The final project is to choose a topic and solve it with a MATLAB program. The program should be original and not a change to pre-existing internet code.

Helmholtz equation

$$A\frac{\partial^2 u}{\partial x^2} + B\frac{\partial^2 u}{\partial y^2} + ku = C\sin(\alpha x)\sin(my) \qquad 0 \le x \le \pi \quad 0 \le y \le \pi$$

$$A, B > 0 \qquad \text{m a specific integer 1,2,3...}$$

$$\text{bc:}\quad u(0,y) = u(x,0) = u(x,\pi) = 0$$

$$\frac{\partial u}{\partial x} - \gamma u = 0 \qquad \gamma > 0 \qquad x = \pi$$

An explicit solution is

$$u = C\sin(\alpha x)\sin(my)$$

$$\alpha = \gamma\tan(\alpha x) \quad \gamma > 0$$

$$C = k - \alpha^2 A - \beta^2 B$$

This is a transcendental equation for α. By graphing the straight line and the tangent function it obviously has one (and only one) solution. This can be found by any nonlinear solver. The parameters should be chosen so that $C \neq 0$ and so there are no eigenvalues. This is always true if $k \le 0$ (modified Helmholtz). For the true Helmholtz ($k > 0$) choose k so that there are no eigenvalues.

This explicit solution should be compared with the numerical solution for a range of wave numbers k with positive and negative values and a sequence of meshes.

There are several choices of problems

1. Set up a finite element code using either triangles or rectangles. The solution of the resultant linear system by the MATLAB built in linear solver. Verify the order of accuracy in both the H^1 and H^2 norms by refining the mesh. What is the effect of the ratio $\frac{A}{B}$and k on the accuracy?

2. Set up a simple finite difference problem and solve the equations by multigrid using various smoothers and V,W and FMG cycles. Verify the rate of convergence. What is the effect of the ratio $\frac{A}{B}$and k on the convergence rate?

3. A third possibility that will be on the web site is to use one of several pre-existing codes and either modify them or extend and describe them in more detail. One possibility is to consider the Helmholtz equation in polar coordinates between two concentric circles (or more generally circular-like geometries).

   See for example

   - the MATLAB built-in pdetools
   - http://dehesa.freeshell.org/FSELIB
   - http://people.scs.fsu.edu/~burkardt/m_src/fem_50/fem_50.html
   - http://www.ce.berkeley.edu/~rlt/feappv/
   - Of interest to me personally is http://www.zib.de/nano-optics/HelmPoleMan/ Possible extension to Master's thesis.

4. Instead of Multigrid use a Krylov method (BICGSTAB or GMRES) in MATLAB.

   Use a preconditioner to improve the convergence rate