

Instrukcja warunkowa **if**

Składnia

```
if (warunek)
{
    blok_instrukcji
}
```

- Jeśli *warunek* jest spełniony, czyli daje wartość *true* (prawda), program wykonuje *blok_instrukcji* zawarty w nawiasach klamrowych.
- Jeśli *warunek* nie jest spełniony, czyli daje wartość *false* (fałsz), *blok_instrukcji* jest pomijany.
- W przypadku, gdy *blok_instrukcji* składa się z pojedynczej instrukcji, nawiasy klamrowe można pominąć.

Instrukcja warunkowa **if**

```
#include <iostream>
using namespace std;
int main()
{
    int i;
    cout << "Podaj liczbę dodatnią ";
    cin >> i;

    if (i <= 0)
        cout << "Podana liczba jest niepoprawna!";

    return 0;
}
```

Instrukcja warunkowa **if**

```
#include <iostream>
using namespace std;
int main()
{
    int i;
    cout << "Podaj liczbę dodatnią ";
    cin >> i;

    if (i <= 0)
    {
        cout << "Podana liczba jest niepoprawna! ";
        i = -i;
        cout << "Twoja liczba po poprawce: " << i;
    }

    return 0;
}
```

Instrukcja warunkowa **if else**

Instrukcja **if** pozwala zdecydować, czy dana instrukcja lub blok instrukcji mają być w ogóle wykonane.

Instrukcja **if else** pozwala zdecydować, która z dwóch instrukcji (lub bloków) ma być wykonana.

Składnia (1)

```
if (warunek_wyboru)
    instrukcja1
else
    instrukcja2
```

Składnia (1) - inny sposób formatowania kodu

```
if (warunek_wyboru) instrukcja1
else instrukcja2
```

Instrukcja warunkowa **if else**

Składnia (2)

```
if (warunek_wyboru)
{
    blok_instrukcji1
}
else
{
    blok_instrukcji2
}
```

- Jeśli *warunek_wyboru* jest spełniony (ma wartość *true*), program wykona *blok_instrukcji1*, a *blok_instrukcji2* pominie.
- W przeciwnym razie, jeśli *warunek_wyboru* nie jest spełniony, program pominie *blok_instrukcji1*, a wykona *blok_instrukcji2*.

Instrukcja warunkowa **if else**

```
#include <iostream>
using namespace std;
int main()
{
    int i;
    cout << "Podaj liczbę ";
    cin >> i;

    if (i > 0)
        cout << "Twoja liczba jest dodatnia.";
    else
        cout << "Twoja liczba jest ujemna lub równa 0.";

    return 0;
}
```

Instrukcja warunkowa **if else**

```
#include <iostream>
using namespace std;
int main()
{
    int i;
    cout << "Podaj liczbę ";
    cin >> i;
    if (i > 0)
        i = i + 2;
        cout << "Wynik: " << i;
    else
        i = i + 5;
        cout << "Wynik: " << i;
}
```

Program powoduje błąd kompilacji. Kompilator widzi pojedyncze **if** z instrukcją **i = i + 2;**. Dalej jest instrukcja z obiektem *cout*. Jak dotąd wszystko w porządku. Teraz jednak kompilator dochodzi do niepowiązanego z niczym **else**, więc zgłasza błąd składniowy.

Instrukcja warunkowa **if else**

Jeśli do kodu dodamy nawiasy klamrowe, uzyskamy zamierzony efekt:

```
#include <iostream>
using namespace std;
int main()
{
    int i;
    cout << "Podaj liczbę ";
    cin >> i;
    if (i > 0)
    {
        i = i + 2;
        cout << "Wynik: " << i;
    }
    else
    {
        i = i + 5;
        cout << "Wynik: " << i;
    }
}
```


Konstrukcja **if else if else**

W programach komputerowych często mamy więcej możliwości niż proste "tak" i "nie". Instrukcję **if else** można rozbudować tak, aby obsługiwała także takie sytuacje.

Składnia (1)

```
if (warunek_wyboru1)
    instrukcja1
else
    if (warunek_wyboru2)
        instrukcja2
    else
        instrukcja3
```

Konstrukcja **if else if else**

Składnia (1) - inny sposób formatowania kodu

```
if (warunek_wyboru1)
    instrukcja1
else if (warunek_wyboru2)
    instrukcja2
else
    instrukcja3
```

Konstrukcja **if else if else**

Składnia (2)

```
if (warunek_wyboru1)
{
    blok_instrukcji1
}
else
{
    if (warunek_wyboru2)
    {
        blok_instrukcji2
    }
    else
    {
        blok_instrukcji3
    }
}
```

Konstrukcja `if else if else`

```
#include <iostream>
using namespace std;
int main()
{
    int i;
    cout << "Podaj liczbę ";
    cin >> i;

    if (i > 0)
        cout << "Twoja liczba jest dodatnia.";
    else
        if (i < 0)
            cout << "Twoja liczba jest ujemna.";
        else
            cout << "Twoja liczba jest równa 0.";

    return 0;
}
```

