

# LabView

## 3

**Dr Piotr Sitarek**

**Katedra Fizyki Doświadczalnej, Politechnika Wrocławska**

# Temat na dziś

---

Programowanie w środowisku (cd.)

ni.com



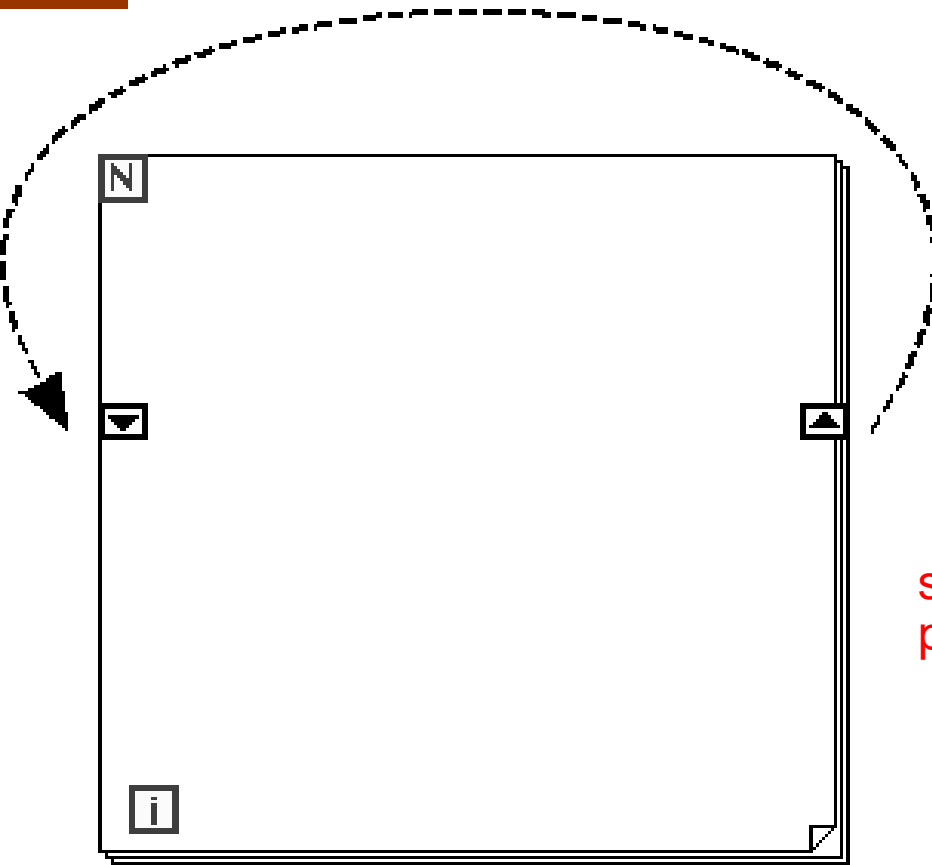
(część materiałów zaczerpnięta ze strony  
producenta)

RONALD W. LARSEN, *LabVIEW for Engineers*

---

# Struktury

- **Shift Registers** - dostęp do danych z poprzedniej pętli

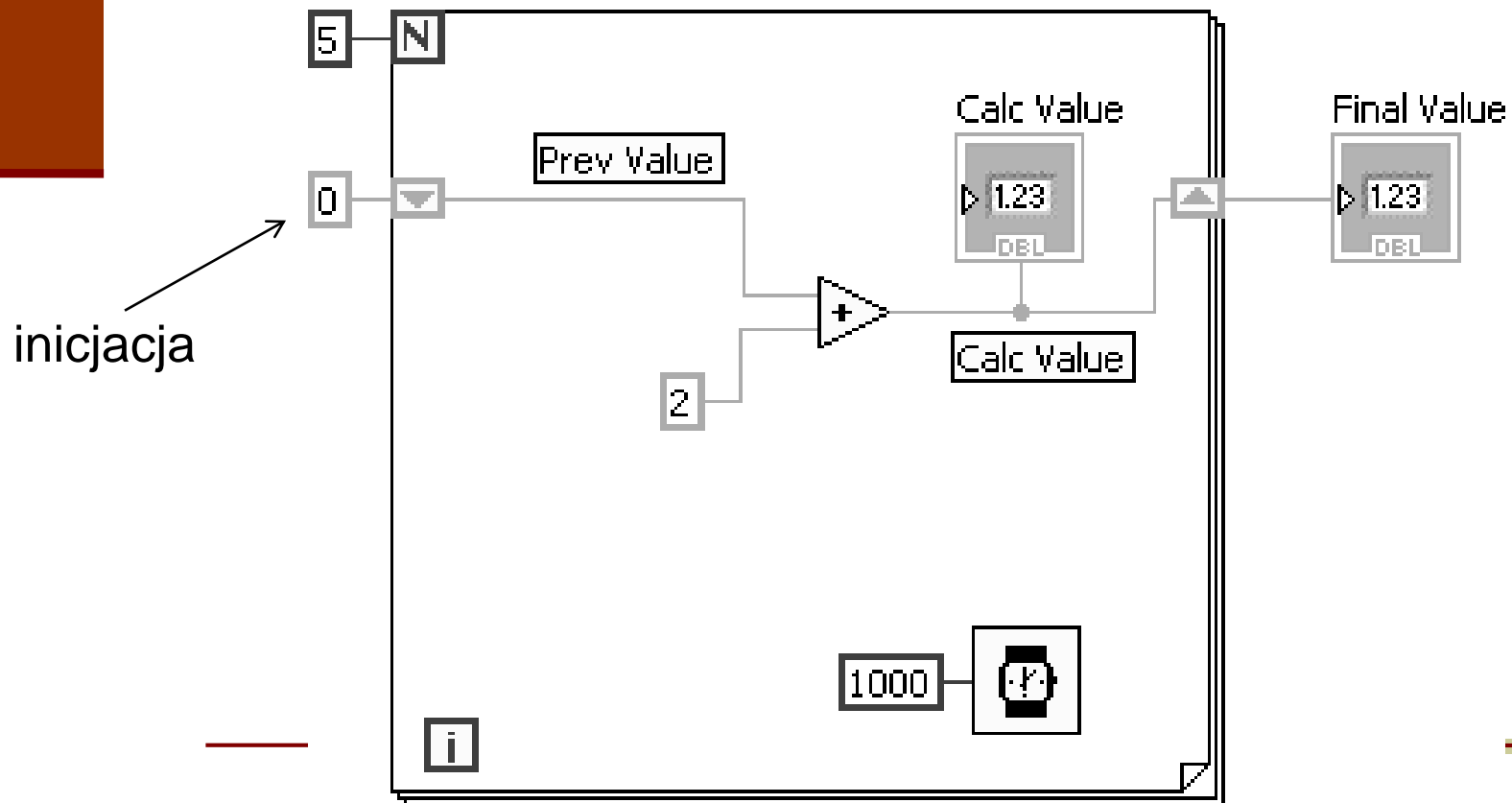


wysyła wartość końcową na początek kolejnej pętli – nie jest potrzebne fizyczne połączenie (przewód)

