Lab 5

Math 9830 Timo Heister, heister@clemson.edu

Note: Unless specifically asked to submit a solution, just work on the exercises and keep track of your progress in your journal.

1. Make yourself familiar with step-4 (I added a copy of step-4 to the class github repository). The step-4 documentation is at `https://www.dealii.org/current/doxygen/deal.II/step_4.html`. Go read it.

2. Run step-4 in 2d and 3d and visualize the result in ParaView. Try to reproduce the pictures in the tutorial (warp by scalar in 2d and isosurfaces in 3d).

3. Think about the reasons for "dimension independent programming" in deal.II. Summarize your thoughts with a list of bullet points.

4. Change the geometry in `make_grid` to be an L-shape domain. Check that this works. Now modify it to not use the 3d L-shape (cube with a corner cut out), but an extruded version of the 2d L-shape. Note that we did this in an earlier lab but not in a dimension-independent way.

5. Take a look at the assembly of a single cell and create a table in your journal to estimate the cost of the assembly for FE_Q degree 1,2,3,4 in 2d and 3d. As an estimate for the cost, count how often `cell_matrix(i, j) +=` is executed. You can do this by writing some code that keeps track of this count, or you can do everything on paper and make a formula.

6. Change the geometry to the unit square/cube ($\Omega = [0,1]^d$) and change the boundary condition to be
$$u(x,y) = \sin(\pi x y)$$
and the right-hand side to be
$$f = -\triangle u$$
for the u above. For this, compute the negative divergence of ∇u on paper.
With this setup, our computed solution u_h will be an approximation to the function u above (this is called the "method of manufactured solutions", which we will talk about in more detail). You can do the convergence of the mean like we did in step-3, except that we can now compute the error as we know the mean of u to be
$$\int_\Omega u \, \mathrm{d}x \approx 0.524663067575319$$
in 2d. Did I compute this value correctly? Check your convergence rate against this value (you can compute the error in the code by using this value!).
Submit your .cc file with this change on Canvas.
Bonus: Also do this in 3d by specifying a different u that also depends on z and repeat the process.