *****  " << endl;
cout << "    ||    " << endl;
```

```

    *
  ***
 *****
  ||

```



А без команди **endl** ми б отримали ось такий варіант ялинки:

```
cout << "    *    ";
cout << "   ***   ";
cout << "  *****  ";
cout << "   ||   ";
```

```

*           ***          *****      ||

```

Ще можна коментувати код –
не виконувати ці команди

```
// cout << "    *    ";
```

або так

```
/* cout << "    *    ";
```

```
cout << "   ***   "; */
```

Для того, щоб додати два числа ми маємо їх кудись записати :)

Для зберігання цифр та текстів використовуються змінні:

```
int a = 2;  
int b = 3;  
int c = a + b;  
cout << c;
```



Виведеться: 5

Тут команда `int` перед змінною `a` говорить програмі, що в змінну `a` можна записувати лише цифри

~~`int x = "Car";`~~ неправильно!

`int y = 12;` правильно!

А як нам записати текст в змінну?

```
string a = "Cat";
```

А що буде, якщо додавати не цифри,
а тексти "Cat"+" "+ "Dog"?

```
#include <iostream>
```

```
#include <string>
```

```
using namespace std;
```

```
int main()
```

```
{
```

```
    string a = "Cat";
```

```
    string b = "Dog";
```

```
    string c = a + " " + b;
```

```
    cout << c << endl;
```

```
    system("pause");
```

```
    return 0;
```

```
}
```

Виведеться: Cat Dog



Ага!

Якщо додавати **цифри** - то вони додаються, як в математиці.

А якщо додавати **тексти** - то вони склеюються в один рядок.

І для роботи з текстом нам потрібна спеціальна бібліотека **<string>**, бо без неї ми не зможемо вивести рядок на екран.

А як в змінну записати щось таке:

$$\frac{4 + 4}{2}$$



Дуже просто:

~~int a = 4 + 4/2;~~ **неправильно!** (a = 6)

int a = (4 + 4)/2; **правильно!** (a = 4)

А значення змінної можна потім змінювати на інше?

Звичайно:

int a = 1; зараз a = 1

a = 10; тепер a = 10

a = a + 2; тепер a = 12

Так-так, `a = a + 2` комп'ютер
прекрасно розуміє і збільшує
значення змінної `a` на `2`



Як ввести **число** з клавіатури?

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

Як ввести **текст** з клавіатури?

```
string b;  
cin >> b;
```

Як додати два числа?

```
int a, b, c;  
cin >> a;  
cin >> b;  
c = a + b;  
cout << c;
```

Ого!

Тепер ми можемо додавати (+),
віднімати (-), множити (*) і ділити (/)
будь-які числа, що вводяться з
клавіатури.

А ми можемо перемножити
два тексти?

```
string a = "Cat";  
string b = "Dog";  
string c = a * b;  
cout << c << endl;
```

Ні, так не можна, виникає помилка, бо
комп'ютер не знає як множити текст
на текст.



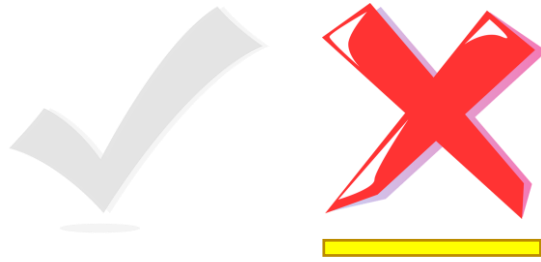
Як перевірити, чи **a** менше **50**?

