Syllabus for Scientific Computing Candidacy Exam in Computer Science Dept.

It is understood that “Scientific Computing” is an enormous field, potentially encompassing everything from simulation of physical systems, to study of numerical errors, to applied mathematics, to scientific visualization, and beyond. Students will obviously focus on some sub-area, and the committee when evaluating the candidate should acknowledge this and be flexible.

Relevant subjects are listed in boldface. Suggested texts are listed afterwards.

**Algorithms and Data Structures:** the basic knowledge expected of any Computer Science student. *Algorithm Design*, J Kleinberg, E Tardos (2005 Addison-Wesley).

**Linear Algebra:** Golub + Van Loan, *Matrix Computations*, Johns Hopkins 1989; or Haber, *Applied Numerical Linear Algebra*; or *Applied Linear Algebra* by Lloyd N. Trefethen; or just about any other text in applied linear algebra.


**Data Mining & Machine Learning:** Duda, Hart and Stork. *Pattern Classification*.


**OVERVIEW:**

The candidate should be able to demonstrate a thorough knowledge of at least three of the boldfaced topics above, and a rudimentary knowledge of the others.

The student should be familiar with **data structures and algorithms** such as lists, trees, arrays + strings, collections, dictionaries, queues, stacks, graphs, and sorting, as well as time-bound analysis of associated algorithms.
The student should be aware of **probability and statistics** as applied to scientific problems, such as: noise in statistical sampling; parameter estimation, including error estimation.

The student should have a basic understanding of various techniques for **numerically solving linear systems**, such as Gaussian Elimination with pivoting, as well as iteration methods such as Gauss-Seidel; zero-finding techniques such as Newton’s method.

The student should be familiar with the techniques, strengths, and weaknesses of various forms of **optimization**, including convex + constrained optimization, stochastic gradient descent, conjugant gradient, etc [Eric, anything to add here?]

The student should have a basic understanding of the effects of **numerical errors** on computations, including roundoff error, truncation error, and propagation of errors in floating-point computations, as well as an understanding of convergence and stability of numerical algorithms.

**Suggested Curriculum:**

Required courses for candidacy exam:

1. Research Seminar (ICS200)
2. Computer Systems Architecture (CS 250A)
3. Distributed Computer Systems (CS 230)
4. Parallel Computing (CS 242)
5. Queueing Theory (CS 231/Net Sys 251)
7. Data Structures (CS 261)
8. Intro to AI (CS 271)

It is recommended that students take six more graduate courses, at least three of which must be from *outside* ICS, in disciplines appropriate for the area of research to which the student intends to apply scientific computing. For example: physics, math, chemistry, biology, etc. Other recommended courses within ICS include, but are not limited to:

- Info retrieval, Filtering, and Classification (CS 221)
- Internet Technology (CS 232)
- Network Security (CS 203/Net Sys 240)
- Data Mining (CS 285)