shift register dodajemy klikając prawym przyciskiem myszy na ramkę pętli

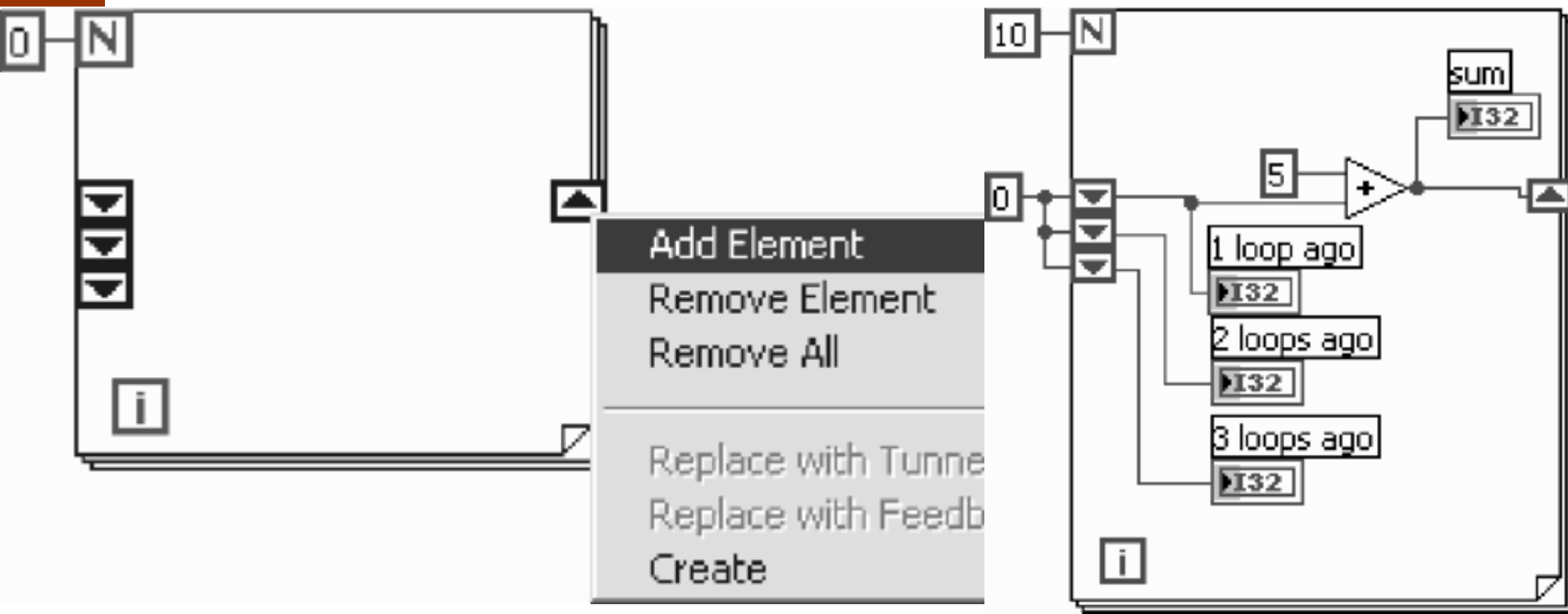
# Struktury

- ❑ **Shift Registers** - dostęp do danych z poprzedniej pętli



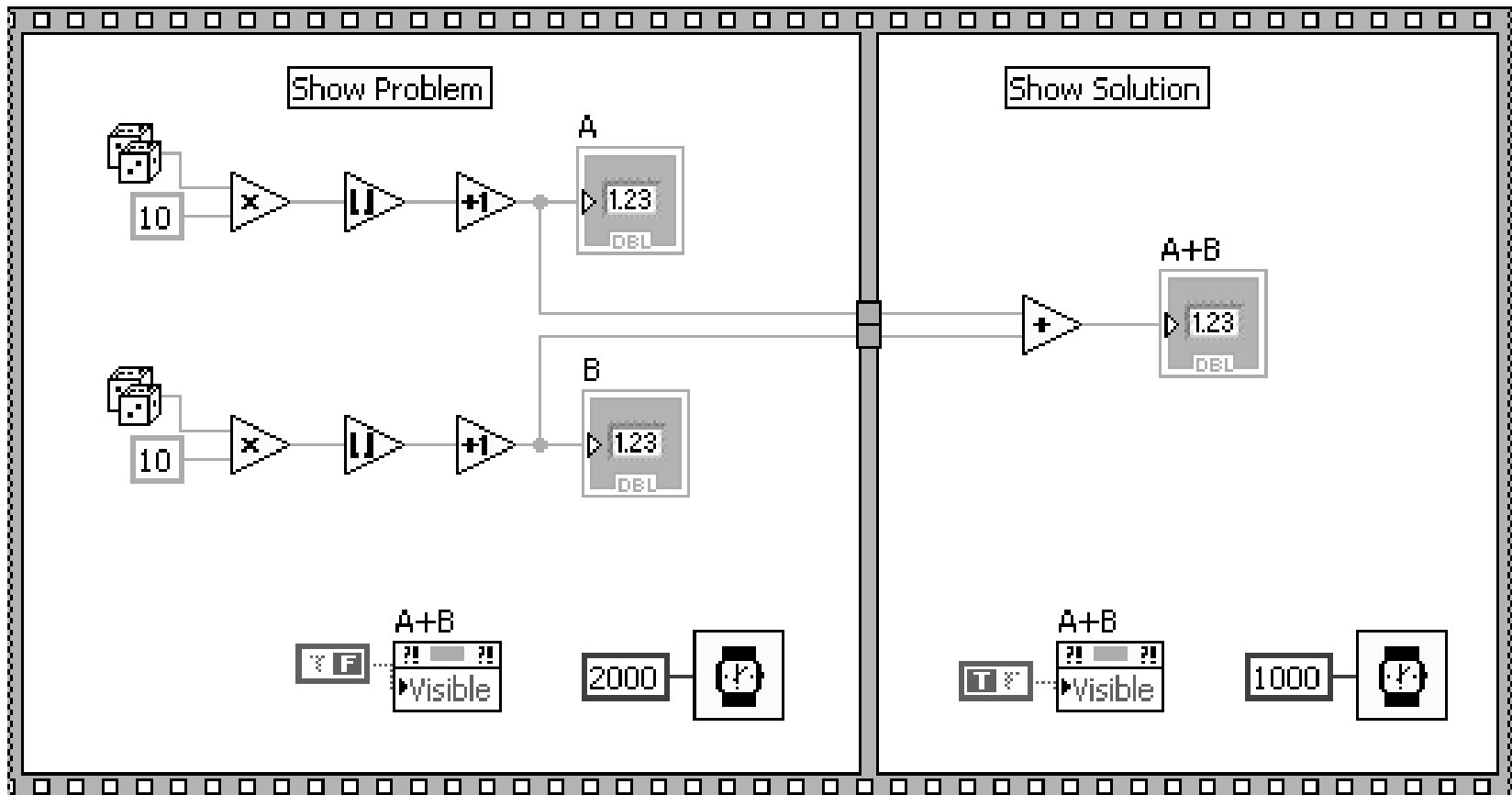
# Struktury

- ❑ **Stacked Shift Registers** - dostęp do danych z poprzednich pętli



