```
%%%%%%%%% Math 430 - Homework 7 %%%%%%%%%%%%
%
% MONTES Virginie
% Due November 6
%
%%Problem 2:
```

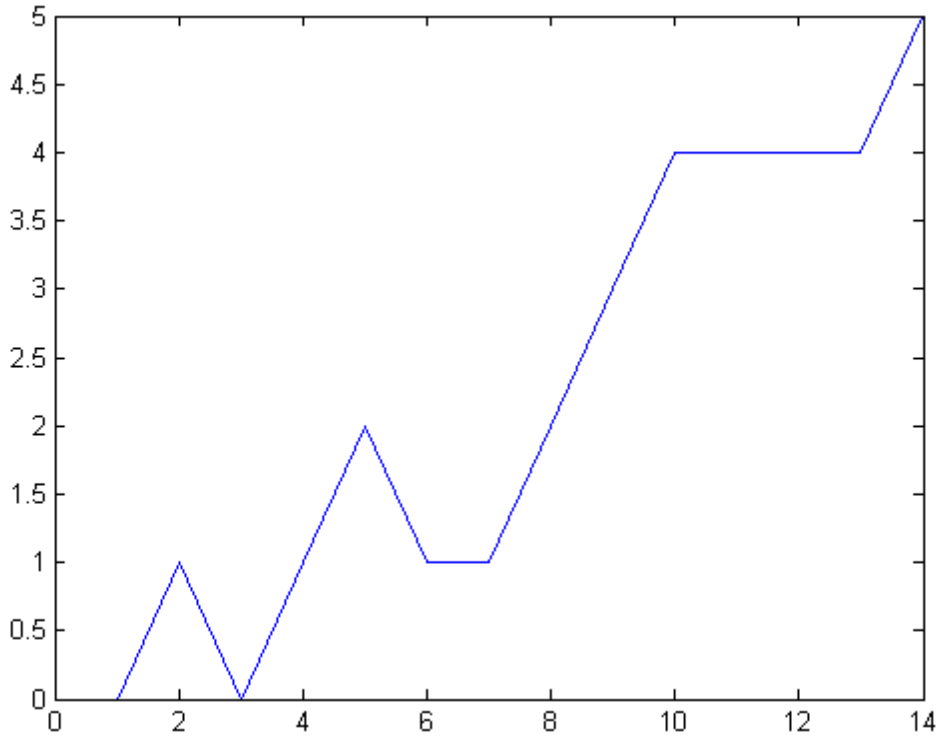
Mars rover problem : from the HW6

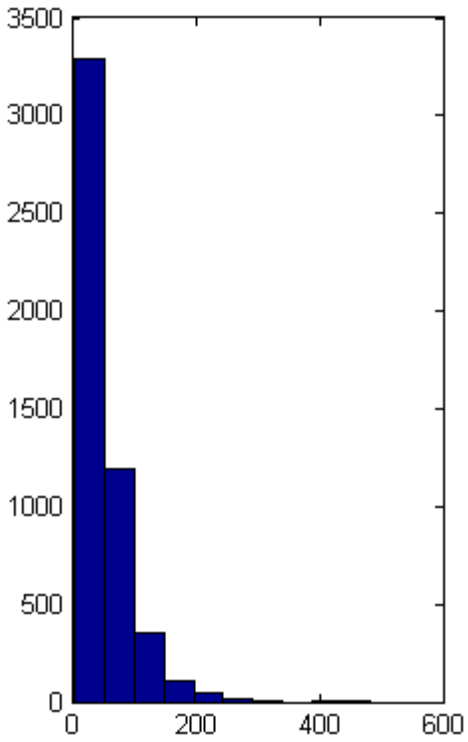
```
% 4 directions

clear all
close all


M1= 5000;  % the size of MC run

% setting up the simulation results vectors
fall1 = zeros(M1,1);
duration1 = zeros(M1,1);

durmax1 = 500;   % max number of step allowed - for graphing purposes
X1 = zeros(durmax1,1);

upper1 = 5;      % stopping conditions for the random walk
lower1 = -5;
left1 = -5;
right1 = 5;

for m1 = 1:M1,
    t1 = 1;
    X1(1) = 0;
    while ( (X1(t1) < upper1) & ( X1(t1) > lower1) & (t1 < durmax1) & (X1(t1) < ri
       t1 = t1+1;
       X1(t1) = X1(t1-1) + (round(rand)*2 -1)/2 + (round(rand)*2 -1)/2;

    end %   while loop
    if ((X1(t1) == upper1) & (X1(t1) == right1)) ,  fall1(m1) = 1;   end
    duration1(m1) = t1;

end   % Monte Carlo loop

figure(1)
 plot(1:t1, X1(1:t1))

% output the estimates

figure(2)
subplot(1,2,1)
  hist(duration1)

[px1,x1] = ecdf(duration1);
  subplot(1,2,2)
  stairs(x1,px1);

phat1 = sum(fall1)/M1;
meandur1 = mean(duration1);

disp([' Est. Probability of fall (4 directions)= ', num2str(phat1), ', average dur

sigmadur1 = sqrt(var(X1));
```

