

Instrukcja pętli

```
for (licznik; warunek; iterator;)
{
    blok_instrukcji;
}
```

Zasada działania: *licznik* ustawia początkową wartość licznika pętli. *blok_instrukcji* wykonuje się dopóki *warunek* jest prawdziwy. Po każdej iteracji *iterator* zwiększa lub zmniejsza wartość licznika pętli. Jeśli *blok_instrukcji* zawiera tylko jedną instrukcję, wówczas nawiasy { i } mogą być pominięte

Przykład 2.1 Program do obliczania średniej ocen dziesięciu studentów

```
static void p21()
{
    int suma = 0;
    int ocena;

    for (int i = 0; i < 5; i++)
    {
        Console.Write("Podaj ocenę: ");
        ocena = Convert.ToInt32(Console.ReadLine());
        suma += ocena;
    }
    double srednia = suma / (double)5;
    Console.WriteLine("Średnia z ocen wynosi: {0:N}", srednia);
}
```

Zadanie 2.1 Zmodyfikować przykład 2.1 w taki sposób, by program obliczał średnią z ocen pozytywnych.

Zadanie 2.2 Klient banku deponuje na lokacie X letniej kwotę Y . Lokata jest oprocentowana 6% rocznie. Kapitalizacja odsetek odbywa się na koniec każdego roku. Napisać program do obliczania, ile wyniesie wartość tej inwestycji po X latach.

Zadanie 2.3 Napisać program do znajdowania największej (najmniejszej) liczby wśród 5-ciu liczb podanych przez użytkownika.

Zadanie 2.4 Napisać program do obliczania liczby punktów zdobytych przez skoczka narciarskiego. Na liczbę punktów (p), jakie otrzymuje skoczek składają się z dwa elementy: punkty zdobyte za długość skoku (p_d) i punkty zdobyte za styl

$$p = p_d + p_s$$

Punkty zdobyte za długość skoku oblicza się według reguły:

$$p_d = 60 + (d - K) * S,$$

gdzie d oznacza długość skoku, K jest punktem konstrukcyjnym skoczni, a S pewną stałą zależną od wielkości skoczni. W zadaniu należy przyjąć $K = 120$ oraz $S = 1.8$.

Punkty zdobyte za styl oblicza się następująco: 5-ciu sędziów wystawia oceny w skali 0 do 20, najlepsza i najgorsza z tych ocen są odrzucane, a pozostałe są sumowane.

Program powinien wczytać długość skoku (d) i oceny wystawione przez pięciu sędziów i na tej podstawie obliczyć liczbę punktów.

Zadanie 2.5 Napisać program do obliczania silni metodą tradycyjną (nie rekurencyjną).

Zadanie 2.6 Napisać program do sprawdzania czy wczytana liczba jest pierwsza.

Zadanie 2.7 Napisać program, który wczytuje dwie liczby całkowite: a i b , gdzie $b > a$, a następnie wypisuje wszystkie liczby pierwsze z przedziału $[a, b]$.

Wskazówka: Zmodyfikować metodę do sprawdzania, czy liczba jest pierwsza w taki sposób, by jako argument przyjmowała liczbą do sprawdzenia i zwracała wartość `true` lub `false` w zależności od tego czy podana liczba jest pierwsza.

Zadanie 2.8 Napisać program, który oblicza symbol Newtona
$$\binom{n}{k} = \frac{n!}{k!(n-k)!}$$

Wskazówka: Wykorzystać funkcję silnia z zadania 2.5.