LABORATORUL NR. 1

MATLAB (abrevierea de la ***Mat****rix* ***Lab****oratory*) este un mediu interactiv, utilizat pentru calcule ştiinţifice şi inginereşti care permite efectuarea de calcule numerice, reprezentari grafice, prelucrari de date, modelari si simulari, etc.

Elementul de bază cu care operează MATLAB-ul este **matricea**.

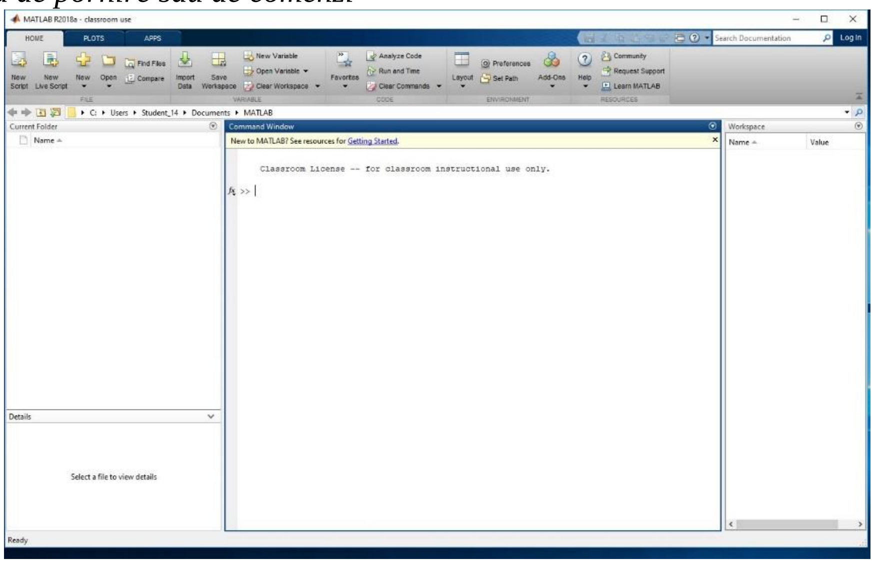
1.1. <u>Lansarea în execuţie</u>

Matlab-ul rulează sub Windows, iar apelarea lui se face selectand cu mouse-ul pictograma Matlab sau indicându-i calea.

La iniţializare, se afişează cursorul mediului, constând în simbolul >>. În dreptul acestuia se pot scrie liniile de comandă.

1.2. <u>Ferestrele de lucru</u>

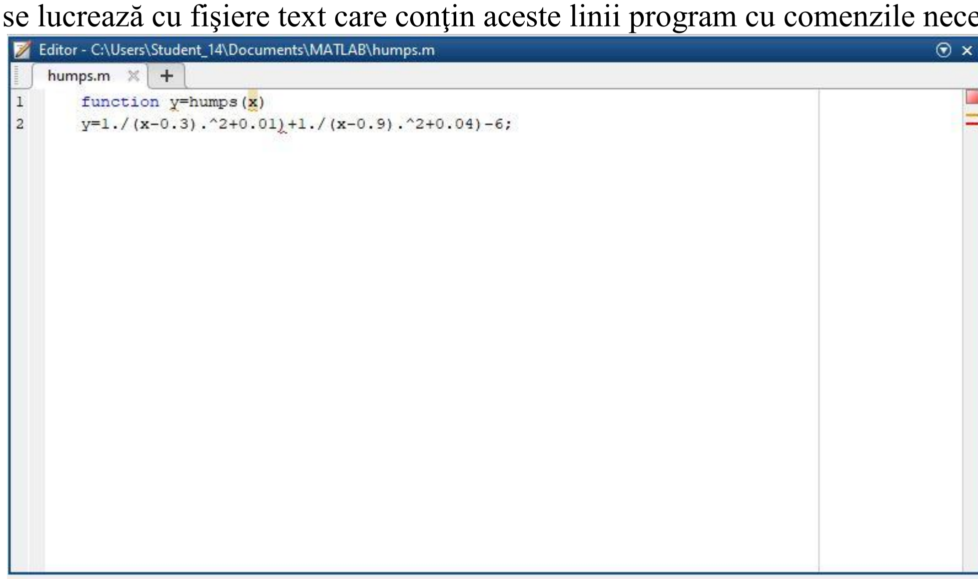
a) Fereastra de pornire sau de comenzi



În această fereastră fiecare linie de comandă este prelucrată imediat şi rezultatul afişat.