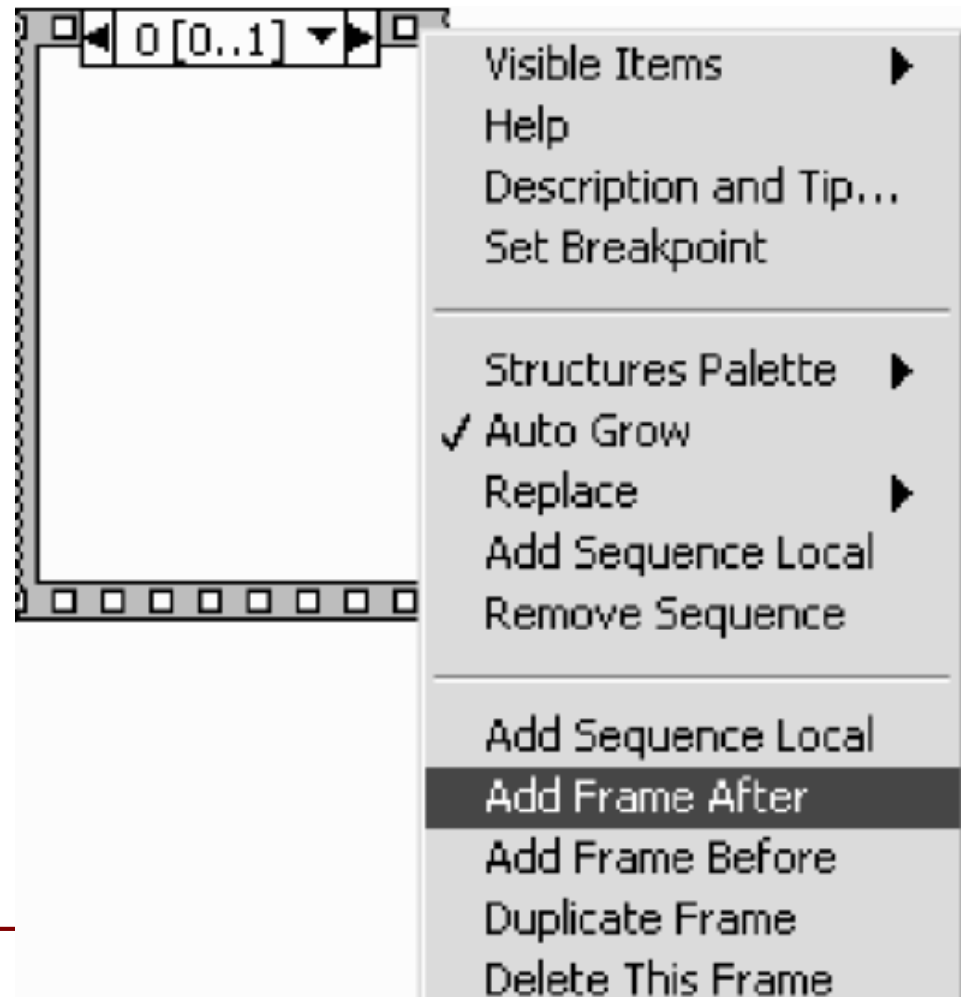
# Struktury

- ❑ **Sequence Structures** – wymusza zachowanie kolejności wykonywanych zadań



# Struktury

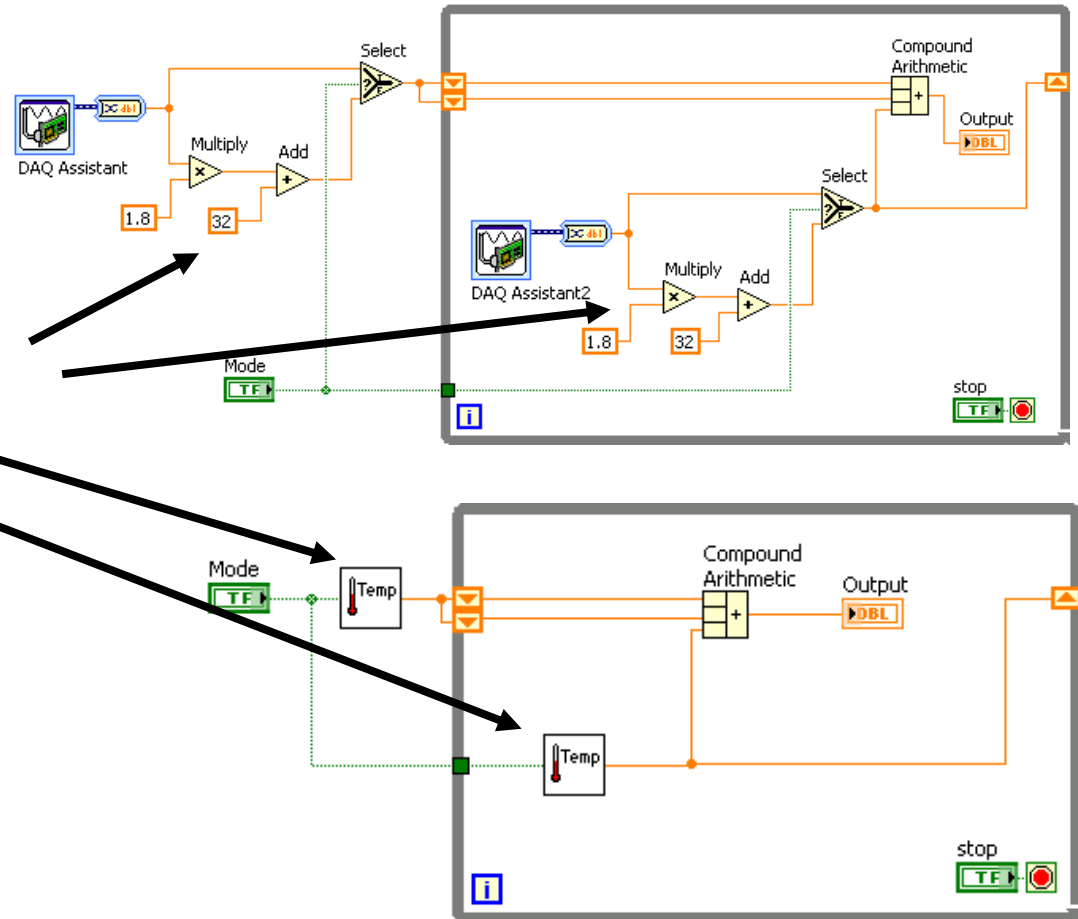
## ❑ Sequence Structures - stacked



można „zwinąć” sequence structure do jednej ramki widocznej w danej chwili (podobnie jak dla str. case)

