

## Schematy blokowe - podstawy

*Schemat blokowy to graficzny zapis algorytmu rozwiązania zadania, przedstawiający opis i kolejność wykonywania czynności realizujących dany algorytm.*

W schemacie blokowym poszczególne operacje przedstawione są za pomocą odpowiednio połączonych skrzynek (klocków, bloków). Połączenia określają kolejność i sposób wykonywania operacji realizujących dany algorytm.

W literaturze informatycznej przyjęto pewne standardowe oznaczenia poszczególnych działań (są to figury geometryczne), ale można również używać innych oznaczeń (muszą one jednak być takie same dla określonego typu operacji).

Przykłady skrzynek do prezentacji algorytmu w postaci graficznej:

Elementy schematu blokowego

| Symbol graficzny  | Nazwa skrzynki (bloku)     | Funkcja  | Opis   |
|---|----------------------------|--|--|
|  | Skrzynka graniczna         | Początek algorytmu lub koniec  | mają kształt owalu. Ze skrzynki START wychodzi tylko jedno połączenie, skrzynka STOP nie ma połączenia wychodzącego. |
|  |                            |  |  |
|  | Skrzynka operacyjna        | Wykonywanie różnych działań, np. sumowania                                     | ma kształt prostokąta.   |
|  | Skrzynka wejścia / wyjścia | Wprowadzanie (czytanie) danych lub wyprowadzanie (drukowanie, pisanie) wyników | jest równoległobokiem, wchodzi i wychodzi z niej jedno połączenie.   |
|  |                            |  |  |
|  | Skrzynka warunkowa         | Sprawdzanie warunku, np. czy $N > 0$   | mają kształt rombu. Ze skrzynki wychodzą tylko dwa połączenia: jedno oznaczone TAK, a drugie NIE.                    |

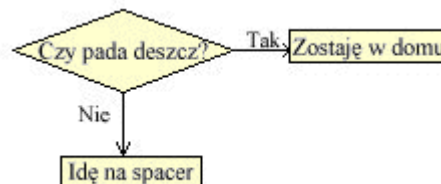
## Zasady budowania schematu blokowego

- Każda operacja jest umieszczona w skrzynce
- Schemat ma tylko jedną skrzynkę "Start" i przynajmniej jedną skrzynkę "Stop"
- Skrzynki są ze sobą połączone
- Ze skrzynki wychodzi jedno połączenie; wyjątek stanowią skrzynki "Stop" (z której nie wychodzą już żadne połączenia) oraz "warunkowa" (z której wychodzą dwa połączenia opisane „Tak” i „Nie” - w zależności od tego czy warunek jest spełniony czy też nie; można wyjść jedną z dwóch dróg).

## Sytuacje warunkowe

Z sytuacjami warunkowymi stykamy się w każdej dziedzinie życia codziennego. Na pytanie "Czy pada deszcz?" odpowiedź może brzmieć "tak" lub "nie". W zależności od tego, czy warunek jest spełniony, czy nie, wybieramy inne rozwiązanie.

Przykład sytuacji warunkowej:



Wiesz już, że z sytuacją warunkową mamy do czynienia wówczas, gdy wynik lub dalsze działanie zależy od spełnienia warunku. Na schemacie blokowym sytuacje warunkowe realizujemy przez skrzynkę warunkową.

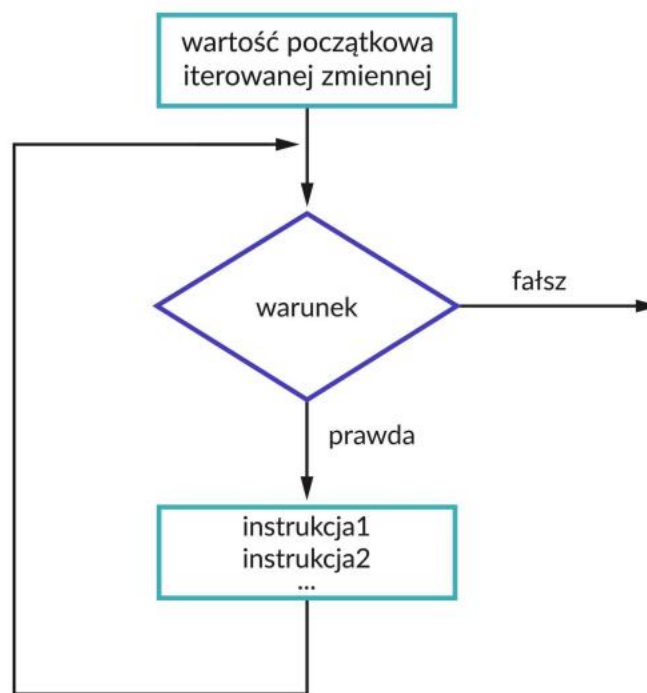
W skrzynce wpisujemy warunek logiczny, stosując znaki "=" równy, "<>" różny, "<" mniejszy, ">" większy, "<=" mniejszy lub równy, ">=" większy lub równy, np:

$(a > 5)$  lub  $(a \leq 20)$ ,  $(a < 5)$  OR  $(a \leq 20)$

Z systemami warunkowymi spotykamy się m.in. w matematyce i fizyce, wtedy gdy wykonywanie działań jest uzależnione od warunku, jakie muszą spełniać liczby, np. mają być nieujemne albo parzyste.

## Iteracja, czyli działanie w pętli

Czasami trzeba wykonać te same operacje na wielu liczbach. W takich przypadkach nie jest konieczne wielokrotne opisywanie działań lub rysowanie takich samych skrzynek. Stosujemy wówczas iterację. Mówimy także, że działania te wykonywane są w pętli. Liczba powtórzeń tych działań może być z góry określona lub zależeć od spełnienia warunku. Iteracja to najczęściej spotykana technika algorytmiczna.



W pierwszym bloku operacyjnym zainicjowana zostaje wartość początkowa iterowanej zmiennej (np. posłodzenie herbaty pierwszą łyżeczką cukru). Dzięki temu możemy zliczać liczbę iteracji. Następnie sprawdzany jest warunek. Jeżeli jest on fałszywy, wychodzimy z pętli i nie powtarzamy więcej operacji w niej zawartych. Natomiast jeżeli warunek będzie spełniony, wykonujemy ciąg odpowiednich instrukcji. Po wykonaniu ciągu instrukcji wracamy do sprawdzenia warunku. Jeżeli jest spełniony, ponownie wykonujemy te same instrukcje itd.