b) Fereastra de editare

Deoarece in fereastra de comenzi, instrucţiunile se introduc linie cu linie, s-a dezvoltat fereastra de editare in care se lucrează cu fişiere text care conţin aceste linii program cu comenzile necesare.



În fereastra de editare se creează fisiere cu programe. Fişierele care conţin instrucţiuni Matlab se numesc fisiere M pentru ca au extensia „.m”. Un astfel de fişier constă într-o succesiune de instrucţiuni Matlab cu posibilitatea apelării altor fişiere M şi a apelării recursive.

Fereastra de editare se deschide prin intermediul secvenţei: File → New → M-file.

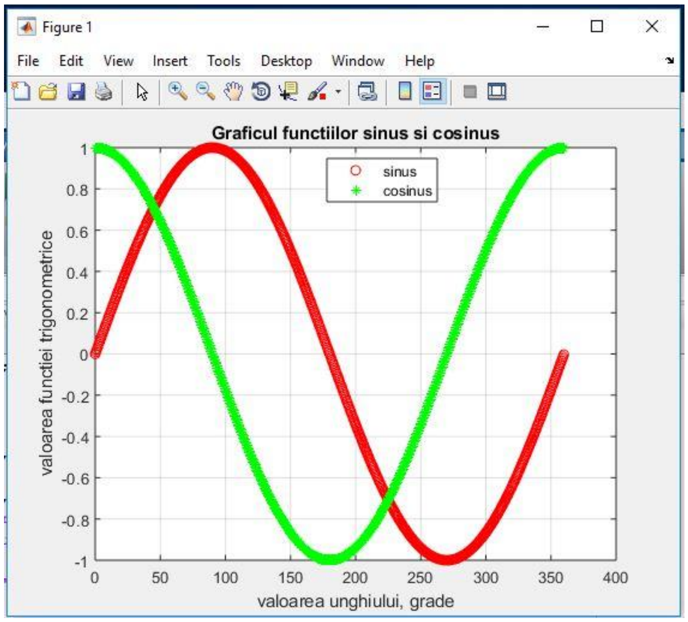
Fişierele se salvează astfel: File →Save as ... → Nume.

OBS. *Numele fişierului trebuie să aibă ca prim caracter o literă, urmată de litere, cifre sau "_". MATLAB retine doar primele 19 caractere din numele variabilei si face distincţie între litere mari si mici.*

Rularea fişierului salvat se poate face în două moduri:

- din fereastra de editare: Debug → Run;
- din fereastra de comenzi: prin tastarea numelui fişierului în dreptul cursorului urmat de apăsarea tastei Enter.

c) Fereastra grafica



1.3. <u>Variabile</u>

Numele variabilelor sunt formate din o literă, urmată de litere, cifre sau "_". Sunt memorate doar primele 19 caractere ale numelor variabilelor. MATLAB este sensibil la tipul de caractere utilizate, mici sau mari.

În MATLAB instrucţiunile sunt de tipul:

variabilă=expresie

sau simplu:

expresie

Terminarea unei instrucţiuni se face cu tasta Enter.

Evaluarea unei expresii produce o matrice afişată pe ecran şi alocată unei variabile, în vederea utilizării ulterioare. Dacă numele variabilei şi semnul "=" sunt omise, este creată automat o variabilă cu numele *ans* (de la answer).

Dacă o linie de comandă se termină cu caracterul ";", instrucţiunea este executată, dar tipărirea rezultatului nu se face.

Pentru afişarea unui text sau valoarea unei variabile se foloseşte funcţia *disp*. Sintaxa acesteia este:

disp(A) – pentru afişarea valorii variabilei A;

disp('afişare text') – pentru afişarea unui mesajului.