# Modularność – podprocedury (SubVi's)

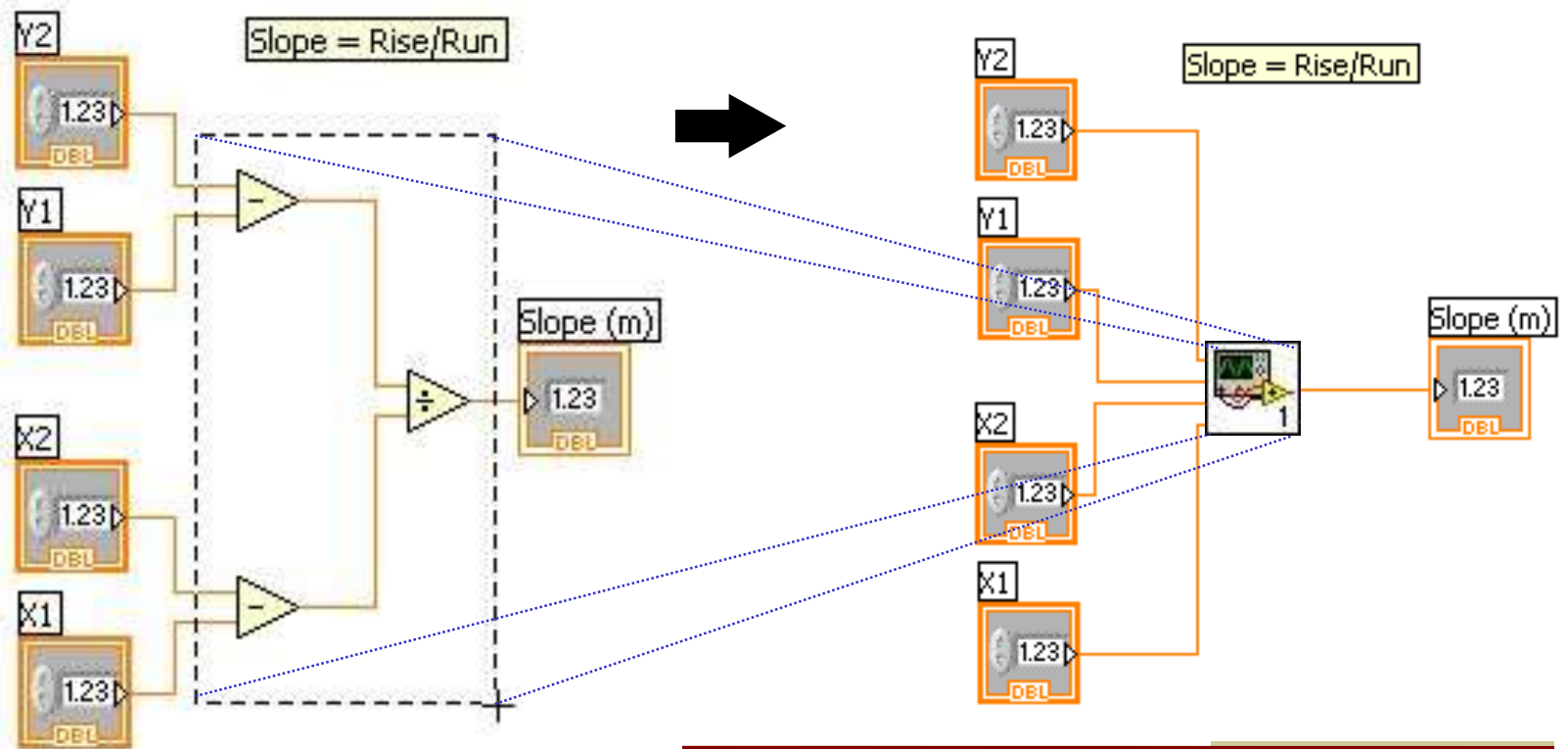
Zamiana powtarzających się fragmentów w podprocedury





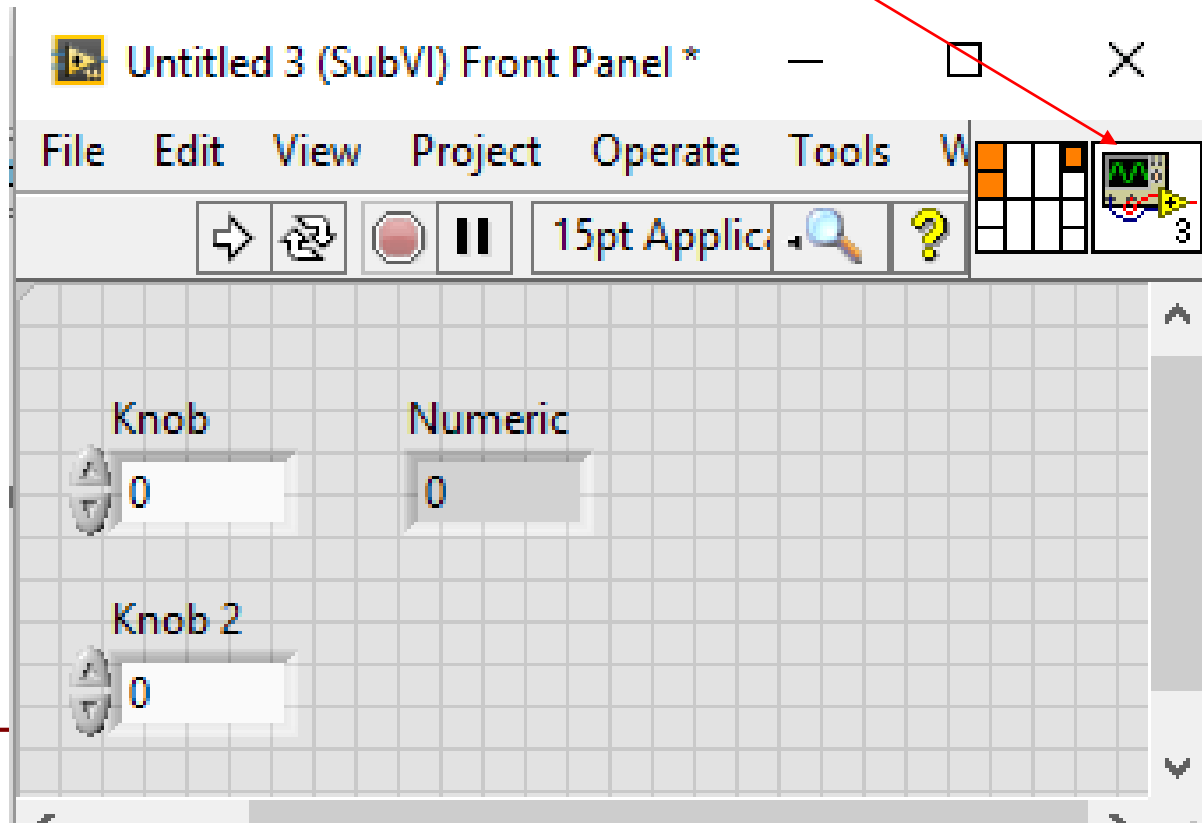
# Tworzenie podprocedur

- Zaznacz obszar, który ma być zamieniony na SubVi
- Wybierz **Edit»Create SubVI** z Edit Menu.

