

## **MATLAB**

Matlab jest językiem programowania, w którym zasadniczo występuje jeden typ danej liczbowej, a jest to macierz liczb zespolonych (szczególnym przypadkiem takiej macierzy jest liczba rzeczywista lub naturalna) w związku z tym nie deklarujemy rodzaju zmiennych. Dodatkowo możemy posługiwać się tablicami znaków „string”, którym jednak poświęcimy tutaj mało uwagi. Matlab ukierunkowany jest na wykonywanie obliczeń i ich wygodną prezentację graficzną. Istnieje wiele bibliotek pozwalających na bardzo złożone obliczenia, które nie będą tutaj również omawiane.

### Wybrane operacje na macierzach

W związku z tym, że Matlab jest ukierunkowany na operacje macierzowe posiada bardzo wiele funkcji operujących na macierzach. Wszystkie operacje podstawowe operują na macierzach

#### Definiowanie macierzy (zawsze w nawiasach prostokątnych)

A=[1 2 4.5 6; 2 5 3 7] średnik oznacza koniec wiersza, spacja rozdziela liczby w wierszu

B=[1 3; 4 5; 10 5; 3 6]

C=[1 2 3 ...

5 6 7] trzy kropki oznaczają kontynuację w następnym wierszu

D=[ 1 3 5

2 4 7] brak kropek jest traktowane jako koniec wiersza macierzy

E=[1:5;1:2:10] generuje macierz, w której pierwszy wiersz zawiera kolejne liczby od 1 do 5 (domyślny krok wynosi 1) a drugi wiersz zawiera liczby od 1 do 10 z krokiem 2 (krok może być ułamkiem)

F=eye(3) definiowanie macierzy jednostkowej (oczywiście wymiar 3x3)

G=ones(4) definiowanie macierzy jedynkowej (oczywiście wymiar 4x4)

H= linspace(0, 2,11) generuje 11 równomiernie rozmieszczonych liczb w zakresie od 0 do 2

#### Odwoływanie się do podmacierzy

B=A(i,j) elementy w i-tym wierszu i j-tej kolumnie

C=A(:,j) oznacza odwołanie się do j-tej kolumny

D=A(i,:) oznacza odwołanie się do i-tego wiersza

E=A(a:b,c:d) oznacza odwołanie do podmacierzy zawartej w wierszach od a do b i kolumnach od c do d

Funkcją przydatną jest funkcja w=size(A) która zwraca ilość wierszy i kolumn macierzy A (w –wektor dwu elementowy lub [w1,w2]=size(A), w1 i w2 odpowiednio ilość wierszy i kolumn)

Wybrane operatory macierzowe (wymiar macierzy dla poszczególnych operacji muszą być zgodne z ogólnie znanymi zasadami z matematyki)

C=A+B dodawanie macierzy

D=A-B odejmowanie macierzy

E=A+2 dodanie do każdego elementu macierzy liczby

F=A\*B mnożenie macierzy

G=A\*2 pomnożenie każdego elementu macierzy przez liczbę

H=A' transpozycja macierzy

$I=A/B$  dzielenie macierzy

$J=A^3$  potęgowanie macierzy (możliwe również wykładniki ułamkowe)

Uwaga występują również operatory tablicowe

$C=A.*B$  mnożenie tablicowe, kropka oznacza, że macierz wynikowa będzie zawierała na odpowiednich pozycjach iloczyny odpowiadających elementów w macierzy A i B

$D=A./B$  dzielenie tablicowe (elementy z A dzielone przez elementy z B)

$E=A.\backslash B$  dzielenie tablicowe (elementy z B dzielone przez elementy z A)

$F=A.^3$  potęgowanie tablicowe (każdego elementu tablicy osobno)

Tablice mogą być argumentami większości funkcji:

$A=[\text{linspace}(0, 2, 50); \text{linspace}(0, 5, 50)]$

$B=\text{sqrt}(A)$  zwraca tablicę pierwiastków

$C=\text{sin}(A)$  zwraca tablicę sinusów

Operacje logiczne ( $<, <=, >, >=, =, \sim$  and, or, not)

$C=A \& B$  (równoważne „ $\text{and}(A, B)$ ”) zwraca macierz, w której na odpowiednich pozycjach wartość 1 oznacza, że na odpowiedniej pozycji elementy w macierzy A i B są niezerowe.

