AE 333/707: PrL 2. Due on 26 Oct 2022. Total=30

This is an optional programming assignment that you may take up in lieu of the 10-mark post-mid-sem quiz. If you submit this, I will NOT evaluate your first quiz. This was relaxed for the pre-mid-sem assessment, but will be enforced for this one.

This is a GRADED INDIVIDUAL work; submission deadline is at the beginning of class on the due date mentioned in the title.

Before the deadline, submit a link in Moodle to a Google Colab document of your work, including detailed formulation/explanation, code listing and the results and discussions. Make sure that the Colab workbook can be EXECUTED by all (TAs and me); please do verify this for hassle-free grading. If you are using any language other than Python, then please submit the file containing your code – i.e., '*.m' file for MATLAB or '*.c' file for C, etc. Needless to say, comments are essential in your code.

Apart from this, you have to submit a hard-copy printout of the same Colab workbook in class by the deadline. If you are using some other language, then submit a proper report of your work – including the formulation, CODE, results and discussions.

You are expected to work independently on this assignment, although you may consult the TAs or the instructor. I may conduct a separate viva to gauge your individual understanding of your own solution! Any hint of plagiarism or academic misconduct will, at a minimum, incur a grade penalty for all parties involved.

1. (0 points) Refer to the 'PrL2 hint' Python code given on Moodle. You will find examples of how to define classes for efficiently representing airfoil geometries, and examples of how objects of these classes can be passed as arguments to functions or other class constructors.

   Most crucially for the next part, the example code also shows how to numerically compute the derivatives of the camber and thickness distribution functions in a general way.

   If you have a more efficient or elegant way, please use it. Otherwise, you can copy-paste from my code for specifying the geometry of a NACA 4-digit airfoil.

2. (20 points) Write a Python class with a method to solve for the flow around an arbitrary airfoil using thin-airfoil theory (with both vortex and source sheets). Set an object of an airfoil specification class obtained in the previous step as an attribute of the current class. The actual method should have the following input arguments only:

   1. The angle of attack, and
   2. A flag to control the plotting behaviour – whether to plot or not (see below).

   The method should be able to do the following:

   - Predict the sectional lift coefficient, and return it as an output.
   - Predict the sectional pitching moment coefficient about the quarter-chord point, and return it as another output.
   - Predict and draw a plot of the pressure coefficient on the surface; whether or not to plot should be controlled by the plotting flag that is specified at input.

3. (2 points) Use the above functions repeatedly in a loop in another program to draw the $c_l - \alpha$ curve and $c_{m,c/4} - \alpha$ for an arbitrary (user-specified) cambered NACA 4-digit airfoil. Note that you would NOT want to plot the pressure distribution for each data point; hence the plotting flag was incorporated in the previous class and should be set to FALSE in the present calls.

4. (8 points) Present a detailed comparison of your results for the particular cambered NACA 4-digit airfoil assigned to your group, with those obtained from XFOIL (run in inviscid mode). In particular, compare the lift slope and the $\alpha_{L=0}$ value for your chosen airfoil with the XFOIL output. Also, compare your prediction of $c_{m,c/4}$ with the XFOIL result as a function of α. Finally, present the pressure coefficient plot for $\alpha = 8°$, and compare with the XFOIL output. Overlay your results on the XFOIL results in the same plot for ease of verification.

5. (0 points) If you wish, you may use the streamline plotting routine developed for PrL1 to visualize the thin-airfoil-theory-simulated flow around a particular airfoil at an interesting AoA, for no extra credit.