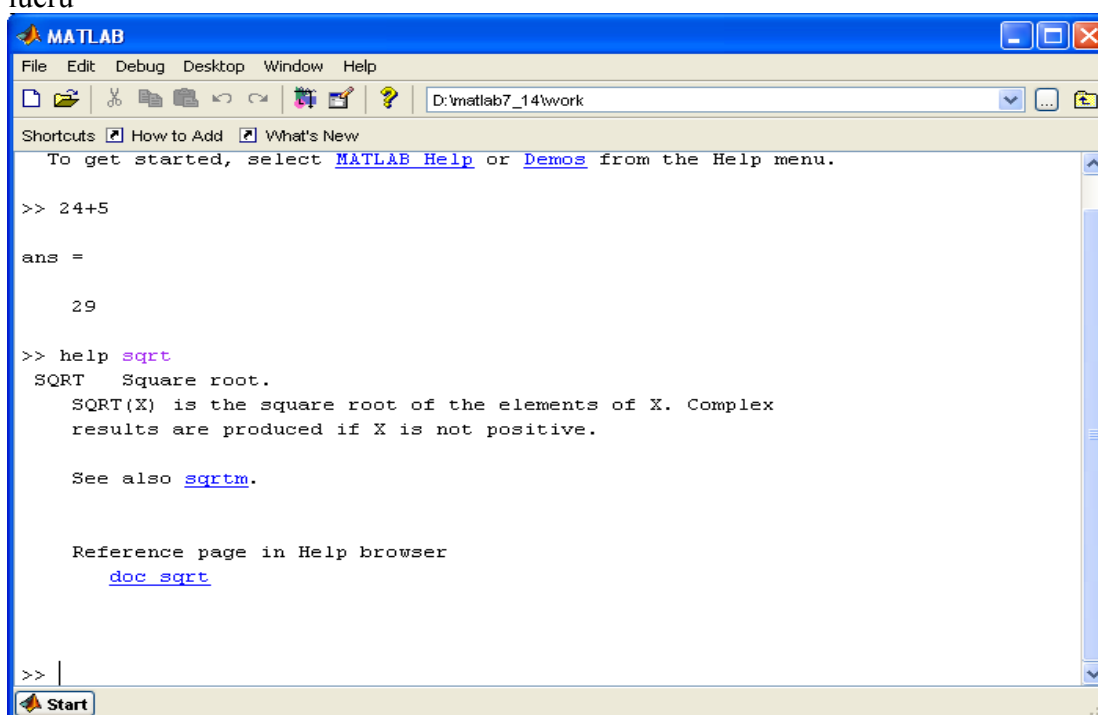


BAZELE PROGRAMĂRII ÎN MEDIUL MATLAB

1. Noțiuni generale

Sistemul de programare **MatLab** (Matrix Laboratory) reprezintă un mediu interactiv de calcul a problemelor ingineresti și științifice având ca un singur tip de date - **matricea**. Programele elaborate în mediul MatLab se salvează în fișiere cu extensia **.m**, numite **M-fișiere**. MatLab este dotat cu un puternic aparat al matematicii de calcul și de reprezentare grafică. Pentru diferite aplicații tehnice MatLab conține pachete specializate numite **Toolbox**-uri cu destinație în domeniul respectiv. Sistemul permite schimbul informației cu editorul de texte Microsoft Word, în particular, a textului programului și rezultatelor textuale, a graficelor.

Lansarea sistemului MatLab are loc cu clic dublu pe iconița și apare fereastra de lucru



Linia care conține caracterul **>>** se numește **linia de comenzi**. După acest caracter poate fi culeasă o comandă sau expresie. Dacă se culege o expresie matematică atunci MatLab-ul poate fi utilizat în regim de calculator. De exemplu, cu expresia

>>24+5

are loc calcularea sumei (la apăsarea tastei ENTER) și cu variabila de sistem **ans** se afișează rezultatul **ans = 29**. Cu comanda **>>help sqrt** se lansează sistemul de ajutorare MatLab, concret asupra funcției standard **sqrt**.

