Parallel Processing Project 2 Report

Team Members: Reem Ooka & Nihar Lodaya

1) Architecture/Algorithm Used and Why?

Parallel Processing:

- The simulation leverages OpenMP for parallel processing to efficiently handle the computational demands of simulating flu spread over a large grid. The use of #pragma omp parallel for is critical in distributing the workload across multiple threads, reducing the total simulation time.

Memory Management:

- Dynamic memory allocation is used to handle a large grid (500x500), with custom overloads for new and delete to track memory usage. This helps in monitoring and managing the memory consumption throughout the simulation, ensuring the program doesn't use more resources than necessary.

Grid-Based Simulation Model:

- The entire population is modeled on a two-dimensional grid where each cell represents a person. This spatial arrangement allows for straightforward calculation of interactions between individuals based on their positions in the grid.
- Boundary conditions are implicitly handled by ensuring neighbor checks (ni and nj in simulateDay) don't reference out of bounds indices, which corresponds to the reduced interaction of individuals at the edges and corners of the grid.

Infection and Recovery Mechanism:

- Each cell on the grid can be either infected (1) or healthy (0). A separate recovery grid tracks how long an infected cell remains sick before recovering, implementing the Ω parameter.
- The simulation iterates over each day, updating the grid based on infection spread (governed by β) and recovery status. This approach models the natural progression and regression of the disease over time.

Explanation for Algorithmic Decisions:

- **OpenMP Usage:** Chosen for its efficiency in handling loops that can be parallelized, such as those iterating over grid cells. This reduces runtime, particularly crucial for large-scale simulations like this one.
- **Dynamic Memory with Custom Tracking:** Since the program could potentially require adjustments based on performance assessments, having an integrated memory tracking mechanism allows for better optimization and debugging.
- **Local Neighbor Checking:** By limiting interaction to immediate neighbors, the model simplifies the calculations required per cycle, making it computationally feasible even for large grids and extended simulation periods.

2) Citations:

None were referred to for this project.

3) Table:

Threads	Alpha	Beta	Omega	Days	Execution Time (s)	Memory Consumption	Final Infection Days
1	0.01	0.15	2	10	5.9265 seconds	3012000	36765
1	0.05	0.25	3	20	9.36742 seconds	3012000	175466
1	0.1	0.5	4	30	13.8676 seconds	3012000	199406
1	0.01	0.25	5	40	17.7251 seconds	3012000	159831
2	0.01	0.15	2	10	5.91828 seconds	3012000	36765
2	0.05	0.25	3	20	9.20447 seconds	3012000	175466
2	0.1	0.5	4	30	13.7937 seconds	3012000	199406
2	0.01	0.25	5	40	17.6985 seconds	3012000	159831
4	0.01	0.15	2	10	5.87743 seconds	3012000	36765
4	0.05	0.25	3	20	9.19861 seconds	3012000	175466
4	0.1	0.5	4	30	13.697 seconds	3012000	199406
4	0.01	0.25	5	40	17.5301 seconds	3012000	159831
						3012000	
8	0.01	0.15	2	10	5.77076 seconds	3012000	36765
8	0.05	0.25	3	20	9.00362 seconds	3012000	175466
8	0.1	0.5	4	30	13.5030 seconds	3012000	199406
8	0.01	0.25	5	40	17.4882 seconds	3012000	159831
16	0.01	0.15	2	10	5.60125 seconds	3012000	36765
16	0.05	0.25	3	20	8.89265 seconds	3012000	175466
16	0.1	0.5	4	30	13.3873 seconds	3012000	199406
16	0.01	0.25	5	40	17.3814 seconds	3012000	159831
32	0.01	0.15	2	10	5.37598 seconds	3012000	36765
32	0.05	0.25	3	20	8.6554 seconds	3012000	175466
32	0.1	0.5	4	30	13.1942 seconds	3012000	199406
32	0.01	0.25	5	40	17.2128 seconds	3012000	159831

4) Explain the behavior of the data in the table and why?

Observations from the Table:

1. **Execution Time vs. Number of Threads:**
   - **Decrease in Execution Time:** As the number of threads increases, the execution time generally decreases. This is expected in parallel computing where distributing the workload across more threads can significantly speed up processing time, especially for computationally intensive tasks like simulating disease spread over a grid.
   - **Diminishing Returns:** The reduction in time becomes less significant as the number of threads increases beyond a certain point. This could be due to overhead from managing more threads or limitations due to other system resources (like CPU cores) that aren't increasing along with the threads.

2. **Effect of Alpha (Initial Infection Rate) and Beta (Infection Probability):**
   - **Higher Final Infection Days with Increased Alpha and Beta:** When either α or β is increased, the "Final Infection Days" typically increases. This suggests that with a higher initial number of infected individuals (α) or a greater probability of transmission (β), the disease spreads more extensively and persists longer within the population.
   - **Beta's Influence:** The variability with different beta values, keeping alpha constant, shows a clear trend where a higher beta leads to longer infection durations and higher final counts, indicating more widespread transmission.

3. **Omega (Days Sick) Impact:**
   - **Increased Omega, Increased Infection Duration:** A higher omega value, which extends the period an individual remains sick, tends to increase the total "Final Infection Days." This is because infected individuals remain contagious for longer, potentially infecting more people before recovering.

4. **Memory Consumption:**
   - **Constant Across Simulations:** The memory consumption remains constant across different simulations. This is expected as the size of the grid and the data structures used (like the grids for current state, next state, and recovery) do not change with the parameters of the simulation but are rather dependent on the fixed grid size (500x500 in this case).

Why We Observe These Behaviors:

- **Parallel Efficiency:** The improvements in execution time with more threads showcase the benefits of parallel processing. However, the overhead associated with managing a large number of threads or the physical limitation of the processor (such as the number of available cores) can limit these gains.
- **Disease Dynamics:** The relationship between α, β, and Ω with the extent of disease spread is consistent with epidemiological models where initial conditions and transmission rates significantly influence the outcome of disease spread.
- **Resource Utilization:** The consistent memory usage underscores that the allocation is primarily determined by the grid size and the number of grids maintained, which do not vary with the simulation parameters.

5) Hardware Specs: **We used the Hancock Lab 0005 system to do our project.**

- Processor (CPU)

  13th Gen Intel(R) Core(TM) i7-13700, 2100 Mhz, 16 Core(s), 24 Logical Processor(s)

- RAM (Memory)

  Installed Physical Memory (RAM)16.0

GB - BIOS

  BIOS Version/Date LENOVO S0IKT15A, 3/3/2023