

C++ - WSTĘP I ĆWICZENIA

Z komentarzem do każdej linii kodu

Materiał dydaktyczny na podstawie plików „Cpp1U-Wstęp” oraz „Cpp1U-Wstęp-Cwiczenia”

Dokument zawiera podstawowe zagadnienia z C++ oraz komplet 24 ćwiczeń. Każda istotna linia kodu została opisana komentarzem // tak, aby uczeń mógł analizować program krok po kroku.

1. Zmienne w C++

Co to jest zmienna?

Zmienna to miejsce w pamięci komputera, w którym przechowujemy dane. Każda zmienna ma typ, nazwę i może mieć wartość początkową.

Najczęstsze typy danych

int - liczby całkowite; double - liczby z częścią ułamkową; char - pojedynczy znak; bool - wartość logiczna true/false; string - tekst.

2. Operatory w C++

Operatory arytmetyczne

+ dodawanie, - odejmowanie, * mnożenie, / dzielenie, % reszta z dzielenia.

Operatory porównania

== równe, != różne, > większe, < mniejsze, >= większe lub równe, <= mniejsze lub równe.

Operatory logiczne

&& oznacza logiczne „i”, || oznacza logiczne „lub”, ! oznacza negację.

3. Inkrementacja i dekrementacja

Znaczenie

x++ i x-- zmieniają wartość po użyciu jej w wyrażeniu, a ++x i --x zmieniają ją przed użyciem.

4. Instrukcja warunkowa if / else

Znaczenie

Instrukcja warunkowa pozwala wykonywać fragment kodu tylko wtedy, gdy podany warunek jest prawdziwy.

Przykład: deklaracja i inicjalizacja zmiennych

```
int wiek = 18;      // Tworzymy zmienną całkowitą wiek i zapisujemy w niej wartość 18.
double cena = 9.99; // Tworzymy zmienną typu double i przypisujemy wartość 9.99.
string imie = "Adam"; // Tworzymy zmienną tekstową imie i zapisujemy napis "Adam".
```

Przykład: wczytywanie danych

```
int liczba;           // Tworzymy zmienną całkowitą liczba.  
cin >> liczba;       // Pobieramy liczbę z klawiatury i zapisujemy ją w zmiennej liczba.  
cout << "Podales: " << liczba; // Wyświetlamy tekst oraz wartość pobraną od użytkownika.
```

Przykład: dzielenie całkowite

```
cout << 7 / 2;       // Obie liczby są typu int, więc wynik dzielenia całkowitego to 3, a nie 3.5.
```

Przykład: modulo

```
int a = 10;          // Zmienna a ma wartość 10.  
int b = 3;           // Zmienna b ma wartość 3.  
cout << a % b;       // % zwraca resztę z dzielenia 10 przez 3, czyli 1.
```

Przykład: warunek if

```
int n = 5;           // Tworzymy zmienną n o wartości 5.  
if (n > 0) {         // Sprawdzamy, czy n jest większe od zera.  
    cout << "Dodatnia"; // Ta linia wykona się tylko wtedy, gdy warunek jest prawdziwy.  
}                   // Koniec bloku instrukcji if.
```

Ćwiczenia 1-24 z pełnym komentarzem do kodu

Zadanie 1. Dodawanie liczb

Treść: Napisz program, który dodaje liczby 5 i 3 oraz wypisuje ich sumę.

```
#include <iostream>           // Dołączamy bibliotekę obsługującą wejście i wyjście.
using namespace std;         // Pozwala używać cout i cin bez przedrostka std::

int main() {                 // Rozpoczynamy główną funkcję programu.
    int a = 5;               // Tworzymy zmienną a i przypisujemy jej wartość 5.
    int b = 3;               // Tworzymy zmienną b i przypisujemy jej wartość 3.
    cout << a + b << endl;   // Dodajemy a i b, wypisujemy wynik i przechodzimy do nowej linii.
    return 0;                // Kończymy program informacją o poprawnym wykonaniu.
}                             // Koniec funkcji main.
```

Zadanie 2. Wczytywanie liczby

Treść: Wczytaj liczbę całkowitą i wypisz ją w formacie: Podales: LICZBA.

```
#include <iostream>           // Dołączamy bibliotekę potrzebną do cin i cout.
using namespace std;         // Używamy standardowej przestrzeni nazw.

int main() {                 // Początek programu.
    int liczba;              // Tworzymy zmienną całkowitą liczba.
    cin >> liczba;           // Pobieramy liczbę wpisaną przez użytkownika.
    cout << "Podales: " << liczba << endl; // Wyświetlamy tekst oraz pobraną liczbę.
    return 0;                // Kończymy program poprawnie.
}                             // Koniec funkcji main.
```