$D=A | B$  (równoważne „ $\text{or}(A, B)$ ”) zwraca macierz, w której na odpowiednich pozycjach wartość 1 oznacza, że na odpowiedniej pozycji elementy w macierzy A lub B są niezerowe.

$E\sim A$  (równoważne „ $\text{not}(A)$ ”)

$F=A<B$  zwraca macierz, w której na odpowiednich pozycjach wartość 1 oznacza, że na odpowiedniej pozycji elementy w macierzy A lub B spełniają podany warunek.

Przykłady stosowania

Usunięcie z macierzy elementów mniejszych niż 100

$x=x(x>100)$  zwraca wynik w postaci wektora elementów spełniających ten warunek

Usunięcie liczb nieskończonych z macierzy

$D=D(\text{isfinite}(D))$  zwraca wynik w postaci wektora elementów skończonych

Wybór z macierzy tych kolumn, których trzeci wiersz ma element większy niż 2

$L=x(3,:)>2$

$x=x(:,L)$

lub skrótowo  $x=x(:, x(3,:)>2)$

### Funkcje zaokrąglania

ceil –zaokrąglenie w górę

floor –zaokrąglenie w dół

fix –zaokrąglenie ujemnych w górę a dodatnich w dół

round –zaokrąglenie do najbliższej całkowitej

find(x)- zwraca indeksy niezerowych elementów macierzy

find(x warunek)- (np  $i=\text{find}(x<2)$ ) zwraca indeksy elementów macierzy spełniających zadany warunek.

## Skrypty i funkcje

Skrypt zbiór tekstowy zawierający instrukcje, nie posiada nagłówka, zmienne mają charakter globalny.

Funkcje – fragment programu identyfikowany nazwą, zmienne wewnątrz mają charakter lokalny chyba, że zostaną w specjalny sposób zadeklarowane jako globalne

Sposób definiowania

```
function wartości-funkcji=nazwa-funkcji(par1,...,parN)  
komentarz  
ciąg instrukcji
```

funkcja może być wywoływana bezparametrowo `function []=fun()` (nie pobiera i nie zwraca wartości)

`function [A, B]=oblicz(V,t0,tk);` -deklaracja nagłówka funkcji, wszystkie zmienne mogą być macierzami)

Wywołanie funkcji

`[A, B]=oblicz(V,t0,tk)` (jako parametry wejściowe można wprowadzić także wartości liczbowe)

Operacje na plikach

`dir` -zawartość aktualnego katalogu

`cd` – przejście do innego katalogu

`save` – zapisanie wszystkich zmiennych do pliku (`save nazwa_pliku` (w katalogu domyślnym), `save ścieżka i nazwa_pliku`

`save ścieżka i nazwa_pliku i nazwy zmiennych`

`load` – odczyt danych (analogicznie do `save`)

`path` – pozwala na dodanie ścieżki dostępu (np `path(path,'c:\matlab5')` tworzy nowa lokalizację gdzie automatycznie będą znajdowane skrypty matlaba)

Inne funkcje

`%` znak procent rozpoczyna komentarz w danej linii

`;` - średnik nie jest obowiązkowy na końcu linii, informuje on interpreter czy wynik danej linii ma być wyświetlony na ekranie (brak średnika powoduje wyświetlanie).

`disp('Podaj wymiary macierzy')` wyświetla komentarz na ekranie

`disp(A)` wyświetla zmienną `A` na ekranie

`n=input('Podaj ilość wierszy macierzy A: ');` czyta macierz z klawiatury równocześnie wypisując informacje

Matlab ze względu na operacje macierzowe w większości nie wymaga zastosowania pętli (nie są polecane) jednak są one dostępne

```
while warunek  
instrukcje  
end
```

```
for i=1:n,  
instrukcje  
end
```

## Funkcje graficzne

Rysowanie na jednym wykresie wielu funkcji

```
subplot(m,n,p)
m – liczba wykresów w pionie
n – liczba wykresów w poziomie
p numer wykresu
subplot(2,1,1)
plot(x)
subplot(2,1,2)
plot(y)
```

Istnieją również inne sposoby wywołania tej funkcji.