Dacă linia de comandă este mai lungă decât o linie ecran, se poate utiliza secvenţa "…" pentru indicarea continuării scrierii liniei de comandă pe următoarea linie ecran.

Ex: S=1+2+3+...Enter
4+5+6 Enter
R: S=21

Obs. Pentru curăţarea ecranului se foloseşte funcţia *clc*, iar pentru ştergerea variabilelor din memoria programului se foloseşte funcţia *clear all*.

Expresiile utilizează operatori aritmetici uzuali:

+	Adunare
-	Scădere
*	Multiplicare
/	Împărţire
\	Împărţire la stânga
^	Ridicarea la o putere
( )	Operatorul de specificare a ordinii de evaluare

Variabile speciale si constante Matlab

Variabilele speciale si constantele nu pot fi declarate si sunt accesibile global, în orice fisier-M.
Variabilele speciale si constantele introduse în mod uzual în MATLAB sunt:
ans – variabila creata automat, reprezentând rezultatul unui calcul pentru care nu s-a alocat un nume;
pi – variabila permanenta, care are alocata valoarea 3,14159265358;
$i = \sqrt{-1}$ – variabila folosita pentru introducerea numerelor complexe (z =x + *i*y);
inf – variabila folosita pentru reprezentarea lui plus infinit, rezultat al împartirii 1.0/0.0;
NaN – variabila folosita pentru reprezentarea lui Not-a-Number, rezultat al împartirii nedefinite 0.0/0.0;

1.4. Numere şi expresii aritmetice

Pentru reprezentarea numerelor zecimale se utilizează **“.”**.

Formatul de afişare al numerelor

MATLAB-ul afişează numerele cu 4 zecimale (setare implicită). Această setare se poate modifica cu ajutorul comenzii:

format opţiune

Opţiune	Rezultat	Exemplu
short	4 zecimale	4/3 → 1.3333
long	14 zecimale	4/3 → 1.33333333333333
short e	4 zecimale + exponenţială	4/3 → 1.3333e+000
long e	14 zecimale + exponenţială	4/3 →1.333333333333333e+000
bank	Două zecimale	4/3 → 1.33
rat	Număr raţional	4/3 → 4/3

Aplicaţie:

1. Să se evalueze funcţia $h = \dfrac{2x + 5 - [43{,}9/2{,}5x - x^3(3\pi/4 - 7)]}{2x + 5}$ pentru x=4,2. Să se afişeze rezultatul în format raţional, folosind funcţia disp.

1.5. Matrici, vectori şi scalari

MATLAB-ul utilizeaza numai un singur tip de obiecte, matrice numerice rectangulare, cu elemente reale sau complexe. În MATLAB, o matrice este un tablou dreptunghiular de numere a cărei dimensiune este dată de o pereche de numere (m x n) care exprimă numărul liniilor si al coloanelor (întotdeauna în această ordine). Scalarii sunt asimilaţi matricelor *1x1* (1linie x 1coloana), iar matricile cu o singură linie sau coloană sunt de fapt vectori (*1*linie x *n* coloane – vector linie; *m* linii x *1* coloana – vector coloana).

Reguli pentru construirea matricelor:

- matricea trebuie numită;
- elementele matricei sunt cuprinse între “[]”;
- elementele unei linii trebuie separate prin blank-uri sau virgule;
- liniile se separa prin “;”.

Ex. Scrieţi în Matlab următoarea matrice: $A = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$.

M: A= [1 2;3 4]

R: A =

```
1     2
3     4
```

Elementul unei matrice A, aflat la intersectia dintre linia m si coloana n este identificat cu notatia *A(m,n)*. Astfel: *b = A(2,1)* conduce la un rezultat în Matlab: b =

3

Elementele dintr-o linie a matricii, de exemplu toate elementele din linia m se notează *A(mi,:)*. Astfel: *c = A(2,:)* conduce la un rezultat în Matlab: c =

3 4

Elementele dintr-o coloană a matricii, de exemplu toate elementele din coloana n se notează *A(:,n)*. Astfel: *d = A(:,1)* conduce la un rezultat în Matlab: d =

1

3

Generarea vectorilor şi a matricelor uzuale

1) Generarea vectorilor linie

a) Un vector cu puţine elemente se introduce în Matlab element cu element.

