In this assignment, we will solve the Laplace equation over some domain $\Omega \in \mathbb{R}^d$, subject to Dirichlet boundary conditions

$$\nabla^2\phi = 0 \quad \text{with} \quad \phi(\partial\Omega) = f \tag{1}$$

Let Ω be the unit box in two dimensions, and let ϕ satisfy the following boundary conditions

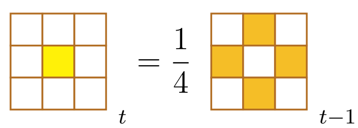
$$\begin{array}{rclcll} & & \phi(x,0) & = & \sin(\pi x) & 0 \le x \le 1 \\ & & \phi(x,1) & = & e^{-\pi}\sin(\pi x) & 0 \le x \le 1 \\ \phi(0,y) & = & \phi(1,y) & = & 0 & 0 \le y \le 1 \end{array} \tag{2}$$

The exact solution is given by

$$\phi(x,y) = e^{-\pi y}\sin(\pi x) \tag{3}$$

We will solve this equation using the Jacobi iterative scheme on a uniform grid:

- make an initial guess for ϕ over a discretization of Ω
- apply the boundary conditions
- use the Jacobi update, which replaces each cell with the average of its four nearest neighbors



(4)

- stop when a convergence criterion is met

1. Write a sequential solver that accepts the number N of sample points along the x axis as an argument and solves the above equation on an N^2 grid.
2. Reimplement the grid update using n threads, where n is accepted as a command line argument
3. Reimplement the grid update using MPI
4. For extra credit, build a hybrid solver: let there be m MPI tasks, each of which uses n threads for the update. With some care, you should be able to *overlap* the grid update with the communications among neighboring MPI tasks, effectively eliminating the communication cost.