Zagnieżdżona instrukcja **if else**

Składnia

```
if (warunek_wyboru1) instrukcja1
else if (warunek_wyboru2) instrukcja2
    else if (warunek_wyboru3) instrukcja3
    ...
    else instrukcjaN
```

Przykład: Program obliczający stopień na podstawie liczby otrzymanych punktów według kryteriów:

```
0...49 - 2
50...59 - 3
60...69 - 3.5
70...79 - 4
80...89 - 4.5
90...100 - 5
```

Zagnieżdżona instrukcja **if else**

```
#include <iostream>
using namespace std;
int main()
{
    int punkty;
    float stopien;
    cout << "Podaj liczbę punktów ([0,100]): ";
    cin >> punkty;

    if (punkty<=49) stopien=2;
    else if (punkty<=59) stopien=3;
        else if (punkty<=69) stopien=3.5;
            else if (punkty<=79) stopien=4;
                else if (punkty<=89) stopien=4.5;
                    else stopien=5;

    cout << "Twoja ocena: " << stopien;
    return 0;
}
```

Operator wyboru ?:

Składnia

wyrażenie1 ? wyrażenie2 : wyrażenie3

Jeśli *wyrażenie1* jest prawdziwe, wartością całego wyrażenia z operatorem wyboru jest *wyrażenie2*.

W przeciwnym wypadku wartością całego wyrażenia jest *wyrażenie3*.

Przykład 1: $5 > 3 ? 6 : 2$

5 jest większe od 3, więc całe wyrażenie przyjmie wartość 6.

Przykład 2: $5 == 3 ? 6 : 2$

5 nie jest równe 3, więc całe wyrażenie przyjmie wartość 2.

Operator wyboru ?:

```
#include <iostream>
using namespace std;
int main()
{
    int a, b, c;
    cout << "Podaj dwie liczby całkowite: " << endl;
    cin >> a >> b;
    c = a > b ? a : b;
    cout << "Większa z liczb " << a << " i " << b << " to " << c;
    return 0;
}
```

Przykładowy wynik działania programu:

```
Podaj dwie liczby całkowite:
2
6
Większa z liczb 2 i 6 to 6
```


Operator wyboru `?:` i instrukcja `if else`

Ten sam program z wykorzystaniem `if else` zamiast `?:`

```
#include <iostream>
using namespace std;
int main()
{
    int a, b, c;
    cout << "Podaj dwie liczby całkowite: " << endl;
    cin >> a >> b;
    if (a > b) c = a;
    else c = b;
    cout << "Większa z liczb " << a << " i " << b << " to " << c;
    return 0;
}
```

Instrukcja **switch**

Składnia

switch (wyrażenie)

```
{  
  case wartość1 : {instrukcja1; break;}  
  case wartość2 : {instrukcja2; break;}  
  ...  
  default      : {instrukcjaN; break;}  
}
```

- Najpierw obliczane jest *wyrażenie* umieszczone w nawiasach po słowie **switch**.
- Jeśli jego wartość odpowiada którejś z wartości podanej w jednej z etykiet **case**, wykonywane będą instrukcje począwszy od tej etykiety.
- Ich wykonywanie kończy się po napotkaniu instrukcji **break**. Działanie instrukcji **switch** zostaje wówczas zakończone.

Instrukcja **switch**

switch (wyrażenie)

```
{  
  case wartość1 : {instrukcja1; break;}  
  case wartość2 : {instrukcja2; break;}  
  ...  
  default      : {instrukcjaN; break;}  
}
```

- Jeśli wartość *wyrażenia* nie zgadza się z żadną z wartości w etykietach **case**, wykonywane będą instrukcje po etykiecie **default**.
- Etykieta **default** jest opcjonalna, tzn. może zostać pominięta. Wówczas, jeśli w zbiorze etykiet **case** nie ma żadnej etykiety równej wartości *wyrażenia*, instrukcja **switch** nie będzie wykonana.
- Instrukcje występujące po etykiecie **case** nie muszą kończyć się instrukcją **break**. Jeśli jej nie umieścimy, to będą wykonywane instrukcje znajdujące się pod następną etykietą **case**.

Instrukcja switch

Przykład: Program obliczający stopień na podstawie liczby otrzymanych punktów według kryteriów:

0...49 - 2

50...59 - 3

60...69 - 3.5

70...79 - 4

80...89 - 4.5

90...100 - 5

Instrukcja `switch`

```
#include <iostream>
using namespace std;
int main()
{
    int punkty;
    float stopien;
    cout << "Podaj liczbę punktów ([0,100]): ";
    cin >> punkty;
    punkty = punkty/10;
    switch (punkty)
    {
        case 5 : {stopien = 3; break;}
        case 6 : {stopien = 3.5; break;}
        case 7 : {stopien = 4; break;}
        case 8 : {stopien = 4.5; break;}
        case 9 :
        case 10 : {stopien = 5; break;}
        default : {stopien = 2; break;}
    }
    cout << "Twoja ocena: " << stopien;
}
```