Ex: Să se scrie în matlab următorul vector linie: $a = |2\ 4\ 6\ 8|$

M: a=[2 4 6 8]

R: a =

```
2    4    6    8
```

b) Un vector a cărui limite (amin şi amax) şi pasul dintre elemente (pas) se cunosc se generează cu următoarea instrucţiune:

A=amin:pas:amax

Numărul de elemente ale vectorului rezultat este: $N = \left| \frac{a\,max - a\,min}{pas} \right| + 1$

Numărul de elemente ale unui vector se obţine cu funcţia *length* a cărei sintaxă este: length(x).

Ex: Să se genereze un vector linie cu limitele intervalului 0 şi 48 şi pasul 4.

M: d=0:4:48
a= length (d)

R: d =

```
 Columns 1 through 7
  0     4     8     12     16     20     24
 Columns 8 through 13
 28    32    36     40     44     48
a =13
```

Obs. Daca valoarea pasului se omite, atunci acesta este considerat implicit egal cu 1.

Ex.: M: e=-5:7

R: e =

-5 -4 -3 -2 -1 0

c) dacă se cunosc limitele intervalului (amin şi amax) şi numărul de elemente (N) ale vectorului se foloseşte instrucţiunea:

A=linspace(amin,amax,N)

Pasul dintre elementele acestui vector este: $pas = \frac{a\,max - a\,min}{N-1}$

Ex.: Să se genereze un vector linie cu limitele amin=0, amax=20 şi numărul de elemente N = 15.

M: f=linspace(0,20,15)

R: f =

Columns 1 through 10

0 1.4286 2.8571 4.2857 5.7143 7.1429 8.5714 10.0000 11.4286 12.8571

Columns 11 through 15

14.2857 15.7143 17.1429 18.5714 20.0000

Obs. Dacă nu se specifică numărul de elemente N, atunci acesta este considerat implicit egal cu 100.

Ex.: g=linspace(0,20)

Obs. Deşi generează acelaşi număr de elemente, prima secvenţă controlează pasul şi poate modifica limita superioară a intervalului, iar a doua controlează numărul de elemente şi menţine limitele impuse.

Ex.: M: a=0:4:50

R: a =

0 4 8 12 16 20 24 28 32 36 40 44 48

M: b=linspace(0,50,13)

R: b =

Columns 1 through 10

0 4.1667 8.3333 12.5000 16.6667 20.8333 25.0000 29.1667 33.3333 37.5000

Columns 11 through 13

41.6667 45.8333 50.0000

2) Generarea vectorilor coloană

a) Un vector cu puţine elemente se introduce în Matlab element cu element.

Ex: Să se scrie în Matlab următorul vector coloană: $a = \begin{vmatrix} 1 \\ 3 \\ -7 \\ 4{,}5 \end{vmatrix}$

b) Se crează un vector linie prin intermediul modalităţilor descrise mai sus şi apoi se foloseşte operatorul transpunere „'”.

Ex: Să se genereze un vector coloană ce conţine 4 elemente in intervalul -2:2.

3) Generarea matricilor speciale

a) Matricea unitate

Este o matrice care are toate elementele egale cu 1. Forme posibile:

U=`ones(n)` este o matrice *nxn* cu elemente de 1.

U=`ones(m,n)` sunt matrici *mxn* cu elemente de 1.

U=`ones(size(A))` are aceeaşi dimensiune cu o matrice A şi are elemente de 1.

b) Matricea zero

Este o matrice care are toate elementele egale cu 0. Forme posibile:

O=zeros(n) este o matrice *nxn* de zerouri.

O=zeros(m,n) sunt matrici *mxn* de 0.

