Matrix Computations - Math 221 - Fall 1999 - T Th 3:30 - 5 in 71 Evans Hall
Professor: Jim Demmel, 737 Soda Hall, 643-5386, demmel@cs. A number of lectures will also be given by Prof. Beresford Parlett, 799 Evans Hall, 642-6655, parlett@math.
Teaching Assistant: Plamen Koev, 868 Evans Hall, plamen@math.
Class Home Page: http://www.cs.berkeley.edu/~demmel/ma221
Text: *Applied Numerical Linear Algebra*, SIAM, 1997.
Recommended Text: *Templates for the Solution of Linear Systems*, R. Barrett et al. SIAM, 1994. Describes standard Krylov-space iterative methods for solving $Ax = b$, with guidance on choosing and algorithm and software. Postscript and html versions of the book, and software, arew available http://www.netlib.org/templates/.
Prerequisites: Good knowledge of linear algebra, programming experience, numerical sophistication at level of Ma. 128ab or equivalent.
Other Reading

1. *Numerical Linear Algebra*, L. N. Trefether and D. Bau, SIAM, 1997. This book is also aimed at a first year graduate audience, but has a more pure mathematical flavor than the main text.

2. *Matrix Computations*, G. Golub and C. Van Loan, 3rd Ed. Johns Hopkins Press, 1996. Very complete, if not encyclopedic, book on matrix computations.

3. *Fundamentals of Matrix Computations*, David Watkins, Wiley, 1991. Very readable beginning graduate level textbook.

4. *LAPACK Users' Guide*, E. Anderson et al. SIAM 1995 (2nd Edition). Describes widely used library of dense numerical linear algebra software. Software and text also available at http://www.netlib.org/lapack

5. *The Algebraic Eigenvalue Problem*, J. Wilkinson, Clarendon Press. Somewhat dated but still excellent comprehensive presentation of numerical linear algebra.

6. *Matrix Perturbation Theory*, G. W. Stewart and J.-G. Sun, Academic Press, 1990. Comprehensive account of perturbation theory for linear algebra problems.

7. *Perturbation Theory for Linear Operators*, T. Kato, Prentice Hall. Comprehensive account of analytic perturbation theory for eigenvalues and eigenvectors; chapter 2 covers the finite dimensional case, which is the subject of this course.

8. *Numerical Methods for Least Squares Problems*, Åke Björck, SIAM, 1996. Comprehensive work on methods for linear least squares problems.

9. *Solving Least Squares Problems*, C. Lawson and R. Hanson, Prentice Hall. Survey of methods for linear unconstrained and constrained least squares problems. Older classic, supplanted by Björck's book.

10. *The Symmetric Eigenvalue Problem*, B. Parlett, Prentice Hall. Algebraic and analytic theory of symmetric matrices and algorithms

11. *Accuracy and Stability of Numerical Algorithms*, N. J. Higham, SIAM, 1996. Comprehensive presentation of rounding error analysis for linear equations and least squares problems.

12. *Rounding Errors in Algebraic Processes*, J. Wilkinson, Prentice Hall. Older classic, supplanted by Higham's book.

13. *LINPACK User's Guide*, J. Dongarra, C. Moler, J. Bunch, G. Stewart, SIAM - user's manual for earlier linear equation solving library; predecessor to LAPACK

14. *Iterative Methods for Sparse Linear Systems*, Yousef Saad, PWS Publishing, 1996. Expands on iterative methods in chapter 6 of the textbook.

15. *Iterative Methods for Solving Linear Systems*, Anne Greenbaum, SIAM. , 1997. Another good, modern account of iterative methods.

16. *A Multigrid Tutorial*, W. Briggs, SIAM, 1987. A good introduction to how multigrid methods work.

Grading: Final: 50%, Programs: 30%, Homework: 20%.

Homework or programs turned in late will receive only half credit.

You may work together on homework, but it should be turned in individually. On the other hand, it is all right to discuss programs with one another, but work should be done individually. Programs will be of two kinds, Fortran, C or C++ (your choice), and Matlab. Matlab software related to the course is available on the class homepage. Assignments to be written in Fortran/C/C++ will use subroutines from libraries like LAPACK or CLAPACK. Matlab is an interactive, user-friendly interface to a large body of numerical and graphics software, including linear algebra, and is widely used for testing and prototyping algorithms. You may find it convenient to do mixed language programming (e.g. calling a Fortran routine from C or C++, or a C routine from Matlab).

Computer Resources: There is a campus site license for Matlab, so it should run everywhere. You will also need a WWW browswer like Netscape; this should also be widely available. If you have difficulty finding appropriate resources, please see me and I will give you an account on socrates; you may prefer the resources in your own department. Math grad students must use socrates accounts. Accounts, manuals, and documentation standards will be handed out later.

Matlab documentation is available from at least three sources. First, there is an extensive online help facility (just type "help commandname" or "help" in Matlab). Second, a brief manual is available on the class homepage. Third, comprehensive manuals can be inspected or purchased in the Computer Center in Evans Hall.

Syllabus:

The standard problems whose numerical solution we will study are systems of linear equations, least squares problems, eigenvalue problems, and singular value problems. Techniques for dense and sparse problems will be covered; it is impossible to cover these areas comprehensively, but students should still come to appreciate many state-of-the-art techniques and recognize when to consider applying them. We will also learn basic principles applicable to a variety of numerical problems, and apply them to the three standard problems. These principles include (1) matrix factorizations, (2) perturbation theory and condition numbers (3) effects of roundoff error on algorithms, including properties of floating point arithmetic (4) analyzing the speed of an algorithm, and (5) choosing the best algorithm for the mathematical structure of your problem, and (6) engineering numerical software.

In addition to discussing established solution techniques, open problems will also be presented.