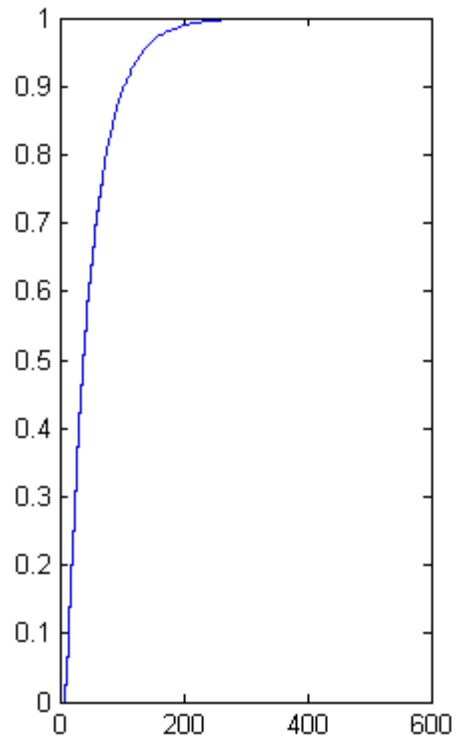
```
 mofe1 = 2*sigmadur1/sqrt(M1);
  disp(['95% CI for the mean flight length ',...
       num2str(meandur1 - mofe1), ' to ', ...
       num2str(meandur1 + mofe1)]);


   % Mars rover problem: all directions

M2 = 5000;  % the size of MC run

% setting up the simulation results vectors
fall2 = zeros(M2,1);
duration2 = zeros(M2,1);

durmax2 = 5000;   % max number of step allowed - for graphing purposes
X2 = zeros(durmax2,1);

upper2 = 5;     % stopping conditions for the random walk
lower2 = -5;
left2 = -5;
right2 = 5;

for m2 = 1:M2,
    t2 = 1;
    X2(1) = 0;
    while ( (X2(t2) < upper2) & ( X2(t2) > lower2) & (t2 < durmax2) & (X2(t2) < ri
       t2 = t2+1;
       X2(t2) = X2(t2-1) + rand - rand ;

    end %   while loop
    if ((X2(t2) == upper2) & (X2(t2) == right2)) ,  fall2(m2) = 1;   end
    duration2(m2) = t2;

end   % Monte Carlo loop

figure(3)
 plot(1:t2, X2(1:t2))

% output the estimates

figure(4)
subplot(1,2,1)
  hist(duration2)

[px2,x2] = ecdf(duration2);   % this command does empirical CDF
  subplot(1,2,2)
  stairs(x2,px2);

phat2 = sum(fall2)/M2;
meandur2 = mean(duration2);

disp([' Est. Probability of fall (all directions) = ', num2str(phat2), ', average
sigmadur2 = sqrt(var(X2));

 mofe2 = 2*sigmadur2/sqrt(M2);
  disp(['95% CI for the mean flight length ',...
       num2str(meandur2 - mofe2), ' to ', ...
       num2str(meandur2 + mofe2)]);


%Comparing the two scenarios to obtain a confidence interval for the
%difference of the average time before falling off the plateau
```

