



Arkusz kalkulacyjny w praktyce

dr Marcin Bernaś
Katedra Informatyki i Automatyki
Wydział Budowy Maszyn i Informatyki

Arkusz kalkulacyjny

- jest to program komputerowy przedstawiający dane w tabeli składającej się z komórek zawierających głównie liczby, ale także teksty oraz formuły;
- każda komórka posiada swoje unikalne współrzędne będące numerem kolumny (np. A, B) oraz wiersza (np. 1, 2) do której możemy się odwołać (np. =A1 – komórka równa wartości z A1),
- dane można przedstawiać w postaci graficznej - wykresu,
- arkusze umożliwiają także programowanie za pomocą języków programowania i makropoleczeń.

Przykład wyliczenia wartości funkcji kwadratowej

Teksty w
komórkach: a, b, c

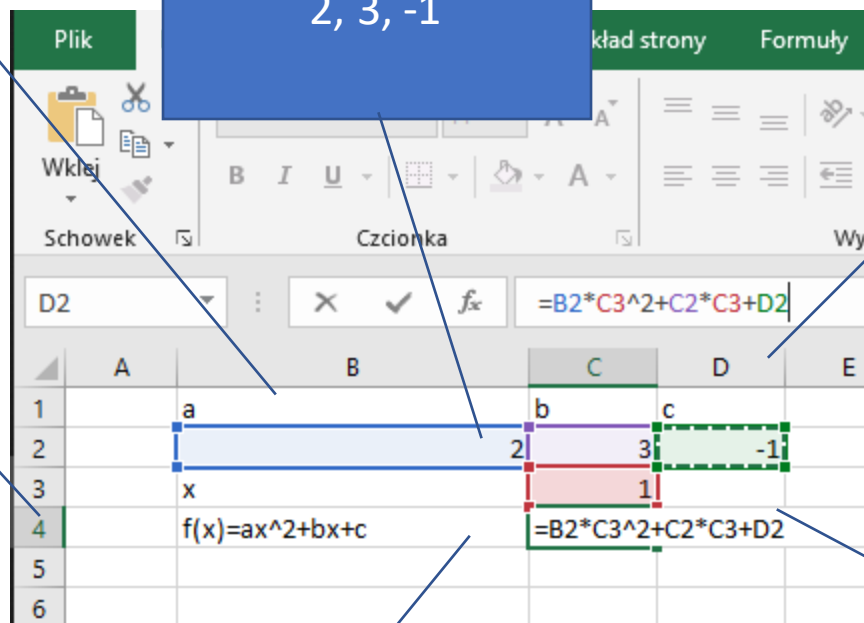
Wartości liczbowe:
2, 3, -1

Numer wiersza 4

Numer kolumny D

Opis funkcji kwadratowej,
którą chcemy wyliczyć

Formuła funkcji kwadratowej którą
chcemy wyliczyć gdzie wartości a, b, c
oraz x wskazane są przez komórki B2,
C2, D2 oraz C3



	A	B	C	D	E
1		a	b	c	
2		2	3	-1	
3		x	1		
4		$f(x)=ax^2+bx+c$	$=B2*C3^2+C2*C3+D2$		
5					
6					

Przydatne informacje

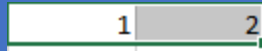
Formuła w komórce zaczyna się od znaku
=

Przy przeciąganiu komórki zawierającej **=A1+1** w prawo komórka zmieni wartość na **=B1+1**

Przy przeciąganiu komórki zawierającej **=A1+1** w dół komórka zmieni wartość na **=A2+1**



Przeciągając jedną cyfrę w prawo lub w dół powielamy ją



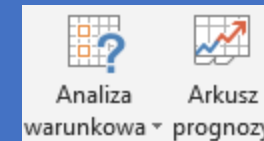
Przeciągając dwie cyfry w prawo lub w dół tworzymy sekwencję

Umieszczając znak \$ przy literze lub cyfrze zapobiegamy jego modyfikacji. Zapisy =A1, =\$A1 oraz =A\$1 są dozwolone

W formułach można stosować operacje dodawania (+), odejmowania (-), mnożenia (*), dzielenia (/) i potęgowania (^)



W formułach można stosować różne funkcje jak *suma*, *średnia* czy funkcje warunkowe *jeżeli*



Jeżeli szukasz wyniku skorzystaj z tych funkcji

Jak wyliczyć wartość funkcji w wielu punktach?

