CS601: Software Development for Scientific Computing

Programming Assignment 4 - Unstructured Grids and Finite Element Method of Solving PDEs

Due: 7/11/2021

The objective of this assignment is to gain a hands-on experience with:

1. Computing PDEs on unstructured grids using the Finite Element Method. In particular, you will solve the steady-state, 2D-heat diffusion problem.

1 Problem Statement

Compute the temperature distribution using Finite Element Method in an isotropic, rectangular, two-dimensional plate with dimensions 600 cm $\times$ 400 cm (see Figure 1 for reference).

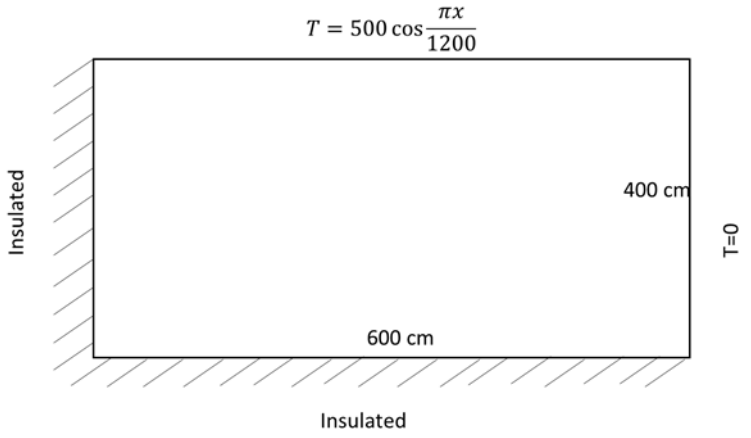

Figure 1: Isotropic rectangular plate with insulated boundaries and specified boundary temperatures

The following information is given:

- Thermal conductivity of the material is $K = 30W/mC$.
- The boundaries at x=0 and y=0 are insulated (no heat flux condition).
- The boundary at x=600cm is maintained at zero temperature.
- Note that there is no convection and radiation heat transfer. The internal heat generation i.e. $f = 0$.

The discretization details are shown in Figures 2 and 3.

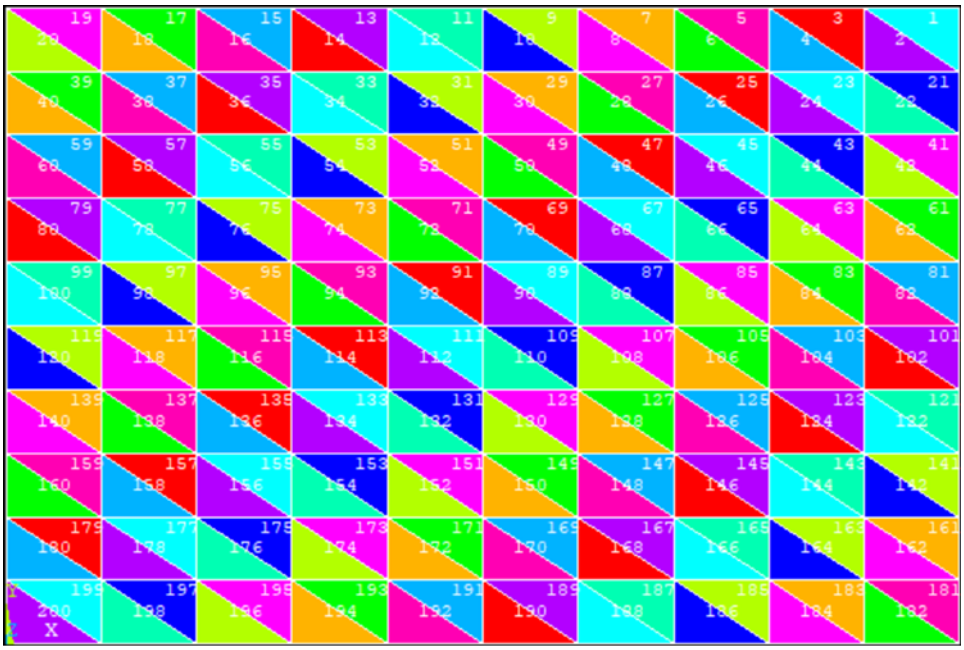

Figure 2: Discretization of the rectangular plate with triangular elements. Element IDs are shown.

