

Wprowadzenie do programowania w języku C++

Jeżeli ktoś nie zainstalował środowiska programistycznego Code::Blocks

otwiera stronę <https://www.jdoodle.com/online-compiler-c++/>

Praca w edytorze

Kodowanie polega na pisaniu instrukcji zgodnie z obowiązującymi regułami.

W dalszej kolejności kod jest kompilowany, czyli tłumaczony na język zrozumiały przez komputer, a następnie wykonywany przez komputer.

Wzorzec programu zawiera:

```
#include <iostream> --> odwołanie do biblioteki, służącej do obsługi strumieni we/wy
using namespace std; --> określenie przestrzeni nazw
```

```
int main() {           --> funkcja główna typu całkowitego
```

```
// tu pisze się kod programu
```

```
return 0;              --> polecenie return 0 oznacza zakończenie działania funkcji
main bez błędu.
```

```
}
```

// -> rozpoczyna jednowierszowy komentarz

komentarze wielowierszowe zawarte są między znakami /* i */

Każde polecenie kończymy średnikiem ;

Instrukcja `cout`

Instrukcja `cout` wyświetla to, co napisane jest po symbolu `<<` np. tekst

`cout<<"to jest tekst"<<endl;`

np. wartość zmiennej `a`

`cout<<a<<endl;`

(**`endl`** oznacza przejście do nowego wiersza)

Operatory

Operatory arytmetyczne	
Operator	Działanie
$a * b$	Mnożenie
a / b	Dzielenie
$a + b$	Dodawanie
$a - b$	Odejmowanie
$a \% b$	Reszta z dzielenia (modulo)
$++a$	Preinkrementacja (zwiększenie o 1)
$a++$	Postinkrementacja (zwiększenie o 1)
$--a$	Predekrementacja (zmniejszenie o 1)
$a--$	Postdekrementacja (zmniejszenie o 1)

Operatory logiczne

Operator	Działanie
<code>a && b</code>	true, gdy <code>a = true</code> i <code>b = true</code>
<code>a b</code>	true, gdy <code>a = true</code> lub <code>b = true</code>
<code>!a</code>	true, gdy <code>a = false</code>
<code>a < b</code>	true, gdy <code>a</code> mniejsze od <code>b</code>
<code>a <= b</code>	true, gdy <code>a</code> mniejsze lub równe <code>b</code>
<code>a > b</code>	true, gdy <code>a</code> większe od <code>b</code>
<code>a >= b</code>	true, gdy <code>a</code> większe lub równe <code>b</code>
<code>a == b</code>	true, gdy <code>a</code> równe <code>b</code>
<code>a != b</code>	true, gdy <code>a</code> różne od <code>b</code>

Typ `bool` umożliwia przechowywanie wartości logicznej (prawda lub fałsz).
 Typ `bool` może być używany do deklarowania zmiennych, które przechowują wartość logiczną (1 - prawda, 0 - fałsz).

TYPY DANYCH

Typy arytmetyczne

Parametry typów zależą od kompilatora i platformy. Rozmiar minimalny (gwarantowany) określany w celu zapewnienia przenosiwości. Dla typów rzeczywistych (`float` i `double`) minimum oznacza zbliżenie do zera. Charakterystyka typów arytmetycznych (kompilator g++).

Nazwa	Rozmiar minimalny	Rozmiar	Minimum	Maksimum
char	1 bajt	1 bajt	-128	127
unsigned char	1	1	0	255
short int	2	2	-32768	32767
unsigned short int	2	2	0	65535
int	2	4	-2147483648	2147483647
unsigned int	2	4	0	4294967295
long int	4	4	-2147483648	2147483647
unsigned long int	4	4	0	4294967295
long long	4	8	-9223372036854775808	9223372036854775807
unsigned long long	4	8	0	18446744073709551615
float	4	4	1.17549e-38	3.40282e+38
double	8	8	2.22507e-308	1.79769e+308
long double	12	12	brak danych	brak danych

Typ tekstowy

Standard C++ definiuje pełnowartościowy typ `string`.

Przykład deklarowania, inicjalizowania, dodawania i wyprowadzania

```
#include <string>
using namespace std;
...
string s1 = "Jan", s2 = "Kowalski";
string txt = s1 + " " + s2;
cout << txt;
```