

Mathematics 5490: Parallel Computing
Craig C. Douglas, <mailto:cdougla6@uwyo.edu>, <http://www.mgnet.org/~douglas>

Syllabus, Spring 2009

Online syllabus: Throughout the entire semester, this syllabus will be online at the URL

<http://www.mgnet.org/~douglas/Classes/na-sc/2009s-notes/syllabus.pdf>

Course Description: A one semester self contained course on parallel computing. Review of parallel architectures, hardware accelerators, programming paradigms, communications methods, applications, algorithms, how to buy or build a supercomputer, and how different scales of parallelism affect performance.

Prerequisites: Permission of the instructor.

Classrooms: Ross Hall 247 (MW, 1:20-2:35)

Class web page: <http://www.mgnet.org/~douglas/Classes/na-sc/2009s-index.html>

TA: none

Office hours: Monday 4:30-5:30 and Tuesday 8:30-9:30 or by appointment (send me email or call for an appointment). Please call 766-6580 before coming to Ross Hall 227.

Textbook: J. Dongarra, G. Fox, K. Kennedy, and L. Torczon, W. Gropp, I. Foster, A. White, The Sourcebook of Parallel Computing, Computer Architecture and Design series, *Morgan Kaufmann*, San Francisco, 2002. ISBN-10: 1558608710, ISBN-13: 978-1558608719. Amazon sells it at a discount

Suggested Readings:

- C. C. Douglas, G. Haase, U. Langer, A Tutorial on Elliptic PDE Solvers and their Parallelization, vol. 16, Software, Environments, and Tools (SET) series, *Society of Industrial and Applied Mathematics* (SIAM), Philadelphia, 2003.
- W. Gropp, E. Lusk, and A. Skjellum, Using MPI, *MIT Press*, Cambridge, 1997.
- W. Gropp, E. Lusk, R. Thakur, Using MPI-2, *MIT Press*, Cambridge, 1999.
- G. E. Karniadakis and R. M. Kirby II, Parallel Scientific Computing in C++ and MPI: A Seamless Approach to Parallel Algorithms and Their Implementation, *Cambridge University Press*, 2003 (with a cdrom of software).

Homework: There will be homework in the form of pencil and paper analysis as well as programming assignments. Each assignment is due at latest when stated on the class homework web page, which you are responsible for checking daily. Late work will only be accepted by prior arrangement unless there are exception circumstances.

Exams/Projects: There will be quizzes and exams in this course. In addition, there will be at least one project that you will be responsible for presenting your results to the class in a formal lecture using the LCD beamer and computer presentation. No blackboard presentations allowed.

Grading: 100% of the grade will come from the exams/projects (50%) and the homework (50%).

Advice:

- It is a good idea to already know Matlab to prototype algorithms and at least one legacy programming language (e.g., C, C++, or Fortran). I will teach C at the very beginning of the course in about one class lecture and suggest that the class use C.
- Do not be late to class in case I give an unannounced quiz in the first 10 minutes.
- Take notes in class. My lecture notes will not be online during the semester.

Cheating Policy: Getting caught cheating or plagiarizing will result in a failing grade and possibly much worse, including expulsion from the university and legal proceedings against you. I have zero tolerance for cheaters. I will enforce whatever the latest university policy is. When in doubt, ask me first. Life is too short to experience the penalties of getting caught cheating.