COMP1200-MatLab - Lab 04
Due 11:59pm – Thursday – September 22
Submit `Lab04.m` and `devPlan04.txt` **via Blackboard**

Before you start writing your program:

Read these instructions carefully. The **`Lab04.txt`** file. at the assignment link is an incomplete development plan. You are to save the file and edited by adding your name and the date and by completing: 1. STATE THE PROBLEM and 2. DESCRIBE THE INPUT AND OUTPUT REQUIREMENTS. Use the development plan as a guide when writing the m-script file solution for the following problem. **This file must be saved as a .txt file.**

NOTE: You will see later that the spelling and casing of file names is very important in MATLAB. Your submitted file(s) MUST be spelled and cased as instructed. [-5 points per file for not doing so.]

***Problem*:**

***Program*: Lab04.m**

On a hot Saturday afternoon, you and your friends notice an empty baby swimming pool on the lawn of your apartment complex. So, why not see if you can fill it with water from water balloons thrown from your second floor balcony.

In Lab03, you will continue computing the distance a water balloon will travel given the balloon launch angle (theta) in degrees, balloon launch velocity (v) in min/sec, and the thrower's height in feet.

You are to write a MATLAB program (m-script file) that find the distances of the water balloons that hit, fall short of, and go past the pool. Some statements in Lab03 can be used in Lab03. Do not include statements from Lab02 that do not apply to Lab04.

INPUT: You will, again, use the `rand()` function to build a 1 x 7 vector of random numbers for each of the following: angle theta, velocity, and thrower's height. Note the minimums and maximums given below are CONSTANT values.

	minimum	maximum	
theta	5	85	degrees
velocity	1	30	ft/sec
thrower's height	4.5	7.0	feet

Compute a vector of horizontal distances using the input vectors and combine the four vectors to create a table (2-D matrix). Find the distances values in the matrix that hit, fall short of, and go past the pool.

Display
- the matrix containing theta, velocity, thrower's height, and distance with column headers.
- the distances that are too short. If the vector is empty, display a message.
- the distances that are too long. If the vector is empty, display a message.
- the distances that hit the pool. If the vector is empty, display a message.

Determine if the average distance is too short, too long, or hits the pool and display an appropriate message.

***Problem Constants*:**

```
BAL_HEIGHT = 12;    % balcony height in feet
G = 32;             % gravitational acceleration in ft/s2
POOL_CENTER 35      % distance to the center of pool
```

and the minimum and maximum values

***Problem Inputs*:**

balloon launch angle (theta) in degrees
balloon launch velocity (v) in ft/sec
thrower's height in feet

Problem Outputs:

distances that a water balloon will travel
the matrix containing theta, velocity, thrower's height, and distance with column headers.
the distances that are too short

the distances that are too long
the distances that hit the pool
average distance
message about average distance: Too short! `< POOL_CENTER - 1`
Too long! `> POOL_CENTER + 1`
It's a hit!

Instructions:

- ☐ Insert comments at the top and throughout each file
  - Include the follow comments at the beginning of this (and ALL) files.
    - **%** your name
    - **%** assignment number
    - **%** date you completed the assignment
    - **%** a short narrative about what the file does
  - Use your development plan algorithm steps as a guide for comments throughout each file
- ☐ Use `clc` and `clear all` at the beginning of your program.
- ☐ Use descriptive variable names.
- ☐ Use Sample Input/Output as a guide.
- ☐ No extra output, i.e., use semicolons!
- ☐ The input values are vector of random number.
- ☐ Compute horizontal distance(s).
  - It helps to break the large equation into parts and solve for each.
  - The results of some parts will be a vector. Keep this in mind when using multiplication. Note: Squaring a variable is multiplication.
  - Combine the parts to compute the distance.
- ☐ The output should be labeled
  - Use `disp()` to print the labels and column headers strings.
- ☐ Output:
  - Numbers should have two decimal places using the format command.
  - Use `disp()` to print the labels and column headers strings.
  - Display a blank line where necessary to separate output.

-5 points per file for absence of any of these required comments at the top

New commands:
`find`
`isempty`
`if-elseif-else`

Run your problem several times to observe values in all categories.

Sample Input/Output: **NOTE: You numbers may be different.**

Example 1:

```
Compute a distance for each set of input
    theta     velocity    thrower_ht  distance
    47.59       16.91        6.89       17.61
    62.32        2.41        6.84        1.29
    19.34       17.03        6.55       20.36
    31.92        8.97        6.32        9.35
    20.02        8.00        4.94        8.41
    30.75        8.05        5.40        8.16
    37.31        5.47        4.97        4.95

Distances that are too short
    17.61
     1.29
    20.36
     9.35
     8.41
     8.16
     4.95

No distances are too long

No balloons hit the pool.

Average distance:
    10.02

The average distance is too short.
```

Example 2:

```
Compute a distance for each set of input
    theta     velocity    thrower_ht  distance
    74.11        1.13        4.75        0.33
    12.82       23.23        5.77       27.80
    77.64       25.61        5.96       11.50
    13.64       27.59        6.41       34.72
    46.36       29.62        4.71       38.67
    16.45       15.65        6.15       18.20
    49.75        8.87        5.79        7.38

Distances that were too short
     0.33
    27.80
    11.50
    18.20
     7.38

Distances that were too long
    38.67

Distances that hit the pool
    34.72

Average distance:
    19.80

The average distance is too short.
```

Submit via Blackboard:

`Lab04.m`	MATLAB script file
`devPlan04.txt`	development plan