

# 5. Mediul de programare MATLAB

## 2. Comanzi pentru grafice speciale :

Reprezentare grafică cu bare verticale: **bar(x,y,'tip\_linie\_culoare')**

Reprezentare grafică cu bare orizontale: **barh(x,y,'tip\_linie\_culoare')**

Reprezentare grafică în trepte (pentru semnale cuantizate): **stairs(x,y, 'tip\_linie\_culoare')**

Reprezentare grafică sub formă discretă (pentru semnale discrete): **stem(x,y,'tip\_linie\_culoare')**

Reprezentare grafică statistică: **pie(x)**

Reprezentare grafică cu histograme: **hist(y)** – MATLAB divide intervalul în care se înscriu datele experimentale din vectorul  $y$  în 10 segmente și apoi reprezintă numărul de date din fiecare segment. **hist(y, n)** –  $n$  reprezintă numărul de segmente în care se realizează divizarea datelor.

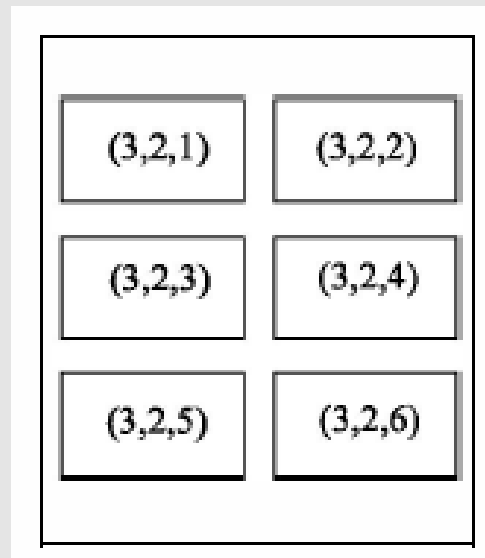
# 5. Mediul de programare MATLAB

## 3. Divizarea ferestrei grafice:

### **subplot(m,n,p)**

- divide fereastra grafică în  $m \times n$  subferestre.
- $p$  – reprezintă numărul subferestrei și poate lua valori între 1 și  $m \times n$ .
- subfereastra din stânga sus se notează cu 1, iar cea din dreapta jos cu  $m \times n$ .

Exemplu:  $m=3, n=2 \Rightarrow p=3 \times 2=6$

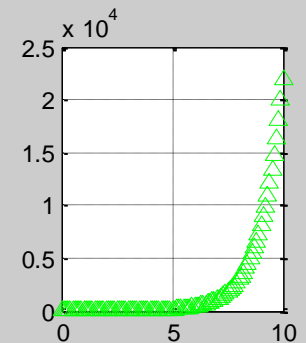
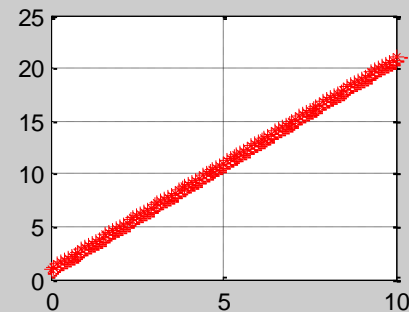


# 5. Mediul de programare MATLAB

## Exemplu:

Să se reprezinte grafic funcția  $y=2x+1$  în prima jumătate a ferestrei grafice, respectiv funcția  $z=e^x$  în ultima șesime a ferestrei grafice.  $x$  ia valori în intervalul  $0 \leq x \leq 10$ , cu pas 0,01.

```
x=0:0.1:10;  
y=2*x+1;  
z=exp(x);  
subplot(2,2,1);plot(x,y,'*r');  
grid;  
subplot(2,3,6);plot(x,z,'^g');  
grid;
```