1. Zaznacz dwie komórki z 1,2

2. Złap za kwadracik w prawym dolnym rogu i przeciągnij w prawo

3. Wartości zachowają stałą różnicę między sobą (w przykładzie 1)



	B	C	D	E
a		b	c	
	3	-1	5	
x:	1	2		
f(x)=ax^2+bx+c	C4+\$D\$2			

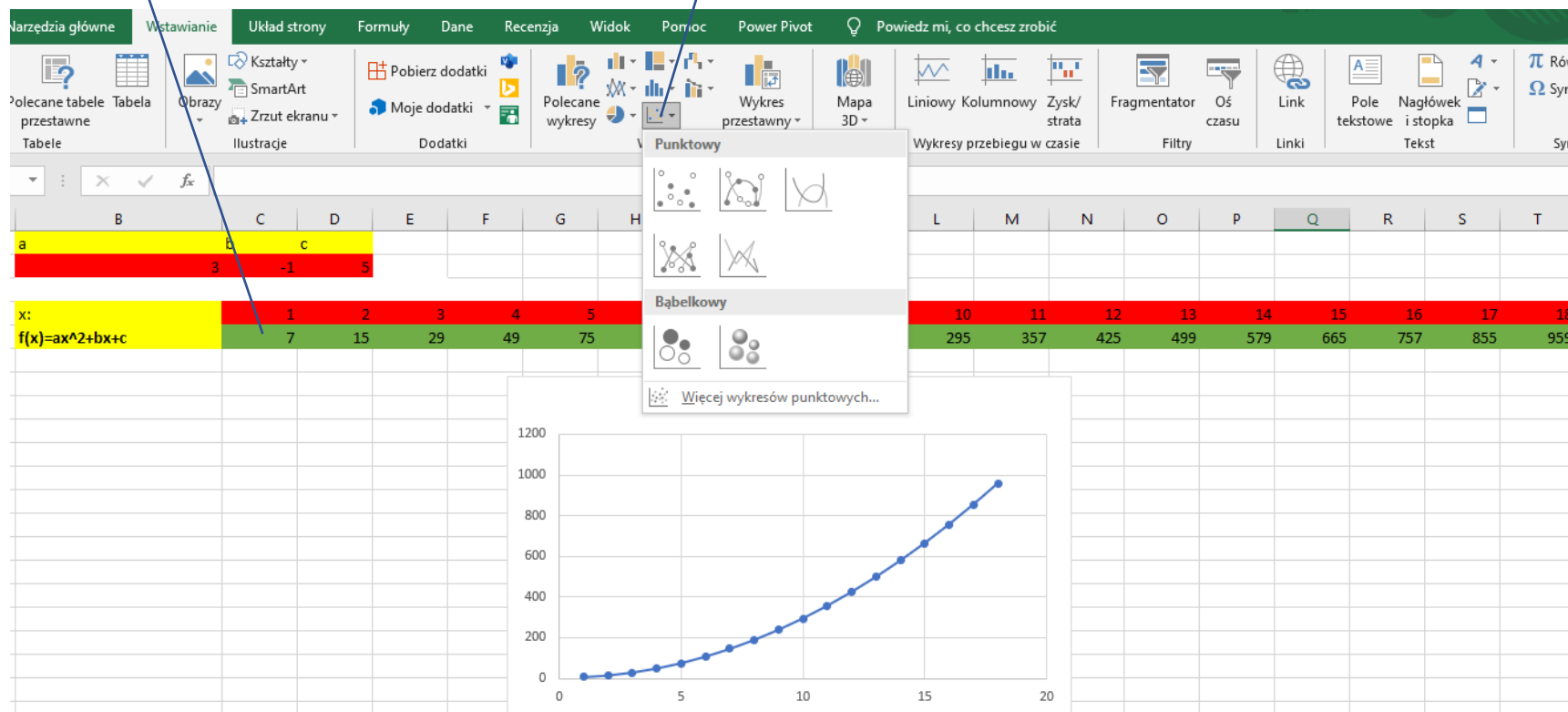
1. Modyfikujemy wartości komórek w formule:
 A) komórki dla wartości stałych jak współczynniki a, b, c zamieniamy na wyrażenia z dolarami np. \$B\$2
 B) wartość liczona dla różnych wartości x nie poprzedzamy wyrażeniem \$

