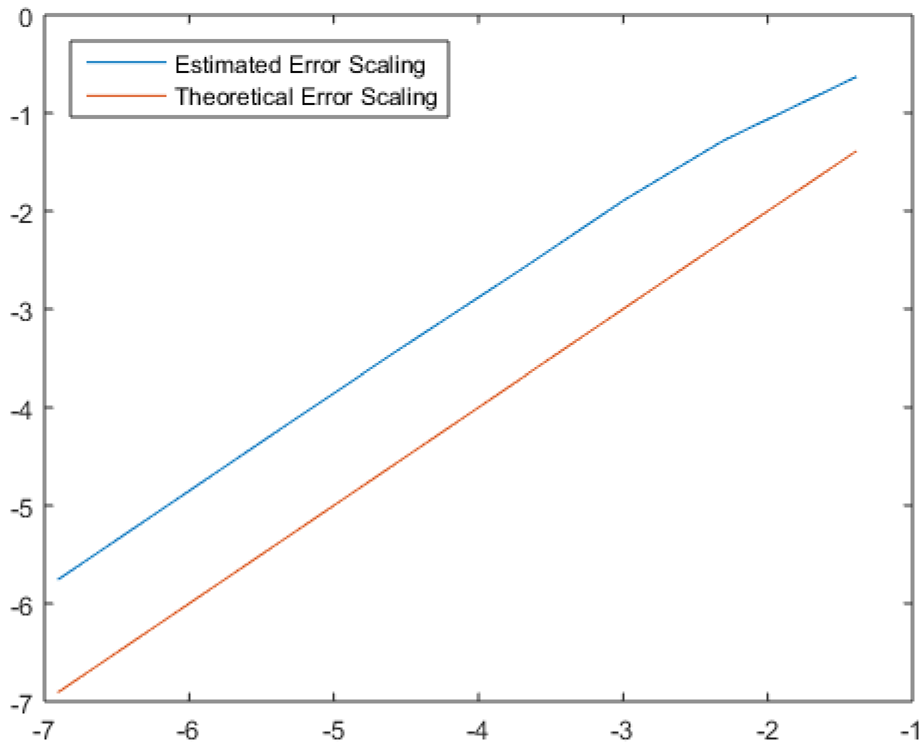
```
% Computer Lab 6
% Exercise 1 (c.)
% NB the following code is very inefficient but hopefully clear at
 least
h = [0.25; 0.1; 0.05; 0.025; 0.01; 0.005; 0.001]; % vector of
 successively smaller step sizes
M = 10000; % number of simulations/paths per value of h
mu = 1;
T = 1;
sigma = 1;
X_0 = 1;
N = round(T./h); % number of grid points per simulation
E_h = zeros(length(h),1); % store MC estimates of E[|X_T - X_T^h|]

for ii = 1:length(h)
    temp = zeros(M,1); % for MC estimator
    for jj = 1:M
        X = zeros(N(ii),1); % vector to store the path
        dB = sqrt(h(ii,1)).*normrnd(0,1,N(ii),1); % Brownian
 increments
        X(1,1) = X_0;
        % simulate the path of X for this value of h using Milstein
 scheme
        for n = 1:N(ii,1)-1
            X(n+1,1) = X(n,1) + h(ii,1)*mu*X(n,1)...
                + sigma*X(n,1)*dB(n+1,1) + ...
                0.5*sigma*sigma*X(n,1)*(dB(n+1,1).^2 - h(ii,1));
        end
        dB(1,1) = 0;
        B_T = sum(dB);
        X_T = X_0*exp((mu-(sigma^2)/2)*T + sigma*B_T);
        temp(jj,1) = abs(X_T - X(N(ii,1),1));
    end
    E_h(ii,1) = mean(temp); % MC estimate of E[|X_T - X_T^h|]
end
% To see the convergence order make a log-log plot
plot(log(h),log(E_h));
hold on;
plot(log(h),log(h)); % Lines should be parallel both with slope 1 -
 why?
legend('Estimated Error Scaling','Theoretical Error
 Scaling','Location','NorthWest');
disp('The estimates of covergence order of the scheme are:')
diff(log(E_h))./diff(log(h)) % estimates of convergence order
```

```
The estimates of covergence order of the scheme are:

ans =

   0.706079789999368
   0.885531473249973
   0.994577780780124
```

```
0.967446666419542
0.984965399047143
0.998899467265363
```



Published with MATLAB® R2015a