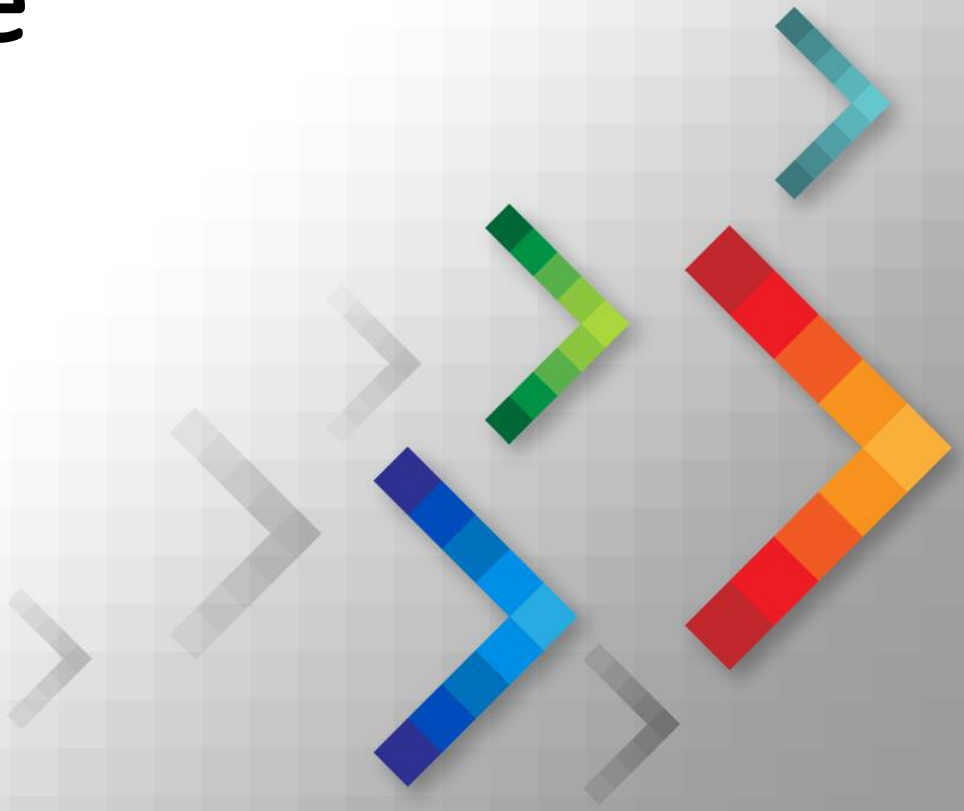
MiGra

PODRĘCZNIKI DO INFORMATYKI

Programowanie w języku C++

Grażyna Koba



Kilka definicji:

Program komputerowy to ciąg instrukcji języka programowania, realizujący dany algorytm.
Język programowania to zbiór określonych instrukcji i zasad składni, używanych do zapisania tzw. **kodu źródłowego** programu.

Program może występować w dwóch postaciach:

- jako **program źródłowy** (postać zrozumiała dla programisty),
- jako **program wynikowy** (kod maszynowy, program wykonywalny) – zapisany w postaci ciągu instrukcji procesora, zrozumiały dla komputera.

Źródło: G. Koba, *Teraz bajty. Informatyka dla szkół podstawowych. Klasa VIII,* Migra, Wrocław 2018
G. Koba, *Teraz bajty. Informatyka dla szkół ponadpodstawowych. Zakres podstawowy. Klasa I,* Migra, Wrocław 2019

Kilka definicji:

Program w języku wysokiego poziomu (program źródłowy)

```
#include <iostream>
using namespace std;

int main()
{
    int a, b, s;
    cin >> a;
    cin >> b;
    s = a + b;
    cout << s;
    return 0;
}
```

Tłumaczenie (translacja)

Wygenerowany kod programu w języku wewnętrznym – fragment (program wynikowy)

pokazany w symbolicznej formie

```
0x00401374      call    0x416610 <__main>
0x00401379      lea     -0x20(%ebp),%eax
0x0040137C      mov     %eax,(%esp)
0x0040137F      movl    $0x1,-0x58(%ebp)
0x00401386      mov     $0x477900,%ecx
0x0040138B      call    0x446984 <std::istream::operator>>(int&)>
0x00401390      sub     $0x4,%esp
0x00401393      lea     -0x24(%ebp),%eax
0x00401396      mov     %eax,(%esp)
0x00401399      mov     $0x477900,%ecx
0x0040139E      call    0x446984 <std::istream::operator>>(int&)>
0x004013A3      sub     $0x4,%esp
```

Źródło: G. Koba, *Teraz bajty. Informatyka dla szkół podstawowych. Klasa VIII,* Migra, Wrocław 2018
G. Koba, *Teraz bajty. Informatyka dla szkół ponadpodstawowych. Zakres podstawowy. Klasa I,* Migra, Wrocław 2019

Kilka definicji:

Proces tłumaczenia programu napisanego w języku programowania wysokiego poziomu na język wewnętrzny komputera nazywamy **translacją**. Może on przebiegać w formie **kompilacji** lub **interpretacji**.