2. Zaznacz formułę. Złap za kwadracik w rogu i przeciągnij w prawo

Wyniki i wizualizacja wartości funkcji

Otrzymujemy wartości funkcji $f(x)$ dla x w przedziale $[1,18]$

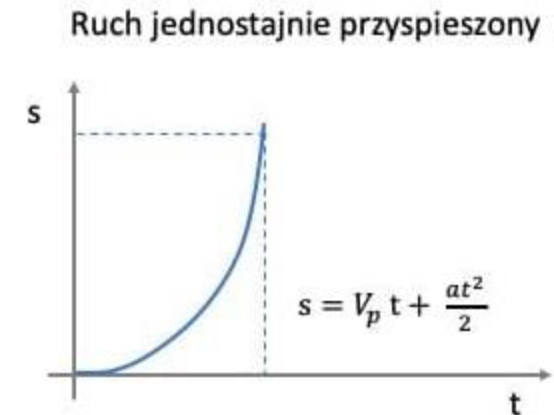
Zaznaczając naszą tabelę i wybierając *wstawienie* -> *wykreś punktowy* możemy obejrzeć wykres funkcji



Zastosowanie

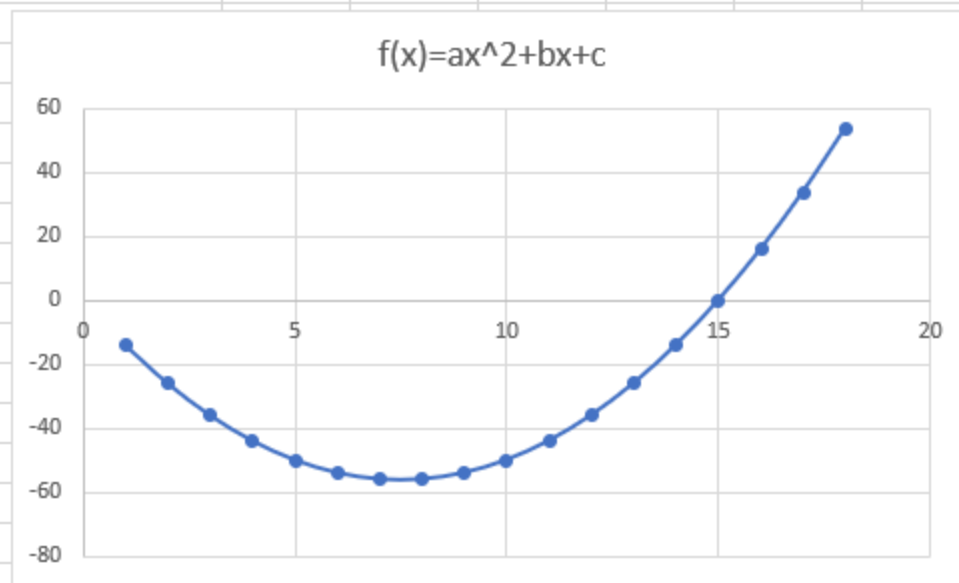
Zmieniając a, b, c możemy analizować dowolne funkcje kwadratowe

Modyfikując formułę można analizować różne zjawiska np. ruch jednostajnie przyspieszony



