% In-Lecture #4 Assignment related to Homework 5.1 Steepest Descent

% This assignment introduces steepest descent algorithms.

% Please see Figure 6.15 at the top of page 116 in JSK's *Software Receiver*
% *Design* book.

% Also, see the introduction to the steepest descent algorithms marker board explanation from the homework #5 hints for problem 5.1.

% Also, see problem 2.1 on Spring 2016 Midterm #2.

% Consider performing an iterative minimization of objective function
% $J(x) = x$^2 – 14x + 49 = (x - 7)^2
% via the steepest descent algorithm (JSK equation (6.5) on page 116).
% $x[k+1] = x[k] - \mu \left.\frac{dJ(x)}{dx}\right]_{x=x[k]}$

% a. Visualize and analyze the shape of the objective function $J(x)$.
% 1) Plot $J(x)$ for $5 < x < 9$. Give the Matlab code for your answer.

```
x = [5 : 0.01 : 9];
J = x.^2 - 14*x + 49;
figure;
plot(x, J);    %% At end of document
```

% 2) Describe the plot.
% *Answer:* It's a concave up parabola (bowl)
% 3) How many local minima do you see?
% *Answer:* 1 at $x = 7$
% 4) Of the local minima, how many are global minima?
% *Answer:* The local minimum is also a global minimum.

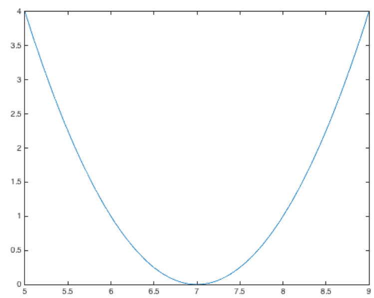
% b. As first step in deriving steepest descent update equation,
% compute the first derivative of $J(x)$ with respect to x.
% *Answer:* dJ(x)/dx = 2x - 14

% c. Implement the steepest descent algorithm in Matlab with $x[0] = 5$.
% 1) What value of x did steepest descent reach in 50 iterations with mu=0.01?
% *Answer:* $x = 6.2568$
% 2) What value of x did steepest descent reach in 50 iterations with mu=0.1?
% *Answer:* $x = 7.0$
% 3) Is the above value the global minimum of $J(x)$? Why or why not?
% *Answer:* Yes, the objective function has only one minimum.

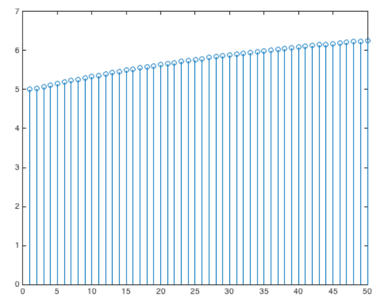
```
% polyconverge.m find the minimum of J(x) via steepest descent
N=50;                           % number of iterations
```

```
mu=0.01;                          % algorithm stepsize
x=zeros(1,N);                     % initialize sequence of x values to zero
x(1)=5.0;                         % starting point x(1)
for k=1:N-1
  x(k+1)= x(k) - (2*x(k)-14)*mu;      % update equation
end
figure;
stem(x);              % to visualize approximation
x(N)
```

Plots for mu = 0.01

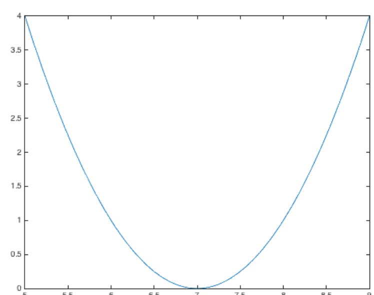


Plot of $J(x)$ vs. x

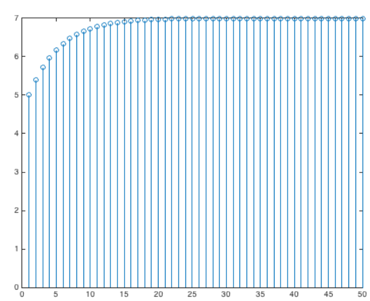


Plot of x vs. iterations

Plots for mu = 0.1



Plot of $J(x)$ vs. x



Plot of x vs. iterations