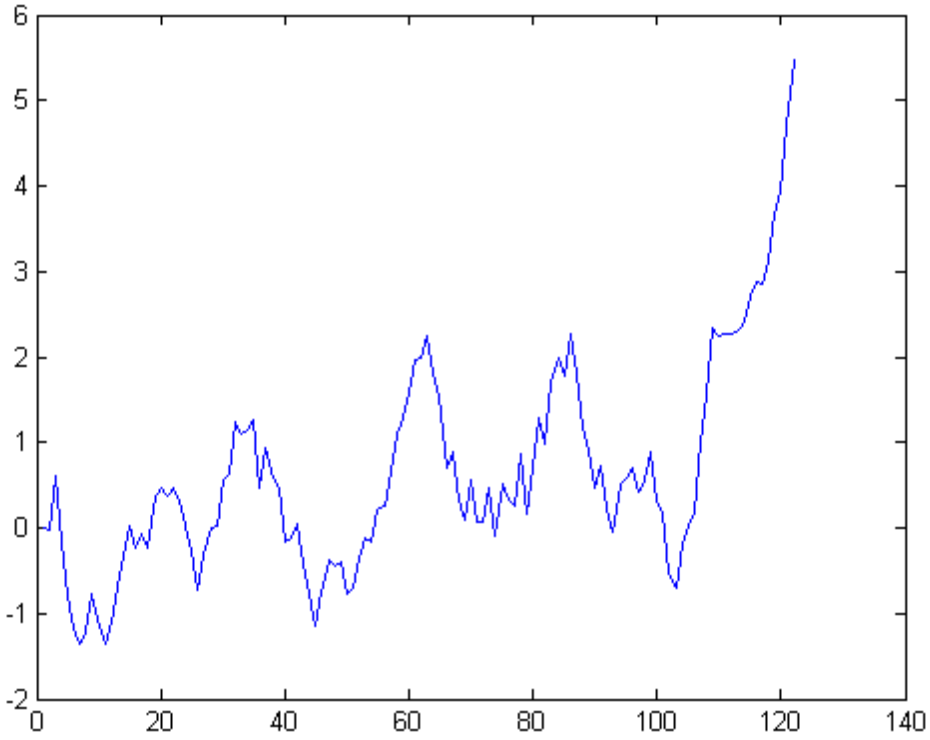
```
mofe3 = 2*sqrt(sigmadur1^2/M1 + sigmadur2^2/M2);
 disp(['95% CI for the mean difference of the average time ',...
       num2str(meandur1 - meandur2 - mofe3), ' to ', ...
       num2str(meandur1 - meandur2 + mofe3)]);


   % There is a evidence that one scenario provides a longer time on
   % average because the result that we obtain is negative. We can say that
   % the scenario 2 (all directions) provides a longer time.
```

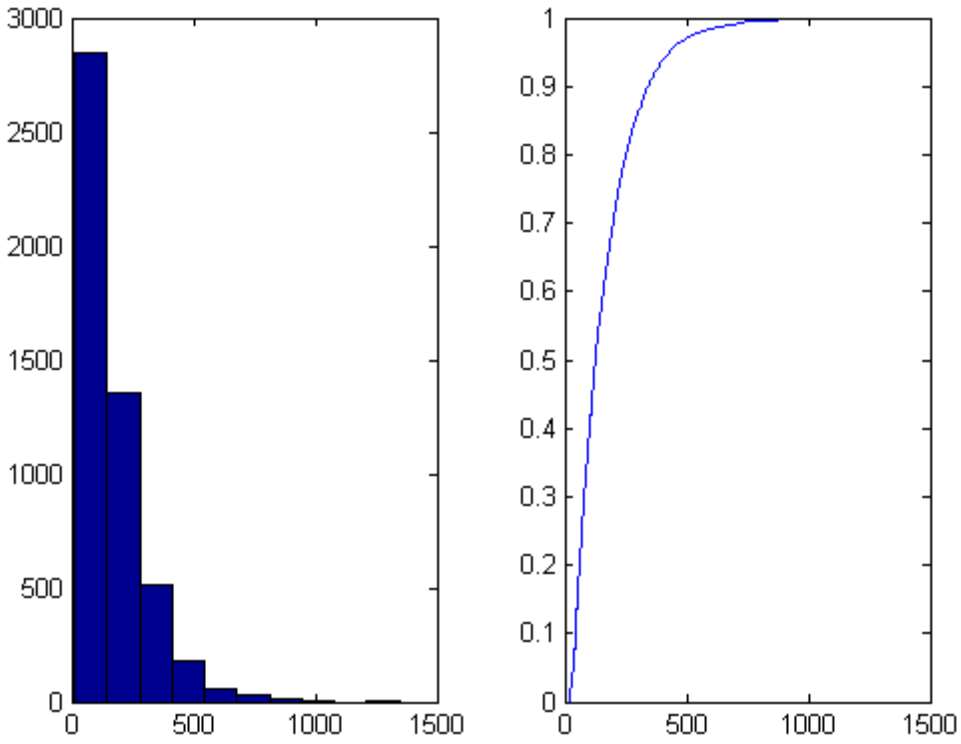
```
 Est. Probability of fall (4 directions)= 0.495, average duration= 51.1112
95% CI for the mean flight length 51.053 to 51.1694
 Est. Probability of fall (all directions) = 0, average duration= 167.6376
95% CI for the mean flight length 167.6068 to 167.6684
95% CI for the mean difference of the average time -116.5922 to -116.4606
```




3500
3000
2500
2000
1500
1000
500
0
0
200
400
600


1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
200
400
600


6
5
4
3
2
1
0
-1
-2
0
20
40
60
80
100
120
140



Published with MATLAB® 7.9