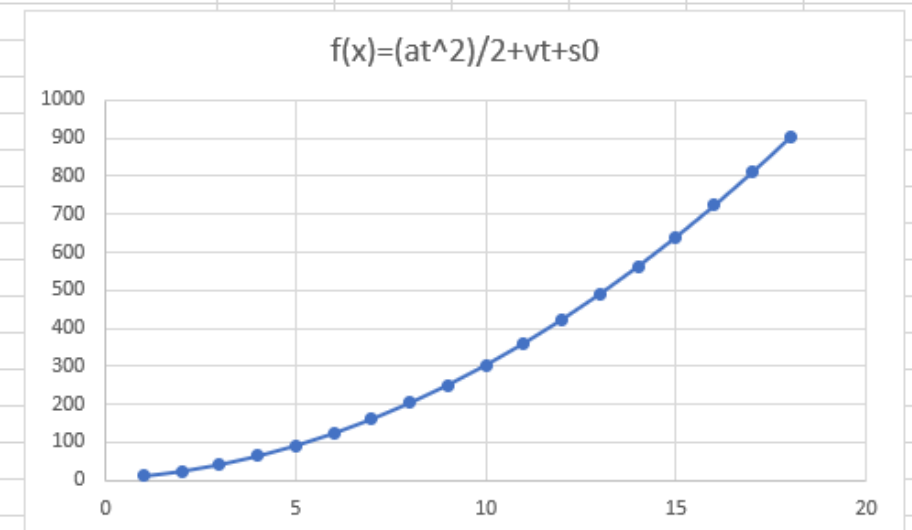
a	b	c
1	-15	0

x:	1	2	3	4	5
$f(x)=ax^2+bx+c$	-14	-26	-36	-44	-50



	B	C	D	E	F	G	H
a		v	s0				
	5	5	3				

t:	1	2	3	4	5	6
$f(x)=(at^2)/2+vt+s0$	10,5	23	40,5	63	90,5	123



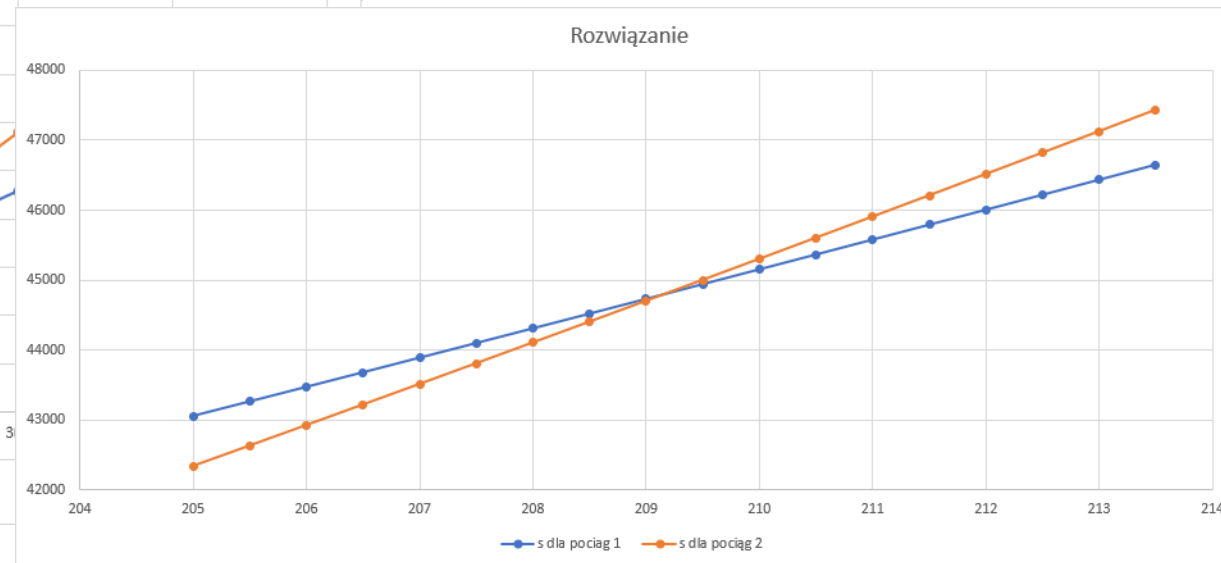
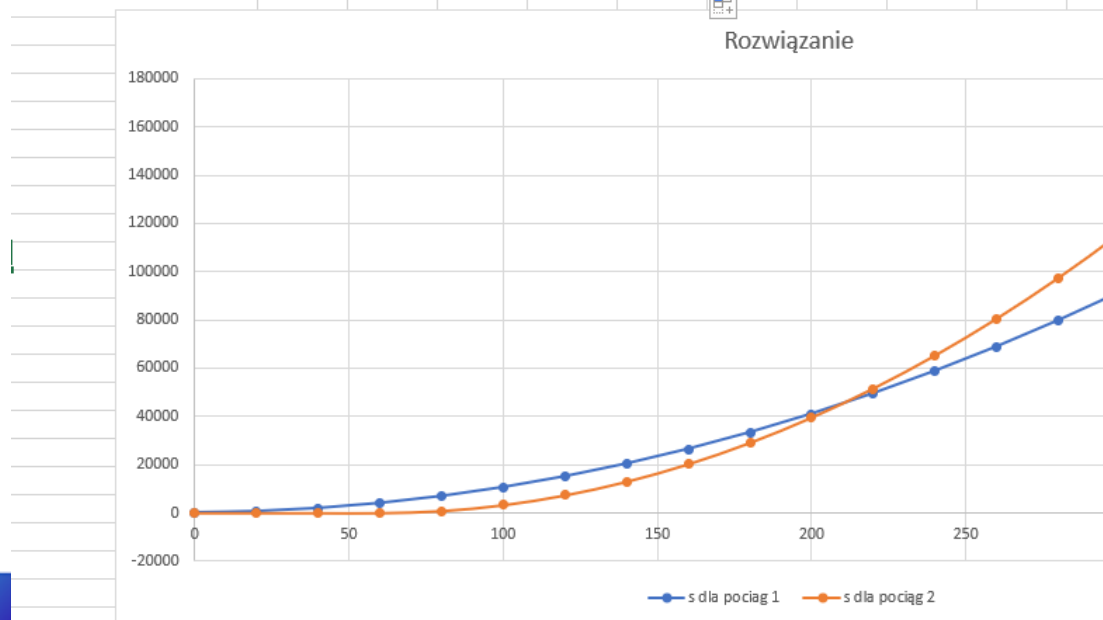
Co z zadaniami?