Dacă se dorește executarea comenzii sau a expresiei fără afișare atunci comanda se finalizează cu caracterul „;”. Identificarea variabilelor se realizează în mod tradițional ca o succesiune de litere latine, cifre și caracterul „_”, obligatoriu începându-se cu o literă.

Atenție! În MatLab literele mari și mici nu sunt identice

Ștergerea ferestrei de comenzi – prin executarea comenzii **clc**.

Funcții MATLAB de interes general

- **help** – furnizează informații despre MATLAB și funcțiile acestuia.

Sintaxă:

help nume – furnizează informații despre nume (poate fi un nume de funcție sau un nume de director).

Exemplu:

help fft – furnizează informații despre transformata Fourier discretă.

- **who** – listează numele variabilelor din spațiul de lucru.
- **whos** – furnizează informații suplimentare referitoare la variabilele din spațiul de lucru (nume, dimensiune etc.).
- **format** – stabilește formatul extern de afișare al numerelor pe ecran.

Sintaxa:

format opțiune – parametrul opțiune poate fi:

- short – 5 cifre // formatul implicit
- long – 15 cifre
- short e – 5 cifre + exp (puteri ale lui 10)
- long e – 15 cifre + exp (puteri ale lui 10)

Exemple:

format short

x=pi → x =

3.1416

format long

x → x =

3.14159265358979

O matrice poate fi introdusă prin atribuire, de exemplu cu expresiile

```
>> a=[1 2 3; 4 5 6]
```

```
>> x=[1 -2 5 -4]
```

se formează matricea **a** cu două linii și trei coloane și vectorul-linie **x** cu patru elemente. Elementele matricei sunt incluse în paranteze pătrate. Elementele unei linii se separă cu spațiu sau virgulă, iar liniile – cu caracterul „;”. Pentru selectarea unor elemente se utilizează indecșii incluși în paranteze rotunde și separați prin virgule, de exemplu, **a(k,3)** determină elementul al treilea din linia cu indicele k, iar **a(:,3)** – coloana a treia în întregime.

În MatLab pot fi utilizate atât mărimi reale, cât și mărimile complexe cu caracterele i sau j în calitate de unitate imaginară:

```
>> a=1+2i      a = 1.0000 + 2.0000i
```

```
>> b=1-3i      b = 1.0000 - 3.0000i
```

Pentru determinarea lungimei vectorului se utilizează funcția **length** :

<pre>>> k=length('Mesajul meu') k = 11</pre>	<pre>>> X=[1 0 -3 6 7 13]; >> k=length(X) k = 6</pre>	<pre>>> k=length([1 4 7 ; 2 0 -2]) k = 3</pre>
--	--	--

iar dimensiunile matricei - funcția **size** :

<pre>>> X=[1 0 -3 ; 6 7 13] >> k=size(X) k = 2 3</pre>	<pre>>> [m, n]=size(X) m = 1 n = 6</pre>	<pre>>> size([2 4 7]) ans = 1 3</pre>
---	--	---

2. Tipuri de operații MatLab

Operații aritmetice

Denumire	Notatie	Exemplu	Notă
Adunare	+	A+B	
Scădere	-	A-B	
Împlicare vectorială	*	A*B	Se realizează conform regulilor prelucrării vectoriale
Împărțirea vectorială la stânga (B la A)	\	A\B	
Împărțirea vectorială la dreapta (A la B)	/	A/B	
Ridicare la putere vectorială	^	A^k	
Transponare vectorială	'	A'	
Multiplicare scalară	.*	A.*B	Se prelucrează element cu element
Împărțirea vectorială la stânga (B la A)	.\	A.\B	
Împărțirea vectorială la dreapta (A la B)	./	A./B	
Ridicare la putere vectorială	.^	A.^k	
Transponare vectorială	.'	A.'	

