

Opracował: Artur Borowiec; Politechnika Rzeszowska 2016

SMath Studio - elementy programowania

1. W programie Smath Studio można zastosować:

- * instrukcje grupowania- `line()`,
- * instrukcje warunkową - `if()`,
- * instrukcje iteracyjne:
 - pętla warunkowa - `while()`,
 - pętla zakresowa - `for()`.

XI. Instrukcja grupowania

Instrukcja grupowania pozwala wydzielić grupę instrukcji.

Definicję grupy uzyskujemy:

- * z panelu bocznego Programowanie > line
- * klawisz `[]` (nawias kwadratowy zamykający).

Grupa reprezentowana jest pionową linią obok instrukcji.

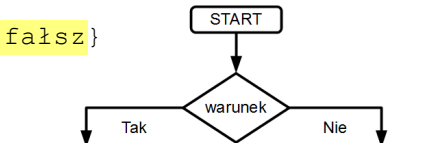
<code>instrukcja1</code>	Grupowanie poprawia przejrzystość zapisu
<code>instrukcja2</code>	algorytmu. Możliwe jest definiowanie
<code>instrukcja3</code>	wielopoziomowych grup instrukcji.

XII. Instrukcja warunkowa

Instrukcja warunkowa `if()` `wymagane{warunek;prawda;fałsz}`

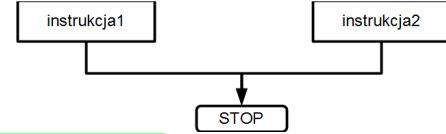
Zastosowanie instrukcji warunkowej pozwala rozdzielić algorytm na dwie wersje (ścieszki).

Wynik sprawdzenia warunku instrukcji musi być



albo prawdą (1), albo fałszem (0).

- * warunek musi zawierać operator logiczny
- * warunek może być zdaniem logicznym

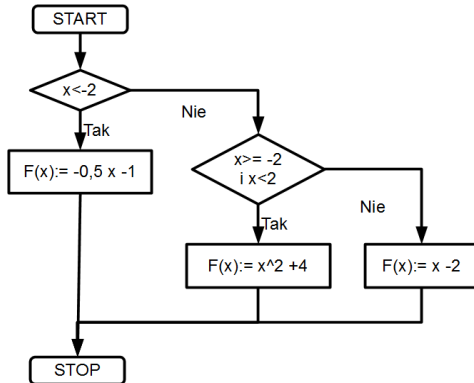


Przykład zastosowania instrukcji warunkowej (dwie wersje) do narysowania wykresu funkcji $F(x)$ określonej w przedziałach.

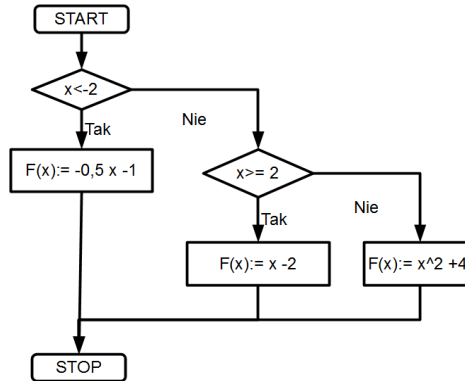
$$F(x) := \begin{cases} -\frac{x}{2} - 1 & \text{dla } x \in (-\infty, -2) \\ -x + 4 & \text{dla } x \in [-2, 2) \\ x - 2 & \text{dla } x \in [2, \infty) \end{cases}$$

$$F(3) = \begin{cases} -2,5 \\ -5 \\ 1 \end{cases}$$

Wersja I



Wersja II



```

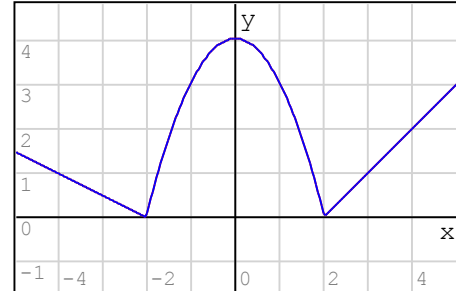
F1(x):= if x < -2
        |  -x / 2 - 1
        |  else
        |  | if (x ≥ -2) ∧ (x < 2)
        |  |   -x2 + 4
        |  |   else
        |  |     x - 2
    
```

