

# VIII. Pętla for

## 8.1. Składnia pętli for(...)

Jeśli uważasz, że opanowałeś dobrze materiał z poprzednich rozdziałów najwyższy czas poznać składnię pętli for(...) i jej zastosowanie.

Pętłę for można podzielić na cztery części:

- Inicjacja początkowych wartości zmiennych (A)
- Ustalenie warunku kończącego pętlę (B)
- Zwiększenie (zmniejszenie) licznika pętli (C)
- Powtarzany blok instrukcji (D);

Zapis uogólniony pętli for(...) wygląda następująco:

```
for(A;B;C)
{//początek powtarzanego bloku instrukcji
  D
}//koniec powtarzanego bloku instrukcji
//lub
for(A;B;C) D;
```

Parametry **A**, **B** i **C** oddzielamy **zawsze** średnikami.

Prosty przykład omawianej pętli:

```
for ( int i = 1; i <= 10; i++)
{
  //Powtarzany blok instrukcji
}

//lub
```

```
for(int i = 1; i <= 10; i++) jedna_powtarzana_instrukcja;
```

Zapis **int i = 1** jest to inicjacja początkowej wartości zmiennej. Zapis ten oznacza, że tworzymy zmienną typu **int**, nazywamy ją **i** oraz nadajemy jej wartość początkową równą **1**. Kolejną część pętli for stanowi warunek. W tym przypadku jest to **i <= 10** i oznacza tyle, że dopóki warunek jest prawdziwy, to ma wykonywać blok instrukcji. Trzecią częścią, bez której pętla for nie byłaby sobą to zwiększenie bądź zmniejszenie wartości zmiennej. W tym przypadku jest to zapis **i++**. Zmienna ta jest zwiększana za każdym razem po wykonaniu wszystkich instrukcji z bloku.

## 8.2. Pętla for(...) a jej użyteczność

Pętla jest jednym z ważniejszych elementów programowania i często bez niej nie da się napisać ciekawego programu. Jeśli chcemy policzyć średnią z określonej liczby liczb, wczytać określoną ilość danych z pliku lub wypisać określoną ilość danych na ekran, pętla for jest do tego po prostu idealna. By dobrze rozumieć działanie pętli spójrzmy jeszcze raz na prosty przykład:

```
#include <conio.h>
#include <iostream>
int main()
{
    using namespace std;
    int i = 10;

    for(i; i > 0; i--)
        cout << "Rakieta startuje za " << i << " sek." << endl;
    cout << "Start zgodny z planem";

    getch();
    return 0;
}
```

Jak już wiesz pętla wykonuje się dopóki warunek pętli jest prawdziwy ( $i > 0$ ). Jeżeli  $i = 0$  to pętla zakończy działanie. **Inaczej mówiąc jeżeli warunek pętli jest prawdziwy(true) to pętla będzie się wykonywać, a jeżeli będzie nieprawdziwy(false) to zakończy(lub nie rozpocznie wcale) swojego działania.**

Takie rozumowanie pociąga za sobą pewien szczegół, który może umknąć naszej uwadze, spójrzmy na kolejny przykład:

```
#include <conio.h>
#include <iostream>
int main()
{
    using namespace std;
    int i = 10;

    for(i; i; i--)
        cout << "Rakieta startuje za " << i << " sek." << endl;
    cout << "Start zgodny z planem";

    getch();
    return 0;
}
```

Dlaczego pętla się kończy gdy  $i = 0$ , przecież nie ma tam warunku określającego takie właśnie zakończenie pętli. Jednak przypomnę pewien szczegół omawiany we wcześniejszych lekcjach, **wartość 0 symbolizuje fałsz (false), natomiast wartość niezerowa symbolizuje prawdę (true)**, a ponieważ wartość 0 symbolizuje fałsz to warunek pętli dla  $i = 0$  też jest fałszywy co oznacza jej zakończenie. Reasumując pętla wykona się tylko wtedy gdy warunek pętli będzie prawdziwy, czyli nie przybierze wartości **false(0)**.

## 8.3. Analizujemy przykłady

Przeanalizuj teraz dokładnie działanie poniższych programów. Przepisz je i uruchom, a następnie wykonaj kilka eksperymentów modyfikując jego działanie. Eksperymenty są zalecane przy nauce programowania. Warto też wprowadzać celowo po jednym błędzie w kodzie, żeby zapoznawać się stopniowo z komunikatami, jakie zaczniesz nam pokazywać kompilator jeśli zapis będzie nieprawidłowy lub jak zaczniesz zachowywać się nasz program. Nie bój się eksperymentów, to one dają Ci doświadczenie!

### 8.3.1. Liczenie średniej ocen

```
#include <iostream>
#include <conio.h>
using namespace std;
int main()
{
    int ilosc;
    float ocena;
    float srednia;
    cout << "Podaj ilosc ocen: ";
    cin >> ilosc;
    srednia = 0;
    for (int i = 1; i <= ilosc; i++)
    {
        cout << "Podaj ocene nr " << i << ": ";
        cin >> ocena;
        srednia += ocena;
    }
    if (ilosc > 0) srednia /= ilosc;
    cout << "Srednia ocen to: " << srednia << endl;
    getch();
    return(0);
}
```

Na uwagę zasługuje linijka:

```
for (int i = 1 ; i <= ilosc ; i++)
```

a dokładnie **int i = 1**; W C++ wprowadzono, możliwość zadeklarowania zmiennej w ciele **for-a**, co jest bardzo pomocne. Zmienna taka istnieje tylko dla tej pętli, co oznacza jej usunięcie po zakończeniu działania pętli.

## 8.4. Ćwiczenia

1. Napisz program, który na ekranie wyświetli takie o to dane:

1,3,9,27 (kolejne potęgi liczby 3).

2. Napisz program obliczający silnie danej liczby