CHEM 284:
High Performance Computing for the Molecular Sciences

Instructor: Taylor Barnes, tbarnes1@berkeley.edu
Spring 2026: The course will first be taught in Spring 2026, and will include 15 weeks of instruction.
Credit Hours: 3

Course description: This course introduces parallel programming concepts commonly encountered in computational molecular sciences codes, including distributed-memory parallelization, shared-memory parallelization, and GPU parallelization. Students will become familiar with MPI, OpenMP, and CUDA, while also learning specific strategies for addressing performance challenges associated with key computational chemistry algorithms. Special emphasis is placed on the execution of machine learning techniques within a high performance computing environment, as well as challenges arising from processing large amounts of data.

Learning objectives for this course: Upon successfully completing this course, students will understand the parallelization strategies commonly employed within the domains of computational chemistry and machine learning, and will be able to design and develop intelligently parallelized codes that make proper use of modern tools and libraries.

Contribution of this course to the broader curricular objectives: This course serves as a core requirement of the MSSE curriculum, and covers essential parallelization tools used throughout many subdomains of computational molecular science. Additionally, it builds upon the machine learning skills introduced in Chem 277B, with the goal of ensuring that students are able to efficiently and productively use these concepts within a high performance computing context.

Course format: This is an online course consisting of 3 hours of asynchronous recorded lectures per week, 1 hour of synchronous online discussion per week, and 2 hours of synchronous online lab work every other week. GSIs will go over homework assignments and practice exercises, prepare students for their homework assignments, and post answer guides to homework assignments after they are submitted. Outside class work should comprise 6 hours per week, for an average total of 11 hours per week.

Prerequisites:
- Admission to the MSSE Program
- Chem 277B: Machine Learning Algorithms or instructor approval

Software competency/access: No commercial software licenses are required for this course. Students should have access to and be proficient in using bCourses. Free IDEs, tools, and libraries for Python and C++ are available and are recommended for this course. Unix or Linux software is also required and is freely available. Students will be provided with recommendations of which IDEs, libraries, tools, and Unix/Linux distribution to use. Students will need access to a computing cluster with GPU nodes (such as NERSC's Perlmutter system).

Reading List and Resources:
These textbooks are provided as supplemental options for the student. They are not

required, but are strongly recommended to get the most out of the course.

- Deep Learning with PyTorch by Eli Stevens, Luca Antiga, and Thomas Viehmann.

All needed course materials will be included in the bCourses site which may include links to external public domain resources.

Grading: There will be 2 programming projects ("Project 1" and "Project 2") and 5 problem sets, which will be graded by GSIs or faculty. Final grades for the course will be weighted as follows:

- 50% problem sets
- 30% programming projects
- 20% completion of synchronous discussion and lab assignments

This course does not include a final exam as it emphasizes hands-on learning and real-world application of computational techniques, making projects a more effective assessment tool than traditional exams.

Assignment Types:

- Students will complete 5 problem sets which are due every 3-4 weeks, and cover material from the preceding 3-4 weeks. Each problem set is weighted at 10% of the course grade.
- There are two programming projects required for the course each weighted at 15% of the course grade. Programming projects are larger assignments integrating concepts from approximately half the course. Project 1 is due in week 9 and Project 2 is due during finals week.
- Discussion and Lab are online sessions in which students put concepts learned in class into action alongside their classmates. Attendance is required. Each discussion and lab will have an assignment that is to be completed during the session and turned in at the end of the session. Some assignments will be completed collaboratively with a team.

Course Attendance: Each student is required to view each lecture recording. Missing a synchronous session (discussion and lab) will result in no credit being awarded for the missed session unless the absence is pre-arranged with the instructor. Accommodations may also be made for emergencies.

Accommodations for Religious Creed: Accommodations will be made for observances and practices associated with religious creed. For more information, see the campus's Religious Creed Policy.

Late work: For problem sets and programming projects, late work will be accepted for a penalty of 10% per day until a week after the assignment due date. Problem sets and programming projects turned in more than a week late will not be accepted without pre-approval from the instructor.

Accommodations for Disabilities: UC Berkeley is committed to creating a learning environment that meets the needs of its diverse student body including students with disabilities. If you have a disability, or think you may have a disability, you can work with the Disabled Students' Program (DSP- https://dsp.berkeley.edu/) to request an official accommodation. If you already have an accommodation letter from DSP, please check to make sure that the letter is submitted through the DSP system (there is no need to email a separate copy). If you would like to set up an individual meeting to discuss your accommodations, please contact the instructors.