# 5. Mediul de programare MATLAB

## 5.12. Grafică în MATLAB – 3D

### 1. Reprezentarea liniilor în spațiu:

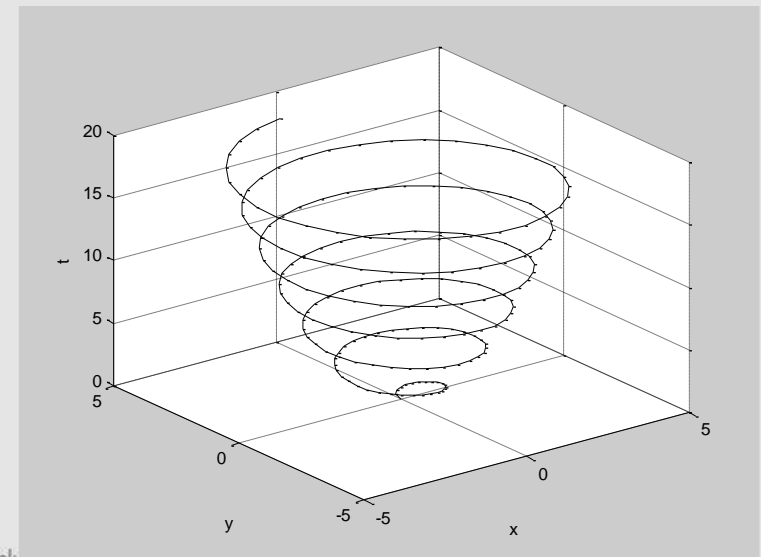
Sintaxa: `plot3(x,y,z,'tip_linie_culoare')`

–  $x,y,z$  – vectori de aceeași dimensiune ce conțin coordonatele punctelor.

### Exemplu:

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

```
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')
```



# 5. Mediul de programare MATLAB

## 1. Reprezentarea suprafețelor:

Sintaxa: **mesh(x,y,z)**  
**surf(x,y,z)**

Graficele 3D sunt create în 3 etape:

1. Crearea unei rețele în planul x-y, care acoperă domeniul funcției.

**[X,Y]=meshgrid(x,y)**

x,y –vectori care divid domeniul x, respective y

2. Calculul valorii funcției z pentru fiecare punct din rețea.

3. Generarea graficului.

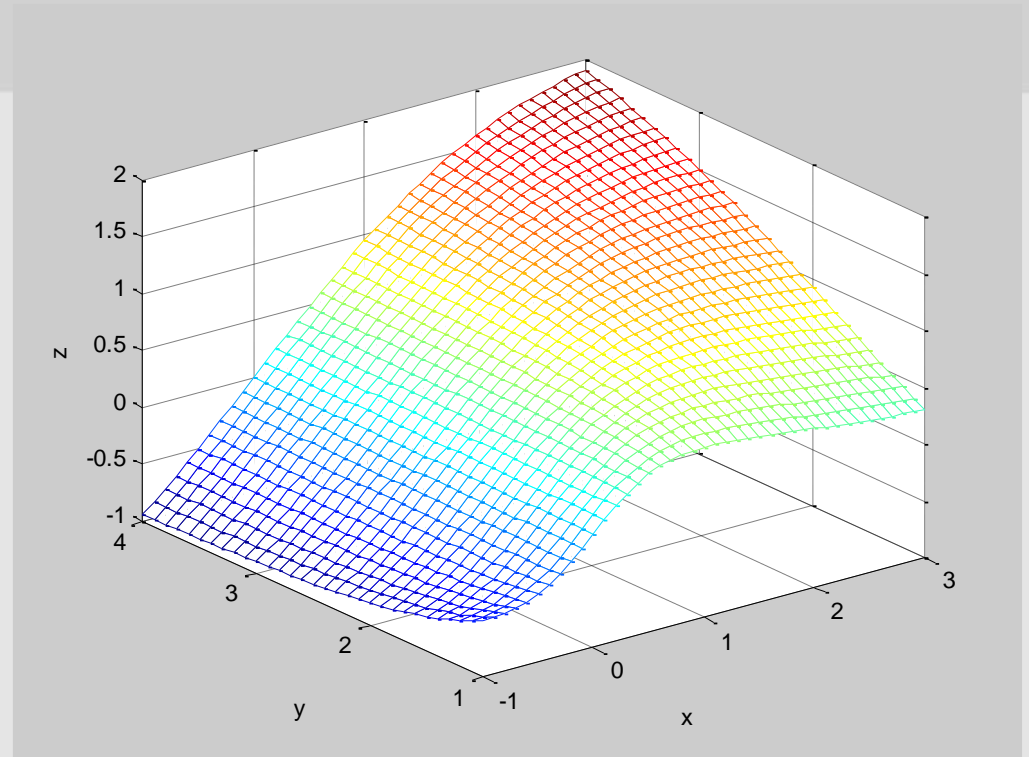
Funcția **colorbar** - afișează în fereastra grafică harta curentă a culorilor, sub forma unei bare așezate lângă grafic.