Zadanie 3. Różne typy danych

Treść: Zadeklaruj zmienne typu double, bool i char, a następnie wypisz ich wartości.

```
#include <iostream>           // Dołączamy bibliotekę wejścia i wyjścia.
using namespace std;         // Używamy przestrzeni nazw std.

int main() {                 // Początek programu.
    double wzrost = 1.75;    // Zmienna double przechowuje liczbę z częścią ułamkową.
    bool pelnoletni = true;  // Zmienna bool przechowuje wartość logiczną true.
    char litera = 'K';       // Zmienna char przechowuje pojedynczy znak K.
    cout << wzrost << endl;   // Wypisujemy wartość zmiennej wzrost.
    cout << pelnoletni << endl; // Wypisujemy true jako 1.
    cout << litera << endl;   // Wypisujemy znak K.
    return 0;                // Kończymy program poprawnie.
}                             // Koniec funkcji main.
```

Zadanie 4. Reszta z dzielenia

Treść: Oblicz resztę z dzielenia 10 przez 3.

```
#include <iostream>           // Biblioteka do obsługi cout.
using namespace std;         // Używamy przestrzeni nazw std.

int main() {                 // Początek programu.
    int a = 10;              // Pierwsza liczba ma wartość 10.
```

```
int b = 3;           // Druga liczba ma wartość 3.
cout << a % b << endl; // Operator % oblicza resztę z dzielenia, czyli 1.
return 0;           // Kończymy program poprawnie.
}                   // Koniec funkcji main.
```

Zadanie 5. Dzielenie dokładne

Treść: Wykonaj dzielenie 7.0 przez 2 i wypisz wynik.

```
#include <iostream>       // Biblioteka do wypisywania danych.
using namespace std;      // Używamy przestrzeni nazw std.

int main() {              // Początek programu.
    double x = 7.0 / 2;    // 7.0 wymusza dzielenie zmiennoprzecinkowe, więc wynik to 3.5.
    cout << x << endl;    // Wyświetlamy wartość 3.5.
    return 0;             // Kończymy program poprawnie.
}                          // Koniec funkcji main.
```

Zadanie 6. Mnożenie i odejmowanie

Treść: Pomnóż 4 i 5, a następnie odejmij 2.

```
#include <iostream>       // Biblioteka do obsługi cout.
using namespace std;      // Używamy przestrzeni nazw std.

int main() {              // Początek programu.
    int a = 4;            // Pierwsza liczba ma wartość 4.
    int b = 5;            // Druga liczba ma wartość 5.
    cout << a * b - 2 << endl; // Najpierw 4*5=20, potem 20-2=18.
    return 0;             // Kończymy program poprawnie.
}                          // Koniec funkcji main.
```

Zadanie 7. Porównanie równości

Treść: Sprawdź, czy 5 jest równe 5 i wypisz 1 lub 0.

```
#include <iostream>       // Biblioteka do wypisywania danych.
using namespace std;      // Używamy przestrzeni nazw std.

int main() {              // Początek programu.
    int a = 5;            // Tworzymy zmienną a o wartości 5.
    cout << (a == 5) << endl; // == sprawdza równość; prawda zostanie wypisana jako 1.
    return 0;             // Kończymy program poprawnie.
}                          // Koniec funkcji main.
```

Zadanie 8. Porównanie większe od

Treść: Sprawdź, czy 3 jest większe od 7.

```
#include <iostream>       // Biblioteka do cout.
using namespace std;      // Używamy przestrzeni nazw std.

int main() {              // Początek programu.
    int a = 3;            // Zmienna a ma wartość 3.
    int b = 7;            // Zmienna b ma wartość 7.
    cout << (a > b) << endl; // Sprawdzamy 3 > 7; fałsz zostanie wypisany jako 0.
    return 0;             // Kończymy program poprawnie.
}
```

```
} // Koniec funkcji main.
```

Zadanie 9. Operator różne od

Treść: Sprawdź, czy 10 jest różne od 5.

```
#include <iostream> // Biblioteka do cout.
using namespace std; // Używamy przestrzeni nazw std.

int main() { // Początek programu.
    cout << (10 != 5) << endl; // != sprawdza różność; 10 jest różne od 5, więc wynik to 1.
    return 0; // Kończymy program poprawnie.
} // Koniec funkcji main.
```