Pociąg wyjeżdża ze stacji A do B z prędkością $v_1=5\text{m/s}$ oraz przyspieszeniem $a_1=2\text{m/s}^2$. Po 60 sekundach z tej samej stacji wyjeżdża drugi pociąg z prędkością $v_2=1\text{m/s}$ oraz przyspieszeniem 4m/s^2 . Kiedy się spotkają? **Otrzymujemy rozwiązanie przybliżone $\sim 209,2\text{ s}$**

t:	0	20	40	60	80	100	120	140	160	180	200	220	240
s dla pociąg 1	0	500	1800	3900	6800	10500	15000	20300	26400	33300	41000	49500	58800
s dla pociąg 2	0	0	0	0	840	3280	7320	12960	20200	29040	39480	51520	65160

$$= (\$B2 * F\$4^2) / 2 + \$C2 * F\$4 + \$D2$$

$$= (\$B3 * (F\$4 - 60)^2) / 2 + \$C3 * (F\$4 - 60) + \$D3$$



Algorytmy

W arkuszu można w prosty sposób wykonać algorytm o ile nie wymagane są zapętlenia. Przykładem realizacji takiego algorytmu jest znajdowanie miejsc zerowych funkcji kwadratowej.

Algorytm wymaga w oparciu o tabelę z poprzedniego przykładu:

- wyznaczenia delty

$$=C2^2-4*B2*D2$$
- sprawdzenia czy delta jest większa lub równa 0

$$=JEŻELI(D4>=0;"TAK";"NIE")$$
- w przypadku wartości dodatniej wyznaczenie miejsc zerowych

