

Algorytmy

Aby zrozumieć ten dział dotyczący algorytmów musimy najpierw zrozumieć pojęcie co to jest algorytm?

Algorytm – jest to ciąg czynności bądź sposób postępowania, który prowadzi do wykonania określonego zadania lub rozwiązania problemu w określonym czasie.

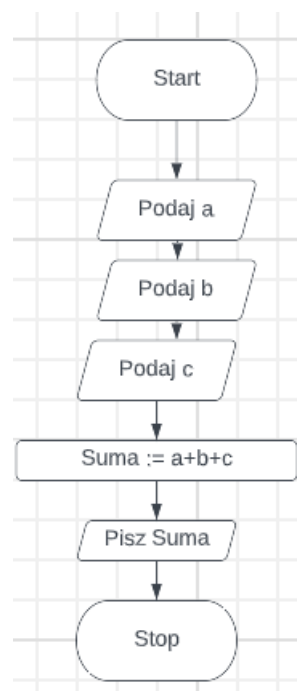
Algorytmika – dział zajmujący się badaniem algorytmów.

Schemat blokowy


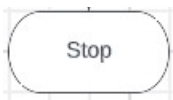
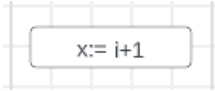


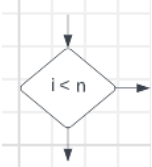
Schemat blokowy – diagram, na którym algorytm jest opisywany dzięki figurom geometrycznym, połączone liniami zgodnie z kolejnością wykonywania czynności wynikających z przyjętego algorytmu rozwiązania zadania – pozwala dostrzec ważne etapy algorytmu i logiczne zależności między nimi.

Na schemacie blokowym poszczególne operacje są opisywane za pomocą klocków, bloków – połączonych ze sobą strzałkami.

Przykład schematu blokowego:



Elementy schematu blokowego:

Symbol graficzny	Nazwa	Funkcja	Opis
	Blok graniczny	Oznacza początek algorytmu	Owalny kształt, wychodzi z niego tylko jedna strzałka, żadna nie może do niego prowadzić
	Blok graniczny	Oznacza zakończenie algorytmu	Owalny kształt, prowadzi do niego jedna strzałka, żadna z niego nie wychodzi
	Blok operacyjny	Służy do zapisania wykonywanych operacji na przykład działań algebraicznych, operacji podstawienia itp.	Prostokąt, w jego wnętrzu zapisujemy wykonywane operacje.
	Blok wejścia	Służy do wprowadzenia danych	Równoległobok, wchodzi do niego i wychodzi jedna strzałka
	Blok wyjścia	Służy do wyprowadzenia wyniku operacji	Równoległobok, wchodzi do niego i wychodzi jedna strzałka
	Blok warunkowy	Służy do sprawdzenia warunku	Romb, zapisujemy w jego wnętrzu warunek do sprawdzenia. Wychodzą z niego dwie strzałki: pierwsza określa operacje wykonywaną w przypadku spełnienia warunku, druga w przypadku nie spełnienia warunku

Pseudokod

Pseudokod - to połączenie języka naturalnego z elementami języka programowania.

Przykłady:

1. Algorytm wczytuje dwie liczby i sprawdza, która jest większa.

Pseudokod:

Start

Wczytaj(a,b)

Jeżeli $a > b$ to

Wypisz(a)

W przeciwnym razie

Wypisz(b)

Koniec

2. Algorytm wczytuje i sumuje n liczb.

Start

Suma:=0

Podaj(n)

$i:=0$

Dopóki $i \leq n$ wykonuj:

Wczytaj(a)

Suma := Suma + a

$i := i + 1$

Wypisz(Suma)

Koniec

3. Algorytm wczytuje i dodaje 10 liczb.

Start

$i := 0$

Dopóki $i < 10$

Wczytaj(a)

$\text{Suma} := \text{Suma} + a$

$i := i + 1$

Koniec

Rodzaje algorytmów:

- Algorytm liniowy – to taki, w którym nie określono żadnych warunków. Jest też nazywany sekwencyjnym, gdyż każdy z kroków w tym algorytmie następuje sekwencyjnie, czyli wykonanie jednej sekwencji powoduje przejście bezpośrednio do następnej.