# 5. Mediul de programare MATLAB

## Exemplu:

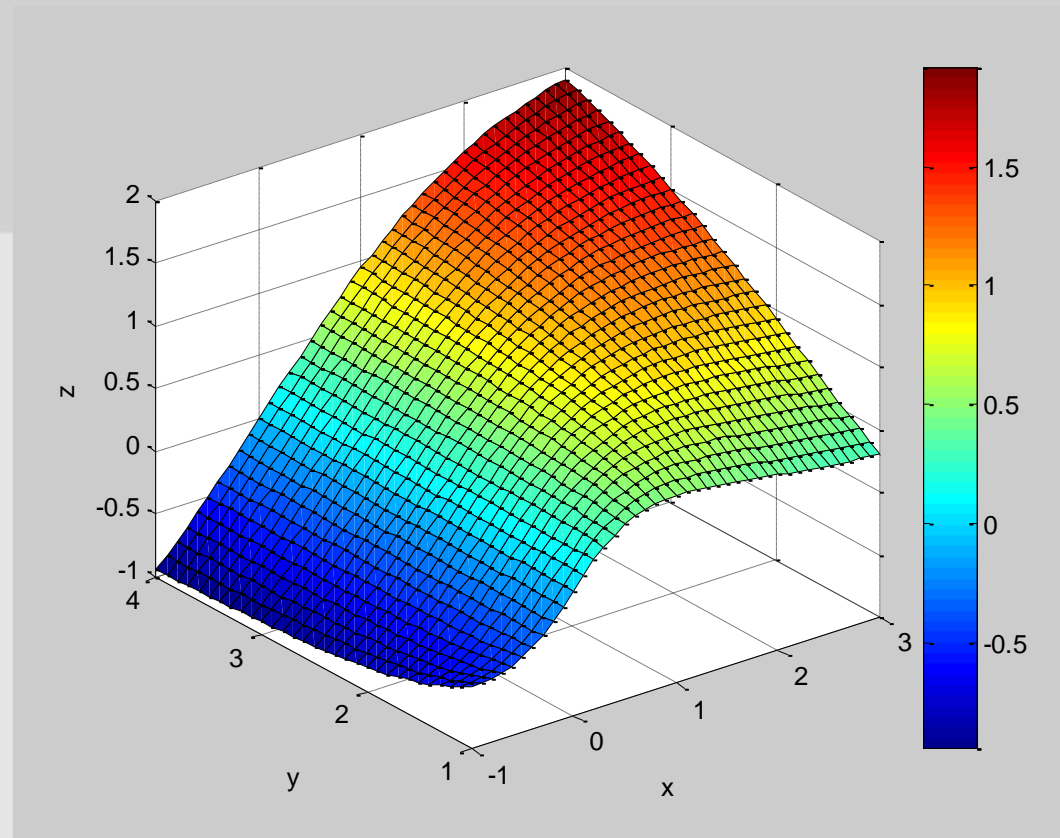
Să se reprezinte grafic, în spațiul 3D, funcția  $z = \frac{xy^2}{x^2+y^2}$ , în domeniul  $-1 \leq x \leq 3$  și  $1 \leq y \leq 4$ . Pasul dintre elemente va fi egal cu 0,1.

```
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')
```



# 5. Mediul de programare MATLAB

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



# 5. Mediul de programare MATLAB

## 5.13. Instrucțiuni de control logic

### 1. Operatori relaționali

Un **operator relațional** compară două matrice sau două expresii matriceale, element cu element, și returnează o matrice de aceeași dimensiune cu a matricelor care se compară, cu elementele 1 când relația este adevărată și zero când relația este falsă.

Operator relațional	Semnificație
<	Mai mic
<=	Mai mic sau egal
>	Mai mare
>=	Mai mare sau egal
==	Identic
~=	Diferit



# 5. Mediul de programare MATLAB

Sintaxa de utilizare a operatorilor relaționali:

**rezultat = expresie\_1 operator expresie\_2**

Unde: *rezultat* – o matrice de elemente 0 și 1 conform comparației realizate; *expresie\_1*, *expresie\_2* – matricele care se compară; *operator* – operatorul relațional folosit.

