Background: Solving the Laplace and Poisson Equation

For this assignment, you will be solving the following equation for steady state conduction heat transfer:

$$\nabla^2 T = -\dot{q}_{gen} \,, \tag{1}$$

where T is temperature, $\dot{q}_{gen}$ is volumetric heat generation (W/m^3), and ∇^2 is the Laplacian (which can also be denoted by $\triangle$). In Cartesian coordinates, $\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}$.

The steady-state heat conduction equation is an example of a Laplace/Poisson equation (its Laplace when $\dot{q}_{gen} = 0$ and Poisson otherwise). Laplace and Poisson equations come up all of the time in engineering and science. They also can describe an electric potential in the absence of a changing magnetic field (Poisson), a gravitational potential (Poisson), potential fluid flow (Laplace).

Your assignment 2D ?

how many points/divisions?

Assume a constant k. Specify k in input file?

- Derive the the steady state heat conduction equation. In addition, discretize the equation and put it in matrix form. Write up this analysis and scan it as a pdf in order to upload with your program.
- Write a program that solves the 2D steady state heat conduction equation with heat generation. A uniform mesh is fine. Use 2nd order central differencing.
- Your program should have an option to solve the equation either by LU Decomposition or by the Gauss-Seidel method. The Gauss-Seidel method should be a function implemented in the solvers.* file (i.e. you should add a Gauss-Seidel method to your function you created for HW 1).
- Your program should read in values from an input file. The input file should allow the user to specify the values for dx, dy, number of mesh points in x, number of mesh points in y, the value for the heat generation, the type of "solver" – i.e. Gauss-Seidel or LU Decomposition, the type and value of boundary conditions (Neumann or Dirichlet) for each boundary (North, East, South, West). You may use 1st order differencing for the Neumann conditions.
- Your program should write out the final results to a Tecplot file called output.dat (or whatever name you like). You are allowed to plot only the internal points and not the boundary values for your LU decomposition case.
- Create a makefile to compile all of your programming files.

2 matrices 1 for x other for y?

Tecplot

Tecplot is the "go to" plotting program for computational fluid dynamics and heat transfer researchers. There is a lot of functionality and it makes nice looking plots. Its one disadvantage is that it does not handle very large data sets all that well. Luckily for us, our data sets will not be that big. You will have to get used to Tecplot as you will definitely be using it in this class and probably even in your research. There are many ways of writing out Tecplot output files and the the output depends on the structure of your mesh. Since our mesh is a simple Cartesian, structured mesh, we can use a simple POINT format for our Tecplot files. They look something like:

```
VARIABLES = "X" "Y" "Temperature"
zone I=100 J=100 SOLUTIONTIME = 0.00, F=POINT
x y T
x y T
x y T
.
.
.
.
```

Office hour Questions

- Assume we know all boundary node conditions?
- constant k?
- how to put 2D eq in matrix when you have Δx & Δy

- for derivation
- 2D
- grid points for matrix.

Assume Δx = Δy = h

This example has 100 mesh points in the x direction (I) and 100 mesh points in the y direction (J). The x points are incremented first, keeping the y fixed. The SOLUTIONTIME is not that big of a deal for this example because this example is a steady state example. Nevertheless, it will become more important for unsteady problems. The x, y, and T will be the data values for x, y, and temperature.

What to submit

Submit to the ICON dropbox your write-up that performs the derivation and discretization of the heat equation, the main program, your solvers.* file/header files, your input file, and the makefile that you used. Zip them up in a zip file to upload.