Zadanie 10. Operator logiczny &&

Treść: Sprawdź, czy 5 jest większe od 0 i jednocześnie mniejsze od 10.

```
#include <iostream> // Biblioteka do cout.
using namespace std; // Używamy przestrzeni nazw std.

int main() { // Początek programu.
    int a = 5; // Zmienna a ma wartość 5.
    cout << (a > 0 && a < 10) << endl; // && wymaga, aby oba warunki były prawdziwe; wynik to 1.
    return 0; // Kończymy program poprawnie.
} // Koniec funkcji main.
```

Zadanie 11. Negacja bool

Treść: Utwórz bool o wartości true i wypisz jego negację.

```
#include <iostream> // Biblioteka do cout.
using namespace std; // Używamy przestrzeni nazw std.

int main() { // Początek programu.
    bool x = true; // Zmienna logiczna x otrzymuje wartość true.
    cout << !x << endl; // ! odwraca wartość logiczną, więc true zmienia się na false, czyli 0.
    return 0; // Kończymy program poprawnie.
} // Koniec funkcji main.
```

Zadanie 12. Instrukcja if - liczba dodatnia

Treść: Sprawdź, czy liczba 5 jest dodatnia.

```
#include <iostream> // Biblioteka do cout.
using namespace std; // Używamy przestrzeni nazw std.

int main() { // Początek programu.
    int n = 5; // Zmienna n ma wartość 5.
    if (n > 0) { // Sprawdzamy, czy n jest większe od zera.
        cout << "Dodatnia" << endl; // Jeśli warunek jest prawdziwy, wypisujemy komunikat.
    } // Koniec bloku if.
    return 0; // Kończymy program poprawnie.
} // Koniec funkcji main.
```

Zadanie 13. Parzysta czy nieparzysta

Treść: Wczytaj liczbę całkowitą i sprawdź jej parzystość.

```
#include <iostream>           // Biblioteka do cin i cout.
using namespace std;         // Używamy przestrzeni nazw std.

int main() {                 // Początek programu.
    int n;                   // Tworzymy zmienną całkowitą n.
    cin >> n;                // Pobieramy liczbę od użytkownika.
    if (n % 2 == 0) {         // Reszta 0 z dzielenia przez 2 oznacza liczbę parzystą.
        cout << "Parzysta" << endl;    // Wypisujemy komunikat dla liczby parzystej.
    } else {                 // Ten blok wykona się, gdy warunek if jest fałszywy.
        cout << "Nieparzysta" << endl; // Wypisujemy komunikat dla liczby nieparzystej.
    }                       // Koniec if/else.
    return 0;                // Kończymy program poprawnie.
}                             // Koniec funkcji main.
```

Zadanie 14. Dodatnia, ujemna czy zero

Treść: Wczytaj liczbę i sklasyfikuj ją względem zera.

```
#include <iostream>           // Biblioteka do cin i cout.
using namespace std;         // Używamy przestrzeni nazw std.

int main() {                 // Początek programu.
    int n;                   // Tworzymy zmienną całkowitą n.
    cin >> n;                // Pobieramy liczbę od użytkownika.
    if (n > 0) {             // Sprawdzamy, czy n jest dodatnie.
        cout << "Dodatnia" << endl;    // Wypisujemy komunikat dla liczby dodatniej.
    } else if (n < 0) {      // Jeśli nie, sprawdzamy, czy n jest ujemne.
        cout << "Ujemna" << endl;    // Wypisujemy komunikat dla liczby ujemnej.
    } else {                // Jeśli oba poprzednie warunki są fałszywe...
        cout << "Zero" << endl;    // ...liczba musi być równa zero.
    }                       // Koniec całej instrukcji warunkowej.
    return 0;                // Kończymy program poprawnie.
}                             // Koniec funkcji main.
```

Zadanie 15. Różnica i iloraz

Treść: Wczytaj dwie liczby całkowite. Wypisz różnicę oraz iloraz tylko wtedy, gdy druga liczba nie jest zerem.

```
#include <iostream>           // Biblioteka do cin i cout.
using namespace std;         // Używamy przestrzeni nazw std.

int main() {                 // Początek programu.
    int a, b;                // Tworzymy dwie zmienne całkowite.
    cin >> a >> b;           // Pobieramy od użytkownika dwie liczby.
    cout << "Roznica: " << (a - b) << "\n"; // Obliczamy i wypisujemy różnicę a-b.
    if (b != 0) {            // Sprawdzamy, czy druga liczba nie jest zerem.
        cout << "Iloraz: " << (a / b) << "\n"; // Dzielimy całkowicie a przez b i wypisujemy wynik.
    } else {                 // Ten blok wykona się, gdy b jest równe zero.
        cout << "Nie mozna dzielic przez 0\n"; // Informujemy o niedozwolonym dzieleniu przez zero.
    }                       // Koniec if/else.
    return 0;                // Kończymy program poprawnie.
}                             // Koniec funkcji main.
```