**Exemplu:**

x=5<8

x =  
1

**Exemplu:**

Să se determine elementele din matricea A care sunt mai mari sau egale cu 4.

A =  $\begin{vmatrix} 2 & 4 & 6 \\ -3 & 5 & 8 \\ 1 & 2 & 3 \end{vmatrix}$

x=4<[2 4 6; -3 5 8;1 2 3]

x =  
0 0 1  
0 1 1  
0 0 0

# 5. Mediul de programare MATLAB

## 2. Operatori logici

Operator relațional	Semnificație	
&	AND (ȘI)	O expresie care utilizează operatorul & este adevărată dacă ambii operanzi sunt adevărați. În termeni numerici, expresia este adevărată dacă ambii operanzi sunt nenuli.
	OR (SAU)	O expresie care utilizează operatorul   este adevărată dacă unul dintre operanzi este logic adevărat sau dacă ambii operanzi sunt adevărați. În termeni numerici, expresia este falsă dacă ambii operanzi sunt nuli.
~	NOT (NU)	O expresie care utilizează operatorul NOT, ~, neagă operandul. În termeni numerici, orice operand nenul devine nul și orice operand nul devine unu.

# 5. Mediul de programare MATLAB

## Exemplu:

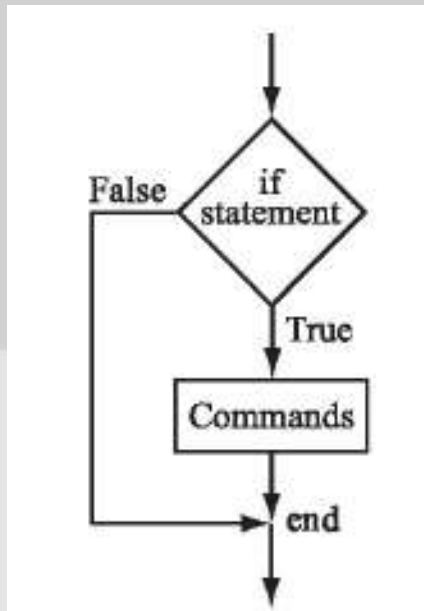
```
u = [1 0 2 3 0 5];  
v = [5 6 1 0 0 7];  
a=u & v  
b=u | v  
c=~u
```

## Rezultat:

```
a = 1 0 1 0 0 1  
b = 1 1 1 1 0 1  
c = 0 1 0 0 1 0
```

# 5. Mediul de programare MATLAB

## 3. Instrucțiunea if



**if** expresie logică  
    grup\_de\_instrucțiuni MATLAB  
**end;**

- Dacă **expresia logică** este adevărată, se execută grupul de instrucțiuni din interiorul ciclului if. Dacă **expresia logică** este falsă, se trece la prima instrucțiune care urmează după end.

# 5. Mediul de programare MATLAB

## 3. Instrucțiunea if

### Exemplu:

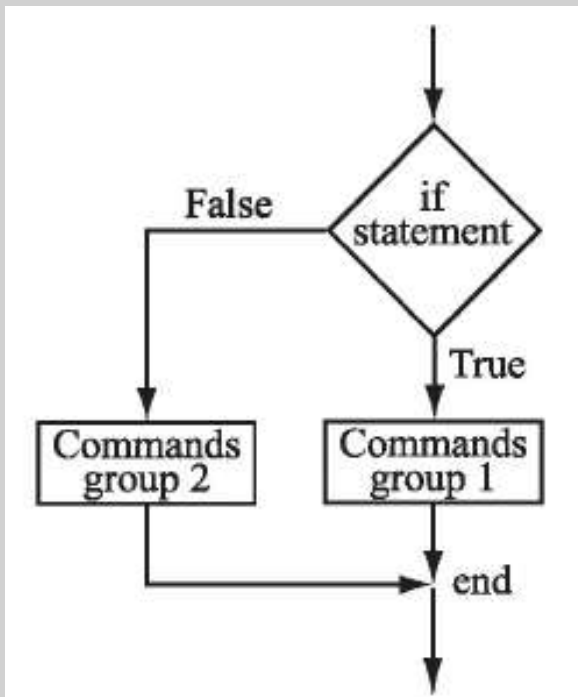
Un angajat efectuează în cadrul normei de bază 40 h / săptămână. Pentru aceste ore este plătit cu 50 lei / h. Pentru orele suplimentare, angajatul are un tarif orar cu 50% mai mare. Să se realizeze un program care, pe baza numărului de ore efectuate lunar (o luna = 4 săptămâni), să calculeze salariul lunar al angajatului.

```
tarif_h=50;  
h=input('Introduceti numarul de ore lucrate intr-o luna=');  
Salar=h*tarif_h;  
if h>40  
    Salar=Salar+(h-40)*0.5*tarif_h;  
end;  
disp('Salar lunar='); disp(Salar);
```