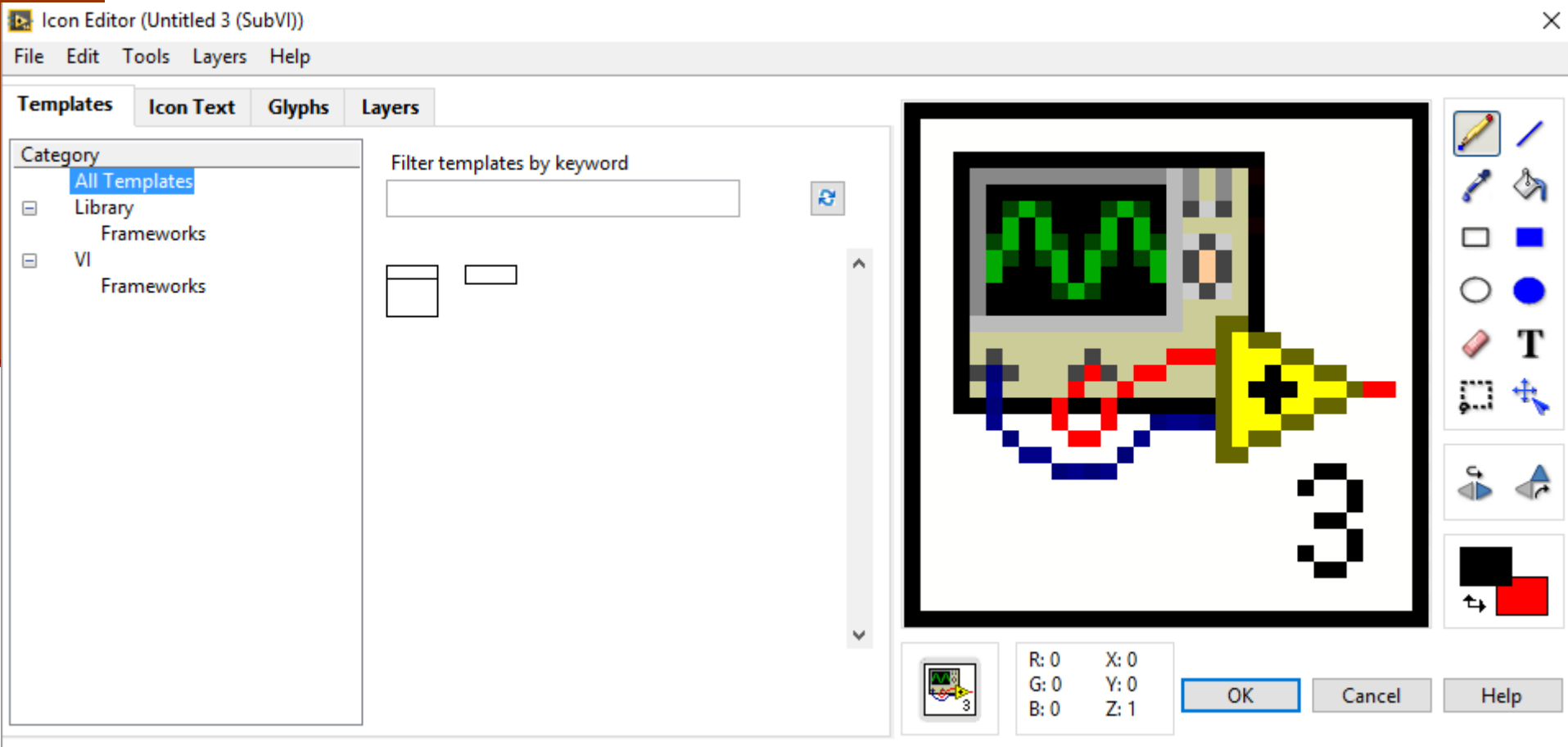


# Tworzenie podprocedur

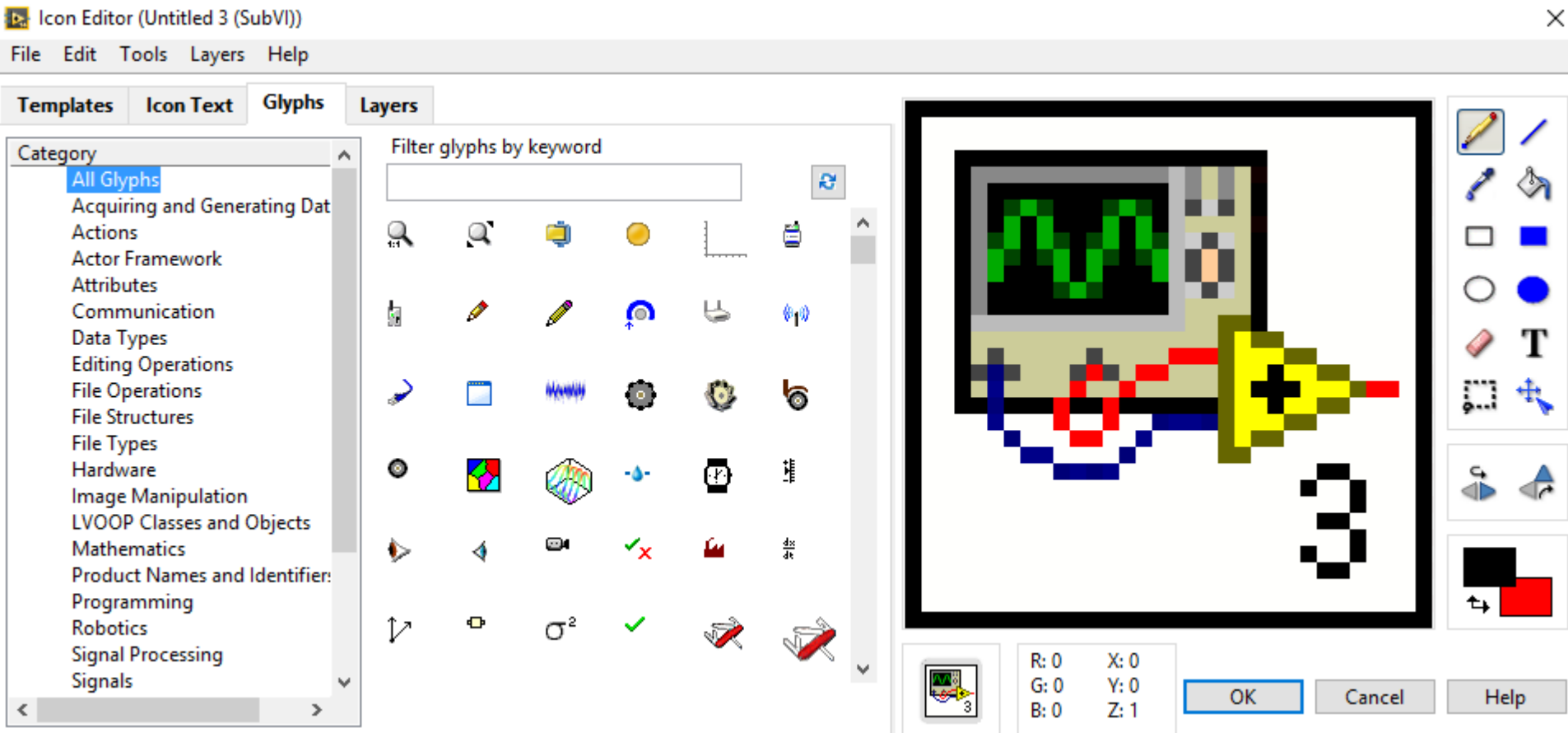
ikonka reprezentująca podprocedurę jest konfigurowalna  
– np. po podwójnym kliknięciu