O=zeros(size(A))are aceeaşi dimensiune cu o matrice A şi are elemente de 0.

c)Matricea identitate

Este o matrice cu elementele de pe diagonala principală egale cu 1 iar restul elementelor sunt nule. Notaţia matematică *I* provine de la denumirea matricii şi nu este folosită în MATLAB, pentru evitarea unor confuzii. Se utilizează sintaxa:

`I=eye(m,n)` – această funcţie returnează o matrice identitate *mxn*.

`I=eye(n)` – se referă la o matrice identitate pătratică *nxn*.

Aplicaţii:

2. Scrieţi în Matlab următoarea matrice: $A=\begin{pmatrix} 1 & 5 & 7{,}5 \\ 75 & 81 & 9 \\ 12 & 3\pi & 8{,}3 \end{pmatrix}$. Afişaţi folosind funcţia disp următoarele: a) elementul din linia 1, coloana 1 a matrice A; b) elementul din linia 2, coloana 1 a matrice A; c) elementul din linia 1, coloana 2 a matrice A; d) elementul din linia 3, coloana 3 a matrice A; e) elementul din linia 1, coloana 1 a matrice A; f) toate elementele din coloana 1 a matricei A; g) toate elementele din linia 3 a matricei A; h) toate elementele din coloanele 1 şi 2 a matricei A; i) toate elementele din liniile 2 şi 3 a matricei A.

3. Să se genereze un vector linie de timp (notat cu t) care să contorizeze un interval cuprins între 0 şi 5 ore din 20 în 20 minute. Să se afişeze numărul de unităţi de timp. Să se transforme acest vector în vector coloană.

1.6. Calcul numeric cu Matlab

În MATLAB, calculele aritmetice asupra tablourilor de date pot fi: operaţii după regulile calculului matriceal (operaţii cu matrice) şi operaţii după regulile calculului scalar (operaţii cu tablouri). În tabelul 1 sunt prezentaţi operatorii aritmetici din MATLAB.

Operaţia	**Scalari**	**Matrice**	**Tablouri**
Adunarea	+	+	.+
Scăderea	-	-	.-
Înmulţirea	*	*	.*
Împărţirea la stânga	\	\	.\
Împărţirea la dreapta	/	/	./
Ridicarea la putere	^	^	.^
Transpunerea	‘	‘	.’

Tabelul 1. Operatorii aritmetici MATLAB

Ordinea operaţiilor aritmetice este aceeaşi cu cea cunoscută în matematica elementară, a operaţiilor aritmetice standard.

Ordinea	**Operaţia**
1	parantezele
2	ridicarea la putere
3	înmulţirea şi împărţirea
4	adunarea şi scăderea

Tabelul 2. Ordinea operaţiilor aritmetice

a) Operaţii aritmetice cu scalari

În tabelul 3 sunt prezentate operaţiile aritmetice între doi scalari, fiind prezentată atât forma algebrică cât şi forma MATLAB.

Operaţia	**Forma algebrică**	**Forma MATLAB**
Adunare	a+b	a+b
Scădere	a-b	a-b
Înmulţire	a x b	a*b
Împărţire la dreapta	a:b	a/b
Împărţire la stânga	b:a	a\b
Ridicare la putere	a^b	a^b

Tabelul 3. Forma MATLAB a operatorilor scalari

b) Operaţii aritmetice cu tablouri

Operaţiile cu tablouri sunt operaţii aritmetice între elementele situate în aceeaşi poziţie a tablourilor, cunoscute sub numele de operaţii element cu element. Pentru a preciza că o operaţie se efectuează element cu element între componentele a două matrice de aceeaşi dimensiune, se utilizează operatorul corespunzător operaţiei precedat de punct. Dacă unul dintre operanzi este un scalar, acesta operează cu fiecare element al tabloului.

c) Operaţii aritmetice cu matrice

Operaţiile uzuale de algebră liniară cu matrice sunt simbolizate cu semnele grafice prezentate în tabelul 1 şi se efectuaează după regulile cunoscute din calculul matriceal.

Aplicaţii:

4. Să se scrie în Matlab matricile: $A=\begin{pmatrix}1 & 2\\ 3 & 4\end{pmatrix}$, $C=\begin{pmatrix}5 & 6\\ 7 & 8\end{pmatrix}$. Să se efectueze următoarele operaţii şi să se explice rezultatul: a=A*C; b=A.*C; c=A.^C; d=A./C.