Kompilacja – przetłumaczenie **całego programu** na język zrozumiały dla procesora, tak by ten program mógł być wykonywany przez komputer.

Interpretacja – tłumaczenie programu tworzonego w jednym z języków programowania **instrukcja po instrukcji**, tak by każda wywołana instrukcja była wykonana przez komputer.



Aby utworzyć program w języku C++:

1. Piszemy kod źródłowy programu (implementujemy program) – korzystając z edytora wbudowanego do środowiska programistycznego wybranego języka programowania.
2. Zapisujemy program w pliku.
3. Kompilujemy program, zwykle korzystając z opcji **Kompiluj** (**Compile**). Otworzy się okno z informacją o przebiegu kompilacji.
4. Jeśli program został skompilowany, uruchamiamy go, zazwyczaj korzystając z opcji **Uruchom** (**Run**). Program uruchomi się, zwykle w osobnym oknie.

Źródło: G. Koba, *Teraz bajty. Informatyka dla szkół podstawowych. Klasa VIII,* Migra, Wrocław 2018
G. Koba, *Teraz bajty. Informatyka dla szkół ponadpodstawowych. Zakres podstawowy. Klasa I,* Migra, Wrocław 2019

Podstawowa struktura programu w języku C++

```
#include <iostream>
using namespace std;
…                    // deklaracje stałych
…                    // deklaracje funkcji
int main ()
{                    // część wykonawcza programu
   return 0;
}
```

Dołączenie biblioteki standardowej

Informacja o korzystaniu z biblioteki standardowej

Źródło: G. Koba, *Teraz bajty. Informatyka dla szkół podstawowych. Klasa VIII,* Migra, Wrocław 2018
G. Koba, *Teraz bajty. Informatyka dla szkół ponadpodstawowych. Zakres podstawowy. Klasa I,* Migra, Wrocław 2019

Deklarowanie zmiennych

1. W języku C++ małe i wielkie litery w nazwach instrukcji, zmiennych itp. mają różne znaczenie. nazwach zmiennych można stosować litery (bez polskich znaków diakrytycznych), cyfry i znak podkreślenia. Nazwa nie może zaczynać się od cyfry.

- W języku C++ zmienne można deklarować w dowolnym miejscu, ale zawsze przed ich pierwszym użyciem.

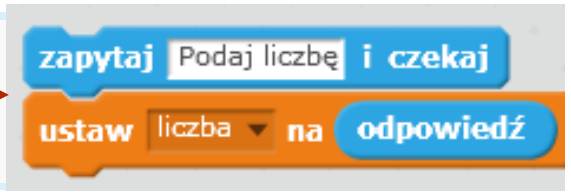
```
int liczba;
int i, j, Suma;
```

```
float Wartosc;
float a, b, X, Y;
```

Instrukcje wejścia/wyjścia

`cin` jest **obiektem** reprezentującym standardowe **wejście** programu.

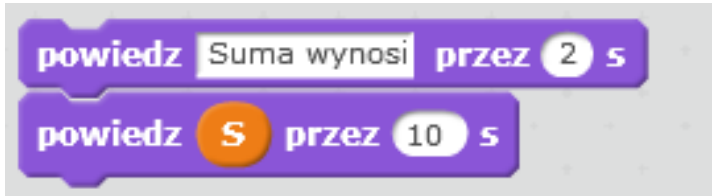
```
cin >> liczba;
cin >> a >> b >> W;
```



Operator **>>** oznacza wprowadzenie danych do odpowiedniej zmiennej, podanej po jego prawej stronie.

`cout` jest obiektem reprezentującym standardowe wyjście programu. Operator **<<** oznacza wyprowadzenie wartości podanej po jego prawej stronie.

