Matlab script(hw3.m):

```
%I used example in the Optimization lecture  as a basis for
%the code.

clear
clc
close all

% Setting up the -2 to 3 range
h = 0.1;
x = -2:h:3;
y = -2:h:3;

% Define the test function
[X, Y] = meshgrid(x, y);
f = (X.^4 + Y.^4) / 4 - (X.^3 + Y.^3) / 3 - X.^2 - Y.^2 + 4;

% The optimization parameters
K = 100; % Number of iterations
ix = zeros(K, 2); % The index of the grid point at each iteration (2D)
ix(1, :) = randi([1, numel(x)], 1, 2); % Start at a random point in the grid
ff = zeros(K, 1); % The objective function along the search path

for k = 1:K
    i = randi(4); % Generate a random number (1 to 4 for 2D)
    ixtemp = ix(k, :);

    % Move in a random direction (up, down, left, right)
    if i == 1 && ixtemp(1) < length(x)
        ixtemp(1) = ixtemp(1) + 1;
    elseif i == 2 && ixtemp(1) > 1
        ixtemp(1) = ixtemp(1) - 1;
    elseif i == 3 && ixtemp(2) < length(y)
        ixtemp(2) = ixtemp(2) + 1;
    elseif i == 4 && ixtemp(2) > 1
        ixtemp(2) = ixtemp(2) - 1;
    end

    % If the cost function at the new point is smaller, accept the move
    if f(ixtemp(2), ixtemp(1)) < f(ix(k, 2), ix(k, 1))
        ix(k + 1, :) = ixtemp;
    else
        ix(k + 1, :) = ix(k, :);
    end

    ff(k) = f(ix(k, 2), ix(k, 1));
end

% The rest of this just creates 2 diagrams as seen in the lecture
% slides(Optimazation slide 43)
figure;
contour(x, y, f);
hold on;
scatter(x(ix(:, 1)), y(ix(:, 2)), 'ok', 'filled');
hold off;
xlabel('x');
ylabel('y');
title('Hill diagram');


figure;
scatter(x(ix(:, 1)), y(ix(:, 2)), 'ok', 'filled');
xlabel('x');
ylabel('y');
title('Scatter Plot');
```

Output on Matlab

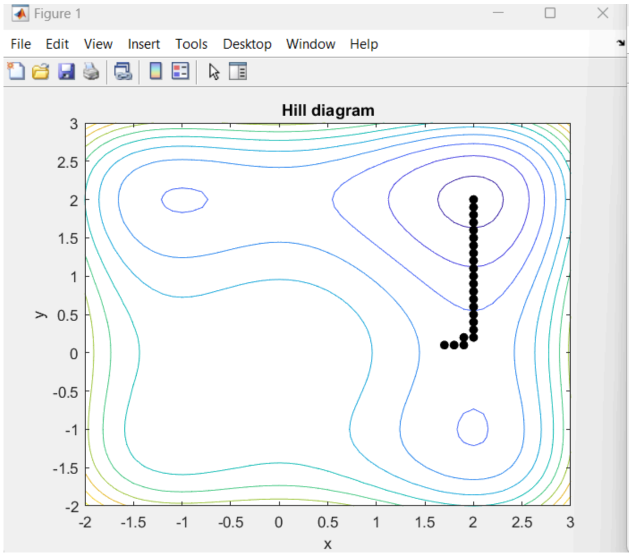
Figure 1
File Edit View Insert Tools Desktop Window Help
Hill diagram
3
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
y
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2
2.5
3
x

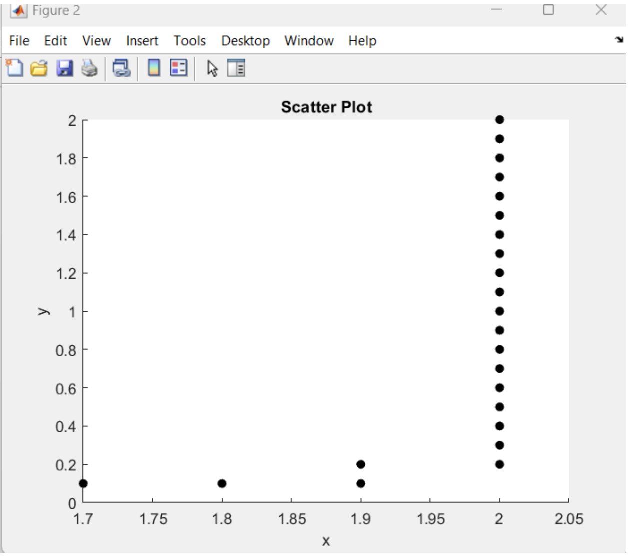
Figure 2
File Edit View Insert Tools Desktop Window Help
Scatter Plot
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
y
1.7
1.75
1.8
1.85
1.9
1.95
2
2.05
x