1.7. Funcţii uzuale ale Matlab

Functia	**Descriere**	**Exemplu**
pow2(x)	Calculeaza puterea lui 2	2^5 →pow2(5) → 32
^	Ridicare la putere a numerelor sau matricelor Obs. $x=\sqrt[n]{a} \Rightarrow x=a\wedge(1/n)$	2^2→2^2 → 4 $\sqrt[3]{27}$ →27^(1/3) → 3
sqrt(a)	Calculează radicalul de ordinul 2 dintr-un număr ($x=\sqrt{a}$)	$\sqrt{81}$ →sqrt(81) → 9
exp	Calculeaza exponentiala ($x=e^a$, e=2,718281...)	e^2 →Exp(2) → 7.3891
log	Calculeaza logaritmul natural	ln 7.3891→Log(7.3891) → 2
log2	Calculeaza logaritmul in baza 2	$log_2 4$→Log2(4) → 2
log10	Calculeaza logaritmul in baza 10	lg 100→Log10(100) → 2

Funcţii trigonometrice

Funcţia	Descriere	
Sin	**Calculează sinusul argumentului**	Sin(pi/2) → 1
Asin	**Calculează arcsinusul argumentului**	Asin(1) → 1.5708
Sinh	**Calculează sinusul hiperbolic al argumentului**	Sinh(2+2i) →-1.5093 + 3.4210i
Asinh	**Calculează arcsinusul hiperbolic al argumentului**	Asinh(2) → 1.4436
Cos	**Calculează cosinusul argumentului**	Cos(pi) → -1
Acos	**Calculează arccosinusul argumentului**	Acos(-1) → 3.1416
Cosh	**Calculează cosinusul hiperbolic al argumentului**	Cosh(2+2i) →-1.5656 + 3.2979i
Acosh	**Calculează arccosinusul hiperbolic al argumentului**	Acosh(2) → 1.3170
Tan	**Calculează tangenta argumentului**	Tan(pi/4) → 1
Atan	**Calculează arctangenta argumentului**	Atan(1) → 0.7854
Tanh	**Calculează tangenta hiperbolică a argumentului**	Tanh(2+2i) → 1.0238 - 0.0284i
Atanh	**Calculează arctangenta hiperbolică a argumentului**	Atanh(2) → 0.5493 + 1.5708i

1.8. Rezolvarea sistemelor de ecuaţii liniare

Fie sistemul:

$$3x + 2y - z = 3$$
$$4x - 3y + 2z = -4$$
$$6x - 2y + 5z = 7$$

Sa se determine x, y, z.

Pentru rezolvare în Matlab, se scrie sistemul cu ajutorul matricelor de forma: $A \cdot X = B$ unde:

A – matricea coeficienţilor necunoscutelor; coeficienţii aceleiaşi neconoscute se regăsesc pe aceeaşi coloană ;

X – matricea necunoscutelor;

B – matricea formată din termenii liberi;

$$A = \begin{bmatrix} 3 & 2 & -1 \\ 4 & -3 & 2 \\ 6 & -2 & 5 \end{bmatrix}, \quad X = \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \quad B = \begin{bmatrix} 3 \\ -4 \\ 7 \end{bmatrix}$$

Rezolvarea sistemelor de ecuaţii liniare se poate face prin două metode:

a) prin împărţirea matricelor

AX=B rezulta X=A\B

b) prin folosirea matricei inverse

$A^{-1}AX = A^{-1}B$ rezulta $X = A^{-1}B$ (scriere în M: X=inv(A)*B)

Aplicaţii:

5. Să se rezolve următorul sistem de ecuaţii:

$$2c + 2a = 5 - b$$
$$42a + 3c + 2b = 7$$
$$a - 6 = -3b - 3c$$

1.9.Reprezentări bidimensionale (2D)

Reprezentarea datelor în coordonate linaire se face utilizând funcţia plot a cărei sintaxă de apelare este:

plot(x,y,'tip_linie culoare')

Este posibila setarea culorii, a tipului de linie, a markerului (de exemplu simbolul "+" sau "o") când plotati datele folosind funcţia plot.

