

Key Deliverables:

- 5 “Check your answer” questions in Canvas (**10 pts**)
 - **You only get 3 attempts** (correct answers carry over to next attempt) at the quiz.
 - **There is a 5 minute “cool-down” between attempts.** Use this time to debug and reevaluate your code. Once you compute a new answer, make sure to critically think about it before submitting again.
 - **No additional attempts will be provided for any reason.**
- 1 .zip file containing the following (naming convention lastname_firstname_CC5.zip) (**15pts**):
 - Functioning Matlab code (.m file(s) - if using multiple files, please name driver code main.m)
 - Published code (.pdf file)
 - Flowchart sketch (.pdf file)
- All numeric quiz answers must be **computed** in your code!

The Balloon Problem

We are going to model the flight of a high altitude balloon being launched from Boulder (elevation 1,624 m) with an initial velocity of 2 m/s in the z direction (vertically)! The balloon will have 4 phases of flight:

1. Ascent prior to popping: the balloon is full of helium (Buoyancy force) and ascends (for the balloon $C_D \approx 0.5$) until 2,100 m when the balloon is popped.
2. Ballistic phase prior to parachute deployment: the balloon instantaneously when the altitude is reached. This removes not only the lifting force, but also reduces the mass of the system (since there will be no more helium). Since the balloon was the primary source of drag, and it is now gone, you can assume drag is negligible (**HINT**: only one force is nonzero during this phase).
3. Controlled descent after parachute deployment at 1950 m: On the return back to Earth, once the balloon drops below pop altitude again, a parachute is deployed ($C_D \approx 1.03$), which will add a new drag force (**HINT**: think about the free body diagram and ensure you’re applying the drag in the correct direction).
4. The balloon reaches the initial launch elevation and should no longer be moving, with no forces on it for the remainder of the time (**HINT**: determine which values of the state vector should be zero).

The constants required for this problem are the following:

- $m_{payload} = 450$ kg (assume structure and balloon material mass are negligible)
- $r_{balloon} = 17$ m (assume same radius for parachute)
- $\rho_{air} = 1.225$ kg/m³ (assumed to be constant)
- $\rho_{helium} = 0.1786$ kg/m³ (assumed to be constant)
- $g = 9.81$ m/s² (assumed to be constant)

To solve this problem, we use the equations of motion and Newton's 2nd Law. We can write $\sum F = m_{sys} \cdot a$, where m_{sys} is the total mass of the system, and the net force ($F_{net} = \sum F$) is determined by summing the 3 vertical forces acting on the balloon. The forces are those associated with gravity, buoyancy, and drag, with equations corresponding to each force defined as follows:

$$F_{gravity} = m_{sys}g$$

$$F_{buoyancy} = \rho_{air}Vg$$

$$F_{drag} = \frac{1}{2}C_D A_c \rho_{air} v^2$$

where ρ_{air} is the air density, V the volume of the balloon, C_D is the coefficient of drag, A_c is the cross-sectional area of the balloon, and v is the balloon's vertical velocity.

Problem 1: Write a function for the Constants

Create a function called `getConst` that creates a structure called `const` that contains all constants given and will be calculated to solve this problem (masses, volumes, etc.)

Problem 2: Write a function for ode45 to integrate (eventually)

The inputs should be t and X (your state vector consisting of height, h , and velocity, v), and `const`. Use `if` and `elseif` statements to determine which stage you are in and calculate m , $F_{gravity}$, $F_{buoyancy}$, and F_{drag} , accordingly to calculate F_{net} and the resulting value of acceleration. You should also determine when you've fallen back to Boulder.

Problem 3: Run ode45 and plot

Run your Balloon Equation of Motion function! Set up your initial conditions (1×2 column vector), and use a time span ranging from 0 to 150 seconds. Plot the height and velocity as functions of time - you may do one of the following:

- use `subplot` to create a 2×1 figure
- use `yyaxis` to create 1 figure with 2 axes

Quiz Questions

You will be asked to assess the velocity and/or height at 4 different times in the quiz. Be sure to calculate these times in your code using `interp1`.

You will be asked to determine at what time the balloon returns to the initial height.

Reflection Question

Use your plots and knowledge of how `ode45` worked to comment on the "end-state" (i.e., after the balloon has returned to the initial height). Consider the final values of altitude and vertical velocity. Do these make *physical* sense? If they do not, **hypothesize** a "post-processing" solution (for once `ode45` has finished running) - **DO NOT CODE THIS IN YOUR CODE - IT WILL CAUSE ISSUES WITH POSSIBLE QUIZ ANSWERS.**

Please write out the answers to these questions in the comments of your Matlab script. This should be about 1 paragraph in length.

Coding Rubric

	Excellent (100%)	Above Average (80%)	Average (70%)	Below Average (50%)
Requirements and Delivery (2pts)	<ul style="list-style-type: none"> Completed 90-100% of the requirements Delivered on time, and in correct format. 	<ul style="list-style-type: none"> Completed 80-90% of the requirements Delivered on time, and in correct format. 	<ul style="list-style-type: none"> Completed 70-80% of the requirements Delivered on time, and in correct format. 	<ul style="list-style-type: none"> Completed <70% of the requirements Delivered on time, but not in correct format.
Coding Standards (2pts)	<ul style="list-style-type: none"> Includes full header* Excellent use of variables (no global or unambiguous variable naming) 	<ul style="list-style-type: none"> Includes full header* Good use of variables (1-2 global or ambiguous variable naming) 	<ul style="list-style-type: none"> Includes incomp. header* Fine use of variables (3-5 global or ambiguous variable naming) 	<ul style="list-style-type: none"> No header Poor use of variables (many global or ambiguous variable naming)
Documentation (2pts) Comment your code	<ul style="list-style-type: none"> Clearly documented Specific purpose noted for each function and/or section 	<ul style="list-style-type: none"> Well documented Specific purpose noted for each function and/or section 	<ul style="list-style-type: none"> Some documentation Purpose noted for each function and/or section 	<ul style="list-style-type: none"> Limited to no documentation
Runtime (1pt)	<ul style="list-style-type: none"> Executes quickly, without errors 	<ul style="list-style-type: none"> Executes without errors, over 1 min runtime 	<ul style="list-style-type: none"> Executes with warnings/errors 	<ul style="list-style-type: none"> Does not execute
Efficiency (2pts)	<ul style="list-style-type: none"> Easy to understand, and maintain 	<ul style="list-style-type: none"> Logical, without sacrificing readability and understanding 	<ul style="list-style-type: none"> Difficult to follow 	<ul style="list-style-type: none"> Difficult to follow, huge and appears patched together
Figure Quality (2pts)	<ul style="list-style-type: none"> Easy to understand, labels and legend present 	<ul style="list-style-type: none"> Easy to understand, lacks labels and legend 	<ul style="list-style-type: none"> Difficult to understand, labels and legend present 	<ul style="list-style-type: none"> Difficult to understand, lacks labels and legend
Reflection Questions (4pts)	<ul style="list-style-type: none"> Easy to understand, fully thought out 	<ul style="list-style-type: none"> Easy to understand, mostly thought out 	<ul style="list-style-type: none"> Hard to understand, effort made 	<ul style="list-style-type: none"> Hard to understand, lacking effort

* Header includes author name(s), assignment title, purpose, creation date, revisions (applicable to **group** projects only)