MATH 609 Numerical Analysis
Programming assignment #2
Basic iterative methods for linear systems

1. Problem Formulation

Write a program for solving the system $Ax = b$ by 1) Jacobi, 2) Gauss-Seidel, 3) SOR (with some choice of the parameter ω, you can experiment with the choice), and 4) SSOR. Use the standard for SOR splitting of A, namely, $A = D - L - U$, where D is the diagonal of A, L is strictly lower triangular and U strictly upper triangular matrices. Do not store the matrix A.

2. Specifications

(1) For stopping the iterations use any of the conditions:

$$\|x^{(m+1)} - x^{(m)}\|_2/\|x^{(0)}\|_2 < TOL, \qquad \|r^{(m)}\|_2/\|r^{(0)}\|_2/ < TOL,$$

where $r^{(m)} = b - Ax^{(m)}$ is the residual of the m-th iterate, or any other criterion you consider appropriate. Set $TOL = 10^{-6}$ or $TOL = 10^{-12}$ if using double precision.

(2) Solve the following linear systems:

Example 1:
$x_0 = 0,\ k_{i-1}(x_i - x_{i-1}) + k_i(x_i - x_{i+1}) + c_i x_i = b_i,\ i = 1, ..., n,\ x_{n+1} = 0$, which after the elimination of x_0 and x_{n+1} is written in the form $Ax = b$ with $x \in R^n$. Solve for the following systems for $n =\ 20,\ 40$:

(a) $k_i = 1$, $i = 0, \ldots, n$, $c_i = 0$ and $b_i = 1/(n+1)^2$, $i = 1, \ldots, n$;

(b) $k_i = 1$, $i = 0, \ldots, n/2$ and $k_i = 10$, $i = n/2+1, \ldots, n$ and $c_i = 0.1/(n+1)^2$ and $b_i = 1/(n+1)^2$, $i = 1, \ldots, n$;

(c) The solution x_i of (1) is an approximation of $u(t_i)$, $t_i = i/(n+1)$ with $u(t)$ the solution of a boundary value problem $-u'' + cu = b$ on the interval $(0, 1)$ with boundary conditions $u(0) = u(1) = 0$. If you plot x_i, it should look like an approximation to the solution of this equation. (There is no requirement to plot this, but it is good for comfirmation purposes.)
Solve the exact equation explicitly (e.g., with an easy choice of c and b), and compute the norm of the error, $E_n = \sum_{i=0}^{n+1} |u(t_i) - x_i|^2$. How does E_n change as the number of iterations increases? How does E_n change as n increases? Note that n can be thought of as the "resolution" of our approximation.

Example 2:

(a) The unknowns are given as a two dimensional array $x_{ij}, i, j = 0, \ldots, n+1$ that satisfy the system

$$(4 + h^2)x_{i,j} - x_{i-1,j} - x_{i+1,j} - x_{i,j-1} - x_{i,j+1} = h^2,$$
$$x_{0,j} = x_{n+1,j} = x_{i,0} = x_{i,n+1} = 0.$$

Here $h = 1/(n+1)$ so that the system represents a finite difference approximation of the boundary value problem $-\Delta u + u = 1$ in $\Omega = (0, 1) \times (0, 1)$ and $u = 0$ on the boundary of Ω. Take $n = 8,\ 16,\ 32$.

(b) In all problems take a r.h.s. $b = h^2(1, 1, \ldots, 1)^t$ and $x^0 = (0, 0, \ldots, 0)^t$ or $x^0 = b$.

(3) The report should be in the specified latex-format and must contain tables with the number of iterations for all cases (plots are welcome as well). Compare the number of iterations for these iteration methods. Submit your printed LaTeX-generated PDF, along with printed out version of your code (which do not need to be put into LaTeX). Email your code to the

TA with instructions on how to run it on a test case.

Remark. If you are getting to big differences in the iteration count then there is something wrong either with your implementation or with the estimates of the maximum and minimum eigenvalues of A you have used.

The penalty for delaying the programming assignment is 5 pts per day (out of 100).