Operații relaționale			Operații logice		
Denumire	Notatie	Exemplu	Denumire	Notatie	Exemplu
Mai mare	>	A > B	Logică NOT	~	~A
Mai mic	<	A < B	Logică AND	&	A & B
Mai mare și egal	>=	A >= B	Logică OR		A B
Mai mic și egal	<=	A <= B			
Identic egal	==	A == B			
Deferit	~=	A~=B			

3. Funcții matematice MatLab

Denumire	Notatie	Exemplu	Notă
Modulul	abs (X)	abs(-13)=13	
Unghiul	angle (X)	>> angle(3+4i) ans = 0.9273	Complexul $X=a+bi$ se reprezintă în forma $r \cdot e^{ij}$, unde $a = r \cos j$, $b = r \sin j$
Partea reală și imaginară	real (X) imag (X)		
Conjugată numărului complex	conj (X)	>>conj(2+3i) ans = 2.00-3.00i	
Funcții de rotundire	ceil (X)	>>ceil(2.4) ans =3	Rotungește la cel mai apropiat întreg spre +∞
	fix (X)	>>fix(2.4) ans = 2	Trunchiază partea zecimală
	floor (X)	>>floor(2.6) ans = 2	Rotungește la cel mai apropiat întreg spre -∞
	round (X)	>>round(2.6) ans = 3 >>round(2.4) ans = 2	Rotungește la cel mai apropiat întreg
Restul împărțirii întregi	mod (X,Y)	>> mod(7,2) ans=1	
Semnul argumentului	sign (X)		
Rădăcină pătrată	sqrt (X)	>> sqrt(5) ans = 2.2361	
Exponenta e^x	exp (X)	>> exp(1) ans = 2.7183	$e^{x+iy} = e^x(\cos y + i \sin y)$
Exponenta binară 2^x	pow2 (X)		
Logaritm natural, în baza 2, în baza 10	log (X) log2 (X) log10 (X)		X>0
Funcții trigonometrice	sin (X) cos (X) tan (X) cot (X) csc (X) sec (X)		Argumentul X se exprimă în radiani !!!!

Funcții trigonometrice inverse	asin(X) acos(X) atan(X) acot(X) acsc(X) asec(X)	
Funcții hiperbolice	sinh(X) cosh(X) tanh(X) coth(X) csch(X) sech(X)	$sh(X)=(e^X-e^{-X})/2$, $ch(X)=(e^X+e^{-X})/2$
Funcții hiperbolice inverse	asinh(X) acosh(X) atanh(X) acoth(X) acsch(X) asech(X)	

O expresie matematică include operanzi, operații aritmetice, funcții standard și paranteze rotunde.

4. Regimuri de programare

Trecerea în regimul MatLab de programare selectăm meniul **File** din fereastra de comenzi. La această comandă editorul propune două opțiuni: **New** (crearea unui nou M-fișier) sau **Open** (deschiderea fișierului existent cu extensia **.m**).

Există două feluri de M-fișiere: *M-scenariu (sau script)* și *M-funcție*.

M-scenariu – reprezintă un fișier care conține o succesiune de instrucțiuni și comentarii (linii care se încep cu caracterul %). Fișierul *M-funcție* poate avea parametri de intrare și ieșire, inclusiv și variabile interne locale. *M-funcție* poate fi apelată din alte programe, se include în biblioteca de funcții ale sistemului în formă de fișiere textuale.

Antetul *M-funcției* are forma:

function [*<parametrii_ieșire>*]= *<nume_funcție>* (*<parametri_intrare>*)

Atenție! Numele fișierului trebuie să fie identic cu numele funcției

De exemplu, funcția de calcul a ariei și perimetrului dreptunghiului cu laturile *a* și *b* se programează în fișierul *drept.m*

```
drept.m
function [s,p]=drept(a,b)
% Calcul arie și perimetru
% al dreptunghiului cu laturile a și b

s=a*b;
p=2*(a+b);
```

Pentru setarea valorilor variabilor *a* și *b* elaborăm următorul fișier script

```
cdrept.m
% Setarea valorilor laturile a și b
% apelul funcției drept
a=5;
b=6;

[s,p]=drept(a,b);      %Apelul funcției

rez=[s p];      % Formarea vectorului cu rezultate de
calcul

disp(rez);      % Afisare rezultat
```

Lansarea fișierului script se realizează prin culegerea denumirii fișierului (cu litere mici și fără extensia **.m**) în linia de comenzi:

```
>> cdrept
      30      22
```

Din variabilele *s* și *p* a fost creat vectorul rezultat *rez*. Afisarea valorilor vectorului *rez* se realizează cu comanda MatLab **disp**.

