Overview: This course will cover parallel computing in scientific computing applications using distributed and shared memory programming on multicore and massively parallel processors, as well as general purpose graphics processing units (GPGPU). Application interfaces include MPI, OpenMP, CUDA, OpenCL, and other parallel scientific computing libraries.

Texts: The content of this course is based on material from:
- *An Introduction to Parallel Programming* by Peter S. Pacheco
- *Heterogeneous Computing with OpenCL* by Benedict Gaster, Lee Howes, David R. Kaeli, Perhaad Mistry, and Dana Schaa

Prerequisites: CAAM 519, or familiarity with scientific computing, Unix/Linux, \LaTeX, C/C++, gdb, valgrind, and similar tools.

Homework: Grades are based on homework and projects. All homework must be typeset in \LaTeX with citations for outside resources used and submitted using Git version control.

All assignments involve coding exercises. If you run out of time and your code does not work, you should explain briefly in your report what you think the remaining issues are.

Late policy: Late assignments will be penalized 25%. No work will be accepted more than one class period (two days) late without prior arrangement (e.g. one 'late' per assignment.)

Honor Code: It is an honor code violation to turn in code or solutions which have, in all or in part, been copied from another student (including computer codes). It is also an honor code violation to consult solutions to assignments from previous sections of this or similar classes.

If you have a documented disability that may affect academic performance, you should make sure this documentation is on file with Disability Support Services to determine the accommodations you need; and meet with me to discuss your accommodation needs.