Syllabus
MATH663/AMSC663 Fall 2007
Advanced Scientific Computing I (ASC I)

Instructors: Aleksey Zimin, Radu Balan
Office Hours: Tuesday, 2-3pm; or by appointment otherwise
Contact Information: alekseyz@ipst.umd.edu, CSS Building 224, Room 4317, x52348
rvbalan@math.umd.edu, Math Building, Room 2308, x55492 / CSCAMM, Room 4131, x51217

• Project Design Proposal
  o By the end of the first month each student must find a faculty advisor and identify a suitable project that includes a deliverable suite of software that is designed to carry out a computational scientific task, propose appropriate algorithms, languages, and platforms for the development of this software, write a short project proposal that also includes a scientific justification, and present the proposal orally. The proposal must also include the description of the computational facilities to be used in the project.
  o Project description should also include outline of a test problem that will be used for verification.
  o Meeting with both instructors and scientific advisor at least one week before oral presentation.
  o Proposal should be no more than 5 pages.
  o Oral presentations at the end of September. Plan on 30 minutes, including questions and discussion.
  o Encourage your scientific advisor to attend the oral presentation. Will require attendance of scientific advisor at end-of-semester presentations.

• Code Development
  o modularity, portability, memory management
  o post-processing, restarting, and writing to databases
  o interactivity
  o scientific visualization
  o documentation and version management tools
  o debugging and profiling tools

• Implementation of Parallel Algorithms (time permitting)
  o use of distributed memory and local caches
  o masking communication costs, load balancing, granularity
  o parallel numerical linear algebra for full and sparse matrices
  o basic parallel algorithms for PDE, e.g. simple domain decompositions

• Project Progress Report
  o Each student must give a written and oral report on the state of his or her project, explain how the software has been developed and tested, give his or her current vision of the finished product, and detail how that vision has evolved over the course of the project.