**Rezultat:** Introduceti numarul de ore lucrate intr-o luna=50  
Salar lunar=  
2750

# 5. Mediul de programare MATLAB

## 4. Instrucțiunea if – else



```
if expresie logică  
    grup_de_instrucțiuni MATLAB 1  
else  
    grup_de_instrucțiuni MATLAB 2  
end;
```

- Dacă **expresia logică** este adevărată, se execută grupul de instrucțiuni dintre if și else, iar apoi se sare la end. Dacă **expresia logică** este falsă, se trece la grupul 2 de instrucțiuni dintre else și end.

# 5. Mediul de programare MATLAB

## 4. Instrucțiunea if – else

### Exemplu:

Un student obține la testele din cadrul unei discipline următoarele punctaje: P1=60, P2=70 și P3=54. Să se construiască un program MATLAB, care să calculeze media celor 3 note și, dacă punctajul este mai mare de 60 să comunice mesajul: disciplina a fost promovata. In caz contrar se va comunica: disciplina nu a fost promovata.

```
P1=60;  
P2=70;  
P3=54;  
media=(P1+P2+P3)/3;  
if media>60  
    disp('Disciplina a fost promovata');  
else  
    disp('Disciplina nu a fost promovata');  
end;  
disp('media='); disp(media);
```

### Rezultat:

Disciplina a fost promovata  
media=  
61.3333

## 5. Instrucțiunea for

```
for index = expresie
    grupul_de_instrucțiuni
end;
```

unde:

- index este numele contorului;
- expresie = valoare initiala:pas:valoare finala (dacă pasul se omite, atunci este implicit 1)
- grup\_de\_instrucțiuni este orice expresie MATLAB.

```
for k=1:4:11
    x=2*k^2
end;
```

```
x =
    2
x =
   50
x =
  162
```



# 5. Mediul de programare MATLAB

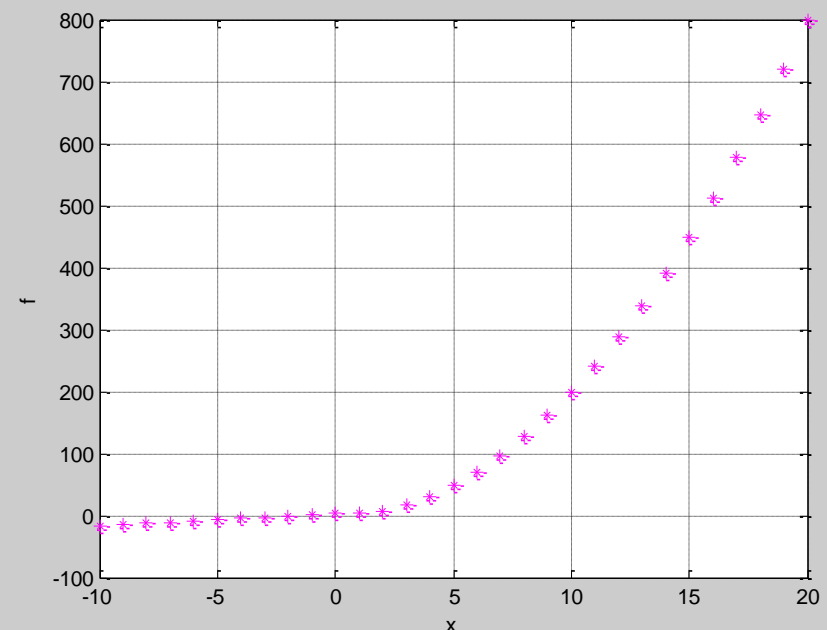
## 5. Instrucțiunea for

### Exemplu:

Calculați și reprezentați grafic funcția  $f$  pentru toate valorile întregi pentru care este definită.

