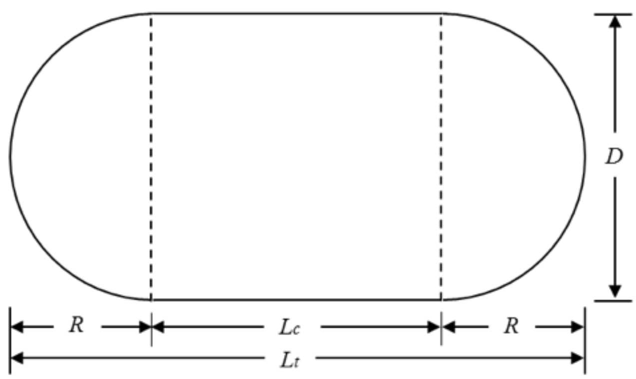
Engineering 13300 HW 10 – MA 2
MATLAB 2 – Individual Tasks

Guidelines for Tasks 5-6:
Tasks 5-6 are individual tasks. You may seek help from classmates, the instructional team or others but the work you submit should be your own. If you collaborate with others and use information developed together or by someone else, ALWAYS document and reference that material. Each student is responsible for submitting their own assignment to Gradescope.

Task 5 (of 6) Tank Volume

Open the template file **ENGR133_Fa21_MATLAB_Template.m**. Modify the file to create a script file that solves the following problem:

Cylindrical steel tanks have many outdoor uses. They are used in oil and gas refining, food production, farming, liquified gas storage, and more. Cylindrical tanks can have flat end caps, elliptical end caps, or spherical end caps.



Your company uses cylindrical tanks with spherical end caps. You are working on a design for a tank fill measurement system that can be used in tanks that are installed horizontally. A probe will measure the height of the fluid in the tank, and that fluid height will be used to determine the volume of the liquid in the tank.

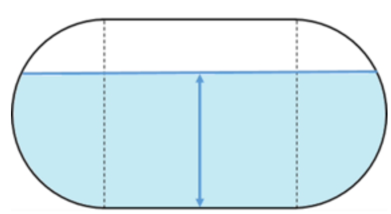
The tanks have the geometry shown in the figure. The two hemispherical end caps are equivalent to one sphere. The end caps and the cylindrical center section have the same radius, R. The tank length, L_t, is the sum of the tank diameter, D, and the length of the cylindrical center, L_c. All lengths are interior measurements. The tank wall thickness is not required for this application.

When a tank is installed horizontally, the fluid volume at any fluid height within the tank can be calculated using the function

$$V_f(h) = \frac{\pi h^2(3R - h)}{3} + L_c\left(R^2 \cos^{-1}\left(\frac{R - h}{R}\right) - (R - h)\sqrt{2Rh - h^2}\right)$$

Where

V_f = fluid volume
h = fluid height (between 0 and the tank diameter, inclusive of both; measured from the tank bottom)
R = tank radius
L_c = length of cylindrical section of the tank



It is important to never overfill a tank. Liquid in outdoor industrial tanks can expand with ambient air temperature changes, so the tank must be empty enough to accommodate the expansion. Your measurement system must ensure that the fill process will shut down properly and not allow a tank to overfill. The first step to achieve this is to set a safety percent, which is the maximum percentage of the tank volume that can be filled.

The fluid height is not measured instantaneously in this system. Instead, the programmer sets an increment, and the fluid height is measured in those increments. For example, if the increment is 0.5m and the tank is empty at the start of the fill, then the volume will be calculated at 0m, 0.5m, 1.0m, 1.5m, etc. until the tank is properly filled.

To ensure the tank does not overfill, you must stop the fill before the fluid volume passes the safety volume. You will keep the volume from exceeding the safety limit by setting a maximum tolerance. To find the maximum allowable tolerance, V_{tol}, you will calculate the difference in fluid volume at two different fluid heights, $R + 0.5\Delta h$ and $R - 0.5\Delta h$:

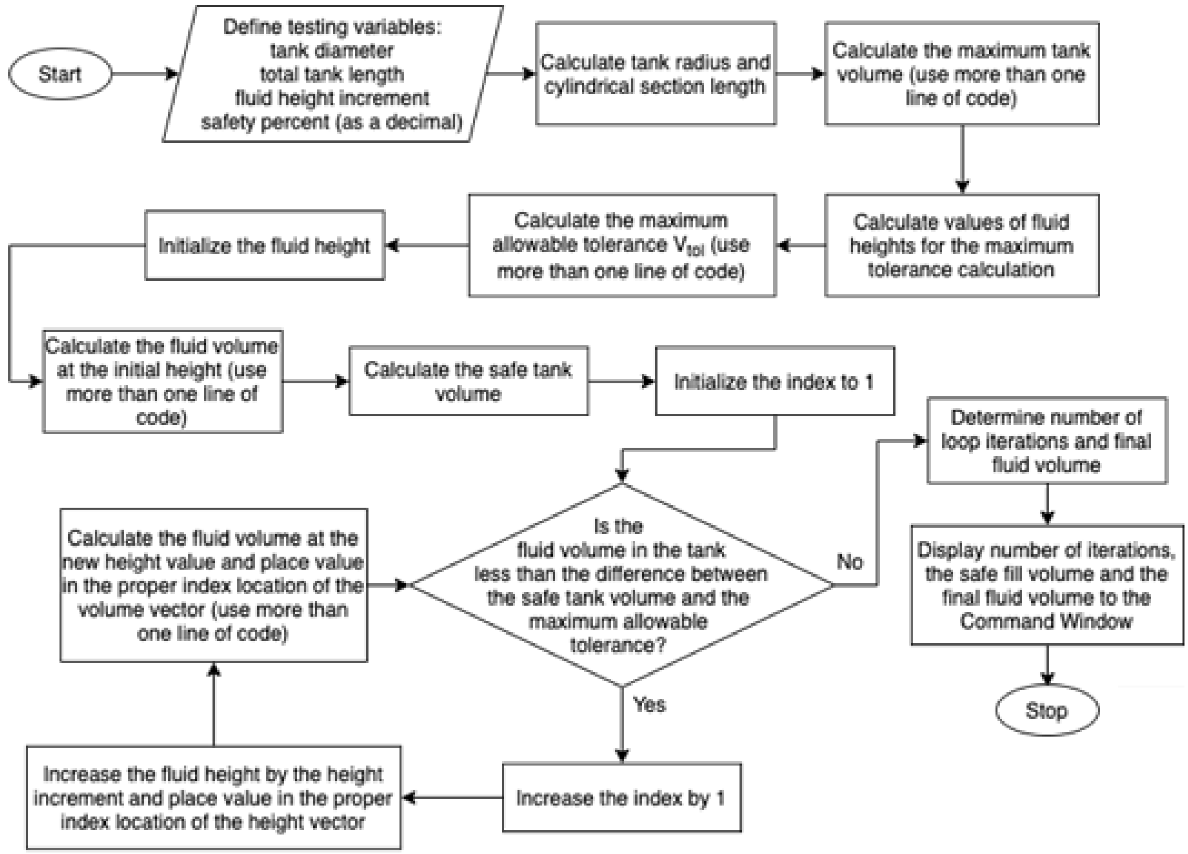
$$V_{tol} = V_f(R + 0.5\Delta h) - V_f(R - 0.5\Delta h)$$

Using the volume function V_f above, where Δh is the fluid height increment and R is the tank radius.

For testing purposes, you will use your company's most popular tank size, which has a diameter of 4.1 meters and a total tank length of 20.5 meters. Assume a fluid height increment of 0.25 meters and that the tank can be safely filled to 80% capacity (0.8 as a decimal). Also assume that the tank is completely empty when filling begins.

To meet the safety needs of the system, you will write a script that translates the flowchart below. Your script will

- Define the testing values as variables in the `INITIALIZATION` section
- In the `CALCULATIONS` section,
    - Calculate the requested values,
    - Create a vector of fluid height values and another vector of corresponding volumes using a while loop,
    - Stop adding elements to the vector as close as possible to the fill capacity without exceeding that value
- In the `OUTPUTS` section,
    - Display the number of iterations, safe fill volume and final fluid volume.



Publish your script as a PDF and name it `Ma2_Task5_`*`username`*`.pdf`. [refer the 'how to publish' demo video]

