

## MATLAB

### Znaki specjalne

% odtąd komentarz  
 ... w następnej linii będzie kontynuacja  
 : cały zakres współrzędnej  
 . operacja tablicowa (po elementach)  
 ; nie wyświetlać wyniku operacji  
 ' transpozycja (macierzy lub wektora)  
 [] macierz pusta;  
 w wywołaniach funkcji: brak argumentu  
 (parametru aktualnego)

### Tworzenie wektorów i macierzy

$x=[1 \ 2 \ -7]$  lub  $x=[1, \ 2, \ -7]$  wektor wierszowy  
 $y=[1; \ 2; \ -7]$  wektor kolumnowy  
 $A=[1 \ 2; \ -3 \ 5]$  macierz  
 $b=[1:5]$  lub  $b=1:5$  w b kolejne liczby naturalne od 1 do 5  
 $c=[-12:3:5]$  w c liczby całkowite z przedziału  $[-12, \ 5]$  z krokiem 3  
 $D=[-3:2:6;3:7]$  macierz D o wymiarach  $2 \times 5$   
 $\text{zeros}(m,n)$  macierz samych zer o wymiarach  $m \times n$   
 $\text{ones}(m,n)$  macierz samych jedynek o wymiarach  $m \times n$   
 $\text{rand}(m,n)$  macierz liczb pseudolosowych z przedziału  $[0, 1]$  według rozkładu jednostajnego o wymiarach  $m \times n$   
 $\text{zeros}(n), \text{ones}(n), \text{rand}(n)$  forma skrócona dla macierzy kwadratowych  $n \times n$   
 $\text{eye}(n)$  macierz jednostkowa  $n \times n$

### Wybór elementów

$A(2,1)$  element  $a_{21}$  macierzy A  
 $A(1,:)$  1. wiersz macierzy A  
 $D(:, [2 \ 4])$  2. i 4. kolumna macierzy D  
 $E=[A \ [ \ 0; \ -1]; \ x],$   
 $F=[y \ [A(:,2);0] \ [1:3]']$  tworzenie macierzy złożonej

### Działania

$A', x'$  transpozycja macierzy lub wektora  
 $\text{inv}(A)$  lub  $A^{-1}$  odwrotność macierzy kwadratowej  
 $E+F, x+y'$  dodawanie macierzy lub wektorów  
 $E * F$  mnożenie macierzy  
 $x * y$  iloczyn skalarny wektorów  
 $E .* F, x .* y'$  mnożenie tablicowe macierzy lub wektorów  
 $A^2$  potęgowanie macierzy (tutaj:  $A * A$ )  
 $A.^3$  potęgowanie tablicowe (po elementach)

Uwaga! W Matlabie wykonana będzie niedozwolona w matematyce operacja dodania wektora wierszowego do

kolumnowego:

$$[1 \ 3] + \begin{bmatrix} 4 \\ 5 \end{bmatrix}$$

Jest to pewien skrót, bardzo niebezpieczny dla początkujących. Nie stosować!

$\text{sum}(x), \text{sum}(A)$  suma elementów wektora, suma po kolumnach macierzy  
 $[q,i]=\text{min}(x)$  najmniejszy element wektora x; i - jego indeks  
 $[Q,I]=\text{min}(A)$  najmniejsze elementy w kolumnach macierzy A; I - tablica zawierająca numery wierszy w których występują  
 $[q,i]=\text{max}(x),$   
 $[Q,I]=\text{max}(A)$  analogicznie do  $\text{min}(x), \text{min}(A)$  w odniesieniu do największych elementów

### Operatory porównania

$w == v$  w równe v  
 $w \sim v$  w nie równe v  
 $w < v$  w mniejsze od v  
 $w > v$  w większe od v  
 $w \leq v$  w mniejsze lub równe v  
 $w \geq v$  w większe lub równe v

### Operatory i funkcje logiczne

$w | v$  w lub v  
 $w \& v$  w i v  
 $\sim v$  negacja v

### Instrukcja warunkowa i pętle

```
if <warunek>
    <instrukcje>
elseif <warunek>
    <instrukcje>
else
    <instrukcje>
end
```

```
while <warunek>
    <instrukcje>
end
```

```
for <zmienna> = <wektor>
    <instrukcje>
end
```