Przykład – Obliczanie obwodu prostokąta:

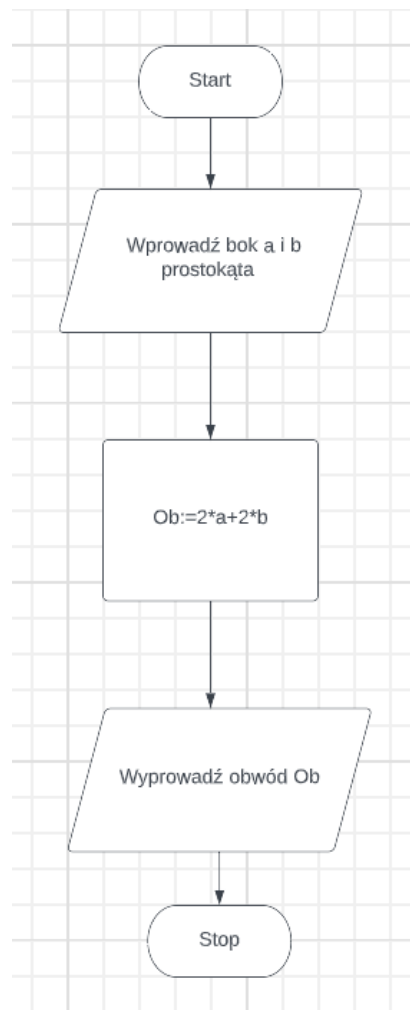
Algorytm liniowy w postaci listy kroków:

Dane: bok a i b

Lista kroków:

1. Początek algorytmu
2. Podaj bok a
3. Podaj bok b
4. oblicz obwód: $ob := 2 * a + 2 * b$
5. Wyprowadź wartość ob
6. Koniec algorytmu

Algorytm liniowy w postaci schematu blokowego:



- Algorytm warunkowy - to taki, w którym wykonanie instrukcji uzależnione jest od spełnienia lub niespełnienia warunku.

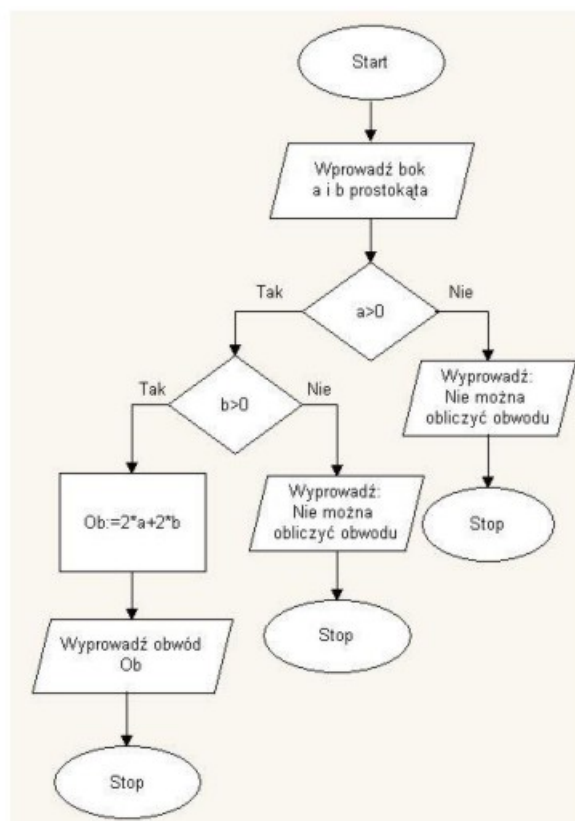
Obliczanie obwodu prostokąta:

Algorytm liniowy w postaci listy kroków:

Dane: bok a i b

1. Początek algorytmu
2. Podaj bok a
3. Podaj bok b
4. Czy bok $a > 0$? jeśli tak idź do kroku 5, jeśli nie podaj komunikat wyjściowy: "nie można obliczyć obwodu" i zakończ algorytm.
5. Czy bok $b > 0$? jeśli tak idź do kroku 6 jeśli nie podaj komunikat wyjściowy: "nie można obliczyć obwodu" i zakończ algorytm.
6. Oblicz obwód $Ob = 2 \cdot a + 2 \cdot b$
7. Wyprowadź wartość Ob
6. Koniec algorytmu

Algorytm liniowy w postaci schematu blokowego:



- Algorytm iteracyjny

Iteracja - nazywamy instrukcję powtarzania danego ciągu operacji. Liczba powtórzeń może być ustalona przed wykonaniem instrukcji lub może zależeć od spełnienia pewnego warunku, który jest sprawdzany w każdej iteracji. Iteracja inaczej zwana jest pętlą.

a) Pętla z licznikiem

Pętla, w której ilość powtórzeń n jest ustalona z góry. Ilość ta jest "kontrolowana" przez tzw. zmienną sterującą, która z kolei jest inkrementowana, czyli zwiększana o jeden.

