C3 Review Test

Time: 3:15pm to 5:00pm (1 hour 45 minutes) [1 hour 30 minutes for test + 15 minutes for submission]

30 marks

Instructions

- **Resubmissions** post 5:00pm will not be considered. Do NOT resubmit on Google classroom in any case. For any genuine reason, resubmit through mail to me and Ritesh (pro2017003@iiita.ac.in) with justification.
- Penalty for late submission:
  5:00pm - 5:14pm: 10%
  5:15pm - 5:29pm: 20%
  5:30pm - 5:59pm: 40%
  6:00pm - 7:59pm: 60%,
  8:00pm - 11:59pm: 75%, thereafter 100%
- Penalty for Plagiarism: 100% + 5 marks deduction from C3.
- Upload the final code as a single file (code/zip), with your Roll Number as the filename.

Problem Statement:

1) Write a Parallel program using pthread to parallelize the given sequential program (attached: solver.c) that solves a system of linear equations using Gaussian Elimination. [20 marks]

2) Write a distributed program using MPI for the same sequential program that solves a system of linear equations using Gaussian Elimination. [10 marks]

(bonus 5 marks for implementing MPI+PThread version of the program)

Note: You need to parallelize the given sequential program. Using any other algorithm to parallelize will not be considered.

(Problem description in next page)

Problem Description

Consider the following system of equations:

$$\begin{aligned}
a_{0,0}x_0 + a_{0,1}x_1 + a_{0,2}x_2 + \quad &\ldots \quad + a_{0,n-1}x_{n-1} &= b_0 \\
a_{1,0}x_0 + a_{1,1}x_1 + a_{1,2}x_2 + \quad &\ldots \quad + a_{1,n-1}x_{n-1} &= b_1 \\
a_{2,0}x_0 + a_{2,1}x_1 + a_{2,2}x_2 + \quad &\ldots \quad + a_{2,n-1}x_{n-1} &= b_2 \\
\ldots \quad &\ldots \quad \ldots &\ldots \\
a_{n-1,0}x_0 + a_{n-1,1}x_1 + a_{n-1,2}x_2 + \quad &\ldots \quad + a_{n-1,n-1}x_{n-1} &= b_{n-1}
\end{aligned}$$

This can also be represented in matrix form as Ax = B

Objective: To find values for the unknowns x_0, x_1, …, x_{n-1}, given the values of $a_{0,0}$, $a_{0,1}$, …, $a_{n-1,n-1}$, and b_0, b_1, …, b_{n-1}.

Approach:

- The idea is to solve these equations by first transforming the matrix A into upper triangular matrix. Usually, this is achieved using Gaussian elimination using the following elementary row operations
  - Swapping two rows,
  - Multiplying a row by a nonzero number,
  - Adding a multiple of one row to another row.

  [visit this wiki page for more information]
- In this sequential program, the same approach is followed but with a subtle difference (for certain reason) that we first swap the i^{th} row with the row below it that has the largest absolute element in the i^{th} column of any of the rows below the i^{th} row.

Input

The input contains several test cases of varying sizes.

- The first line contains one integer (N) representing the number of unknowns and the number of rows/columns of matrix A ($1 \leq N \leq 2048$). Assume N to be a power of 2.
- The next N lines contains N integers that represent the $a_{i,j}$ element of the equation in the matrix A, separated by a single space ($1 \leq a_{i,j} \leq 20$).
- The last line contains N integers separated by a space representing the elements of the b_n vector. The input must be read from a file named input_X.in (where X= 4, 16, 64 etc eg. “input_4.in”)

Output

The output contains only one line with N values separated by a single space representing the vector x_n. The output is written to a file named output.out.