F1(3) = 1

```

F2(x):= if x < -2
        |  -x / 2 - 1
        |  else
        |  | if x ≥ 2
        |  |   x - 2
        |  |   else
        |  |     -x2 + 4
    
```

F2(3) = 1



$\begin{cases} F1(x) \\ F2(x) \end{cases}$

☐ - XIIIa. Instrukcje iteracyjne

Pętla while() wymagane{warunek;instrukcje}

1. Instrukcja iteracyjna pętli warunkowej pozwala powtarzać cyklicznie wybrane instrukcje podczas prawdziwości warunku pętli.
2. Instrukcje powinny modyfikować argumenty warunku pętli.

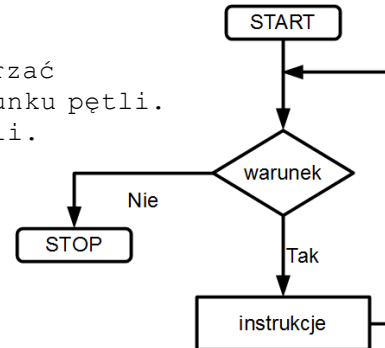
UWAGA:

Źle zdefiniowany warunek może zablokować program.

Przykład zastosowania pętli warunkowej.

```

{ i := 1
  t := -3 Definicja inicjacyjna zmiennych.
while t < 5
  | T1 i := t
    
```



```
i:= i+1
t:= t+2
```

Obliczenie elementów
pierwszego wiersza macierzy T.

$$T = (-3 \ -1 \ 1 \ 3)$$

XIIIb. Instrukcje iteracyjne cd.

Pętla for(3) wymagane {licznik; zakres; instrukcje}

1. Instrukcja iteracyjna pętli zakresowej pozwala powtarzać cyklicznie wybrane instrukcje z podstawieniem aktualnej wartości licznika dla ściśle określonego zakresu powtórzeń pętli.
2. Licznik to zmienna z kolejnymi wartościami zakresu.
3. Zakres pętli definiujemy z wykorzystaniem funkcji zakresowej rang(2) lub rang(3).
4. Liczba powtórzeń to liczba elementów z zakresu.

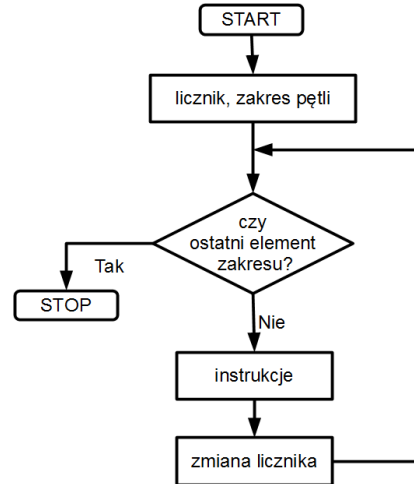
Przykład zastosowania pętli zakresowej.

```
for k ∈ 1..7
    K1 k := k
    K2 k := k2 + 1
```

licznik i zakres pętli
zgrupowane instrukcje
do wykonania w pętli

$$K = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 2 & 5 & 10 & 17 & 26 & 37 & 50 \end{pmatrix}$$

$$k = 7$$



Pętla for(4) wymagane {licznik;warunek;zakres;instrukcje}

1. Instrukcja iteracyjna pętli zakresowej z dodatkowym warunkiem pozwala powtarzać cyklicznie wybrane instrukcje z podstawieniem aktualnej wartości licznika.
2. Licznik jest modyfikowany rekurencyjnie po każdym przejściu pętli.

Przykład zastosowania pętli zakresowej z warunkiem.

licznik; warunek licznika; modyfikacja licznika

for t:=1;t<8;t:=t+1

$T_1 t := t$	zgrupowane instrukcje do wykonania w pętli
$T_2 t := t^2 + 1$	

$T = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 2 & 5 & 10 & 17 & 26 & 37 & 50 \end{pmatrix}$

t = 8

