```
clear;
clc;
x = [1  1 -1 -1;  % Input pattern (row 1)
     1 -1  1 -1]; % Input pattern (row 2)
t = [1 -1 -1 -1]; % Target output for AND function
w = [0 0];
b = 0;
epoch = 0;
convergence = true;
while convergence
    convergence = false;
    for i = 1:4 % Iterate over all input patterns
        % Calculate the net input
        y = b + x(1, i) * w(1) + x(2, i) * w(2);
        % Check if the output is incorrect
        if y * t(i) <= 0
            convergence = true; % Changes are needed
            % Update weights and bias
            for j = 1:2
                w(j) = w(j) + (t(i) * x(j, i));
            end
            b = b + t(i);
        end
    end
    epoch = epoch + 1
end
disp('HEBB NET for AND Function');
disp('Final Weight Matrix:');
disp(w);
disp('Final Bias:');
disp(b);
disp('Number of Epochs:');
disp(epoch);
```