Zadanie 16. Inkrementacja

Treść: Wczytaj liczbę całkowitą, zwiększ ją o 1 operatorem inkrementacji i wypisz wynik.

```
#include <iostream>          // Biblioteka do cin i cout.
using namespace std;         // Używamy przestrzeni nazw std.

int main() {                 // Początek programu.
    int n;                   // Tworzymy zmienną całkowitą n.
    cin >> n;                // Pobieramy wartość n od użytkownika.
    n++;                     // Zwiększamy n o 1 operatorem inkrementacji.
    cout << n << "\n";      // Wypisujemy nową wartość n.
    return 0;                // Kończymy program poprawnie.
}                             // Koniec funkcji main.
```

Zadanie 17. DUZA / MALA

Treść: Wczytaj liczbę. Jeśli jest większa od 100, wypisz DUZA, w przeciwnym razie MALA.

```
#include <iostream>          // Biblioteka do cin i cout.
using namespace std;         // Używamy przestrzeni nazw std.

int main() {                 // Początek programu.
    int n;                   // Tworzymy zmienną całkowitą n.
    cin >> n;                // Pobieramy liczbę od użytkownika.
    if (n > 100) {           // Sprawdzamy, czy liczba jest większa od 100.
        cout << "DUZA\n";    // Jeśli tak, wypisujemy DUZA.
    } else {                 // Jeśli warunek jest fałszywy...
        cout << "MALA\n";    // ...wypisujemy MALA.
    }                       // Koniec instrukcji warunkowej.
    return 0;                // Kończymy program poprawnie.
}                             // Koniec funkcji main.
```

Zadanie 18. Czy liczby są równe?

Treść: Wczytaj dwie liczby całkowite i wypisz wynik porównania.

```
#include <iostream>          // Biblioteka do cin i cout.
using namespace std;         // Używamy przestrzeni nazw std.

int main() {                 // Początek programu.
    int a, b;                // Tworzymy dwie zmienne całkowite.
    cin >> a >> b;           // Pobieramy obie liczby.
    cout << (a == b) << "\n"; // Sprawdzamy równość; wynik będzie 1 dla prawdy lub 0 dla fałszu.
    return 0;                // Kończymy program poprawnie.
}                             // Koniec funkcji main.
```

Zadanie 19. Podzielność przez 3

Treść: Wczytaj liczbę całkowitą i sprawdź, czy jest podzielna przez 3.

```
#include <iostream>          // Biblioteka do cin i cout.
using namespace std;         // Używamy przestrzeni nazw std.

int main() {                 // Początek programu.
    int n;                   // Tworzymy zmienną całkowitą n.
```

```

cin >> n;           // Pobieramy liczbę od użytkownika.
if (n % 3 == 0) {    // Sprawdzamy, czy reszta z dzielenia przez 3 wynosi 0.
    cout << "TAK\n"; // Jeśli tak, liczba jest podzielna przez 3.
} else {            // Jeśli reszta jest różna od 0...
    cout << "NIE\n";  // ...liczba nie jest podzielna przez 3.
}                  // Koniec if/else.
return 0;           // Kończymy program poprawnie.
}                  // Koniec funkcji main.

```

Zadanie 20. Dodatnia i parzysta

Treść: Jeśli liczba jest dodatnia i parzysta, wypisz OK, w przeciwnym razie NIE.

```

#include <iostream>    // Biblioteka do cin i cout.
using namespace std;  // Używamy przestrzeni nazw std.

int main() {          // Początek programu.
    int n;             // Tworzymy zmienną całkowitą n.
    cin >> n;          // Pobieramy liczbę od użytkownika.
    if (n > 0 && n % 2 == 0) { // Sprawdzamy dwa warunki jednocześnie: dodatniość i parzystość.
        cout << "OK\n";    // Oba warunki są prawdziwe, więc wypisujemy OK.
    } else {           // Gdy przynajmniej jeden warunek jest fałszywy...
        cout << "NIE\n";   // ...wypisujemy NIE.
    }                 // Koniec if/else.
    return 0;         // Kończymy program poprawnie.
}                    // Koniec funkcji main.