### Opisywanie wykresów

```
xlabel, ylabel, title,
text(x,y,napis)- x,y współrzędne napisu
figure- tworzy lub aktywuje rysunek
id=figure- tworzy rysunek
id=figure(id)- aktywuje rysunek o identyfikatorze id
close- usuwa obiekt aktywny
close(id)- usuwa obiekt o podanym identyfikatorze
set- pozwala na ustalenie odpowiednich cech obiektu na rodzaj lub grubość linii
axis- pozwala na ustalenie zakresu skali na osiach
figure;
hh2 = plot(T1,pHe,t,y,'w');
set (hh2(1), 'LineStyle','-.');
set (hh2(2), 'LineStyle','-');
axis([0 20 6.6 8.6]);
set (hh2, 'LineStyle','-','LineWidth',2.0); grid;
```

### Trójwymiarowa grafika

```
[X,Y]=meshgrid(1:0.2:2, 1:0.2:2) % tworzenie siatki
Z=X+Y
mesh(X,Y,Z)-wykreślenie wykresu
view zmienia kierunek oglądania aktywnego układu współrzędnych
view(az,el) azymut i elewacja (standartowo az=-37.6 el=30)
```

## Przykłady z komentarzem

### % program wprowadzający dane (skrypt)

```
s=""; % deklaracja pustego znaku
while ~strcmp(s,'t') % sprawdzenie warunku z zastosowaniem funkcji strcmp porównującej ciągi
% znaków
disp('Podaj przedział generacji funkcji') % wyświetlenie tekstu
a=input('Podaj początek przedziału: '); % czytanie z klawiatury wartości do zmiennej
b=input('Podaj koniec przedziału: ');
t=a:0.01:b; %generacja wektora liczb w przedziale od a do b z krokiem 0.01
s=input('Podaj wzór funkcji jednej zmiennej f(t): ','s'); % czytanie nazwy funkcji lub wyrażenia, które
%ma zostać wykonane
x=eval(s,""); % interpretacja odczytanego ciągu i wykonanie operacji matematycznych
h=figure; % stworzenie nowego okna graficznego o uchwycie zapisanym w zmiennej h
plot(t,x); % stworzenie wykresu w aktywnym oknie graficznym
disp('Naciśnij coś');
pause; % zatrzymanie programu
close(h); % zamknięcie okna o uchwycie h
disp('Dziękuję za współpracę');
s=input('Czy chcesz zakończyć t/n ? : ','s'); % sprawdzenie warunku zakończenia pętli
end % słowo kluczowe kończące zakres pętli.
```

### % program macierz

```
% program napisany w stylu języka C, ponieważ w Matlabie całą macierz można wprowadzić jedną
% instrukcją
s="";
A=[]; % deklaracja pustej macierzy
B=[];
while ~strcmp(s,'t')
disp('Podaj wymiary macierzy')
n=input('Podaj ilość wierszy macierzy A: ');
p=input('Podaj ilość kolumn macierzy A: ');
m=input('Podaj ilość wierszy macierzy B: ');
l=input('Podaj ilość kolumn macierzy B: ');
for i=1:n,
    for j=1:p, %stosowanie pętli for
        st=['Podaj element macierzy A(',num2str(i),num2str(j),'): '];
        disp(st)
        A(i,j)=input("");
    end
end
for i=1:m,
    for j=1:l,
        st=['Podaj element macierzy B(',num2str(i),num2str(j),'): ']; % konwersja liczby na znaki aby
% można wyświetlić cały tekst w jednej linii
        disp(st)
        B(i,j)=input("");
    end
end
A=A % instrukcja nie zakończona średnikiem powoduje wyświetlenie jej wyniku na ekranie
B=B
s=input('Podaj wzór wykonywanej operacji macierzowej A operator B: ','s');
C=eval(s,"")
disp('Naciśnij coś');
pause;
disp('Dziękuję za współpracę');
s=input('Czy chcesz zakończyć t/n ? : ','s');
end
```

```
% Program obliczający dynamiczny przebieg C(t) napisany w stylu Matlab - brak pętli
% przykład funkcji, która nie zwraca parametrów
% przykład wywołania oblicz(1,2,0.1,0.2,0,10) (nazwa pliku powinna być taka sama jak funkcji)
% V - objętość zbiornika [m3] % Vg - przepływ objętościowy
% C0 - stężenie początkowe w reaktorze % C - stężenie na wejściu reaktora
function []=oblicz(V,Vg,C,C0,t0,tk);
global Vg V C % deklaracja zmiennej globalnej w funkcji
[T,X]=ode45('rown1',t0,tk,C0,0.000001); %funkcja standartowa do obliczania równań różniczkowych
clear global Vg V C % usunięcie zmiennej globalnej w funkcji
h=figure;
plot(T,X);
disp('Naciśnij coś');
pause;
close(h);
%% Koniec programu
```