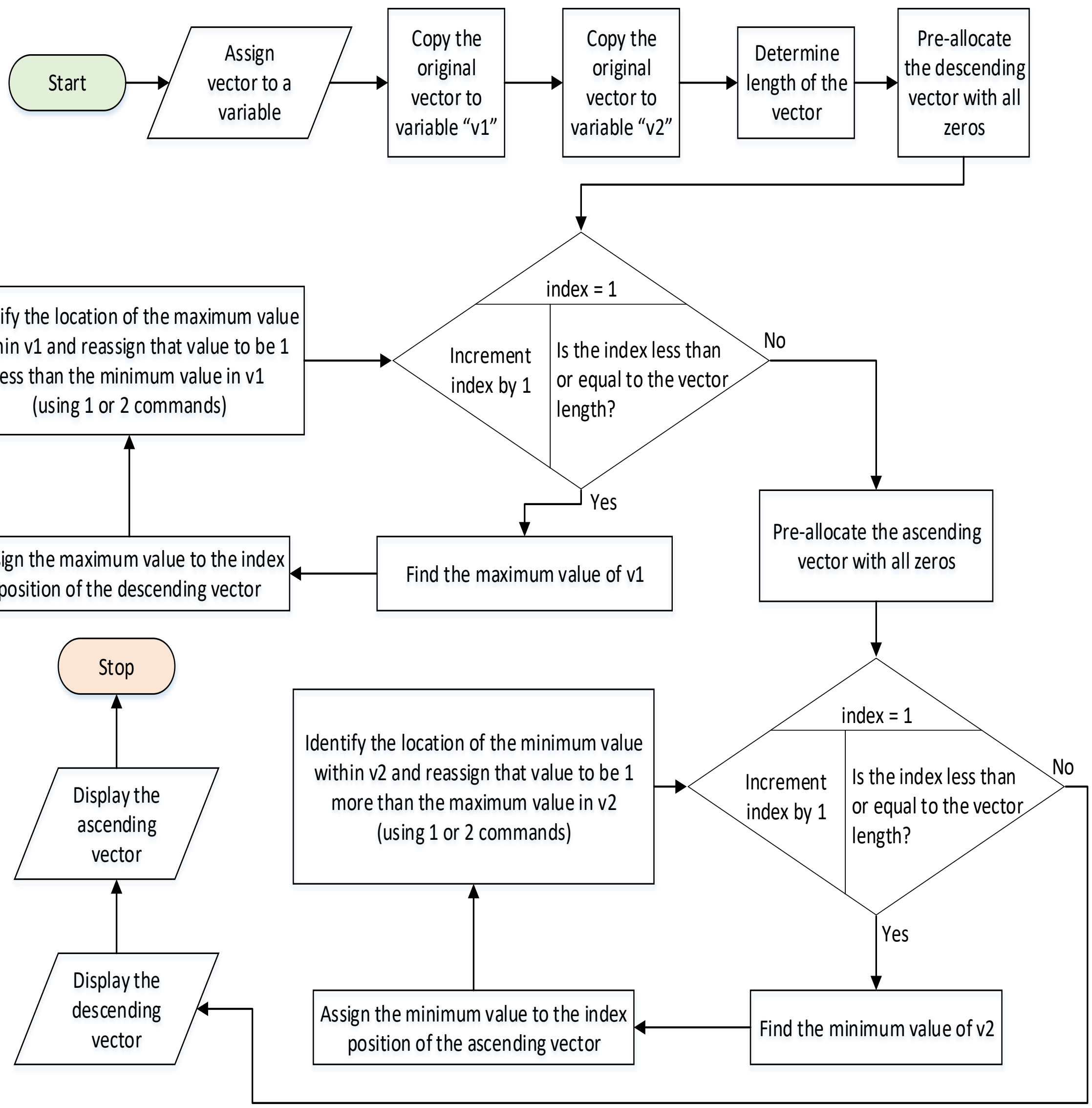
Task 5 Files:

- `Ma2_Task5_`*`username`*`.m`
- `Ma2_Task5_`*`username`*`.pdf` (published file)

Task 6 (of 6) Vector Array Sorting using loops

Open the template file **ENGR133_Fa21_MATLAB_Template.m**. Modify the file to create a script file that solves the following problem:

Sorting algorithms are commonly used in computer programming. Below is a flowchart for a basic vector-sorting algorithm. It first sorts the vector elements into descending order. Then it sorts the same vector into ascending order. It will sort vectors of any size with unique elements (i.e., no repeated numbers).



Problem Steps

1. In the `INITIALIZATION` section, set the initial vector to `v = [10 5 1 8 -9 0 2 3]`.
2. In the `CALCULATIONS` section, create two for loops, one for ascending order and one for descending order. Hint: lookup `zeros` and `length` inbuilt functions
3. Display the sorted array using `fprintf` and `disp` function in the `OUTPUTS` section.
4. **Publish** your script as a PDF and name it `Ma2_Task6_`*`username`*`.pdf`. [refer the 'how to publish' demo video]

Expected output:

```
The vector v is: [10 5 1 8 -9 0 2 3]

Vector sorted in descending order:
    10     8     5     3     2     1     0    -9

Vector sorted in ascending order:
    -9     0     1     2     3     5     8    10
```

Task 6 Files:

- `Ma2_Task6_username.m`
- `Ma2_Task6_username.pdf` (published file)