### Przykłady:

```
for i=1:10
    x(i)=i^2;
end

i=0;
for z=-0.4:0.1:1
    i=i+1;
    x(i)=z^2;
end
```

## Inne pożyteczne polecenia

<code>clear &lt;lista&gt;</code>	usunięcie wymienionych zmiennych; jeśli lista zmiennych jest pusta, usunięte zostaną wszystkie zmienne
<code>who</code>	podanie istniejących zmiennych
<code>disp(x)</code>	wyświetlenie zawartości zmiennej x
<code>clc</code>	wyczyszczenie okna dialogowego
<code>close all</code>	usunięcie wszystkich okien z rysunkami
<code>strcat</code>	konkatenacja stringów

## Tworzenie funkcji

```
function [x,y]=funkcja(a,b)
```

Plik powinien (nie musi) nazywać się 'funkcja.m'.

<code>a, b</code>	argumenty funkcji (parametry wejściowe)
<code>x, y</code>	wartości funkcji (parametry wyjściowe)
<code>return</code>	wyjście z funkcji (domyślnie na końcu pliku *.m)
<code>end</code>	koniec definicji funkcji (gdy jest więcej w pliku)

## Grafika i animacja

<code>plot, plot3</code>	rysunek linii zadanych tablicowo w 2D i 3D
<code>fplot</code>	wykres funkcji jednej zmiennej zadanej analitycznie
<code>axis</code>	sterowanie skalowaniem i wyglądem osi
<code>legend</code>	legenda rysunku
<code>xlabel</code>	opis osi X
<code>ylabel</code>	opis osi Y
<code>hold on off</code>	dodawanie nowych elementów do starego rysunku / rysowanie od nowa
<code>meshgrid</code>	tablice współrzędnych X i Y siatki dla wykresów w 3D
<code>surf</code>	rysunek powierzchni w 3D

### Przykłady:

```
plot(1:3,[2 -1 5])
```

```
fplot(@(x) sin(x),[-3 3])
```

Rysowanie wykresu funkcji  $x \cdot e^{-(x^2+y^2)}$  w 3D dla  $x, y \in [-2, 2]$ .

```
[X,Y] = meshgrid(-2:.2:2, -2:.2:2);  
Z = X .* exp(-X.^2 - Y.^2);  
surf(X,Y,Z)
```

albo:

```
[X,Y] = meshgrid(-2:.2:2, -2:.2:2);  
[Lx,Ly]=size(X);  
for i=1:Lx  
    for j=1:Ly  
        Z(i,j)=fun([X(i,j) Y(i,j)]);  
    end  
end  
surf(X,Y,Z)
```

albo:

```
[X,Y] = meshgrid(-2:.2:2, -2:.2:2);  
Z=arrayfun(@(x,y) fun([x y]),X,Y);  
surf(X,Y,Z)
```

gdzie 'fun.m' to plik następujący:

```
% fun.m  
function fx=fun(x)  
    fx=x(1)*exp(-(x(1)^2+x(2)^2));  
%
```

Zaznaczenie na powyższym wykresie czerwonej kropki w punkcie o współrzędnych  $[1, -1]$ .

```
hold on  
x=1  
y=-1  
z=x*exp(-x^2-y^2)  
plot3(x,y,z,'r.','MarkerSize',25);
```

## Symboliczne liczenie gradientu i hesjanu

### Przykład:

```
syms x1 x2 x3  
x=[x1, x2, x3]  
f=x(1)*x(2)^2+x(3)^4  
grad=jacobian(f,x)  
hes=jacobian(grad,x)
```

albo

```
x=sym('x',[3 1])  
f=funs(x)  
grad=jacobian(f,x)  
hes=jacobian(grad,x)
```

gdzie 'funs.m' to plik następujący:

```
% funs.m  
function fx=funs(x)  
    fx=r1(x)+r2(x);  
end
```

```
function r1x=r1(x)  
    r1x=x(1)*x(2)^2;  
end
```

```
function r2x=r2(x)  
    r2x=x(3)^4;  
end
```