

```
% problema 1, modalitatea 1
```

```
figure(1)
x = 0:0.01:2*pi;
y = exp(x).*sin(x);
plot(x,y)
```

```
% problema 1, modalitatea 2
```

```
functia_ex1_lab11 = inline('exp(x).*sin(x)')
figure(2)
ezplot(functia_ex1_lab11, [0 , 2*pi])
```

```
% problema 2
```

```
t = -2:0.01:2;
x = 1+abs(t); y = abs(1-t.^2);
figure('Name','O functie data parametric')
% folosim si comanda "subplot" pentru a desena 3 grafice diferite in cadrul
% aceleasi figuri numita "Figure 3: O functie data parametric"
subplot(1,3,1); plot(t,x)
title('Graficul functiei x(t)'); grid
subplot(1,3,2); plot(t,y)
title('Graficul functiei y(t)'); grid
subplot(1,3,3); plot(x,y)
title('Graficul functiei y(x)'); grid
axis([0 4 -1 5])
% fiecare grafic va avea titlul ei
% cel de-al treilea grafic are axa Ox de la 0 la 4 si Oy de la -1 la 5
% la fiecare grafic apare si grila de linii orizontale si verticale.
```

```
%{
```

```
% Avem si varianta mai simpla:
```

```
figure(3)
t = -1:0.01:1;
x = 1+abs(t);
y = abs(1-t.^2);
plot(x,y)
%}
```

```
% problema 3
```

```
figure(4)
```

```
% folosim si comanda "subplot" pentru a desena 2 grafice diferite in cadrul  
% aceleasi figuri numita "Figure 4"
```

```
theta1 = 0:0.01:pi; r1 = 2*sin(2*theta1);  
theta2 = 0:0.01:2*pi; r2 = 2*sin(2*theta2);  
subplot(1,2,1)  
polar(theta1,r1)  
subplot(1,2,2)  
polar(theta2,r2)
```

```
% problema 5
```

```
figure(5)  
theta = 0:0.01:2*pi;  
r = sin(2*theta).*cos(2*theta);  
polar(theta,r)
```

```
% problema 6
```

```
figure(6)  
ezplot('(x.^2+y.^2-1).^3+27*x.^2*y.^2', [-1 1 -1 1])
```

```
% problema 7
```

```
figure(7)  
ezplot('(x.^2+y.^2).^2 = x.^2-y.^2', [-1 1 -1 1])
```

```
% problema 9
```

```
x0 = 2; y0 = 3; R = 4  
figure(8)  
ezplot(@(x,y) (x-x0).^2+(y-y0).^2 - R.^2)
```

```
% problema 10
```

```
x0 = input('Introduceti abscisa centrului, x_0 = ');  
y0 = input('Introduceti ordonata centrului, y_0 = ');  
R = input('Introduceti raza cercului R = ');  
figure(9)  
ezplot(@(x,y) (x-x0).^2+(y-y0).^2 - R.^2)
```

```
% problema 11
a = input('Introduceti coeficientul nenul a = ');
b = input('Introduceti coeficientul b = ');
c = input('Introduceti coeficientul c = ');
figure(10)
coef = [a b c];
rad = roots(coef) % am determinat zero-urile cu ajutorul comenzii roots aplicata pentru polinoame
fct_pol = @(x) a*x.^2+b.*x+c
val0 = input('Introduceti valoarea in jurul careia sa cautam solutia = ');
rad_val0 = fzero(fct_pol,val0)
ezplot(fct_pol)

rad_sort = sort(rad)
%{
am scris radacinile in ordine crescatoare astfel incat sa la pot folosi
in determinarea minimului/maximului; le vom folosi scriind rad_sort(1) si rad_sort(2)
%}
if a<0
    val_min = nan
    fprintf('Deoarece %.2f este <0, functia nu admite puncte de minim (local)\n',a)
else
    val_min = fminbnd(fct_pol,rad_sort(1),rad_sort(2))
end

fct_pol2 = @(x) -a*x.^2-b.*x-c
%{
am definit o noua functie = opusul primeia
acum o pot folosi in determinarea maximului
x_maxim(f) = x_minim(-f)
%}
if a<0
    val_max = fminbnd(fct_pol2,rad_sort(1),rad_sort(2))
else
    val_max = nan
    fprintf('Deoarece %.2f este >0, functia nu admite puncte de maxim (local)\n',a)
end

% desenam acum un grafic in care sa apara si graficul functiei si anumite puncte particulare
a1 = input('Introduceti acum intervalul pentru doriti sa desenam graficul functiei. Primul punct = ');
b1 = input('Introduceti acum intervalul pentru doriti sa desenam graficul functiei. Al doilea punct = ');
x = a1:0.01:b1;
y = fct_pol(x);

figure(11)
```

```
plot(x,y)
% !! mai multe grafice in acelasi reper
% folosim comanda: hold on - hold off
hold on
plot(rad_sort,fct_pol(rad_sort),'r*')
plot(val_min,fct_pol(val_min),'gx')
plot(val_max,fct_pol(val_max),'bx')
hold off
%{
    r = red (culoarea liniei (punctelor) desenate cu plot)
    g = green (culoarea liniei (punctelor) desenate cu plot)
    b = blue (culoarea liniei (punctelor) desenate cu plot)
    * = marcarea cu * a liniei (punctelor) desenate cu plot
        (se foloseste, de exemplu, daca desenam doar cateva puncte, pentru a le scoate in evidenta)
    x = marcarea cu x a liniei (punctelor) desenate cu plot
        (se foloseste, de exemplu, daca desenam doar cateva puncte, pentru a le scoate in evidenta)
%}
```

```
% !! mai multe grafice in acelasi reper
% folosim un singur plot:
figure(12)
plot(x,y,rad_sort,fct_pol(rad_sort),'r*',val_min,fct_pol(val_min),'gx',val_max,fct_pol(val_max),'bx')
```

```
% problema 12
% daca vrem doar valori ale functiei definim functia10.m si o apelam
x10 = input('Introduceti valoarea in care vreti sa calculati functia = ');
val_x10 = functia10(x10)
```

```
% daca vrem doar valori ale functiei definim, intr-un mod similar, functia11.m
% si o apelam
x10_var2 = input('Introduceti valoarea in care vreti sa calculati functia = ');
val_x10_var2 = functia11(x10_var2)
```

```
% daca vrem sa reprezentam grafic functia din enunt
figure(13)
x = -1:0.01:1;
y = (-x).*(x<0) + (x.^2).*(x>=0);
plot(x,y,'r:')
% : = marcarea cu o linie punctata a liniei (punctelor) desenate cu plot
```

```
% daca vrem sa reprezentam grafic functia din enunt (modalitatea 2)
figure(14)
x = -1:0.01:1; n = length(x);

for i = 1:n
    if x(i)<0
        y(i) = -x(i);
    else
        y(i) = x(i) ^ 2;
    end
end

plot(x,y,'b--')
% -- = marcarea cu o linie intrerupta a liniei (punctelor) desenate cu plot
```