Zwiększenie tej wartości powoduje odpowiednie wyrażenie (licznik), dla zmiennej sterującej k będzie to $k:=k+1$. Jeżeli np. zmiennej k nadamy wartość początkową 0 to będzie to wyglądało następująco:

$k:=0$ i $k:=k+1$ $0:=0+1$ (zmienna k przyjmuje wartość 1)- pierwsze przejście

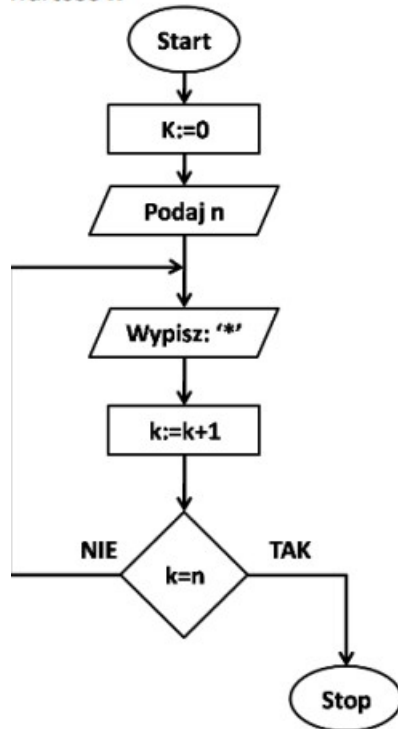
$1:=1+1$ (zmienna k przyjmuje wartość 2)- drugie przejście

$2:=2+1$ (zmienna k przyjmuje wartość 3)- trzecie przejście

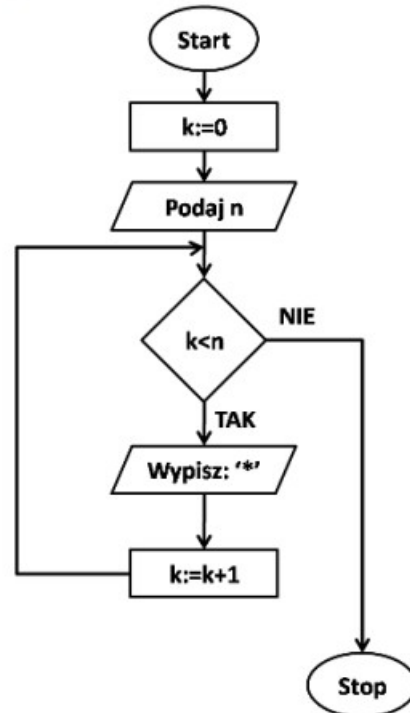
itp. Jeżeli k osiągnie odpowiednią wartość np. $k \leq n$ to wówczas pętla zostaje opuszczona i wykonywana jest dalsza część instrukcji.

Poniżej znajdują się przykłady, które wypiszą szlaczek z n gwiazdek:

Pętla jest wykonywana tak długo, aż k osiągnie wartość n



Warunek jest sprawdzany przed wykonaniem instrukcji

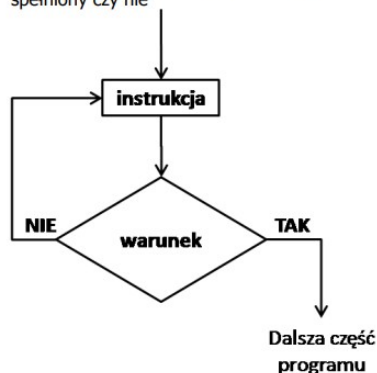


b) Pętle warunkowe

Oprócz pętli z licznikiem istnieją jeszcze inne dwa rodzaje pętli, których działanie jest uzależnione od warunków. Poniżej znajdują się schematy blokowe tych pętli.

Przykład

Najpierw jest wykonywana instrukcja, a następnie jest sprawdzany warunek. Pętla jest wykonywana, aż do spełnienia warunku. Instrukcja zostanie wykonana przynajmniej jeden raz bez względu na to, czy warunek jest spełniony czy nie



Warunek jest sprawdzany na początku, a dopiero później jest wykonywana instrukcja. Pętla jest wykonywana tak długo, jak spełniony jest warunek.

