function FinalCode2(iterations)

input = [0 0 0;0 0 1;0 1 0;0 1 1;1 0 0;1 0 1;1 1 0;1 1 1];

output = [0;1;1;0;1;0;0;1];

bias = [-1 -1 -1 -1];

coeff = 0.7;

rand('state',sum(100\*clock));

weights = -1 +2.\*rand(4,4);

for i = 1:iterations

out = zeros(8,1);

numIn = length (input(:,1));

for j = 1:numIn

neuron1 = bias(1,1)\*weights(1,1) + input(j,1)\*weights(1,2) + input(j,2)\*weights(1,3)+input(j,3)\*weights(1,4);

x2(1) =1/(1+exp(-neuron1));

neuron2 = bias(1,2)\*weights(2,1) + input(j,1)\*weights(2,2) + input(j,2)\*weights(2,3)+input(j,3)\*weights(2,4);

x2(2) =1/(1+exp(-neuron2));

neuron3 = bias(1,3)\*weights(3,1) + input(j,1)\*weights(3,2) + input(j,2)\*weights(3,3)+input(j,3)\*weights(3,4);

x2(3) =1/(1+exp(-neuron3));

x3 = bias(1,4)\*weights(4,1) + x2(1)\*weights(4,2) + x2(2)\*weights(4,3)+x2(3)\*weights(4,4);

out(j) = 1/(1+exp(-(x3)));

delta\_out=(output(j)-out(j))\*out(j)\*(1-out(j));

delta\_neuron1=x2(1)\*(1-x2(1))\*weights(4,2)\*delta\_out;

delta\_neuron2=x2(2)\*(1-x2(2))\*weights(4,3)\*delta\_out;

delta\_neuron3=x2(3)\*(1-x2(3))\*weights(4,4)\*delta\_out;

for k = 1:4

if k == 1

weights(1,k) = weights(1,k) + coeff\*bias(1,1)\*delta\_neuron1;

weights(2,k) = weights(2,k) + coeff\*bias(1,2)\*delta\_neuron2;

weights(3,k) = weights(3,k) + coeff\*bias(1,3)\*delta\_neuron3;

weights(4,k) = weights(4,k) + coeff\*bias(1,4)\*delta\_out;

else

weights(1,k) = weights(1,k) + coeff\*input(j,1)\*delta\_neuron1;

weights(2,k) = weights(2,k) + coeff\*input(j,2)\*delta\_neuron2;

weights(3,k) = weights(3,k) + coeff\*input(j,3)\*delta\_neuron3;

weights(4,k) = weights(4,k) + coeff\*x2(k-1)\*delta\_out;

end

end

end

end

weights

out

end