

Programowanie 1

Sławomir Pluciński (splucinski@pjawstk.edu.pl)

Algorytm

skończony zbiór dobrze zdefiniowanych instrukcji przeznaczony do wykonania określonego zadania, który przy ustalonym stanie początkowym pozwala na uzyskanie odpowiedniego, rozpoznawalnego stanu końcowego w skończonym czasie.

Etapy konstruowania algorytmu

- 1) sformułowanie zadania – ustalamy jaki problem ma rozwiązywać algorytm
- 2) określenie danych wejściowych – ich typu (w typie określamy, czy dane są liczbami rzeczywistymi, całkowitymi, czy znakami, czy też innego typu)
- 3) określenie wyniku oraz sposobu jego prezentacji
- 4) ustalenie metody wykonania zadania (może być kilka metod na rozwiązanie – wybieramy tą, która według nas jest najlepsza)
- 5) zapisanie algorytmu za pomocą wybranej metody (z punktu 4)
- 6) analiza poprawności rozwiązania
- 7) testowanie rozwiązania dla różnych danych (algorytm musi być uniwersalny, aby służyć do rozwiązywania zadań dla różnych danych wejściowych)
- 8) ocena skuteczności algorytmu (praktyczna ocena algorytmu : np. szybkości, skomplikowania)

Sposoby zapisu algorytmu:

- 1) lista kroków
- 2) pseudojęzyk (pseudokod)
- 3) graficzna prezentacja za pomocą schematu blokowego
- 4) zapis w danym języku programowania

Sposoby zapisu algorytmu:

- 1) lista kroków
- 2) pseudojęzyk (pseudokod)
- 3) graficzna prezentacja za pomocą schematu blokowego
- 4) zapis w danym języku programowania

Sposób 1.

Lista kroków charakteryzuje się tym, że każdy wiersz opisujący pojedynczy krok realizowanej czynności jest numerowany.

- 1) pobierz pierwszą liczbę
- 2) pobierz drugą liczbę
- 3) dodaj liczby do siebie
- 4) wynik dodawania podziel przez 2
- 5) wyświetl otrzymaną wartość
- 6) zakończ

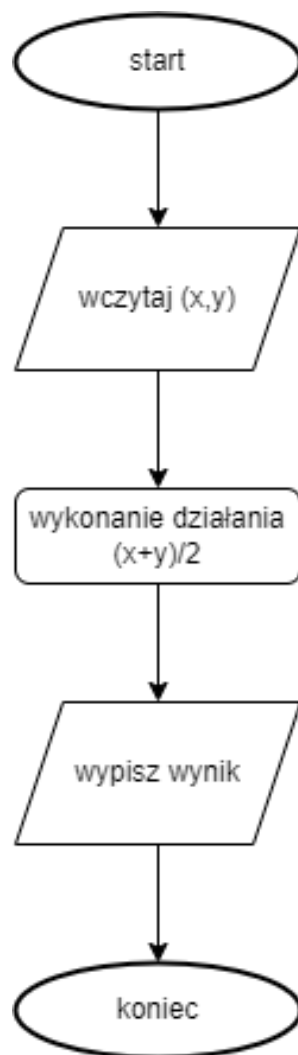
Sposób 2.

Pseudojęzyk (pseudokod) jest metodą pośrednią między zapisem za pomocą listy kroków a zapisem w języku programowania.

- początek
- wprowadzenie liczb x i y rzeczywistych
- wykonanie działania $(x+y)/2$
- pisz wynik
- koniec

Sposób 3.

Zapis za pomocą schematu blokowego.



Sposób 4a.

Problem zapisany w postaci programu w języku C/C++

```
#include <iostream>
int main() {
    float x, y;
    std::cin >> x;
    std::cin >> y;
    std::cout << "Średnia arytmetyczna wprowadzonych liczb wynosi: „  $(x+y)/2$ ;"
}
```

Sposób 4b.

Problem zapisany w postaci programu w języku Java

```
import java.util.Scanner;
```

```
public class Main {
```

```
    public static void main(String[] args){
```

```
        Scanner scanner = new Scanner(System.in);
```

```
        double x = scanner.nextDouble();
```

```
        double y = scanner.nextDouble();
```

```
        System.out.println("Średnia arytmetyczna wprowadzonych liczb wynosi: " + (x+y)/2 );
```

```
    }
```

```
}
```

Sposób 4b.

Problem zapisany w postaci programu w języku Java

```
import java.util.Scanner;
```

```
public class Main {
```

```
    public static void main(String[] args){
```

```
        Scanner scanner = new Scanner(System.in);
```

```
        double x = scanner.nextDouble();
```

```
        double y = scanner.nextDouble();
```

```
        System.out.println("Średnia arytmetyczna wprowadzonych liczb wynosi: " + (x+y)/2 );
```

```
    }
```

```
}
```

Specyfikacją problemu algorytmicznego

Algorytm obliczający sumę dwóch liczb pod warunkiem, że obie są dodatnie.