Collaboration Policy: Unless otherwise instructed, all assignments are to be completed independently and materials submitted as homework should be the result of one's own independent work.

Course requirements: Each student is required to view all of the online lectures, submit all homework assignments and projects, and attend discussion and lab sessions. A laptop, workstation, or access to a UNIX-style account is required, as is installation of an Emacs or VI editor.

Contacting the Instructor and GSIs: All communications with the Instructor and GSIs should be sent via bCourses.

Office hours: The instructor will be available 1 hour per week at **TBD** for one-on-one consultation during office hours or by appointment. The GSIs will be available 4 hours a week at **TBD**. A calendar of these synchronous office hours will be posted on the course website.

Course Schedule:
Course Start Date: January 20th, 2026
Course End Date: May 15th, 2026
The course is scheduled to follow the normal Spring Semester schedule, providing 15 weeks of instruction.

Weekly Topics
Week 1:
Topics: Introduction to Parallel Computing, Scaling, and Using Supercomputers
Activities:
- Read assigned materials for Week 1 in bCourses
- Watch recorded lectures
- Participate in online discussion

Week 2:
Topics: Introduction to MPI
Activities:
- Read assigned materials for Week 2 in bCourses
- Watch recorded lectures
- Participate in online discussion and lab

Week 3:
Topics: Distributed-Memory Parallelization Algorithms
Activities:
- Read assigned materials for Week 3 in bCourses
- Watch recorded lectures
- Participate in online discussion

Week 4:
Topics: Introduction to OpenMP
Activities:
- Read assigned materials for Week 4 in bCourses
- Watch recorded lectures

- Participate in online discussion and lab

Assignments:
- Problem Set 1 (Due Date: TBD)

Week 5:
Topics: Shared-Memory Parallelization Algorithms
Activities:
- Read assigned materials for Week 5 in bCourses
- Watch recorded lectures
- Participate in online discussion

Week 6:
Topics: Profiling and Optimizing Parallel Codes; The Roofline Model
Activities:
- Read assigned materials for Week 6 in bCourses
- Watch recorded lectures
- Participate in online discussion and lab

Week 7:
Topics: Cache Re-use, Matrix Multiplication Techniques, and HPC Architecture
Activities:
- Read assigned materials for Week 7 in bCourses
- Watch recorded lectures
- Participate in online discussion

Assignments:
- Problem Set 2 (Due Date: TBD)

Week 8:
Topics: Key Parallelized Libraries (Boost, BLAS, LAPACK, Eigen3, FFTW3, etc.)
Activities:
- Read assigned materials for Week 8 in bCourses
- Watch recorded lectures
- Participate in online discussion and lab

Week 9:
Topics: Introduction to GPU Computing and CUDA
Activities:
- Read assigned materials for Week 9 in bCourses
- Watch recorded lectures
- Participate in online discussion

Assignments:
- Project 1 (Due Date: TBD)

Week 10:
Topics: Building a Custom GPU-Parallelized Neural Network
Activities:
- Read assigned materials for Week 10 in bCourses
- Watch recorded lectures
- Participate in online discussion and lab

Week 11:
Topics: Introduction to PyTorch
Activities:
- Read assigned materials for Week 11 in bCourses
- Watch recorded lectures
- Participate in online discussion

Assignments:
- Problem Set 3 (Due Date: TBD)

Week 12:
Topics: Managing ML Data: Python vs C++ Fundamental Types and Performance Implications; Managing Large Data in an HPC Context; Serialization and HDF5
Activities:
- Read assigned materials for Week 12 in bCourses
- Watch recorded lectures
- Participate in online discussion and lab

Week 13:
Topics: Parallel Computing With PyTorch
Activities:
- Read assigned materials for Week 13 in bCourses
- Watch recorded lectures
- Participate in online discussion

Week 14:
Topics: More Advanced Operations in PyTorch: Updating Models; Data Augmentation, Convolution, GANs, etc.
Activities:
- Read assigned materials for Week 14 in bCourses
- Watch recorded lectures
- Participate in online discussion and lab

Assignments:
- Problem Set 4 (Due Date: TBD)

Week 15:
Topics: SE(3)-Transformers; RFDiffusion Case Study
Activities:
- Read assigned materials for Week 15 in bCourses
- Watch recorded lectures
- Participate in online discussion

Week 16: RRR Week

Week 17 (Finals Week):
Assignments:
- Problem Set 5 (Due Date: TBD)
- Project 2 (Due Date: TBD)