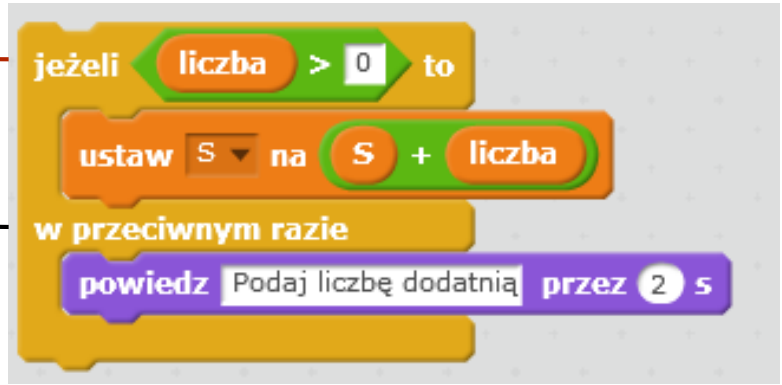
```
cout << P;
cout << "S = " << a+b << "\n ";
cout << "Suma wynosi:" << S << endl;
```



Instrukcja warunkowa

```
if (wyrażenie) instrukcja1; else instrukcja2;
lub
if (wyrażenie) instrukcja1;
```

```
if(liczba > 0)
    cout << S = S + liczba;
else cout << "Wprowadź liczbę dodatnią";
```



określenie	C++
równy	==
różny	!=
mniejszy	<
mniejszy lub równy	<=
większy	>
większy lub równy	>=
alternatywa logiczna	||
koniunkcja logiczna	&&
negacja logiczna	!

Źródło: G. Koba, *Teraz bajty. Informatyka dla szkół podstawowych. Klasa VIII,* Migra, Wrocław 2018
G. Koba, *Teraz bajty. Informatyka dla szkół ponadpodstawowych. Zakres podstawowy. Klasa I,* Migra, Wrocław 2019

Instrukcje iteracyjne

Iteracja polega na wielokrotnym powtarzaniu tej samej operacji (ciągu operacji).
Iterację implementujemy, stosując tzw. **pętlę**. Z pętlą mamy do czynienia, gdy w pewnym kroku algorytmu wracamy do jednego z wcześniejszych kroków, co powoduje, że kroki te mogą zostać wykonane wiele razy.

```
for (wyrażenie_początkowe; warunek; wyrażenie_pętli) instrukcja;
```

```
 8      for(i=0; i<n; i++)
 9      {
10          cout << "Podaj liczbe: ";
11          cin >> liczba;
12          S = S + liczba;
13      }
```

```
S+=liczba
```

Tablice w języku C++

Aby utworzyć zmienne indeksowane w języku C++, musimy zadeklarować specjalną strukturę danych – tablicę.

```
opis_typu nazwa_tablicy [liczba_elementów_tablicy];
```

Do elementów tablicy odwołujemy się, podając nazwę tablicy i indeks elementu umieszczony w nawiasach kwadratowych, np.: *a*[0], *a*[1], …, *a*[*n* - 1] – dla tablicy *n*-elementowej o nazwie *a*. W języku C++ pierwszy indeks jest zawsze równy 0.

Liczbę elementów tablicy musimy określić wcześniej, np. wpisać w deklaracji tablicy jako konkretną wartość.

Na przykład: `int dane[50];`

oznacza zadeklarowanie tablicy o nazwie *dane* składającej się z pięćdziesięciu elementów o indeksach od 0 do 49.

Do elementów tablicy odwołujemy się przez zmienne: *dane*[0], *dane*[1], …, *dane*[49].

Funkcje w języku C++

W języku C++ wszystkie podprogramy nazywane są funkcjami. Dzielimy je na dwie grupy:

- funkcje niezwracające wartości (odpowiednik procedury w języku Pascal)

```
4   const  int N = 10;
5   int a[N];
6
7   void WprowadzDane()
8   {
9       for( int i = 0; i < N; i++)
10      {
11          cout << "Podaj dana nr " << i << ": ";
12          cin >> a[i];
13      }
14  }
```

Funkcje w języku C++

- funkcje zwracające wartość (odpowiednik funkcji w języku Pascal).

```
 4  int Maksimum(int n)
 5  {
 6      int i, a, max;
 7
 8      for(i=0; i<n; i++)
 9      {
10          cout << "Podaj liczbe: ";
11          cin >> a;
12          if(i==0)
13              max = a;
14          else
15              if(a > max)
16                  max = a;
17      }
18      return max;
19  }
```

Źródło: G. Koba, *Teraz bajty. Informatyka dla szkół podstawowych. Klasa VIII,* Migra, Wrocław 2018
G. Koba, *Teraz bajty. Informatyka dla szkół ponadpodstawowych. Zakres podstawowy. Klasa I,* Migra, Wrocław 2019