Chem 4050/5050
Computational Problem Solving in the Chemical Sciences

FL2025

Instructor:	Robert Wexler
E-mail:	wexler@wustl.edu
AI:	Eyob Tewelde
AI E-mail:	eyob@wustl.edu
Lecture Time:	M-W-F 11:00A-11:50A
Recitation Time:	R 11:30A-12:50P
Lecture Room:	Seigle Hall 00206
Recitation Room:	Seigle Hall 00004
Lecture Section:	01
Recitation Section:	A
Credits Hours:	3
Department:	Chemistry
Delivery Mode:	In-Person

Contents

1 Course Description

Have you ever wondered how molecular interactions shape the world around us? Why do certain materials exhibit unique properties? How can we predict and manipulate chemical reactions at the atomic level? These are the mysteries at the heart of chemistry, where understanding the unseen world of atoms and molecules can unlock groundbreaking advances in science and technology. However, one needs specialized numerical methods and computational chemistry skills to explore these questions. This course is designed to bridge this gap. It provides a comprehensive introduction to the mathematical and computational skills necessary to model chemical phenomena at the atomic level. We start by building a strong foundation in mathematical representations of chemical problems, utilizing open-source software tools for problem-solving, data interpretation, and visualization of materials and molecular structures. In the second part of the course, we delve into the fascinating world of atomic-level computer modeling. You'll learn various methodologies, such as Monte Carlo and molecular dynamics. We'll analyze static (thermodynamic and structural) and dynamic properties and their statistical errors. Don't worry if you're new to coding – we'll cover the basics of Python programming in the first few lectures, setting you up for success. By the end of this course, you will be proficient in using computational tools, understanding atomic interactions, and approaching chemical problems with a structured and strategic thought process. Join us to unlock the secrets of the molecular world and transform the way you see chemistry!

1.1 Prerequisites

Chem 106/112A, Math 132, Physics 191, Chem 261

1.2 Learning Objectives

1. **Applied Python for Chemical Equilibria and Bonding**: Enhance problem-solving skills by applying Python programming to solve complex problems in chemical reaction equilibria, chemical bonding, and balancing chemical equations.

2. **Statistical Analysis in Chemical Research**: Develop proficiency in analyzing reaction orders and calibration data using regression and correlation techniques in statistics.

3. **Bridging Mathematics and Chemistry**: Deepen understanding of chemistry's mathematical principles, including numerical methods that connect abstract mathematical theories with tangible chemical phenomena.

4. **Simulation Techniques in Modern Chemistry**: Gain expertise in modeling chemical phenomena such as chemical bonding, nanoparticle shape, lipid interactions, surface adsorption, and polymer solvation using Monte Carlo and molecular dynamics simulations.

5. **Integration and Exploration in Computational Chemistry**: Foster an environment for exploring special topics of interest in modern chemistry and computation, encouraging students to link their interests with computational methods.

1.3 Transferable Skills

Upon successful completion of this course, students will build a skill set that includes:

1. **Enhanced GitHub Skills for Portfolio Development**: Through effectively using GitHub, students will learn to curate and showcase their computational projects. This skill is crucial for building a professional portfolio reflecting their computational chemistry expertise.

2. **In-Depth Simulation Technique Insights**: The course will enable students to understand and interpret the outcomes of various simulation techniques thoroughly. This knowledge is key to appreciating these techniques' diverse applications and nuances in modeling chemical behavior.

3. **Linking Theory with Practice**: Students will learn to draw parallels between measured chemical properties and their computational models. This skill is vital for comprehensively understanding chemistry's theoretical and practical aspects.

4. **Critical Application of Simulation Methodologies**: The course will cultivate a critical and informed approach toward applying different simulation methods. Students will learn how to effectively employ these methodologies and understand their limitations in solving chemical problems, promoting a well-rounded approach to computational chemistry.

These goals are designed to equip students with robust skills and knowledge, preparing them for advanced studies or professional careers in the evolving field of computational chemistry.

2 Texts, Materials, and Supplies

There will be no required textbook. Lecture notes will be made available via the course website: wexlergroup.github.io/comp-prob-solv/.