```

Zadanie 21. Największa z trzech liczb

Treść: Wczytaj trzy liczby całkowite i wypisz największą.

```

#include <iostream>    // Biblioteka do cin i cout.
using namespace std;  // Używamy przestrzeni nazw std.

int main() {          // Początek programu.
    int a, b, c;       // Tworzymy trzy zmienne całkowite.
    cin >> a >> b >> c; // Pobieramy trzy liczby od użytkownika.
    int maxL = a;      // Na początku zakładamy, że największa jest liczba a.
    if (b > maxL) maxL = b; // Jeśli b jest większe od aktualnego maksimum, zapisujemy b jako maksimum.
    if (c > maxL) maxL = c; // Jeśli c jest większe od aktualnego maksimum, zapisujemy c jako maksimum.
    cout << maxL << "\n"; // Wypisujemy największą znaną wartość.
    return 0;          // Kończymy program poprawnie.
}                    // Koniec funkcji main.

```

Zadanie 22. Średnia dwóch liczb

Treść: Wczytaj dwie liczby całkowite i oblicz dokładną średnią typu double.

```

#include <iostream>    // Biblioteka do cin i cout.
using namespace std;  // Używamy przestrzeni nazw std.

int main() {          // Początek programu.
    int a, b;          // Tworzymy dwie zmienne całkowite.
    cin >> a >> b;     // Pobieramy obie liczby.
    double srednia = (a + b) / 2.0; // Dzielenie przez 2.0 daje wynik zmiennoprzecinkowy.
    cout << srednia << "\n"; // Wypisujemy dokładną średnią.
}

```



```

return 0;           // Kończymy program poprawnie.
}                  // Koniec funkcji main.

```

Zadanie 23. Dekrementacja

Treść: Ustaw $x=10$ i wypisz kolejno $x--$, aktualne x , $--x$, aktualne x .

```

#include <iostream>      // Biblioteka do cout.
using namespace std;    // Używamy przestrzeni nazw std.

int main() {            // Początek programu.
    int x = 10;          // Tworzymy x i nadajemy mu wartość 10.
    cout << x-- << "\n"; // Najpierw wypisujemy 10, a potem x zostaje zmniejszone do 9.
    cout << x << "\n";   // Wypisujemy aktualną wartość x, czyli 9.
    cout << --x << "\n"; // Najpierw zmniejszamy x do 8, a potem wypisujemy 8.
    cout << x << "\n";   // Wypisujemy aktualną wartość x, czyli 8.
    return 0;           // Kończymy program poprawnie.
}                        // Koniec funkcji main.

```

Zadanie 24. Warunek wielopoziomowy

Treść: Jeśli liczba jest dodatnia i >50 wypisz A; jeśli dodatnia, ale ≤ 50 wypisz B; w przeciwnym razie C.

```

#include <iostream>      // Biblioteka do cin i cout.
using namespace std;    // Używamy przestrzeni nazw std.

int main() {            // Początek programu.
    int n;               // Tworzymy zmienną całkowitą n.
    cin >> n;            // Pobieramy liczbę od użytkownika.
    if (n > 0 && n > 50) { // Sprawdzamy, czy liczba jest dodatnia i większa od 50.
        cout << "A\n";    // Dla takiej liczby wypisujemy A.
    } else if (n > 0) {    // Jeśli pierwszy warunek nie zadziałał, sprawdzamy tylko dodatniość.
        cout << "B\n";    // Liczba jest dodatnia, ale nie większa niż 50, więc wypisujemy B.
    } else {              // Pozostały wszystkie liczby niedodatnie.
        cout << "C\n";    // Wypisujemy C.
    }                    // Koniec całej instrukcji if / else if / else.
    return 0;            // Kończymy program poprawnie.
}                        // Koniec funkcji main.

```

Najważniejsze rzeczy do zapamiętania

- Operator $=$ przypisuje wartość, natomiast $==$ porównuje dwie wartości.
- Dzielenie dwóch wartości typu `int` daje wynik całkowity. Aby uzyskać część ułamkową, przynajmniej jeden operand powinien być typu `double`, np. $7.0 / 2$.
- Operator $\%$ zwraca resztę z dzielenia i jest szczególnie przydatny przy sprawdzaniu parzystości i podzielności.
- Operator $\&\&$ oznacza, że oba warunki muszą być prawdziwe.
- $x++$ używa najpierw starej wartości, a potem zwiększa zmienną; $++x$ najpierw zwiększa zmienną, a potem używa nowej wartości.
- Przed dzieleniem przez zmienną należy sprawdzić, czy jej wartość nie jest równa 0.