$$=JEŻELI(D5="TAK";(-C2-PIERWIASTEK(D4))/(2*B2);"brak")$$

$$=JEŻELI(D5="TAK";(-C2+PIERWIASTEK(D4))/(2*B2);"brak")$$

Przykład dla różnych a, b, c

Dwa miejsca zerowe

Pierwiastek podwójny

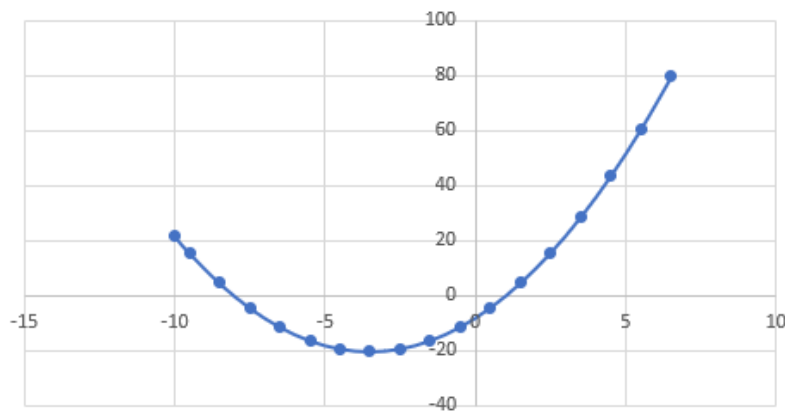
Brak miejsc zerowych

A	B	C
1	7	-8
1.	delta	81
2.	czy delta > 0	TAK
3.	wylicz x1:	-8
	wylicz x2:	1

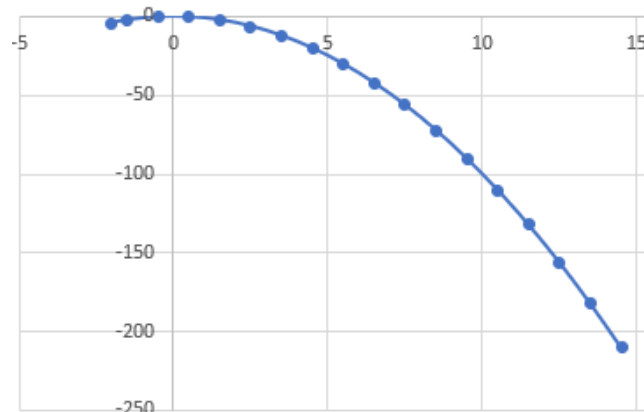
A	B	C
-1	0	0
1.	delta	0
2.	czy delta > 0	TAK
3.	wylicz x1:	0
	wylicz x2:	brak

A	B	C
-1	0	-10
1.	delta	-40
2.	czy delta > 0	NIE
3.	wylicz x1:	brak
	wylicz x2:	brak

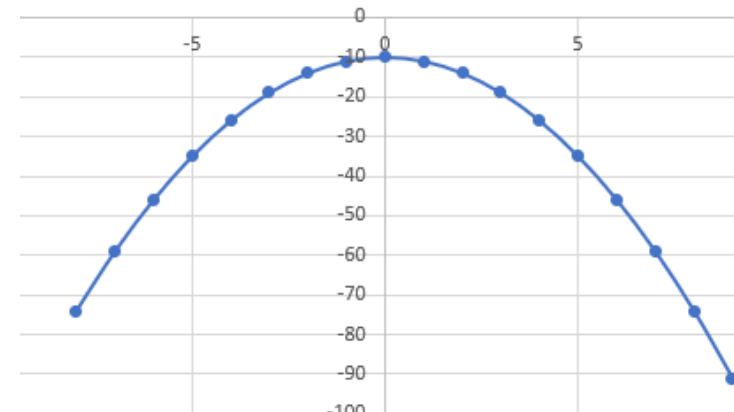
f(x)



f(x)



f(x)



Najczęstsze błędy i jak ich uniknąć

Pamiętaj o kolejności wykonywania działań:
 $2+2*2$ to nie $(2+2)*2$

Nie idź na skróty. Każdy algorytm na swoją ustaloną sekwencję

Pamiętaj o przetestowaniu wszystkich wariantów np. dla 2, 1 i braku pierwiastków.

Jeżeli liczba wariantów jest nieskończona, zbadaj czy działa dla wartości brzegowych

Warto zwizualizować sobie wynik np. wykresem.

Nie komplikuj nadmiernie rozwiązania.

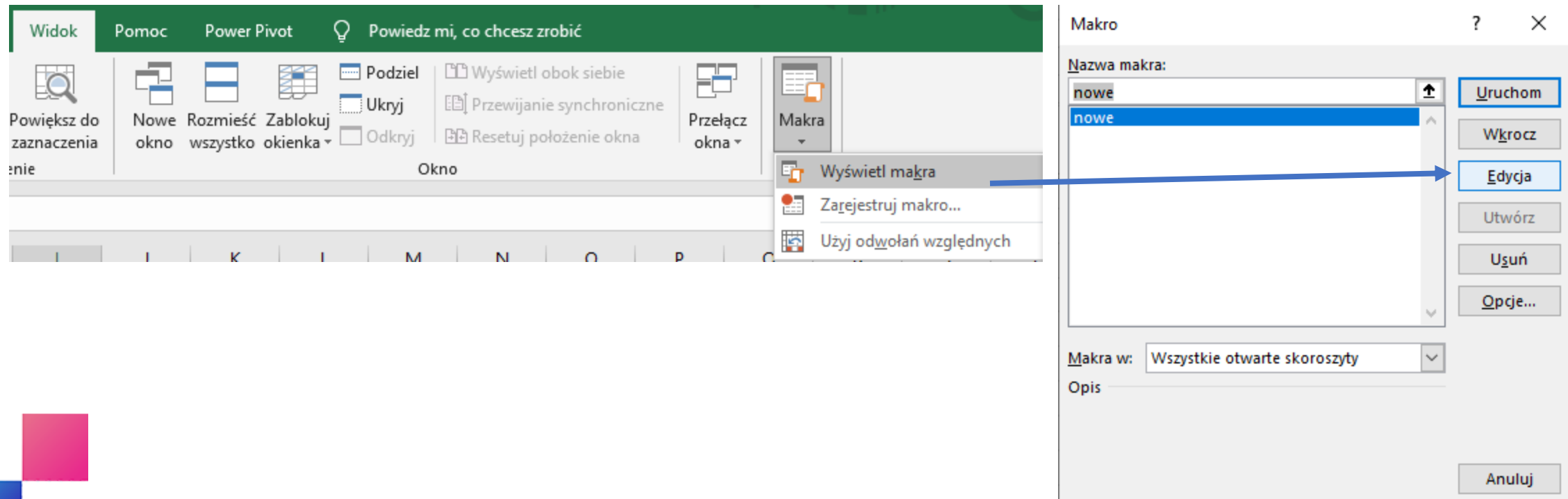
Przy kopiowaniu formuł (przeciąganiu) sprawdź kolejną czy adresy komórek zmieniają się zgodnie z zamierzeniem