$$f = \begin{cases} 2x + 3, & x \in [-10, 2] \\ 2x^2 - 1, & x \in (2, 20] \end{cases}$$

```
x=-10:20;  
for k=1:length(x)  
    if x(k) <= 2  
        f(k)=2*x(k)+3;  
    else  
        f(k)=2*x(k).^2-1;  
    end;  
end;  
plot(x,f,'*m');grid;  
xlabel('x');  
ylabel('f');
```



# 5. Mediul de programare MATLAB

**6. Instrucțiunea while** – se utilizează în situațiile în care repetarea unui set de instrucțiuni este necesară, dar numărul de repetări nu se știe în avans.

- repetarea setului de instrucțiuni se face atâta timp cât o condiție specificată este adevărată.

**while** condiție  
grup\_de\_instrucțiuni MATLAB  
**end;**

unde: - „condiție” este unul dintre operatorii relaționali: =, <, >, <=, >=, - =, și poate să aibă forma expresie\_1 condiție expresie\_2.

```
x=1;  
while x<=15  
    x=2*x  
end;  
disp('Ciclul s-a incheiat');
```

```
x =      x =  
  2      8  
x =      x =  
  4     16
```

Ciclul s-a incheiat

# 5. Mediul de programare MATLAB

## 6. Instrucțiunea while

### Exemplu:

Să se scrie un program, utilizând o buclă while, care calculează suma elementelor vectorului  $a=[6\ 4\ -2\ 1\ -5\ 3\ 3]$ .

```
a=[6 4 -2 1 -5 3 3];  
b=length(a);  
sum=0;  
k=1;  
while k<=b  
    sum=sum+a(k);  
    k=k+1;  
end;  
disp('sum=');disp(sum);
```

Rezultat: sum=  
10

# 5. Mediul de programare MATLAB

## 5.14. Funcții ale utilizatorului

În cazul în care prima linie a unui fișier M conține cuvântul cheie **function** fișierul respectiv intră în categoria **fișierelor de tip funcție**.

Forma generală a primei linii a unui fișier funcție este:

```
function [var_iesire1, var_iesire2 ... ] = nume_funcție(var_intrare1, var_intrare2 ... )
```

unde :

- **function** reprezintă un cuvânt cheie obligatoriu care declară că fișierul este de tip funcție;
- **nume\_funcție** reprezintă numele funcției, adică numele sub care este salvat fișierul, însă fără extensie.
- **var\_iesire1, var\_iesire2** reprezintă parametri de ieșire care trebuie separați prin virgulă și cuprinși între paranteze drepte. Dacă funcția nu are parametri de ieșire parantezele drepte și semnul egal nu mai au sens.
- **var\_intrare1, var\_intrare2** reprezintă parametri de intrare care trebuie separați prin virgulă și cuprinși între paranteze rotunde. Dacă funcția nu are parametri de intrare parantezele rotunde nu mai au sens.

# 5. Mediul de programare MATLAB

**Exemplul 1:** Sa se realizeze o functie care sa calculeze suma si media aritmetica a elementelor vectorului x ce contine urmatoarele elemente: 1,3,5,7,9,10,13,20,34,52.

În fereastra de editare se scrie următorul program (în care se definește modul de calcul al sumei, respectiv a mediei aritmetice a elementelor vectorului x):

```
M1: function[s,m]=medie(x);  
    n=length(x);  
    s=sum(x);  
    m=s/n;  
    disp('Suma elementelor vectorului x este:'); disp(s);  
    disp('Media numerelor este:'); disp(m);
```

Acest program se salvează cu numele **medie.m**. În fereastra de comenzi se definește vectorul x și apoi se apelează funcția medie.

```
M2:    x=[1,3,5,7,9,10,13,20,34,52];  
       medie(x)
```

# 5. Mediul de programare MATLAB

## Exerciții:

1. Să se construiască o funcție a utilizatorului care să permită reprezentarea grafică a  $\ln(x)$  pentru  $x \in [1,5]$ .
2. Să se construiască o funcție a utilizatorului care să permită convertirea temperaturii din  $^{\circ}\text{C}$  în Fahrenheit:  $F=9C/5+32$ . Să se calculeze valoarea în F a temperaturii de  $37^{\circ}\text{C}$ .