CME 108: Introduction to Scientific Computing  
2012-2013 Summer Quarter

Location: Room 200 – 305  
Time: MWF 4:15-5:30

Instructor:  
Arvind Krishna Saibaba (arvindks@stanford.edu)  
PhD Candidate, ICME, Stanford University

Description:  
Numerical computation for mathematical, computational, physical sciences and engineering: error analysis, floating-point arithmetic, nonlinear equations, numerical solution of systems of algebraic equations, banded matrices, least squares, unconstrained optimization, polynomial interpolation, numerical differentiation and integration, numerical solution of ordinary differential equations, truncation error, numerical stability for time dependent problems and stiffness. Implementation of numerical methods in MATLAB programming assignments.

Pre-requisites:  
Calculus (single- and multi-variable), familiarity with linear algebra, programming experience (in any language, not necessarily MATLAB) including fundamental concepts such as loops and conditionals. More formally, MATH 51, 52, 53 or equivalent; prior programming experience (MATLAB or other language at level of CS 106A or higher).

Enrollment options and grading:  
Grades will be based on 6 problem sets (60%), a take-home final exam (40%). For those enrolled in 4 units, they will have to do an additional project which will count as one homework (i.e. 6 psets + project = 60% and take-home exam = 40%).

Problem sets:  
In order to reinforce material that is discussed in class, students will be continuously evaluated by means of weekly problem sets. Each problem set will involve some written parts (emphasizing theoretical concepts) and programming assignments.

Final exam:  
This will be a 24 hour take-home final exam that will cover all the material taught in class. The exam will be open-book, open notes and open internet. However, you may not consult anyone else but the instructor for clarification.

Collaboration policy:  
Collaboration is welcome and is encouraged on the problem sets. However, it is important that you write down the names of your collaborators. Everyone must submit their own write-ups and code must not be duplicated. Sharing code is explicitly prohibited. As mentioned earlier, the exam cannot be discussed with anyone else. General guidelines for acceptable levels of collaboration can be found online at http://cs.stanford.edu/degrees/ug/HonorCode.shtml.
Projects (for students taking the class for 4 units):
This will be useful to take the opportunity to either learn a new topic or to apply topics discussed in class to some cool applications. Highly recommend this option, especially for graduate students.

Office hours:
An hour following each lecture, or by appointment.

References:
There is no required book for this class. Some course notes will be posted and links to several online resources will be provided during the course, as and when necessary. The following books are useful references.


Topics covered: This is just to provide a rough idea and is subject to change.

1. Absolute and relative errors, round-off errors in floating point arithmetic, Taylor series and finite difference approximations.
2. Nonlinear equations in a single-variable, algorithms for root-finding including bisection, secant and Newton-Raphson method, iterative methods by successive refinement and designing termination or convergence criteria.
5. Nonlinear system of equations: Newton-Raphson method, convergence criteria
6. Unconstrained optimization: optimality conditions, iterative algorithms including steepest descent, Newton's method.
7. Interpolation: Fitting data with curves, polynomial and piecewise polynomial interpolation, radial basis function interpolation.
10. Numerical methods for ordinary differential equations: Solution of initial value problems, time step selection based on stability and accuracy considerations, including concept of “stiffness”, solution of boundary value problems with finite differences, with appropriate enforcement of boundary conditions, or via the shooting method.