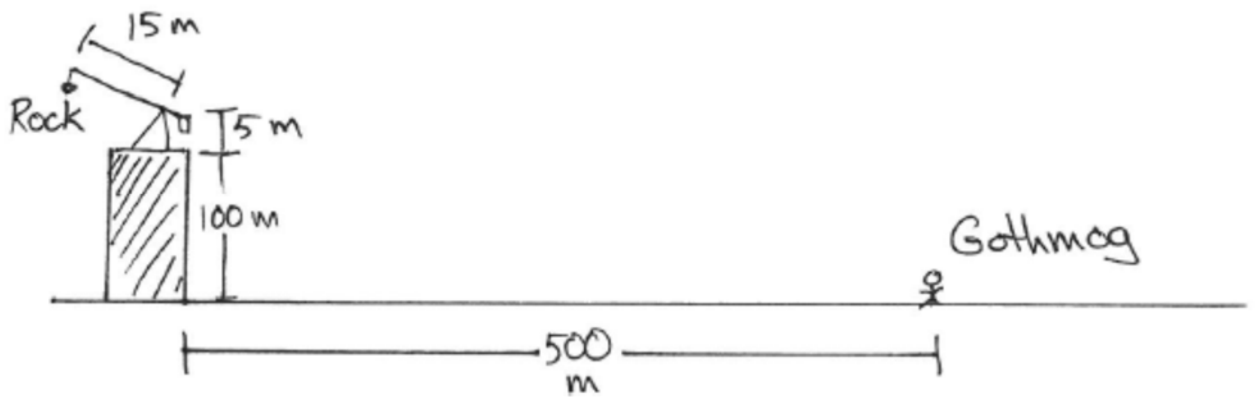
BMME 201 Summer 2018
Programming Assignment #2: Trebuchet
Due: Wednesday, 5/30

Background

Lobbing rocks from a trebuchet of Gondor during the Battle of the Pelennor Fields is tough work. Orcs, battering rams, and siege towers are all over the place, and those big dragons keep trying to pick us up and drop us off the wall. The general said something about a pair of Hobbits with a fancy ring who think that one simply walks into Mordor instead of flying with the eagles, but all I care about is hitting that ugly Orc general Gothmog with this big rock.

The trebuchet launches 100 kg spherical rocks from atop a wall rising 100 m above the battlefield. The pivot on the trebuchet is 5 m above the wall, and the arm is 15 m long. To simplify the problem, you may ignore the sling that holds the rock and assume that the initial vertical position of the rock is always 120 m. Gothmog is at a horizontal distance of 500 m away from the release point of the rock. You may neglect the effects of wind, friction, and Balrogs.



Problem

The trebuchet crew can control the angle of release of the rock and the release velocity. Solve the problem of squishing Gothmog in two ways:

<u>Fix the release angle at 45°</u>

- Calculate the x distance traveled as a function of the release velocity magnitude ranging from 0 to 100 m/s.
- Plot the x distance traveled as a function of release velocity magnitude. On the same plot, draw a horizontal line at the target distance.
- Look at the intersection(s) between the two lines. Hardcode these values in your declarations section.

<u>Fix the release velocity magnitude at 75 m/s</u>

- Calculate the x distance traveled as a function of release angle ranging from -90° to 90°.
- Plot the x distance traveled as a function of release angle. On the same plot, draw a horizontal line at the target distance.
- Look at the intersection(s) between the two lines. Hardcode these values in your declarations section.

Finally, using your results from parts 1 and 2, make a single plot of y vs. x showing the trajectories of the rock. They should all start at the height of the trebuchet and end at the point y = 0 at the distance to the target.

Output

When run, your code should print the answers to the two different situations using `fprintf` with some explanatory text and sufficient significant figures. It should also make three separate plots:

i. Plot of x distance vs. release velocity magnitude (fixed angle of 45°)
ii. Plot of x distance vs. release angle (fixed release velocity magnitude of 75 m/s)
iii. Plot of y distance vs. x distance for all solutions

Strategy

While you can do this problem in several ways, here is a suggestion:

- Define a custom function that calculates the x distance traveled given initial kinematic parameters. This function takes as its input the release velocity magnitude (scalar or vector), release angle (scalar or vector), initial x position, and initial y position. It returns a vector of total x distance traveled.
- Define another custom function that calculates the x and y positions for the trajectory. It is similar to the previous function except that it also takes an input of the number of subdivisions between t = 0 and t = time of flight. It returns two vectors with the x and y positions at each time.
- Now organize your code. Start with a declarations section where you define ALL numerical variables. Next, solve the fixed release angle problem. Then, solve the fixed release velocity problem. Lastly, have one section where you print your results to the command window, and then another section with the three plots.

Note that custom functions are NOT required but a good suggestion. You will not be penalized if you don't do them; however, they will help your code clean and organized.

Scoring

Doing all of the above steps perfectly will receive a maximum of 80/100. For a higher score, implement any of the three changes below:

- (10 pts) Instead of identifying the best velocity/angle by looking at the graph yourself and hardcoding the values into your output code, write code to solve for it automatically. See the hint later. We will test it by changing the variable for the distance to Gothmog ONLY at the beginning of your code in your Declarations, and EVERY plot and output should be correct (e.g., ranges on plots should autoscale, the horizontal line in graphs 1 and 2 should adjust, etc.).

- (5 pts) Now worry about the trebuchet's release position. For example, if it releases the rock at 0°, then the initial (x,y) = (0,120); if it releases at 90°, then (x,y) = (-15,105). You code should account for this effect in ALL calculations.
- (5 pts) Add a small Gothmog at the target location in your trajectory plot. You can either draw a stick figure or insert a graphic. Again, he should move when we change the distance to Gothmog variable if you did the first part of the bonus points.

Theory

Recall the motion of an object discussed in mechanics. Acceleration (*a*), velocity (*v*), and displacement (*s*) are related by derivatives/integrals. For the y component:

$$a_y(t) = g$$
$$v_y(t) = \int a_Y(t)\,dt = gt + v_{y0}$$
$$s_y(t) = \int v_y(t)\,dt = \frac{1}{2}gt^2 + v_{y0}t + s_{y0}$$

where *t* is time, *g* is gravity (-9.81 m/s^2, even in Middle Earth), v_0 is the initial velocity, and s_0 is the initial displacement. Some specialized versions of these equations exist, but you can derive everything necessary in this problem from these three. The analogous equations in the x direction are simpler, since there is no a_x component.

Hints

- Time of flight is determined solely by the y component. To find the total horizontal distance traveled, find the time t_0 when s_y equals 0 and evaluate $s_x(t_0)$.
- Part 2 has two answers. Be sure to find them both.
- When displaying angles using `fprintf`, you can make a degree symbol using `%c` and `char(176)`; e.g., 45°C is `fprintf('%2.0f%c\n',45,char(176))`. You can also copy/paste that character into the .m file from another location.
- For the extra part of finding release velocity and angle automatically, you can use the function `min` with two outputs. The first output is the minimum value, and the second is the index at which this occurs. Look at a plot of (x distance traveled – 500) for the possible velocities or angles, and think of how you can modify it so that the minimum value of the plot is near 0. If you have multiple possible answers, then figure out an automated way to exclude the first answer that you find to get the second.
- Also for the first extra part, the step size of your time variable will determine the number of significant figures in your release angle/velocity.