Linii tip		**Tipurile de marker**		**Culori**	
-	solid	+	Plus	c	cyan
- -	linie lunga întrerupta	o	Cerc (litera o)	m	magenta
:	linie scurta întrerupta	*	steluta	y	yellow
-.	linie punct	x	x	r	red
		s	pătrat	g	green
		d	romb	b	blue
		^	triunghi cu vârful in sus	w	white
		p	pentagon	k	black
		h	hexagon		

Comanda *grid* adaugă linii ajutătoare graficelor.

Comanda *title('text')* – permite precizarea titlului graficului.

Comenzile *xlabel('text'), ylabel('text')* se utilizează pentru precizarea numelui mărimilor reprezentate pe axă, precum şi a unităţilor de măsură folosite.

Reprezentări multiple in acelaşi sistem de coordonate

plot(x1,y1,'tip_linie culoare', x2,y2,'tip_linie culoare') ; grid ;

Ex.: Să se reprezinte grafic variaţia lui y în funcţie de t, reprezentând punctele prin cercuri de culoare roşie.:

t	0	1	2	3	4	5	6	7	8	9	10	11	12
y	0	0.55	1	2	4	7.2	11	14	15.1	16	16	16	16

```
M: t = [0 1 2 3 4 5 6 7 8 9 10 11 12];
   y= [ 0 0.55 1 2 4 7.2 11 14 15.1 16 16 16 16];
   plot(t, y,'or'); grid;
   xlabel('t'); ylabel('y');
```

Aplicaţii:

6. Să se reprezinte grafic funcţia $y = e^{-x^2} \cdot cos(20x)$ pe intervalul x=-2:0,1:2, cu marker tip romb, culoare albastră. Să se denumească axele şi să se dea titlu graficului.

7. Să se reprezinte grafic în acelaşi sistem de axe funcţiile: y=3sin(πx) şi $z = e^{-0,2x}$ ştiind că x ia valori în intervalul [0; 4] cu pasul 0,02. Să se denumească axele şi să se dea titlu graficului.

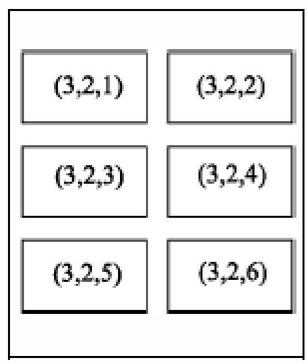
Comenzi pentru grafice speciale

Reprezentare grafica cu bare verticale: bar(x,y,’tip_linie_culoare’)
Reprezentare grafica cu bare orizontale: barh(x,y,’tip_linie_culoare’)
Reprezentare grafica în trepte (pentru semnale cuantizate): stairs(x,y,’tip_linie_culoare’)
Reprezentare grafică statistică: pie(x)

<u>Divizarea ferestrei grafice:</u>

subplot(m,n,p)

- Divide fereastra in m x n ferestre, iar p poate lua valori intre 1 si m x n. Numerotarea ferestrelor incepe din stânga sus şi se continuă de la stânga la dreapta, cea din dreapta jos se notează m x n.
- Exemplu: m = 3, n = 2, p = 6



Aplicatie:

9. Să se reprezinte grafic funcția y = 2x+1 în prima jumătate a ferestrei grafice, respectiv funcția $z = e^x$ în ultima şesime a ferestrei grafice, ştiind că x ia valori în intervalul $0 \le x \le 10$, cu pasul 0,01.

1.10. Reprezentări grafice 3D

<u>Reprezentarea liniilor în spațiu</u>

plot3(x,y,z,’tip_linie_culoare’)

unde x,y,z sunt vectori de aceeasi dimensiune ce contin coordonatele punctelor.

Aplicatie:

10. Să se reprezinte grafic în spațiul 3D, punctele care au drept coordonate x, y şi z, valorile rezultate în urma calculului următoarelor expresii: $x=\sqrt{t} \cdot \sin(2t)$, $y=\sqrt{t} \cdot \cos(2t)$, $z = 0.5\ t$, unde $0 \le t \le 6\pi$ cu pas 0,1.