Fișierul script poate fi apelat direct din linia de comenzi indicând valorile mărilor de intrare în partea dreaptă a funcției:

```
>> [s,p]=drept(3,5)

s=15
p=16
```

5. Programarea structurilor algoritmice

Instrucțiunea de organizare a salturilor condiționate are una din următoarele forme:

if < condiție > <comenzi> end	if < condiție> < comenzi> else < comenzi> end	if < condiție> < comenzi> elseif < comenzi> < comenzi> else < comenzi> end
---	--	--

În lor de condiție se utilizează orice expresie logică construită în baza operațiilor relaționale și logice.

Ciclu cu numărul de repetări cunoscut poate fi organizat cu instrucțiune **for** în felul următor:

```
for V=ValInit:Pas:ValFin  
    <comenzi>  
end
```

unde ValInit, Pas și ValFin – reprezintă valoarea inițială, pasul și valoare finală a parametrului ciclului. Implicit Pas are valoarea 1 și poate fi omis. Se admit cicluri incluse:

<pre>for i=1:n for j=1:m a(i,j)=x(i)^j; end end</pre>	<pre>for i=1:n-1 for k=i+1:n if a(i) < a(k) m=a(i) a(i)=a(k) a(k)=m end end end</pre>
---	--

În antetul ciclului poate fi și un vector, de exemplu, construcția

```
k=1;  
for i=[0 5 7]  
    x(k)=2^i;  
    k=k+1;  
end
```

formează vectorul $X=[1 \ 32 \ 128]$.

Ciclu cu test initial are construcția:

```
while <condiție>
    <comenzi>
end
```

și asigură executarea comenzilor din ciclu atât timp cât condiția este adevărată. Ciclu poate întrerupt cu instrucțiunea **break**:

```
while a<1
    n=n+1
    if n>250
        break
    end
    ...
end
```

Instrucțiunea comutator are construcția:

```
switch <expresie>
case <valoarea 1>
    <comenzi>
case < valoarea 2>
    <comenzi>
    .....
otherwise % poate lipsi
    <comenzi>
end
```

Introducerea datelor de la tastatură se realizează cu instrucțiunea
 $\langle \text{variabila} \rangle = \text{input}(\text{'mesaj'})$

6. Elemente de grafică MatLab

Pentru trasarea graficului unei funcții se utilizează instrucțiunea

plot(x,y,scg)

unde **x** este vectorul argumentului, iar **y** -vectorul funcției, **scg** – este șir de caractere care specifică tipul și culoare curbei. Trasarea mai multor grafice pe ecran se setează cu instrucțiunea

subplot(ngv,ngo,ngc)

unde **ngv** – este numărul de grafice pe verticală, **ngo** – este numărul de grafice pe orizontală, **ngc** – este numărul graficului curent în ordine crescândă.

<pre>ygfrax.m % Procesare unui vector x=0:pi/100:2*pi; y=sin(x); z=cos(x).^2; subplot(121); plot(x,y,'g'); grid; title('Funcția y=sin(x)'); xlabel('x');</pre>
--

```
ylabel('y');  
  
subplot(122);  
plot(x,z,'r');  
grid;  
title('Functia z=cos²x');  
xlabel('x');  
ylabel('z');
```

În acest program se generează vectorul x cu valori de la 0 până la 2π cu pasul $\pi/100$. Pentru valorile vectorului x se calculează funcțiile y și z . Pentru fiecare funcție graficul se construiește într-o fereastră aparte. Cu instrucțiunea **grid** pe grafic se trasează linii verticale și orizontale. Cu **title** se definește titlu, iar cu **xlabel** și **ylabel** se marchează axele graficului.