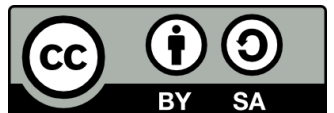


C++

Wprowadzenie

DAMIAN KURPIEWSKI



C++



Język wieloparadygmatowy



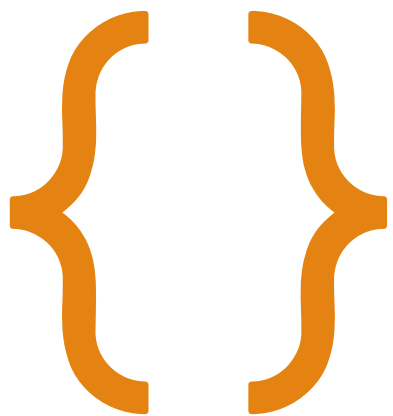
Kompilowany



Statyczna kontrola typów



Mocno ustrukturyzowany kod



C++

KONSTRUKCJE JĘZYKA

Szablon

```
#include <iostream>

using namespace std;

int main() {
    return 0;
}
```

Komentarze



// komentarz jednolinijkowy



```
/*  
komentarz  
wielolinijkowy  
*/
```



/// Komentarz dokumentacyjny

Zmienne



```
int liczba = 34;
```



```
string tekst = "Hello World";
```

0.0

```
double liczbaRzeczywista = 3.4;
```

Operatory arytmetyczne

```
int a = 10;
```

```
int b = 3;
```

```
int suma = a + b;           // 13
```

```
int iloczyn = a * b;       // 30
```

```
int div = a / b;           // 3
```

```
double iloraz = a / (double)b; // 3.3333...
```

```
int modulo = a % b;        // 1
```

Operatory przypisania

```
int a = 10; // 10
```

```
a += 1; // 11
```

```
a -= 1; // 10
```

```
a *= 2; // 20
```

```
a /= 2; // 10
```

```
a %= 4; // 2
```


Operatory porównania

```
int a = 10;
```

```
int b = 5;
```

```
a == b // false
```

```
a != b // true
```

```
a < b // false
```

```
a <= b // false
```

```
a > b // true
```

```
a >= b // true
```

Wejście

```
string tekst;
```

```
cin >> tekst;
```

```
int liczba;
```

```
cin >> liczba;
```

Wyjście

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

```
int a = 10;
```

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

Instrukcja warunkowa

```
int temperatura = 25;

if (temperatura < 10) {
    cout << "Zimno!" << endl;
} else if (temperatura < 20) {
    cout << "Cieplo!" << endl;
} else {
    cout << "Goraco!" << endl;
}
```

Pętla warunkowa while

```
int x = 0;

while (x < 10) {
    cout << "x = " << x << endl;
    x++;
}
```

Pętla licząca for

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

Pętla licząca for
z ujemnym
krokiem

```
for (int i = 10; i > 0; i--) {  
    cout << "i = " << i << endl;  
}
```

Tablice statyczne

```
int tablica1d[5] = {1, 2, 3, 4, 5};
```

```
cout << tablica1d[2] << endl;    // 3
```

```
int tablica2d[2][2] = {{1, 2}, {3, 4}};
```

```
cout << tablica2d[1][1] << endl; // 4
```


Tablice dynamiczne

```
#include <vector>

vector<int> tab;

tab.push_back(10);

cout << tab.size() << endl; // 1

cout << tab[0] << endl; // 10
```

Funkcje

```
int suma(int a, int b) {  
    return a + b;  
}
```

```
cout << suma(2, 5) << endl;
```

Klasy

```
class Prostokat {  
    int wys, szer;  
public:  
    Prostokat(int wysokosc, int szerokosc) {  
        this->wys = wysokosc;  
        this->szer = szerokosc;  
    }  
  
    int pole() {  
        return this->wys * this->szer;  
    }  
  
    bool czyKwadrat() {  
        return this->wys == this->szer;  
    }  
};
```

```
Prostokat prostokat = Prostokat(5, 10);  
cout << prostokat.pole() << endl;  
cout << prostokat.czyKwadrat() << endl;
```