ENED 1091: Engineering Models II
Homework Assignment #1
Due: Week of Feb 26th at the beginning of your Recitation Section

Problem 1: Array Indexing

For this problem, you will need to download the HW1.mat file from Blackboard. Drop it in your current MATLAB folder then type: `load HW1` at the MATLAB command prompt. You should now have the following three variables (x, y, and M) in your workspace with the values shown below. Don't make any modifications to these variables.

```
x = 4    5   -1    5    3   -2    0    2    5    5   -1    5    5    3    4

y = -1   5    5    4    5    3   -2    4    5    3    4    3    1    3   -1

M =
      2      2      3      2     -4
     -5     -2      7      3      0
     -2      5     -3     -2      5
     -5     -5      0      2     -2
     -4     -1     -1      2      1
      4     -1      2     -4     -3
```

For all parts of this problem, don't put a semicolon at the end of your command so you can copy the results as well as your MATLAB command.

a) What MATLAB command will pull out entries 7 thru 12 in vector, x, and store them in a vector called a?
Command and Results:

a = x(7:12)

a =

0 2 5 5 -1 5

b) What MATLAB command will pull out every other entry of the vector, x, starting at entry 4 and ending at entry 12 and store these entries in a vector called b?
Command and Results:

b = x(4:2:12)

b =

5 -2 2 5 5

c) What MATLAB command will pull out the7 in matrix, M, and store it in a scalar variable called c?

Command and Results:

```
c = M(2,3)

c =

     7
```

d) What MATLAB command will pull out row 3 of the matrix, M, and store these values in a row vector called d?

Command and Results:

```
d = M(3,:)

d =

    -2     5    -3    -2     5
```

e) What MATLAB command will pull out column 4 of the matrix, M, and store these values in a column vector called e?

Command and Results:

```
e = M(:,4)

e =

     2
     3
    -2
     2
     2
    -4
```

f) What MATLAB command will pull out the section of matrix, M, shown below and store these values in a matrix called f?

```
    -2     7     3
     5    -3    -2
    -5     0     2
    -1    -1     2
```

Command and Results:

```
f = M(2:5,2:4)

f =
```

```
-2    7    3
 5   -3   -2
-5    0    2
-1   -1    2
```

Problem 2: Functions for Creating Arrays (randi, rand, ones, zeros)

For all parts of this problem, don't put a semicolon at the end of your command so you can copy the results.

a) What MATLAB command will create a vector, a, that has 12 random integers with values that fall within the range [−3 7]?
Command and Results:

```
a = randi([-3,7],1,12)

a =

    5    6   -2    7    3   -2    0    3    7    7   -2    7
```

b) What MATLAB command will create a vector, b, with 500 zeros in it?
Command (Don't paste the results!):

```
b = zeros(1,500)
```

c) What MATLAB command will create a vector, c, with 200 sevens in it?
Command (Don't paste the results!):

```
c = 7*ones(1,200)
```

d) Use the ***rand*** command to create a vector, d, of 10 decimal numbers that fall in the range of 3 to 8.
Command and Results:

```
d = 3+5*rand(1,10)

d =

    6.9760    3.9344    5.4488    5.2279    6.2316    6.5468    6.7734    4.3801    6.3985    6.2755
```

e) What MATLAB command will produce a 3x4 matrix, e, that has random integers with values that fall within the range [−5   5]?
Command and Results:

```
e = randi([-5,5],[3,4])
```

e =
-2 4 5 -1
-5 2 -5 3
-4 -2 -1 3

Problem 3: More Loop Practice

For this problem, you will need the same x and y vectors from Problem 1. Create a script file and include the command: `load HW1` in your script. Use a ***for loop*** to accomplish the following:

- Create a vector z that has entries with the maximum of x and y (i.e.. if x(k) exceeds y(k) then z(k) equals x(k) otherwise z(k) = y(k)). ***Display vector z after the for loop.***
- Count how many entries in vector x are greater than the corresponding entries in y. ***Display the count after the for loop.***
- Count how many entries in vector y are greater than the corresponding entries in x. ***Display the count after the for loop.***
- Create a vector, called Location, that stores the locations (positions in the vectors) where the x and y values are equivalent. ***Display Location after the for loop.***

MATLAB Script and Results:

Script:

```
%% HW1 P3
% Name: Horace
% Date: 19 Feb 2019

%% Code
%clear processor
close all; clc;

load HW1;
z = zeros(1,15);
i = 0;
j = 0;
same = 0;
for k = 1:15
 if x(k) > y(k)
     z(k) = x(k);
     i = i+1;
 elseif x(k) < y(k)
     z(k) = y(k);
     j = j+1;
 else
     z(k) = y(k);
     same = same+1;
     location(same)=k;
```

```
 end
end
fprintf('vector z:');
disp(z);
fprintf('the count that x are greater than the
corresponding entries in y:');
disp(i);
fprintf('the count that y are greater than the
corresponding entries in x:');
disp(j);
fprintf('the location:');
disp(location);
```

Results:

vector z: 4 5 5 5 5 3 0 4 5 5 4 5 5 3 4

the count that x are greater than the corresponding entries in y: 7

the count that y are greater than the corresponding entries in x: 5

the location: 2 9 14

Problem 4: Plotting Data from a Matrix

For this problem, you will need to download the excel file: Height posted on Blackboard. This file has projectile height data versus time for five different launch angles (25°, 35°, 45°, 55°, and 65°) assuming an initial velocity of 100 m/s.

Import the row of time values as a vector (1x191) into MATLAB using either xlsread or the import tool. Import the height data as a matrix (5x191) into MATLAB using either xlsread or the import tool.