# Tworzenie podprocedur

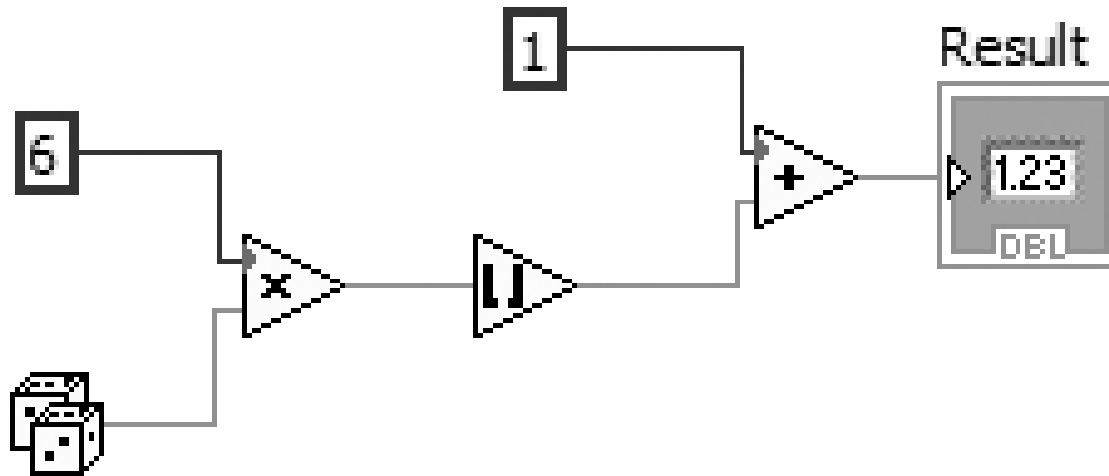


# Tworzenie podprocedur



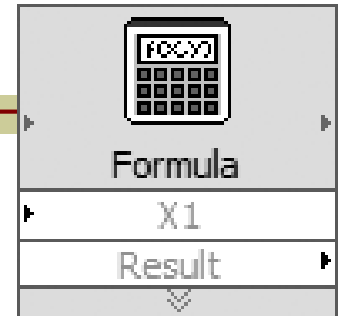
# Funkcje matematyczne

Generator losowy (rozkład normalny)  $0 \leq \text{losowanie} < 1$ .



symulator rzutu kostką

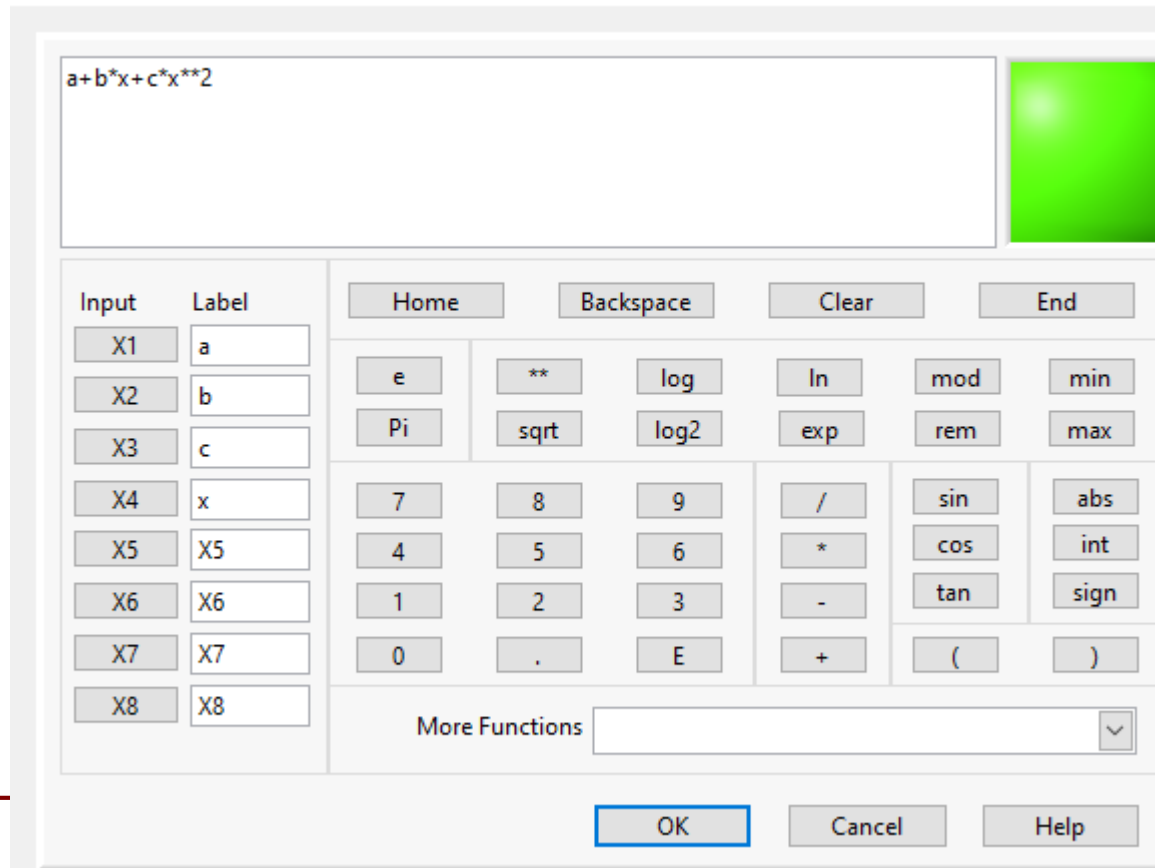
# Funkcje matematyczne



## Formula (express)

- program do wpisywania bardziej skomplikowanych wyrażeń algebraicznych bez konieczności używania poszczególnych funkcji w połączeń między nimi
- w kolumnie label możemy ustanowić nazwy własnych zmiennych
- zielone pole oznacza poprawnie wprowadzone dane

Configure Formula



Input	Label
X1	a
X2	b
X3	c
X4	x
X5	X5
X6	X6
X7	X7
X8	X8

Home    Backspace    Clear    End

e    \*\*    log    ln    mod    min  
Pi    sqrt    log2    exp    rem    max

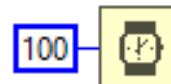
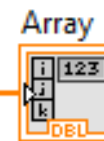
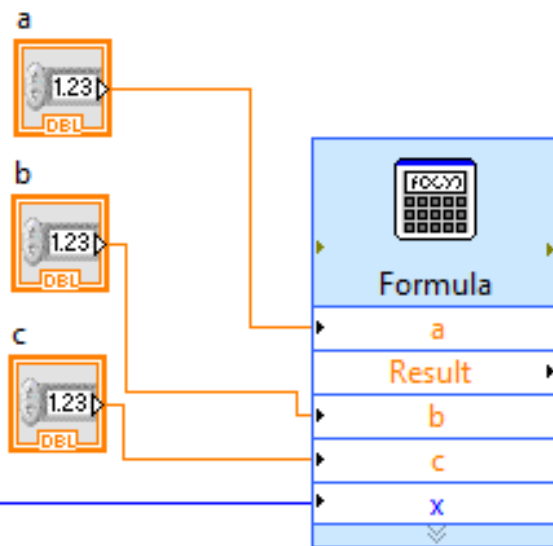
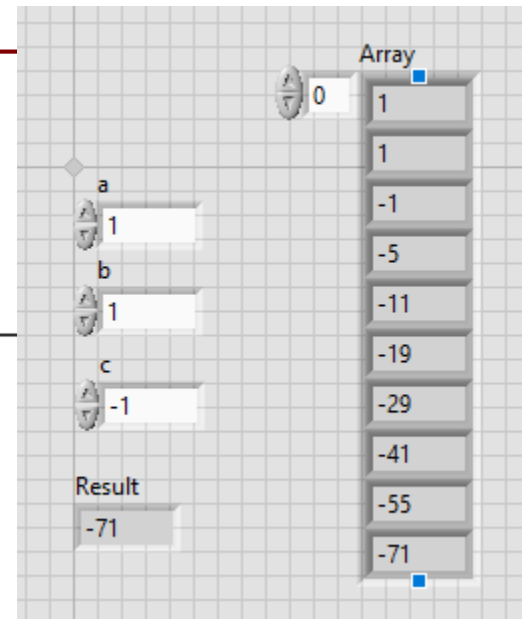
7    8    9    /    sin    abs  
4    5    6    \*    cos    int  
1    2    3    -    tan    sign  
0    .    E    +    (    )

