Assignment: Parallel 1-D Rod Heat Diffusion using OpenMP

1. Objective

In this assignment, you will simulate heat diffusion along a one-dimensional metal rod using the finite difference method. You will first implement a sequential version, then parallelize it using OpenMP.

2. Mathematical Model

We model a rod divided into N discrete points. At each time step, the temperature at each interior point is updated using the following rule:

$$u_i{}^{t+1} = \frac{1}{2}(u_{i-1}{}^{t} + u_{i+1}{}^{t})$$

Where u_i is the array element at index [i]. at a given iteration time t.

Boundary conditions: The first and last points remain fixed throughout the simulation.

3. Implementation Tasks

3.1 Sequential Version

- Use a 1D array (size N) to represent the rod.

- Initialize all temperatures to 0.

- Set the center point to a high temperature (e.g., 100).

- Run the simulation for T iterations for time steps.

- Measure execution time using a high-resolution timer.

3.2 Parallel Version (OpenMP)

- Parallelize the spatial update loop using OpenMP.

- Measure execution time for different thread counts (1, 2, 4, 8).

3.3 (Optional) Advanced Version

- Instead of running for a fixed number of time steps, stop when the maximum change is below a tolerance ε.

- Use OpenMP reduction(max:variable) to compute the maximum difference.

4. Performance Analysis

You must measure and report the following:

- Execution time (sequential vs parallel).

- Speedup = T_sequential / T_parallel.

- Efficiency = Speedup / Number_of_threads.

5. (Optional) Visualization

Save snapshots of the rod temperatures to a CSV file. Using a given Python (matplotlib) file to visualize the diffusion process.

6. Output file

For every 50 timesteps, store one snapshot of time, each snapshot is in a line by itself in the output file and a single snapshot is of the form (this is called a csv format):

$$t_0, u_0, u_1, u_2, \dots, u_N$$

Given that:

N: is the rod length and is equal to 500.

T: is the total time steps and is equal to also 500.

t_i: is a single time snapshot and its values are: [0, 50, 100, 150, 200, 250, 300, 350, 400, 450, 500]

u_i: is a single rod point, where [i] ranges from [0:N]

The output file name should be “heat_output.csv”

And should be in the same path as the python script.

7. Grading Rubric

Correctness: 30%

Parallel Implementation: 40%

Discussion: 30%