(a) At the MATLAB command prompt, type a MATLAB command that will display the first six columns in your height matrix.

Command and Results:

```
%% HW1 P4 (a)
% Name:  Horace
% Date: 19 Feb 2019

%% Code
%clear processor
clear;clc;
```

```
A = xlsread('Height.xlsx',1,'B4:G8');
disp(A)
```

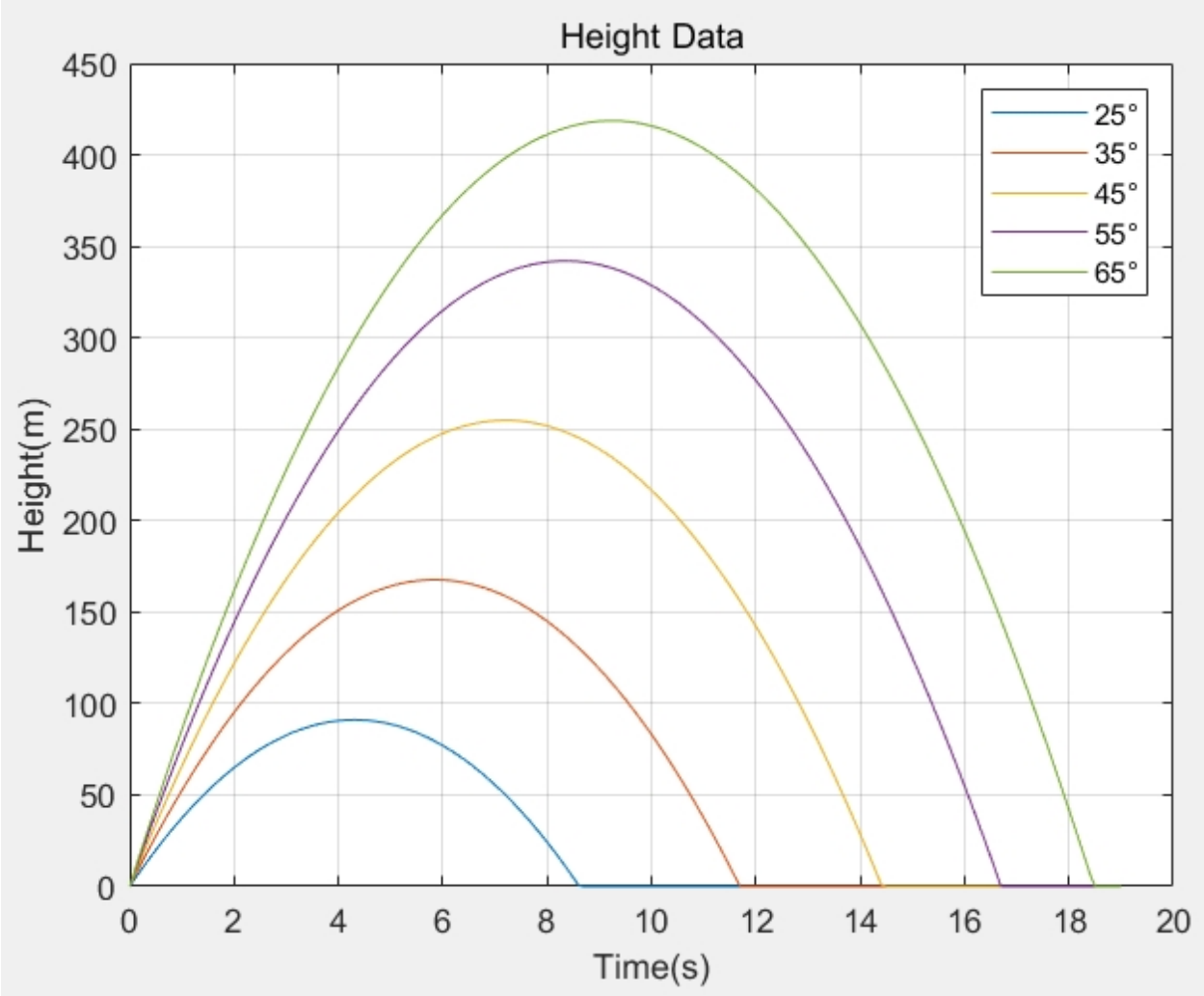
A =

0	4.1771	8.2562	12.2371	16.1199	19.9047
0	5.6867	11.2753	16.7658	22.1583	27.4526
0	7.0220	13.9459	20.7718	27.4995	34.1291
0	8.1425	16.1868	24.1331	31.9813	39.7314
0	9.0140	17.9300	26.7478	35.4675	44.0891

(b) Plot time on the x-axis and height on the y-axis for all five angles. Add a legend showing each launch angle. Add labels **(include units)** and a title.

MATLAB Commands and Plot:

```
T = xlsread('Height.xlsx',1,'B1:GJ1');
A25 = xlsread('Height.xlsx',1,'B4:GJ4');
A35 = xlsread('Height.xlsx',1,'B5:GJ5');
A45 = xlsread('Height.xlsx',1,'B6:GJ6');
A55 = xlsread('Height.xlsx',1,'B7:GJ7');
A65 = xlsread('Height.xlsx',1,'B8:GJ8');
figure(1)
plot(T,A25)
hold on
plot(T,A35)
plot(T,A45)
plot(T,A55)
plot(T,A65)
xlabel('Time(s)')
ylabel('Height(m)')
title('Height Data')
legend('25¡ã','35¡ã','45¡ã','55¡ã','65¡ã')
grid on
```


Height Data
Height(m)
Time(s)
25°
35°
45°
55°
65°
0
50
100
150
200
250
300
350
400
450
0
2
4
6
8
10
12
14
16
18
20