CIEG 675
LAB #2 Due **Monday January 18, 2021**

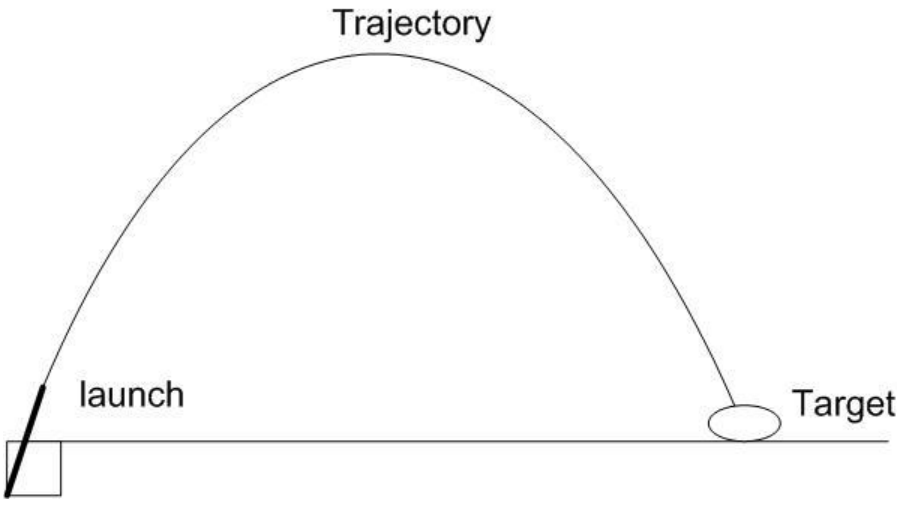
1) There used to be a game on the old Radio Shack computers where you were responsible for providing an initial trajectory angle to a munitions expert so they could try to hit a target with a mortar round at some known distance away (see picture below).

Develop a user-defined **function** that will take as input
the mortar firing angle,
the distance to target and
the initial mortar velocity (those better in matlab can also have the target elevation as an input; others: just assume the mortar launch point elevation and target elevation are the same at zero).

The function should plot to a figure window the mortar trajectory at various stages of flight while using the **pause(n)** function to see the intermediate positions. Assume a HIT if the mortar strikes within 10 meters of the target location. You should indicate at the end of the simulation if the target was hit or not and the distance away from the target the mortar landed. This may include altering the plot, and/or including text on the plot, and/or including text in the command window.

Neglect air resistance, Coriolis, earth surface curvature etc.

We can turn this into a movie with information we learn later



$$X = V_x t = V\cos(\theta)t$$

$$Y = V_y t = V\sin(\theta)t - \frac{1}{2}gt^2$$

2) Write a user-defined function (call it **`circleplot or similar`**) that will take, as input, the coordinates for the center of a circle along with its radius, and then plots the circle. It should also plot a marker in the center of the circle (your choice of marker type and color). Make the `axis equal` so it actually looks like a circle. As a second step, add, as an input to your function, a color (i.e. `'k'` or `'r'`). Then use the `fill` command to fill in the circle with that color. **Demonstrate your function works by including 2 plots of 2 different circles (different center coordinates, different radii and different fill colors).**

WARNING: you can't just use the marker 'o' for this problem.

3) Make a plot of your choosing. Then, using `set` and `get` commands,

- Increase the figure window width and height to something larger than default
- Change the figure Color to [1 1 1] (white)

  NOTE: colors are defined in MATLAB using RGB as a 3x1 vector of numbers corresponding to the level of Red, Green and Blue (in that order) from 0 to 1, which is just the relative level from the typical scale of RGB from 0 to 255. For example, [0 1 0] is green. Check out this really interesting YouTube video if you are further interested in the subject of colors on computer screens: https://youtu.be/LKnqECcg6Gw ☺

- Change the XTicks to something other than what they were
- Change the YTicks to something other than what they were
- Change FontSize of tick labels on the x- and y-axis to 14 pt. (i.e. xticklabel)
- Change the FontName to Times New Roman
- Change the tick direction to out
- Add a title, xlabel and ylabel. Make them bold and FontSize 16
- Turn the box off (must do this last!)
- Save (print) the figure to a folder other than the current folder **WITHOUT** changing the current folder you are in. That is, save to a new folder in the actual save call using the full file pathname