<u>Rezolvare :</u>

```
t=0:0.1:6*pi;
x=sqrt(t).*sin(2*t);
y=sqrt(t).*cos(2*t);
z=0.5*t;
plot3(x,y,t,'k')
grid
xlabel('x'); ylabel('y'); zlabel('t')
```

Aplicatie:

11. Poziția unei particule în mişcare este dată de expresiile.

$x = \left[\frac{(t-15)}{100} + 1\right]\sin(3t)$, $y = \left[\frac{(t-15)}{100} + 1\right]\cos(0.8t)$,$z = 0.4 \cdot t^{3/2}$

Să se reprezinte grafic poziția particulei pentru un interval de timp 0≤t≤30.

<u>Reprezentare grafică a unei suprafețe</u>

mesh(x,y,z)
surf(x,y,z)

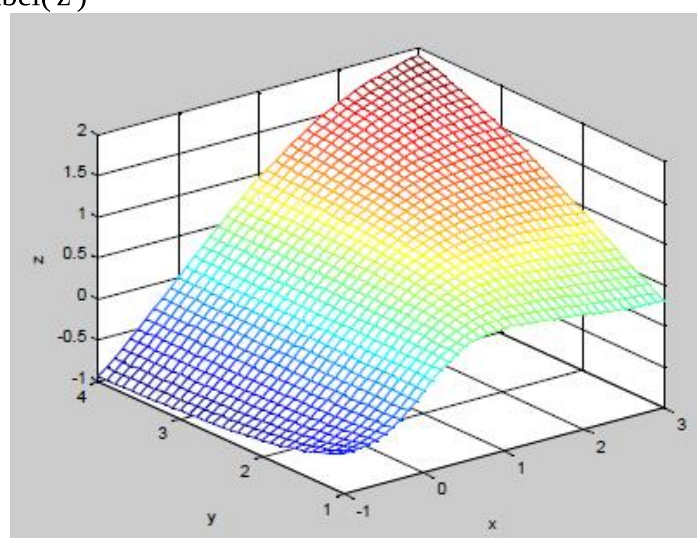
graficele 3D sunt create în 3 etape :

- Crearea unei rețele în planul x-y care acoperă domeniul pe care este definită funcția [X,Y]=meshgrid(x,y)
- Calculul funcției Z pentru ficare punct din rețea (! Atenție funcția va depinde acum de valorile X, Y !)
- Generarea graficului.

Funcția colorbar afiseaza în fereastra grafică harta culorilor sub forma unei bare asezate lângă grafic.

Exemplu: să se reprezinte grafic în spațiul 3D funcția z = $\frac{xy^2}{x^2+y^2}$în domeniul -1≤x≤3 şi 1≤y≤4. Pasul dintre elemente va fi egal cu 0,1.

```
x=-1:0.1:3;
y=1:0.1:4;
[X,Y]=meshgrid(x,y);
Z=X.*Y.^2./(X.^2+Y.^2);
mesh(X,Y,Z);
xlabel('x'); ylabel('y'); zlabel('z')
```



Aplicație:

12. Să se reprezinte grafic în spațiul 3D funcția z =$\frac{y^2}{4} - 2\sin(1{,}5x)$ în domeniul -3≤x≤3 şi -3≤y≤3. Pasul dintre elemente va fi 0,1. Folosiți comanda surf.
13. Să se reprezinte grafic în spațiul 3D funcția z =$\frac{-\cos(2R)}{e^{0.2R}}$, unde R= $\sqrt{x^2 + y^2}$în domeniul -5≤x≤5 şi -5≤y≤5. Pasul dintre elemente va fi egal cu 0,1. Se va utiliza comanda mesh.
14. Să se reprezinte grafic în spațiul 3D funcția z =$x \cdot e^{-x^2-y^2}$, în domeniul -2≤x≤2 şi -2≤y≤2. Pasul dintre elemente va fi 0,2.