MATRIX game - mixed strategies Nash equilibrium

```
clear all
C=[7,15,2,3;4 2 3 10; 5 3 4 12]
m = size(C,1);
n = size(C,2);
c=[zeros(m,1);1];
A= [C', -ones(n,1)]; b=zeros(n,1); Aeq=[ones(1,m),0]; beq=1;
lb= [zeros(m,1);-inf]; ub=[ ];
[sol,Val,exitflag,output,lambda] = linprog(c, A,b, Aeq, beq, lb, ub);
x = sol(1:m)
y = lambda.ineqlin
```

BIMATRIX game - mixed strategies Nash equilibrium

```
C1=[3,3;4 1;6 0]; C2=[3 4;4 0;3 5];
[m,n] = size(C1);
H=[zeros(m,m),C1+C2,ones(m,1), zeros(m,1); C1'+C2',zeros(n,n),zeros(n,1),ones(n,1); ones(1,m), zeros(1,n+2);
zeros(1,m),ones(1,n),0,0];
%X0=[0,1,0,0,1,1,1]'; % m+n+2 vector
X0=[rand(5,1);10-20*rand(2,1)]
%X0=[0,0,1,1,0,10-20*rand(1,2)]';
Ain=[-C2', zeros(n,n),zeros(n,1),-ones(n,1);zeros(m,m), -C1,-ones(m,1),zeros(m,1)]; bin=zeros(n+m,1);
Aeq=[ones(1,m),zeros(1,n+2);zeros(1,m),ones(1,n),0,0];
beq=[1;1]; LB=[zeros(m+n,1);-Inf;-Inf];
UB=[ones(m+n,1);Inf;Inf];
[sol,fval,exitflag,output]=fmincon(@(X) 0.5*X'*H*X, X0, Ain,bin, Aeq,beq,LB,UB)
x = sol(1:m)
y = sol(m+1:m+n)
```