Final Exam : I am tired waiting for take off.

Your goal is to provide the best firing solution (aim parameters) for the launch platform. 125 potential points total (100 for writeup + 25 for the problems solved during the final exam time).

During the in class final time time, each of you **individually** will have to solve slightly modified problems: run your problem on a modified data file, adapt your code to a slightly different problem. If you know what your codes does, you should be done in an hour.

Algorithm evaluation:

If your code runs more than 5 minutes on my rather old computer, I will consider it wrong even if it eventually produce a mathematically correct answer. You should aim for less than 2 minutes calculation time for the problem 3.

Your code should provide **parameters** for the catapult aim (see requirements below). To solve the problem, you need your own model simulation code, but it will not be used in during the evaluation. The simulation of the solution will be done with the instructor's computer and code using the input parameters supplied by your algorithm. So, make sure that physics is done correctly.

I will run your code on my computer with software set identical to the one in a classroom. If you want to check your code, use the computer classes in the library.

Written report requirements:

- All regular requirements outlined in the syllabus.
- Discuss relevant equations, describe your solution, show results.
- All Matlab code/scripts must be present in the carbon copy as well.
- Report should be less than 10 pages long (listings length is not counted).
- Make all your calculations in the S.I. units (m, kg, s).

The problem:

Aren't we all tired to wait for a take off time in an airport. This usually a place of the majors delay. Landing and taking off not only the most dangerous part of an airborne journey, but also the most stressful time for a plane's hardware, this is the time when something is likely to break and, then, repairs are needed which also contributes to delays.

Your start up decided to circumvent all of the above by removing necessity of planes to take off and land. Food, fuel, passengers, and even pilots will be delivered to the planes which never land with the help of a catapult. If someone arrives to a destination, she just takes a parachute and jumps, which is perfectly safe and can be done at your own desire. No more annoying "buckle your belt and switch off your electronics".

The whole enterprise works like this, planes at different heights fly over the catapult location one at a time, you on the ground aim the catapult with certain elevation angle above the horizon (measured in radians) and launch a capsule with a precious cargo toward the plane. The launch parameters, i.e., the speed, the launch time, and the elevation angle, must be calculated in such a way, that the capsule reaches the plane with zero relative velocity. If this is not satisfied, the capsule would crush the plane and itself, and you would lose all money in a law suit against you.

The model:

To succeed, you need to write a capsule flight simulator. A capsule is a sphere with a variable (from a capsule to another capsule) diameter (D); it has the launch/take off speed (V_e), i.e. the final speed of a capsule after escaping from the catapult at zero height; the capsule has the drag coefficient (C_d), and the total, combined with the cargo, mass (m).

With these parameters, a capsule could fly through the stratosphere with much smaller air density, so you need to take this into account, since future planes might fly higher than the current cruising altitude of 10 km. The crude approximation of the air density (ρ) dependence on the altitude (h) above the Earth's surface is

$$\rho(h) = \rho_0 2^{-h/h_{1/2}}$$

where $\rho_0 = 1.2$ kg/m^3 is the density of air at the Earth's surface, and $h_{1/2} = 6800$ m is the altitude at which density of air drops by factor of 2. Assume, that these 2 parameters are constant (it is unlikely that you would expand your business to other planets in any near future). Fortunately, you do not need to account for the curvature of the Earth or change of the acceleration due to gravity ($g = 9.80$ m/s^2). We will assume that only gravity and the air drag are acting on the flying capsule

$$\begin{aligned}
\vec{F} &= m\vec{a} = \vec{F}_g + \vec{F}_d \\
\vec{F}_g &= -mg\hat{y} \\
\vec{F}_d &= -\frac{1}{2}\rho(h) C_d A v \vec{v}
\end{aligned}$$

where $\vec{F}_g$ is the gravitational pull force, $\vec{F}_d$ air drag force, $\hat{y}$ is the unit vector in the 'y' direction (we assume it points up), A is the **cross sectional** area of the capsule, and $\vec{v}$ is the velocity of the projectile.

The plane, to which we need to land our capsule, moves strictly horizontally with a constant velocity in the positive $\hat{x}$ direction, and at time $t_0 = 0$ is directly above the catapult.

We assume that our catapult is surrounded by a flat landscape.

For the following assignments, make sure that you follow the exact specification of the functions.

Problem 1 (20 points):

Write a capsule flight simulator, which takes a given capsule launch parameters and calculates the x and y position of the capsule vs. time. The time **must** span from zero till the time when the capsule hits the ground, i.e., no underground motion and a trajectory truncation in mid-flight are permitted. Clearly, there is some uncertainty. So, the trajectory may be cut when `abs( y(time(end)) ) <= 0.001` m. The function must follow **precisely** the specification below

```
function [time, x, y] = capsule_flight_simulator(m, D, Ve, Cd, ElAng)
```

Where m is the capsule mass, D is its diameter, Ve is the capsule escape/launch speed, Cd is the capsule drag coefficient, and ElAng is the elevation angle in **radians**.

Plot the trajectory of the capsule for the parameter values provided in `capsule_and_plane_problem_datafile1.mat` file supplied at the class web page and additionally set Ve=900 and ElAng=`pi/3` for this function.

Problem 2 (20 points):

Write the function which finds the maximum height which capsule can reach, the time after the launch when the maximum height is reached, the x coordinate and the x velocity component of the capsule at this time. The solution function must comply to the following specification

```
function [MaxHeight, t_at_max, vx_at_max, x_at_max] =
         capsule_max_height_parameters(m, D, Ve, Cd, ElAng)
```

Present the results of this function in your report for the parameter values provided in `capsule_and_plane_problem_datafile1.mat` file supplied at the class web page and additionally set Ve=900 and ElAng=`pi/3` for this function.

Problem 3 (40 points):

Write the function **precisely** following the specification which finds a launch parameters for the capsule and the plane parameters provided in a file.

```
function [ElevationAngle, Vescape, Tlaunch] =
    capsule_launch_solution(capsule_and_plane_parameters_file_name)
```

Where ElevationAngle is the launch elevation angle in radians with respect to horizon and positive x direction, Vescape is the capsule escape velocity at separation with the catapult time, and Tlaunch is the time of launch. The Tlaunch could be negative if needed. Recall that 0 time is the time when the plane is directly above the catapult.

The parameters file will contain variables defining the capsule properties m, Cd, D, and variables regarding the horizontal velocity component of the plane (Vxplane) and its altitude/height (Hplane).

The solution is evaluated by how small will be the following **at the capsule maximum height**

$$E = |\vec{r}_{plane} - \vec{r}_{capsule}|^2 + |\vec{v}_{plane} - \vec{v}_{capsule}|^2 \tag{1}$$

where $\vec{r}$ is the radius vector pointing to the position of either the capsule or the plane.

During the final you will be provided with different parameters files. You algorithm should be able to work with all of them.

Note: if your code takes more than 2 minutes to solve this problem, you are doing it wrong way.

Problem 4 (20 points):

This is a problem to solve at home.

To launch the capsule you need the permission codes from a customer, after all, the customer decides when he wants to departure. This code is delivered to you as a radio voice message. However, a competing company tries to jam the transmission. The voice message recording will have a lot of noise. Each team will get its own voice recording of the code, available on the web site. What is your code (it should be a 6 digit number)?