MAPL 600 / CMSC 760 Fall 2007
Take-Home Exam 1
Show all work.
All work must be your own (i.e., no group efforts are allowed).
If you use a reference book, cite it, or you will lose credit!
Work problems totaling 50 points.
(I'll stop grading after that, so don't hand in extra parts.)
Due: Friday Sept 28, 8am. (See late penalty policy on information sheet.)

1. (10) On the website `http://www.cs.umd.edu/users/oleary/SCSCwebpage/cs_direct` you will find a program `problem4.m` that tests various reordering methods for applying a sparse direct method to a linear system of equations. The methods include `specnd.m`, an implementation of spectral partitioning, available in a freely-available Matlab package. Run the program as is, using either the package (that you search for and download from the web) or your own implementation, following Demmel's notes. Also run the program on the matrix obtained from `load('west0479.mat')`. Discuss the results, comparing the algorithms and identifying their strengths and weaknesses.

2. (10) Does SOR converge for strictly diagonally dominant matrices for $1 \leq \omega < 2$? Give a proof if the answer is yes and a counterexample if the answer is no.

3. Let $\mathbf{A}$ be an $n \times n$ matrix. Denote the rows of $\mathbf{A}$ by $\mathbf{a}_1^T, \ldots \mathbf{a}_n^T$ and the elements of the vector $\mathbf{b}$ by $b_1, \ldots, b_n$. Assume $\mathbf{a}_i^T \mathbf{a}_i = 1$ for $i = 1, \ldots, n$. Consider the following algorithm for solving $\mathbf{A}\mathbf{x} = \mathbf{b}$:

Choose an arbitrary $\mathbf{x}^0$.
For $k = 0, 1, \ldots,$

$\bar{\mathbf{x}}^{(1)} = \mathbf{x}^{(k)}$
For $j = 1, 2, \ldots, n$
$\bar{\mathbf{x}}^{(j+1)} = \bar{\mathbf{x}}^{(j)} - \mathbf{a}_j \mathbf{a}_j^T \bar{\mathbf{x}}^{(j)} + b_j \mathbf{a}_j$
End for
$\mathbf{x}^{(k+1)} = \bar{\mathbf{x}}^{(n+1)}$

End for

3a. (10) Show that this is a stationary iterative method. (Hint: It is sufficient to express it in the form $\mathbf{x}^{(k+1)} = \mathbf{G}\mathbf{x}^{(k)} + \mathbf{c}$.)

3b. (10) Give a sufficient condition on the matrix $\mathbf{A}$ to ensure convergence.

4. This problem involves the *five-point operator matrix* discussed in class, except that we add to it the matrix $\sigma\mathbf{I}$.

Consider the partial differential equation

$$-u_{xx} - u_{yy} + \sigma u = f(x, y),$$

$$u(0, y) = u(1, y) = u(x, 0) = u(x, 1) = 0.$$

Let v_{ij} be our approximation to the value $u(ih, jh)$, $i, j = 0, \ldots, m$, with $h = 1/m$. Then using finite difference approximations to the second derivatives, we can write a linear system $\mathbf{Av} = \mathbf{b}$ for the unknown values v_{ij} as follows:

$$\frac{-v_{i+1,j} + 2v_{ij} - v_{i-1,j}}{h^2} + \frac{-v_{i,j+1} + 2v_{ij} - v_{i,j-1}}{h^2} + \sigma v_{ij} = f(ih, jh),$$

for $i, j = 1, \ldots, m$. (Note that we are abusing notation somewhat by labeling the elements of the vector v with two subscripts. You can think of the elements as stacked in a column vector, but programming is somewhat simpler if you store v as a 2-dimensional array.)

Let $f(x, y)$ be the function corresponding to the true solution $u(x, y) = x^3 + y^3 + \cos(xy)$.

4a. (10) For what values of σ is the matrix strictly diagonally dominant? Justify your answer.

4b. (10) Is the matrix reducible? Justify your answer.

4c. (10) Write a Matlab program to solve this system with $\sigma = 1000$, $m = 20$, using Chebyshev semi-iteration based on the Jacobi splitting. Run three cases, using .31, .61, and .91 as bounds on $\rho(\mathbf{G}_J)$. Start each iteration with the guess $\mathbf{v}^{(0)} = 0$ and do 25 iterations, printing v_{11} and v_{55} at each iteration (so I can easily detect bugs) and printing whatever information you need to get an estimate of the convergence rate of the Chebyshev iteration. Discuss your results: What happens for each of the three bounds on $\rho(\mathbf{G}_J)$? How much work is performed per iteration? (If your work per iteration or your storage is more than $O(m^2)$, you will lose points.)