Sample Solutions for Assignment 5.

Reading: Chapter 3 in the text plus supplemental material on finite elements in 2D.

1. Write a code to solve Poisson's equation on the unit square with Dirichlet boundary conditions:
$$u_{xx} + u_{yy} = f(x,y), \quad 0 < x, y < 1$$
$$u(x,0) = u(x,1) = u(0,y) = u(1,y) = 1.$$
Take $f(x,y) = x^2 + y^2$, and demonstrate numerically that your code achieves second order accuracy. [Note: If you do not know an analytic solution to a problem, one way to check the code is to solve the problem on a fine grid and pretend that the result is the exact solution, then solve on coarser grids and compare your answers to the fine grid solution. However, you must be sure to compare solution values corresponding to the same points in the domain.]

   See the code `poisson.m` on the course web page.

   I used a fine grid with $h = 1/64$ and then solved on grids with $h = 1/32$, $h = 1/16$, and $h = 1/8$. I looked at the maximum difference between the coarse grid values and the fine grid solution at the nodes of the coarse grid, and obtained the following results.

```
  h             maxerr      ratio
3.125e-02    4.595e-05
6.250e-02    2.288e-04    4.979e+00
1.250e-01    9.454e-04    4.132e+00
```

   This looks essentially like second order accuracy. The error appears to be reduced by more than a factor of 4 in going from $h = 1/16$ to $h = 1/32$, but this may be because the computed solution with $h = 1/32$ is more like that with $h = 1/64$ (which is being used for comparison) than it is like the true solution. Probably the actual error reduction is closer to a factor of 4.

2. Now use the 9-point formula with the correction term described in Sec. 3.5 to solve the same problem as in the previous exercise. Again take $f(x,y) = x^2+y^2$, and numerically test the order of accuracy of your code by solving on a fine grid, pretending that is the exact solution, and comparing coarser grid approximations to the corresponding values of the fine grid solution.

   It turns out that you get only *third order* accuracy for this problem, since although f is smooth, the domain has corners, which means that the solution $u(x,y)$ does not have four continuous derivatives and therefore the Taylor series expansion is not valid to this many terms. Note that since u is constant on the boundaries, $u_{xx} = 0$ along the boundaries $y = 0$ and $y = 1$ and $u_{yy} = 0$ along the boundaries $x = 0$ and $x = 1$. Yet, as you approach the corners $(1, 0)$, $(0, 1)$, or $(1, 1)$ from inside the domain, $u_{xx} + u_{yy} = x^2 + y^2$ approaches 1, 1, and 2, respectively. Thus, the largest errors are near these corners.

Using the code `poisson9pt.m` on the course web page for the assigned problem, I got the following results:

h	L_2-norm of error	ratio err(2h)/err(h)
1/4	$7.3e-5$	
1/8	$8.1e-6$	8.9817
1/16	$9.8e-7$	8.2650
1/32	$1.2e-7$	8.5086

3. We have discussed using finite element methods to solve elliptic PDE's such as

$$\triangle u = f \quad \text{in } \Omega, \quad u = 0 \quad \text{on } \partial\Omega,$$

with *homogeneous* Dirichlet boundary conditions. How could you modify the procedure to solve the *inhomogeneous* Dirichlet problem:

$$\triangle u = f \quad \text{in } \Omega, \quad u = g \quad \text{on } \partial\Omega,$$

where g is some given function? Derive the equations that you would need to solve to compute, say, a continuous piecewise bilinear approximation for this problem when Ω is the unit square $(0, 1) \times (0, 1)$.

Let the total number of interior nodes be N and the total number of boundary nodes be M. Write the approximate solution as $\hat{u}(x, y) = G(x, y) + \sum_{k=1}^{N} c_k \varphi_k(x, y)$, where φ_k is a bilinear basis function associated with interior node k, and G matches or approximately matches g on $\partial\Omega$; for example, if we number the boundary nodes as $N + 1, \ldots, N + M$, then we could take $G(x, y)$ to be the piecewise bilinear interpolant of g on $\partial\Omega$:

$$G(x, y) = \sum_{k=N+1}^{N+M} g_k \varphi_k(x, y),$$

where g_k is the value of g at boundary point k. Choose the coefficients $c_1, \ldots, c_N$ so that

$$\int\int_{\Omega} (\hat{u}_{xx} + \hat{u}_{yy}) \varphi_\ell \, dx\, dy = \int\int_{\Omega} f \varphi_\ell \, dx\, dy, \quad \ell = 1, \ldots, N.$$

After using Green's theorem, this becomes

$$-\int\int_{\Omega} (\hat{u}_x \varphi_{\ell x} + \hat{u}_y \varphi_{\ell y}) \, dx\, dy + \int_{\partial\Omega} \hat{u}_n \varphi_\ell \, d\gamma = \int\int_{\Omega} f \varphi_\ell \, dx\, dy, \quad \ell = 1, \ldots, N.$$

Since each φ_ℓ, $\ell = 1, \ldots N$, associated with an interior node is 0 on the boundary, the boundary term vanishes and we are left with

$$-\int\int_\Omega (\hat{u}_x \varphi_{\ell_x} + \hat{u}_y \varphi_{\ell_y})\, dx\, dy = \int\int_\Omega f \varphi_\ell \, dx\, dy, \quad \ell = 1, \ldots, N.$$

Now substituting the expression for $\hat{u}$, we can write

$$-\int\int_\Omega \left[\left(\sum_{k=1}^{N} c_k \varphi_{k_x} + \sum_{k=N+1}^{N+M} g_k \varphi_{k_x} \right) \varphi_{\ell_x} + \left(\sum_{k=1}^{N} c_k \varphi_{k_y} + \sum_{k=N+1}^{N+M} g_k \varphi_{k_y} \right) \varphi_{\ell_y} \right] dx\, dy =$$

$$\int\int_\Omega f \varphi_\ell \, dx\, dy, \quad \ell = 1, \ldots, N.$$

Pulling sums outside the integrals and bringing known terms to the right-hand side, this becomes

$$-\sum_{k=1}^{N} c_k \int\int_\Omega (\varphi_{k_x} \varphi_{\ell_x} + \varphi_{k_y} \varphi_{\ell_y})\, dx\, dy =$$

$$\int\int_\Omega \left[f \varphi_\ell + \sum_{k=N+1}^{N+M} g_k (\varphi_{k_x} \varphi_{\ell_x} + \varphi_{k_y} \varphi_{\ell_y}) \right] dx\, dy, \quad \ell = 1, \ldots, N.$$

The matrix is the same as for a standard bilinear finite element approximation for Poisson's equation with Dirichlet boundary conditions:

$$A_{\ell,k} = \int\int_\Omega (\varphi_{k_x} \varphi_{\ell_x} + \varphi_{k_y} \varphi_{\ell_y})\, dx\, dy.$$

The right-hand side is different for equations corresponding to nodes that are next to the boundary.