Funkcję w ramach formuły można zagnieżdżać:
=jeżeli(A1>10;5;jeżeli(A1<10; ...)

Warto poznawać nowe algorytmy ;)

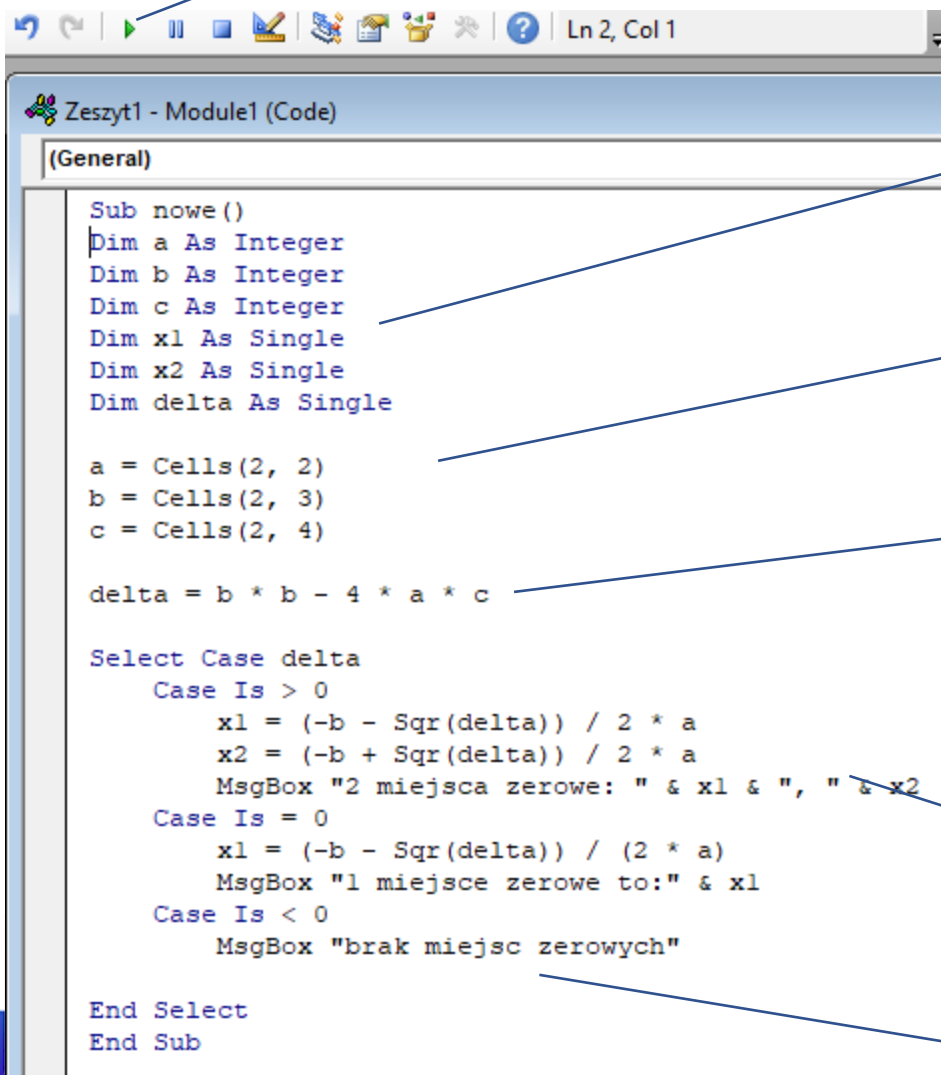
Algorytm można zapisać w postaci makra

W arkuszu można korzystać z dedykowanego języka skryptowego, który oferuje znacznie więcej możliwości. Niestety takie podejście wymaga znajomości składni języka – w tym przypadku Visual Basic for Application. Przykład przedstawiono poniżej:



Przykład

Uruchom makro



```

Sub nowe ()
    Dim a As Integer
    Dim b As Integer
    Dim c As Integer
    Dim x1 As Single
    Dim x2 As Single
    Dim delta As Single

    a = Cells(2, 2)
    b = Cells(2, 3)
    c = Cells(2, 4)

    delta = b * b - 4 * a * c

    Select Case delta
        Case Is > 0
            x1 = (-b - Sqr(delta)) / 2 * a
            x2 = (-b + Sqr(delta)) / 2 * a
            MsgBox "2 miejsca zerowe: " & x1 & ", " & x2
        Case Is = 0
            x1 = (-b - Sqr(delta)) / (2 * a)
            MsgBox "1 miejsce zerowe to:" & x1
        Case Is < 0
            MsgBox "brak miejsc zerowych"
    End Select
End Sub

```

Podobnie jak komórki przechowują wartości tak w skrypcie zmienne a,b,c,x1,x2,delta przechowują wartości typu liczba całkowita (Integer) oraz rzeczywista (Single)

Do zmiennych a, b, c wpisujemy wartości z tabeli. Komórki (Cells) adresowane są numerem wiersza i kolumny (zamiast A=1, B=2, itd.)

Wyliczamy wartość delta

Rozpatrujemy 3 przypadki (Select Case) dla zmiennej delta:

Jeżeli (Case Is) $\text{delta} > 0$ wylicz i wyświetl 2 pierwiastki
 Jeżeli (Case Is) $\text{delta} = 0$ wylicz i wyświetl 1 pierwiastek
 Jeżeli (Case Is) $\text{delta} < 0$ wyświetl brak pierwiastków

MsgBox wyświetla napisy ujęte w cudzysłów („”) oraz zmienne. Do łączenia ich wykorzystuje znak &

Składnia VBA

Dim – definicja zmiennej przechowującej wartość określonego typu

As Integer – liczba całkowita (-1, 0, 1, ..., 10)

As Single – liczba rzeczywista (0.7, -1.02, 5, 10000)

Cells(nr wiersza, nr kolumny) – funkcja zwraca wartość wskazanej przez numer wiersza i kolumny np. Cells(1,1) -> =A1

W VBA podobnie jak w formularzu można korzystać z +, -, *, i /. Znak potęgowania (^) nie zadziała. Znak = umożliwia przypisanie zmiennej wartości np. **a = 1**

Struktura **Select** służy do analizy wielu przypadków

Struktura **If a>0 Then operacja** wykonuje operacje tylko gdy a>0 (warunek spełniony)

W skryptach można wykonywać operacje wielokrotnie np. 10 razy:

For K = 1 To 10
operacja
Next K

Warunki posiadają różną postać równy(=), większy(>), mniejszy(<), większy-równy(>=), mniejszy-równy(<=) oraz różny(<>)

Cells(1,1).Value=1 umieści w komórce A1 wartość 1
Dla przedziału zastosuj **Range** np.
Range(A1:A5).Value=1;

Algorytm Euklidesa

Poniżej przedstawiono algorytm wyliczenia największego wspólnego dzielnika NWD (tzw. algorytmu Euklidesa)

```
Sub NWD()  
Dim a As Integer  
Dim b As Integer  
a = 102  
b = 33  
Dim c As Integer  
Do Until b = 0  
c = a Mod b  
a = b  
b = c  
Loop  
MsgBox "NWD: " & a  
End Sub
```

a, b liczby dla których szukamy NWD

Zmienna na przechowywanie wartości

Powtarzaj pętlę dopóki b nie będzie równe 0

c równe jest reszcie z dzielenia a przez b,
np. $10 \bmod 3 = 3$ reszta **1**

Koniec pętli

Wyświetl wynik NWD

Powodzenia i ... do zobaczenia w październiku w UBB!

