Syllabus
AMSC664/CMSC664 Spring 2016
Advanced Scientific Computing II (ASC II)

This course is a continuation of AMSC 663 (ASC I). This course focuses on issues of algorithm development code development and validation relevant to scientific computing and large-scale computations on high performance computing systems.

**Instructors:** Howard Elman, Radu Balan

**Classes:** Tuesday, Thursday, 9:30am – 10:45pm (or as needed) in CSIC 4122

**Office Hours:** Radu Balan: Tuesday, 12:30pm-1:30pm in CSCAMM; Howard Elman: by appointment

**Contact Information:**

Howard Elman: elman@cs.umd.edu, AV Williams 3125, x52694
Radu Balan: rvbalan@math.umd.edu, Math Building, Room 2308, x55492 / (CSCAMM) CSIC Building, Room 4131, x51217

**Prerequisites:** AMSC 663 (ASC I)

Students’ responsibilities in AMSC664:

- Students will provide weekly statements of their project status. The statements are due before the beginning of the scheduled weekly class period.
- Each student will prepare and give a 25-minute oral presentation between March 1st and March 15th detailing the status of the project. At the time of this presentation the implementation must be in its final stages or complete.
- Students will prepare and give a comprehensive 30-minute final project presentation and submit the final project report at the end of the course. This project report should be comprehensive. Your advisor’s presence at the presentation is required. Students are advised to start preparing this report well ahead of the last class to allow for feedback and corrections to occur before final submission.

Students are responsible for achieving the project goals that were listed in the proposal. Grading takes into account student’s understanding of the project, its implementation, timing of the weekly reports, presentation quality and the final report quality.

The class will meet in CSIC 4122 as needed. In addition to student presentations, there may be meetings in which other presentations are given, on topics including

- Large-scale optimizations, particularly quadratic programs and implementation issues.
- Examples of Parallel Algorithms and Code Development
  - Several examples of algorithm development and implementation from instructors’ areas of research (e.g. OpenMP)
Computational Science Seminars
  o Local and visiting computational scientists representing a broad selection of scientific disciplines will explain the computational aspects of their work with emphasis on code development and validation. Students are expected to listen to these seminars critically.

Students are encouraged to meet with the instructors by appointment to discuss their progress.