Short Exercise #2 1D Heat Diffusion

Assigned: Thursday, October 25, 2018   *Due: Thursday, November 1, 2018*

In this exercise you will be simulating the diffusion of heat in one dimension using MPI. To do this you will be using the heat diffusion equation which for 1 dimension is :

$$\frac{du}{dt} = \alpha \frac{d^2u}{dx^2} \tag{0.1}$$

For this assignment, we will assume the α, the heat constant, is one. We will also be assuming that u is constant at both endpoints. Using a stepsize of h and k,

$$x_i = x_0 + ih \tag{0.2}$$

$$t_i = t_0 + ik \tag{0.3}$$

, discritizing and solving this equation for u at time n+1 and position j yields :

$$u_j^{n+1} = (1 - 2r)u_j^n + ru_{j-1}^n + ru_{j+1}^n \tag{0.4}$$

with

$$r = \frac{k}{h^2} \tag{0.5}$$

Yes there was a lot of hand waving here because at this point we're more interested in programming this thing than worrying about the math. There are numerous sources online that will explain the details if you're interested.

Regarding the above equation, things to note are that the solution at t = t_{n+1} is determined explicitly from the solution at t = t_n and each point is updated based on its neighbors.
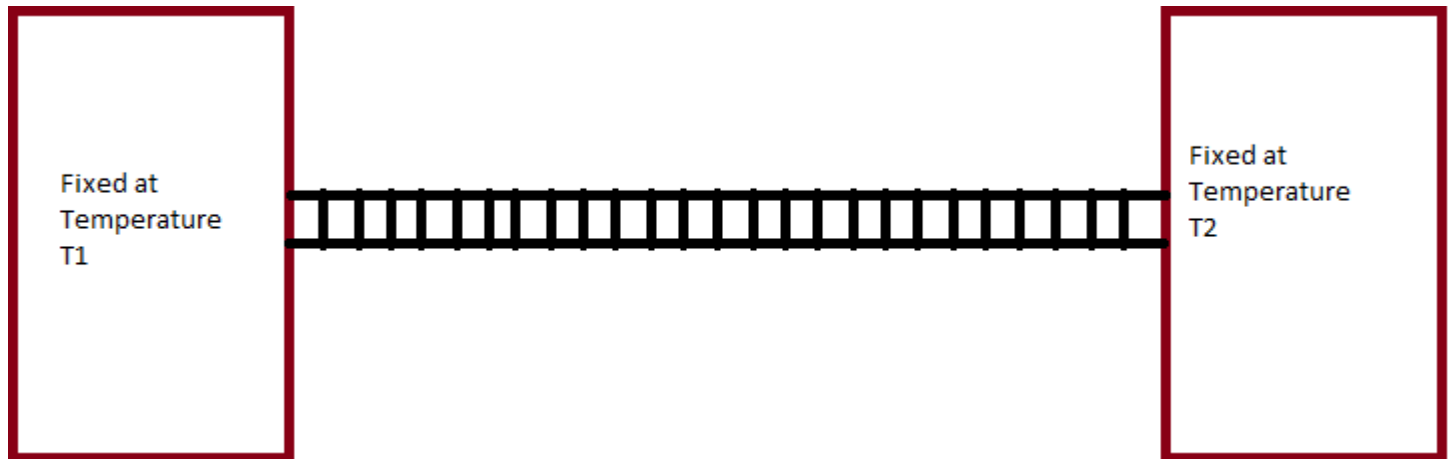

FIGURE 1 – Heat Diffusion Problem

Visually the environment you are trying to simulate is shown in figure 1. In the image you can see two constant heat sources connected with a narrow rod which is subdivided into chunks where you will be calculating the current temperature at each timestep. Figure 2 shows an example of how each chunk's temperature is computed using the information from the previous timestemp.

