Computer Assignment#3:
Computation of Streamline Patterns for Steady Inviscid Incompressible Fluid Flows

Consider a steady 2-dimensional inviscid incompressible flow through a chamber between the inlet and the outlet as shown in Figure 1. You need to numerically find out the streamline pattern within the chamber.

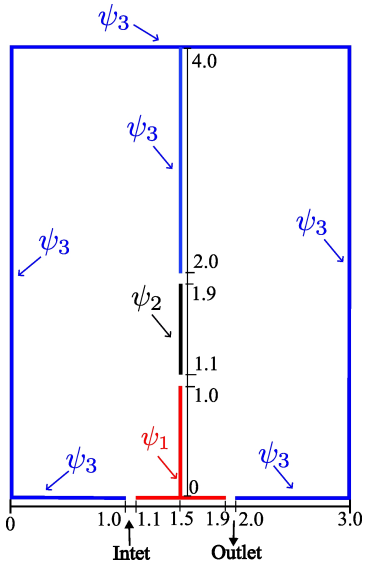

Figure 1

As you know from your fluid mechanics knowledge, you can obtain streamline pattern by solving the steam-function equation,

$$\frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2} = 0.$$

Since the chamber walls are streamlines (i.e. lines of constant ψ), they are assigned ψ_1, ψ_2 and ψ_3 as shown the Figure 1. Note, the assignment of these values of ψ is totally arbitrary as long as continuity equation is satisfied.

- Assume that the inlet, outlet and all slots are 0.1 m (per unit depth) and the chamber is 3 m by 4 m as shown in the Figure 1, where the locations of the inlet, outlets and slots are indicated.

- Compute the solution for three different set of values for boundary conditions:

	ψ_1	ψ_2	ψ_3
Test 1	100	150	300
Test 2	100	200	300
Test 3	100	250	300

- Present the converged solution (with maximum ERROR of 10^{-4}) in a Tabular form for all y grid point values at locations $x = 0.0,\ x = 1.0,\ x = 2.0,\ x = 3.0$.
- Use initial guess values as (a) $\psi = 100.0$, (b) $\psi = 150.0$, (c) $\psi = 200.0$. Observe the effect of the initial data for one set of boundary condition, which should be documented in your report as figures and comments.
- Write your **own code** for the Point Jacobi iterative method and use it for the given problem. Do not used any other method. You will be awarded zero for the assignment if any available package like LAPACK or program listed in the internet or program produced from ChatGPT, is used for the assignment.
- Use $\Delta x = 0.1$, $\Delta y = 0.1$.
- Consider the solution as converged when the maximum ERROR is

$$\text{ERROR} \equiv \frac{\left\|\psi_{ij}^{n+1} - \psi_{ij}^{n}\right\|_2}{\left\|\psi_{ij}^{n+1}\right\|_2} < 10^{-4}$$

  where $\|\psi_{ij}\|_2$ is the L2 norm, which is defined as

$$\|\psi_{ij}\|_2 = \sqrt{\sum_{i=2}^{IM-1}\sum_{j=2}^{JM-1} |\psi_{ij}|^2}$$

  and IM, JM are the maximum values of the indices i, j. In a similar manner, L2 norm for the difference in the solution at a point at two consecutive iterations are denoted as $\left\|\psi_{ij}^{n+1} - \psi_{ij}^{n}\right\|_2$,

$$\left\|\psi_{ij}^{n+1} - \psi_{ij}^{n}\right\|_2 = \sqrt{\sum_{i=2}^{IM-1}\sum_{j=2}^{JM-1} |\psi_{ij}^{n+1} - \psi_{ij}^{n}|^2}$$

- Provide contour plots for converged solutions (corresponding to all the three initial guesses) having maximum ERROR of 10^{-4}.

General Instructions

- **Checklist for submission:**

  1. Flowchart of the *Python*/*C*/*C*$^{++}$ code written by you. No other computer language is allowed. Additionally, web-based compilers must not be used to run the code.
  2. The code[1] with proper inline documentation for each function. (Use meaningful variable names following the class notes).
  3. A "README.txt" file which contains the proper description on how to run your code and get the plots.
  4. Single brief report, in pdf format, where all the above mentioned results in terms of **tables** and **plots** and **comments** are included. The plots submitted by you must be reproducible independently by the TA's from your code(s).

- **Instruction for submission:**

  - You must look at the RUBRIC before submission.
  - Rename your program file as your roll number (example: 204010006.c).
  - Rename the report as your roll number (example: 204010006.pdf)
  - Submit all the **4 documents** as stated in the Checklist above.

- **Notes:**

  - Marks will be given only if the program is working and showing correct result. No grade for steps will be given without results.
  - Assignment will not be evaluated if "instruction for submission" are not followed properly.

- Copying program from each other or from any other source will lead to severe penalty. Submissions will be checked using Stanford University's plagiarism detection software, **MOSS**, which can identify plagiarism even if variable names are altered or program lines are rearranged. The similarity with other's programs should be less than 30%. Make sure about it before submission of the assignment.

– E N D –

[1] You will be given zero mark for the entire assignment if any other computer language(s) is used other than the specified ones.