#### % Funkcja rown1

```
function [Dx]=rown1(t,x)
global Vg V C
Dx=(Vg/V)*(C-x);
```

#### % Program obliczający dynamiczny przebieg C(t) napisany w stylu C – zastosowanie pętli for

```
function [C1,T]=obliczaj(V,Vg,C,C0,dt,t0,tk);
%Funkcja obliczająca przebieg czasowy stężenia soli w zbiorniku
% Sposób wywołania: obliczaj(V,Vg,C,C0,dt,t0,tk) lub [a,b]=obliczaj(V,Vg,C,C0,dt,t0,tk)
% Gdzie:
% V - objętość zbiornika [m3] % Vg - przepływ objętościowy
% C0 - stężenie początkowe w reaktorze % C - stężenie na wejściu reaktora
W=get(0,'ScreenSize'); % odczytanie rozdzielczości monitora
C1=[C0];
tb=t0+dt;
T=[t0,tb];
c2=(Vg/V)*(C-C0)*dt+C0;
C1=[C1,c2];
n=round((tk-t0)/dt);
for i=2:n
    c2=(Vg/V)*(C-c2)*dt+c2;
    C1=[C1,c2];
    tb=tb+dt;
    T=[T,tb];
end;
temp2=figure('menubar','none','numbertitle','off','name',...
' Wykres ', 'position',W,'color',[0 0 0]); % ustalenie niektórych parametrów okna graficznego
h=plot(T,C1);
axis('on');
xlabel('t');
ylabel('C');
title('Wykres czasowy');
set(h,'lineStyle','-','lineWidth',2.0); % odwołanie się do obiektu o uchwycie h i ustalenie stylu i
% grubości linii
grid on % włączenie siatki na wykresie
```

Zadania do wykonania

1. Program obliczający wartości  $x$ ,  $x^2$ ,  $x^3$  dla liczby rzeczywistej  $x$  w przedziale  $[x_p, x_k]$  i zmieniającej się z krokiem  $dx > 0$ . Program należy napisać korzystając z operacji macierzowych ( bez użycia pętli programowych). Wyniki należy przedstawić na wykresie.

2. Program pozwalający na łączenie podanych trzech macierzy w następujący sposób:

a)  $\begin{bmatrix} A & B & C \end{bmatrix} = \begin{bmatrix} D \end{bmatrix}$

b)  $\begin{bmatrix} A \\ B \\ C \end{bmatrix} = \begin{bmatrix} D \end{bmatrix}$

c)  $\begin{bmatrix} A & C \\ B & \end{bmatrix} = \begin{bmatrix} D \end{bmatrix}$

Jeżeli macierze mają niezgodne wymiary to należy je dostosować uzupełniając wierszami lub kolumnami zerowymi. Macierze wejściowe i macierz wynikową wyświetlić na ekranie.

3. Program wczytujący macierz i obliczający sumy poszczególnych wierszy oraz kolumn.

a) Obliczyć sumę wszystkich liczb dodatnich i ujemnych w macierzy.

b) Obliczyć średnią liczb dodatnich i ujemnych.

c) Transponować podaną macierz

4. Program obliczający wartości

a) funkcji  $y = \sqrt[3]{\sqrt{x-1} - \sqrt{ax^2 + 9} + c}$  i wypisujący rozwiązania na ekranie

c=0 x=1 a=-1 y=-1.41

c=0 x=0 a=1 brak rozw.

c=0 x=5 a=-1 brak rozw.

c=3 x=1 a=0 y=0

b) dla funkcji z punktu a narysować wykres w podanym przedziale  $[x_p, x_k]$  jeżeli w tym przedziale są punkty, dla których brak rozwiązania to należy przyjąć wartość 0 w tych punktach.