Specyfikacja:

Zadanie: Oblicz sumę dwóch liczb, pod warunkiem, że liczby te będą dodatnie.

Dane: Wprowadzone przez użytkownika wartości rzeczywiste liczb a i b .

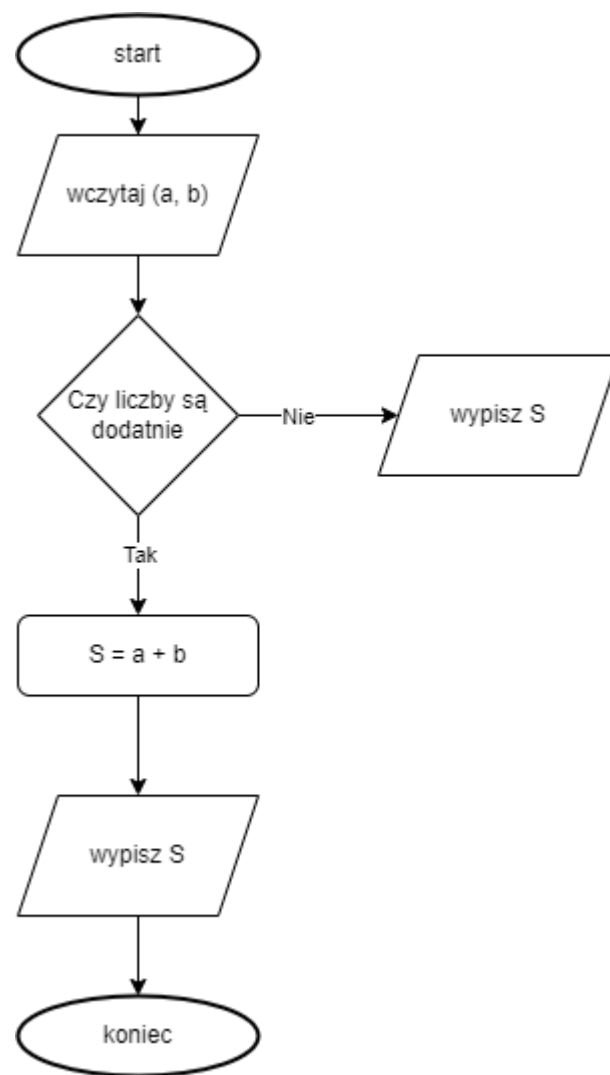
Wynik: Suma liczb a i b , jeżeli obie te liczby były dodatnie, w przeciwnym razie tekst „Liczby nie spełniają warunków zadania”.

1. **Opis słowny:** Wczytaj dwie liczby. Sprawdź, czy liczby są dodatnie. Jeżeli obie są dodatnie, oblicz wynik i wyświetl go na ekranie, w przeciwnym razie wypisz na ekran informację o tym, że liczby nie spełniają warunku zadania

2. Lista kroków:

1. Zaczynij algorytm.
2. Wprowadź wartość dwóch liczb: a, b.
3. Sprawdź, czy liczby a i b są dodatnie.
4. Jeżeli liczby są dodatnie, przejdź do punktu 6.
5. Wypisz na ekran „Liczby nie spełniają warunków zadania”. Przejdź do punktu 8.
6. Oblicz sumę obu wprowadzonych liczb: $S = a + b$.
7. Wyświetl wynik na ekran: S.
8. Zakończ algorytm.

3. Schemat blokowy:



Zmienne

Zarezerwowany obszar w pamięci, który przechowuje daną określonego typu o zadeklarowanej wielkości.

Wartość do zmiennych przypisywana jest przez zastosowanie znaku =
zmienna = wartość;

Zmienne dzielimy z uwagi na ich zasięg na:

- Zmienne globalne
- Zmienne lokalne

Typy zmiennych

W C++ wymagane jest jawne wskazanie jakiego typu jest zmienna po przez wybór odpowiedniego słowa kluczowego.

Typ zmiennej jest definiowany po przez następujący schemat: typ_zmiennej nazwa;

Wyróżniamy następujące typy:

unsigned char	8 bitów	0 .. 255
char	8 bitów	-128 .. 127
short int	16 bitów	-32,768 .. 32,767
unsigned int	32 bits	0 .. 4,294,967,295
int	32 bits	-2,147,483,648 .. 2,147,483,647
unsigned long	32 bits	0 .. 4,294,967,295
long	32 bits	-2,147,483,648 .. 2,147,483,647
float	32 bits	3.4×10^{-38} .. $3.4 \times 10^{+38}$
double	64 bity	1.7×10^{-308} .. $1.7 \times 10^{+308}$
long double	80 bitów	3.4×10^{-4932} .. $1.1 \times 10^{+4932}$
boolean	1 bajt	True / False