WHAT YOU NEED TO DO You need to implement a MPI program that will output the temperature at each grid point after running the equation for a specified number of timesteps. Example execution

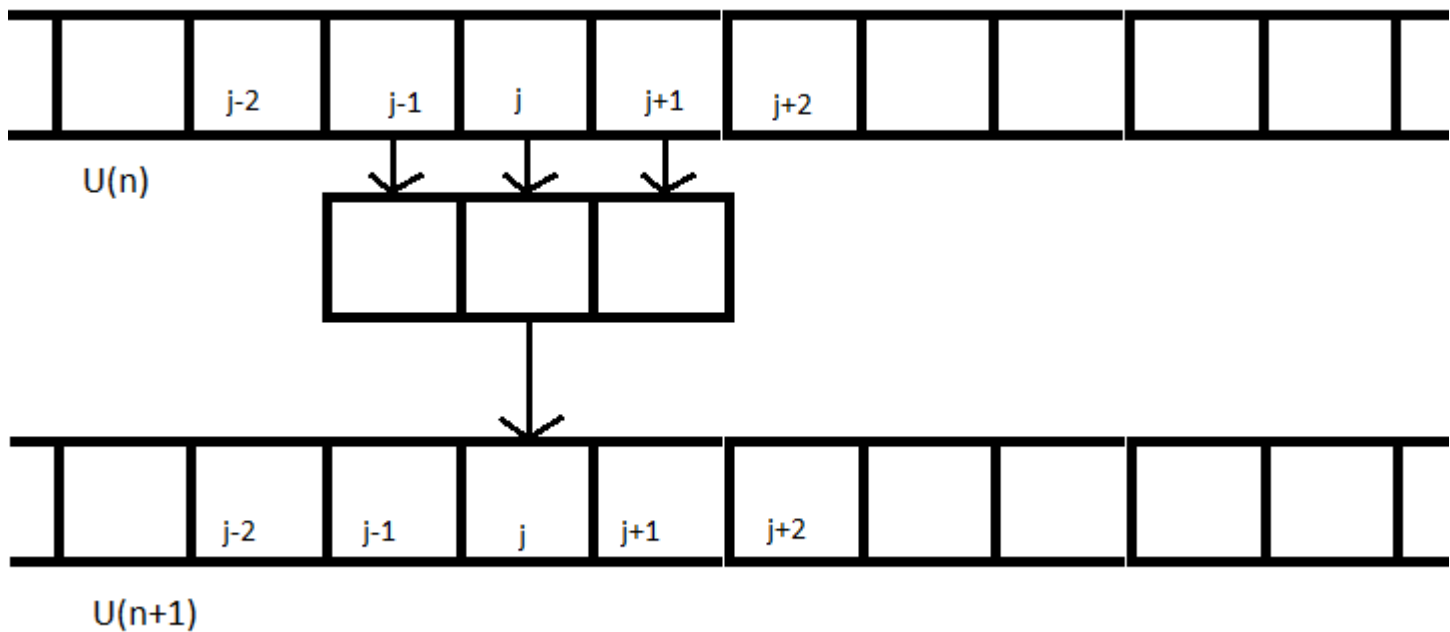mpirun -np numProcs ./heat1D T1temp T2temp NumGridPoints NumTimesteps

FIGURE 2 – 1D Stencil

Your program should determine at runtime how many gridpoints each MPI rank will need to compute. You MAY use blocking SENDS and RECEIVES and any other MPI functionality you deem helpful for this assignment. At the end of execution, your program should write an output file named heat1Doutput.csv listing the temperatures at each grid point in comma separated format. Assume k will always be 1 and h will always be 2. Assume all chunks start with a temperature of 0. Use **double**s for all values.

The main folder of your solution should be named se2 (note capitalization) and should follow the following structure :

```
se2
├── src
│   └── All of your files
└── CMakeLists.txt
```

You MUST use CMake for configuring your project. You will need to use online resources to determine how to correctly link the MPI libraries and refer to needed header files. Do NOT write your own Makefile directly. For submission, compress your solution into se2.zip (NOT .TAR.GZ, or .RAR or anything else). zip is installed on the login node. Submit via canvas.

Example Output Format for 3 grid points
334.2, 330.1, 320.6

Note the spacing and lack of comma on last value. Also do not insert newlines into this file. I'll be using **diff** to grade these so it must match to receive full credit.

For assignment p2, we will be expanding this into 2 or possibly 3 dimensions, so you may want to consider that when designing your solution.