

Serie 1

% EX. a

```
A = [2 3 4 ; 5 1 1; 5 3 2; 4 1 2];  
b=[21; 19; 24; 19];  
x = A \ b ;
```

% EX. b (see files f.m and f2.m)

```
v = f(1,1) ;  
[v,g,h] = f(1,1) ;  
x = [ 1 1 ];  
v = f2(x) ;  
[v,g,h] = f2(x) ;
```

% EX. c

```
[X,Y]=meshgrid(-2:.2:2,-2:.2:2);  
Z=X.^2 + 3 * Y.^2;  
%mesh(Z)  
surf(X,Y,Z)
```

% EX. d

```
fct = 'f2';  
x_opt = fminsearch(fct, [0 ; 0]);  
% x_opt = fminsearch('f2', [0 ; 0]);  
disp('The minimum of f2 is: '); disp(x_opt);
```

% EX. e (see file rosen.m)

```
x =[1 1];  
v = rosen(x) ;  
[v,g,h] = rosen(x) ;
```

%plot rosen

%1st way

```
[X1, X2]=meshgrid(-2:.2:2,-2:.2:2);  
Z=100*(X2-X1.^2).^2 + (1-X1).^2;  
mesh(Z)  
%surf(X1,X2,Z)
```

%2nd way

```
x=[1 1];  
j=1;  
i=1;  
[X,Y]=meshgrid(-2:.2:2,-2:.2:2);  
[m,n] = size([X,Y]); % calcul de la taille de [X,Y]  
% construction de la matrice Z de dessin  
for j=1:m  
    for i=1:m  
        % artifice afin d'utiliser la valeur de la fonction avec le meshgrid
```

```

x(1) = X(1,j);
x(2) = Y(i,1);
Z(i,j)=feval('rosen',x);
i=i+1;
end
j=j+1;
end
mesh(Z)

% fminsearch

fct = 'rosen';
x_opt = fminsearch(fct, [0 ; 0]);
% x_opt = fminsearch('rosen', [0 ; 0]);
disp('The minimum of rosen is: '); disp(x_opt);

```

%%%%FUNCTIONS%%%%%

% f(x,y) returns the value, the gradient and the hessian at point (x,y) for
% the quadratic function $x^2 + 3y^2$

```

function [f, g, h] = f(x,y)
% value of the function at (x,y)
f=x^2 + 3*y^2;
% gradient at (x,y)
g=[2*x ; 6*y];
% hessian at (x,y)
h=[2 0; 0 6];

```

```

function [f, g, h] = f2(x)

% value of the function at point x
f=x(1)^2 + 3*x(2)^2;
% gradient at point x
g=[2*x(1); 6*x(2)];
% hessian at point x
h=[2 0; 0 6];

```

% Rosenbrock function

```

function [f, g, h]=rosen(x)
f=100*(x(2)-x(1)^2)^2 + (1-x(1))^2;
g=[-400*(x(2)-x(1)^2)*x(1)-2*(1-x(1)) ; 200*(x(2)-x(1)^2)];
h(1,1)=2-400*x(2)+1200*x(1)^2;
h(1,2)=-400*x(1);
h(2,1)=-400*x(1);
h(2,2)=200;

```