

# Foundations of Programming and Software Engineering for Molecular Science

# Daily Schedule

09:00 AM - 10:30 AM PT

Instruction

10:30 AM - 10:40 AM PT

Break

10:40 AM – 12:00 PM PT

Instruction

12:00 PM – 01:00 PM PT

Break

1:00 PM – 03:00 PM PT

Zoom Group Work

-- Continue working on assignments if necessary --

- Dr. Pritchard and Dr. Nash are available for office hours from 3:00 PM to 5:00 PM PT
- Incorporate daily group pull requests by next class session (we'll explain what this means!)



# Your Instructors



**Dr. Jessica Nash**



**Dr. Ben Pritchard**



**Dr. Sam Ellis**



# What will you learn?

**Expectations** – At the start of the course, we expect that you understand Python syntax.

By the end of this course, you will be able to answer these questions:

- Broadly, what is Monte Carlo simulation?
- How can I use Monte Carlo simulation to predict the properties of a chemical system?
- What is the Python Standard Library?
- How can I use git to keep a record of my project?
- How can I use GitHub to share code and collaborate on projects?
- What is one of defining features of the array in the library NumPy?
- How can I make visualizations in python?
- What is the difference between an interpreted and a compiled programming language?
- What is the syntax of the C++ programming language?



# What is our goal?

**First and foremost** – our goal is to learn and work together! 😊

In this course, we will be implementing a Monte Carlo simulation of a Lennard Jones fluid.

1. Implementation with the Python Standard Library
2. Implementation with NumPy
3. Implementation in C++

We will compare our simulation to benchmarks computed by the National Institutes of Standards and Technology (NIST).

We will also compare simulation results to experimental data reported by NIST – how accurate are our simulations?

We will compare performance of different implementations of our code. Which is fastest? Python, NumPy, or C++ implementation?

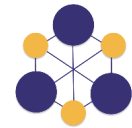


# Assignments

**Individual Assignments** – Individual programming challenges. To be completed on your own (4 total). Quizzes, attendance.

**Group Assignments** – Daily group programming challenges. You should complete one task and review the work of one of your groupmates each day. Submission and review will be on central team repository.

**Presentation** – On the last day of the course, each group should wrap up their project and present to the class (15 minutes per group).



# Grade Breakdown

## Individual Homework: 40%

- 4 individual coding challenges
- Quizzes, assignments, attendance

## Group Assignments: 35%

- 6 group coding assignments

## Participation and Code Review: 10%

- Review group pull requests

## Final project & presentation: 15%

- You should record the final project presentation and turn it in on bCourses. Due at the end of day on the last day of class (we will not meet on Friday, August 11)

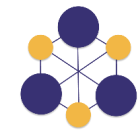


# Course Communication

**Course Website** – Has text lessons and assignments.

**bCourses** – Your homepage for the course. Turn in assignments on bCourses for grading.

**GitHub** – Will be used for peer code review and discussion.





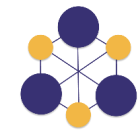
# Course Website

<https://msse-chem-280-2023.github.io/>



# Course Communication

This class involves a team project. Please keep in mind that we want to work together and mentor one another. When you come to this class, please have a **collaborative mindset** (unless it's an individual assignment 😊 ).



# Class Code of Conduct

## Our Pledge

In the interest of fostering an open and welcoming environment, we as students and instructors pledge to making participation in this course and our community a harassment-free experience for everyone, regardless of age, body size, disability, ethnicity, gender identity and expression, level of experience, nationality, personal appearance, race, religion, or sexual identity and orientation.



# Class Code of Conduct

## Our Standards

### Examples of behavior that contributes to creating a positive environment include:

- Using welcoming and inclusive language
- Being respectful of differing viewpoints and experiences
- Gracefully accepting constructive criticism
- Focusing on what is best for the community
- Showing empathy towards other community members

### Examples of unacceptable behavior by participants include:

- The use of sexualized language or imagery and unwelcome sexual attention or advances
- Trolling, insulting/derogatory comments, and personal or political attacks
- Public or private harassment
- Publishing others' private information, such as a physical or electronic address, without explicit permission
- Other conduct which could reasonably be considered inappropriate in a professional setting