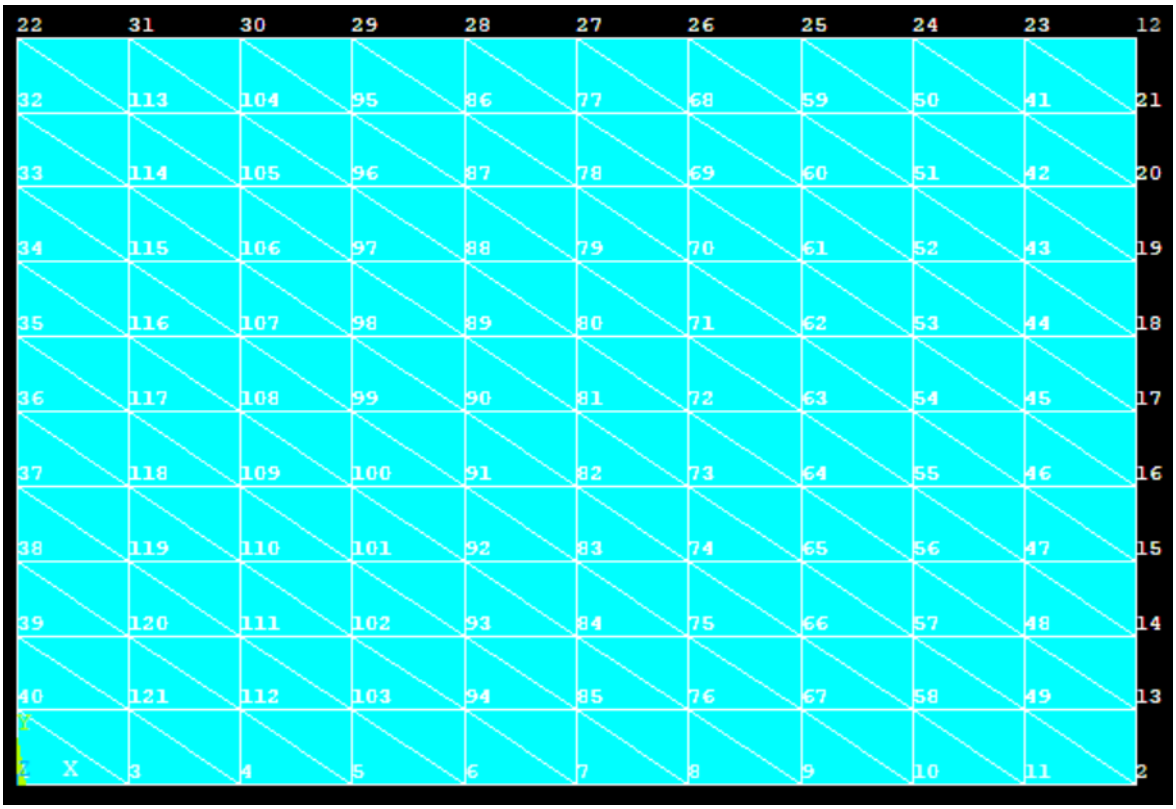

Figure 3: Discretization of the rectangular plate with triangular elements. Node IDs are shown.

2 What you need to do and submit

Refer to the notes on FEM provided on the course webpage.

- clone the *PA4* repository that contains all the files that you need to get started with your assignment (link provided in the discussion forum).
- Modify `FEMain.cpp` and add other files that you think are necessary. Modify the `Makefile` as required (currently the Makefile doesn't consider the folder structure. May have build errors.). Create a target called `team` in the `Makefile` that prints out your team members' names.
- A shell script (this must be written in bash) called `runme` that builds your solution using the `Makefile` and executes it using the parameters provided. This script should take in a single argument: the name of the input files (node and element) without extensions. It is assumed that the node and element file names have the same name but different extensions. E.g. the node and element files are called `fine.node` and `fine.elem` respectively. The script should first print the team name using the `Makefile` that you have created. Then, the script should compile (using the `Makefile`, and run your program (accepting single filename argument). Upon successful execution of your program, you should have an output file generated.
- your program must produce an output file `solution.txt`. Write temperature at each node into this file. The file should contain two fields on each line (blankspace separating them): `nodeID temp`. The output file should not anything else other than this.

You must tag your source code and submit as you have done previously. The tag name to be used is: `cs601pa4submission`*. All tag names are case-sensitive..*