X = [-50 0 0 0 0.5 0.5 0.5 1 1 ;40 0 0.5 1 0 0.5 1 0 0.5];

T = [1 1 1 1 1 -1 -1 -1 -1];

%reprezentare grafica a datelor

figure(1), hold on;

eticheta1 = find(T==1);

etichetaMinus1 = find(T==-1);

plot(X(1,eticheta1),X(2,eticheta1),'or');

plot(X(1,etichetaMinus1),X(2,etichetaMinus1),'xg');

axis([-55 2 -2 45]);

%pune pauza 2 secunde

%creaza perceptron

net = newp([-1 1;-1 +1],1,'hardlims');

view(net);

%afiseaza proprietatile perceptronului legate de vectorul de ponderi + bias

disp('Proprietati legate de vectorul de ponderi:');

disp(net.inputWeights{1});

disp('Proprietati legate de bias:');

disp(net.biases{1});

%initializeaza parametri retelei, implicit ponderile + bias sunt nule

net = init(net);

%seteaza numarul de epoci pentru antrenare

net.trainParam.epochs = 1;

net.inputWeights{1}.learnFcn='learnpn';

net.biases{1}.learnFcn='learnpn';

%antreneaza reteaua

for i=1:100

e=i;

net = train(net,X,T);

figure, hold on;

plot(X(1,eticheta1),X(2,eticheta1),'or');

plot(X(1,etichetaMinus1),X(2,etichetaMinus1),'xg');

axis([-55 55 -55 55]);

plotpc(net.IW{1},net.b{1})

title('Reprezentarea grafica a exemplelor de antrenare si a dreptei de separare');

%simuleaza reteaua pentru datele de intrare

if(isequal(sim(net,X),T))

break

end

if(mod(i,5)==0)

close all

end

end

etichetePrezise = sim(net,X);

isequal(etichetePrezise,T)

e