More Functions

OK    Cancel    Help

# Funkcje matematyczne

## Formuła - przykład



10

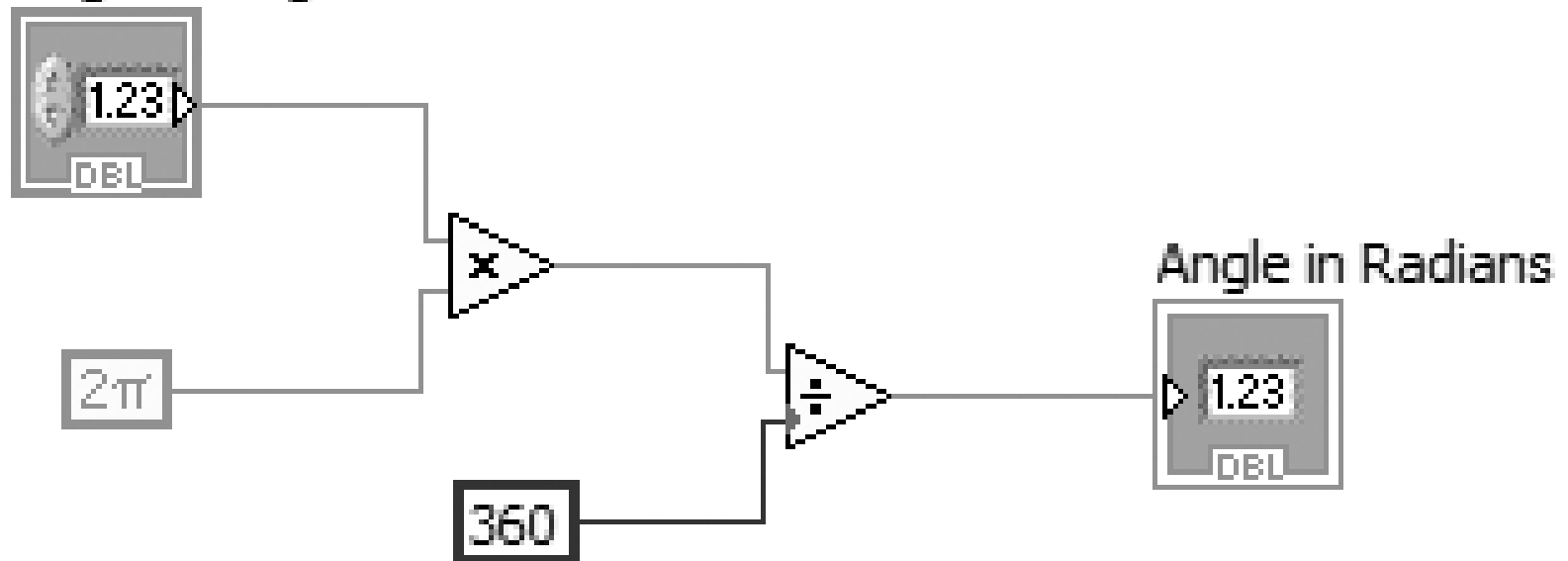
N

i

# Funkcje matematyczne

## Stopnie i radiany

Angle in Degrees

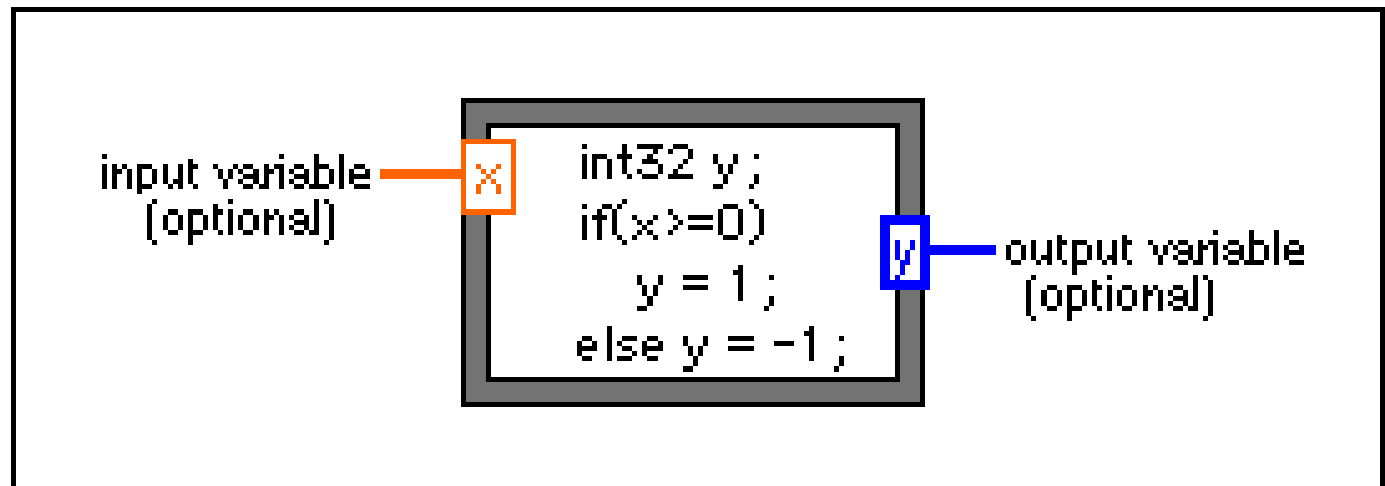




# Formula node

Struktura pozwalająca na wstawianie fragmentu kodu w języku podobnym do C

Wbudowane funkcje: abs, acos, acosh, asin, asinh, atan, atan2, atanh, ceil, cos, cosh, cot, csc, exp, expm1, floor, getexp, getman, int, intrz, ln, lnp1, log, log2, max, min, mod, pow, rand, rem, sec, sign, sin, sinc, sinh, sizeofDim, sqrt, tan, tanh.



# Okno skryptów

Wspomagane jest także inne oprogramowanie – pod warunkiem, że jest zainstalowane na komputerze

Wspomagane narzędzia:

MathScript script node

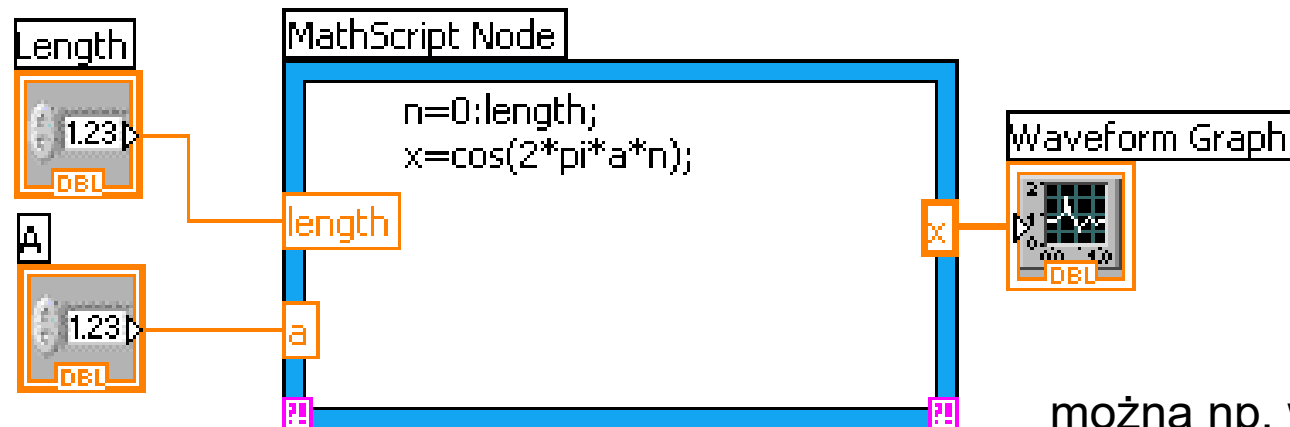
MathSoft software

Mathematica software

MATLAB® software

Maple software

Xmath software

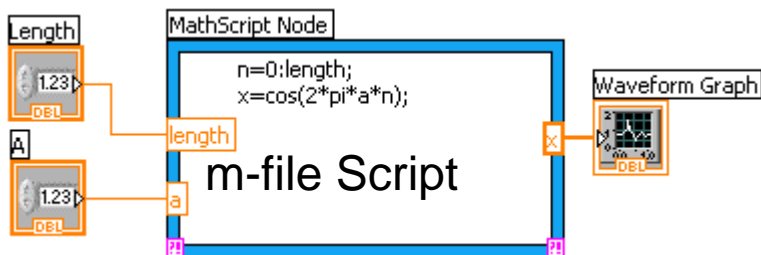


można np. wstawiać kod MatLab'a

# MathScript Window

## Testowanie algorytmów (LabVIEW»Tools»MathScript Window)

- mamy wtedy także dostęp np.  
do panelu MatLab'a



Output Window

-0.00902	-0.95106	-1
-0.95106	-0.80902	-0.58779
-0.30902	-3.1847e-015	0.30902
0.58779	0.80902	0.95106
1	0.95106	0.80902
0.58779	0.30902	-2.455e-016
-0.30902	-0.58779	-0.80902
-0.95106	-1	-0.95106
-0.80902	-0.58779	-0.30902
-3.4296e-015	0.30902	0.58779
0.80902	0.95106	1
0.95106	0.80902	0.58779
0.30902	-3.1847e-015	-0.30902
-0.58779	0.80902	-0.95106
-1	0.95106	-0.80902
-0.58779	0.30902	-3.6745e-015
0.80902	0.95106	1
0.95106	0.80902	0.58779
0.30902	2.4424e-016	-0.30902
-0.58779	-0.80902	-0.95106
-1	-0.95106	-0.80902
-0.58779	-0.30902	-3.9194e-015
0.80902	0.30902	0.58779
0.95106	0.95106	1
0.30902	0.80902	0.58779
-0.58779	4.8916e-016	-0.30902
-1	-0.80902	-0.95106
-0.58779	-0.95106	-0.80902
0.80902	-0.30902	-4.1643e-015
0.95106	0.30902	0.58779
0.30902	0.95106	1

Variables

Partition/Variable	Dimension	Type
Global		
Local		
f	1x1	double array
n	1x201	double array
a	1x201	double array

Variable Workspace

Graphical first? Graph

Legend?

View/Modify Variable Contents

User Commands

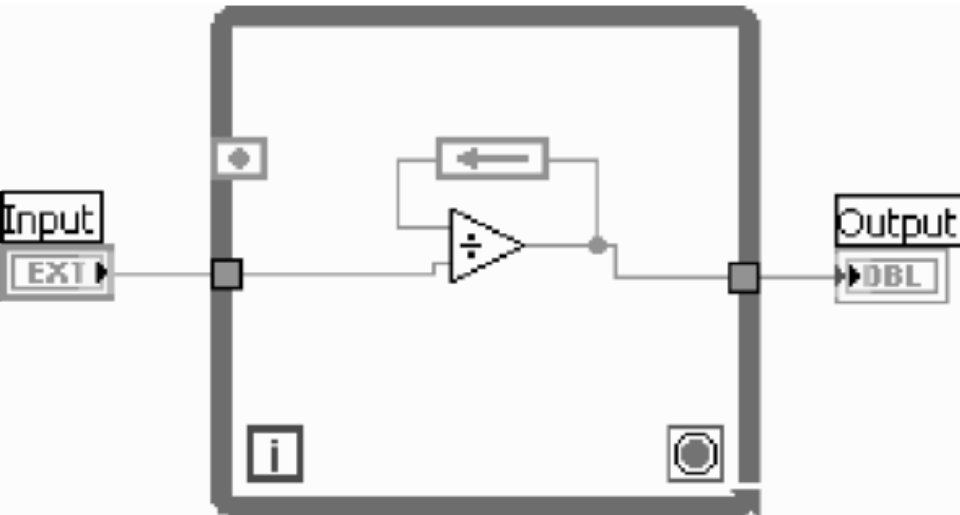
```
x=cos(2*pi*f*n)
```

# Struktury

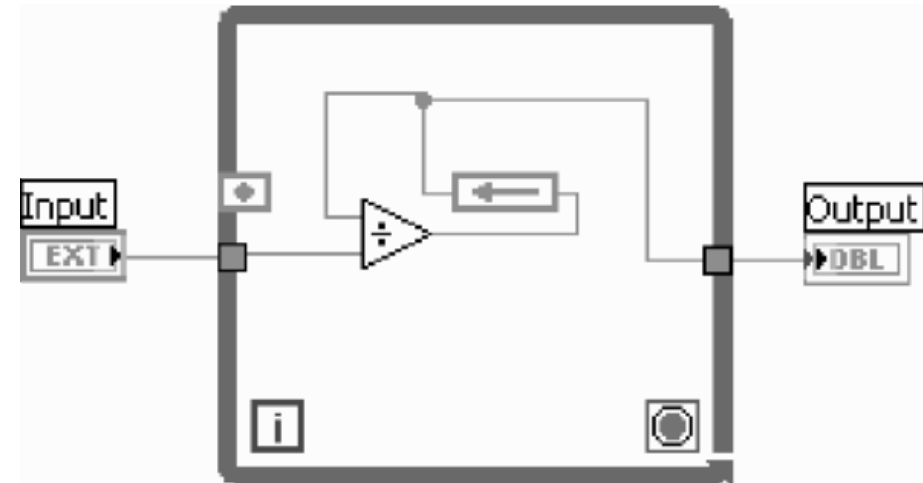
## □ Feedback Nodes

pojawia się automatycznie dla pętli For czy While jeśli wyjście subVI, funkcji czy grupy subVIs są podłączone do tych samych elementów

po



przed



# Zadanie 1

---

Korzystając z express vi „Formula” napisz program liczący wartość wielomianu 3 st:  $y = a + bx + cx^{**2} + dx^{**3}$  (jak w przykładzie na slajdzie 15).

---

# Zadanie 2

---

Korzystając ze struktury „shift register” napisz program liczący  $n!$

- należy wykorzystać indeksator  $i$
  - $n$  powinno definiować ilość iteracji pętli for
  - zwróć uwagę od jakiej wartości zaczyna się  $i$  oraz jaka jest ostatnia wartość  $N$
  - zwróć uwagę na typ danych reprezentujących wynik (może należałoby ograniczyć wartość  $n$ )
-

# Zadanie 3

Korzystając ze struktury „shift register” i poprzedniego zadania napisz program liczący wartość wyrażenia:

$$\sum \frac{1}{(n+2)n!}$$

# Zadanie 4

---

Korzystając ze struktury „shift register” i poprzednich zadań napisz program liczący wartość wyrażenia:

$n!!$

---