```
if (a < 10) {  
    cout << "U mene menshe 10 hrn";  
}
```

Якщо умова правильна (**a** < 10), то виведеться:

U mene menshe 10 hrn

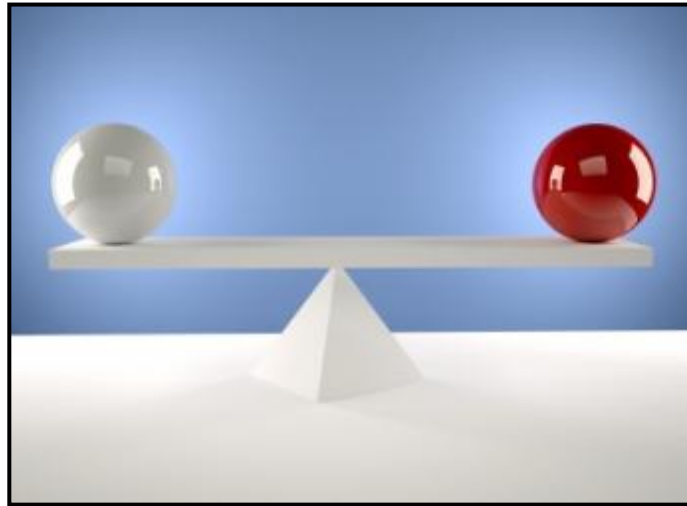
А якщо **a** >= 10 (*більше або дорівнює*)
— **то нічого не виведеться, б-е-е-е**



Тут ми можемо додати ще одну команду (а в іншому випадку)

```
if (a < 10) {  
    cout << "Ya bidnyi";  
}  
else {  
    cout << "Ya bohatyi";  
}
```

Якщо умова правильна ($a < 10$),
то виведеться на екран Ya bidnyi,
а якщо ні - Ya bohatyi



А тепер давайте перевіримо, чи одне число дорівнює іншому

Може так:

~~if(a = 10){ }~~ **неправильно!**

бо **=** це команда для запису в змінну **a** числа **50**

Добре! А якщо так:

if(**a == 50**){ } **правильно!**

бо **==** це команда перевірки, чи в змінній **a** теж знаходиться число **50**

А тепер давайте виведемо на екран комп'ютера 10 раз слово “Hello”

```
for (int i = 0; i < 10; i++){  
    cout << "Hello";  
}
```

Ця команда запускає цикл `for`.

Перший раз цикл запускається зі значенням `i = 0` і виводить “Hello”

На другий раз спрацьовує команда `i++` (збільшити `i` на 1)

і цикл виводить “Hello” (`i = 1`)

і знову виводить “Hello” (`i = 2`)

і знову виводить “Hello” (`i = 3`)

і так поки `i` не стане 10, бо тоді умова (`i < 10`) не спрацює і цикл закінчить свою роботу.

А що буде, якщо **5:2** ?

```
int a = 5/2;
```

Тоді **a** = 2;

Річ у тім, що зміна типу **int** не може зберігати дробову частину числа.

Тип **int** її просто відкидає:

```
int a = 2.9;    // a = 2
```

```
int a = 2.1;    // a = 2
```

Для цифр, які можуть мати дробову частину, є вдвічі більший (подвійний) тип даних:

```
double a = 2.5;    // a = 2.5
```



Генератор випадкових чисел

```
#include <iostream>
#include <time.h>
using namespace std;

int main()
{
    srand(time(0));
    int a = rand() % 100;
    cout << a << endl;
    // 0 – 99
    system("pause");
    return 0;
}
```

А тепер калькулятор!



1. Вводимо два числа **a**, **b**
2. Вводимо **знак** `<string>` (+, -, *, /)
3. Об'являємо змінну **rezultat** = 0
4. Додаємо 4 перевірки **if**
5. І виводимо результат

```
double a, b;  
string znak;  
cout << "Vvedit a: ";  
cin >> a;  
cout << "Vvedit b: ";  
cin >> b;  
cout << "Vvedit znak: ";  
cin >> znak;  
double rezultat = 0;  
  
if (znak == "+") { rezultat = a + b; }  
if (znak == "-" ) { rezultat = a - b; }  
if (znak == "*" ) { rezultat = a * b; }  
if (znak == "/" ) { rezultat = a / b; }  
  
cout << "Rezultat: " << rezultat;
```

Бажаю тобі успіхів!