2.1 Non-Required Texts

- Introduction to Python Computations in Science and Engineering, Kitchin
- Mathematics for Physical Chemistry: Opening Doors, McQuarrie
- Understanding Molecular Simulation: From Algorithms to Applications, Frenkel and Smit
- Statistical Mechanics: Theory and Molecular Simulation, Tuckerman
- A Guide to Monte Carlo Simulations in Statistical Physics, Landau and Binder
- Computer Simulation of Liquids, Allen and Tildesley

3 Grading

Your grade in this course will reflect a diverse assessment strategy designed to cater to different learning styles and strengths. I am committed to providing a variety of assessment methods to allow you to demonstrate your understanding and skills effectively.

3.1 Final Grade Composition

- **Homework Assignments (40% of final grade, 200 points)**: You will have five homework assignments designed to reinforce the concepts learned in class and through readings. These assignments will test your understanding and ability to apply these concepts in practical scenarios.
- **Projects (40% of final grade, 200 points)**: Two major projects will be a significant part of your grade. These projects will allow you to delve deeper into specific areas of interest, apply computational methods to solve complex problems and showcase your analytical and coding skills.
- **GitHub Repository (20% of final grade, 100 points)**: Maintaining a GitHub repository will be crucial to your learning journey. This repository will be a portfolio of your work throughout the course, including code from homework and projects. It will be evaluated on organization, documentation, and the demonstration of your computational skills.
- **Optional Final Paper (100 points)**: Students may submit an optional final paper. It will only be included in the final grade calculation if it improves the overall grade. In that case, the paper will account for 100 of 600 total points (17%), and the grade composition will be adjusted to:
  - Homework Assignments (33%, 200 points)

  - Projects (33%, 200 points)
  - GitHub Repository (17%, 100 points)

The assessment diversity is designed to comprehensively evaluate your skills and knowledge, encompassing theoretical understanding and practical application. Your active engagement and consistent effort in all these components will be key to your success in this course. The grade boundaries will be

%	Grade
≥ 90	A- to A
≥ 80	B- to B+
≥ 70	C- to C+
≥ 60	D- to D+
< 60	F

4 Assignments and Homework

The coursework is designed to blend theoretical understanding with practical application, primarily focusing on Python programming and its application in computational chemistry. **Students are expected to spend approximately six to nine hours per week outside of class on studying, homework, and projects.**

4.1 Homework Assignments

Each homework assignment consists of problems that challenge you to complete and write Python code from scratch. These assignments are crafted to enhance your coding skills, deepen your understanding of course concepts, and apply them to solve real-world chemical problems. Key components of these assignments include:

- **Code Development**: Developing Python code to address specific chemical phenomena or problems.
- **GitHub Integration**: Regularly pushing your code to your GitHub repository will be a dynamic record of your progress and skills.

- **Problem-Solving**: Utilizing your code to analyze and answer questions, linking coding skills with chemical knowledge.

4.2 Projects

Projects in this course are more extensive and in-depth than homework assignments. They are designed to simulate real-world scenarios where you will apply computational methods to conduct chemical research. Each project will involve:

- **Complex Coding Tasks**: Completing and writing multiple Python scripts to address more intricate and comprehensive problems in computational chemistry.
- **GitHub Project Management**: Organizing and maintaining a project directory in your GitHub repository, which will include all your codes, data, and documentation.
- **Computer Experiments**: Using your Python codes to perform computational experiments, analyze data, and derive conclusions.
- **Report Writing**: Compiling your findings, methodologies, and conclusions into a detailed report showcasing your ability to communicate complex scientific concepts effectively.

4.3 Optional Final Paper

The optional final paper challenges students to develop a computational research proposal that complements their interests or current research. Students will choose a system and property of interest. They will then apply what they've learned earlier in the course to choose a computational and analytical method to model or simulate their system and property of interest. Students are required to meet with Prof. Wexler to review an outline of their proposal before proceeding to draft the final paper. Students are encouraged to do any or all of the following:

- Review the literature (e.g., using SciFinder, Web of Science, etc.)
- (Judiciously) Utilize generative artificial intelligence (e.g., ChatGPT, Gemini, etc.)
- Consult their classmates, lab mates, AI, etc.

4.4 GitHub Repository Management

As a crucial component of this course, each student must maintain a GitHub repository that serves as a comprehensive portfolio of their work. This repository will showcase their coding and project work and reflect their ability to professionally organize and document their computational journey.

4.4.1 Final Submission Deadline

All final changes to your GitHub repository must be merged by 11:59 PM Central on December 12, 2025. This deadline ensures ample time to refine your repository and present your work in the best possible light.

4.4.2 Repository Guidelines

- **Adherence to Best Practices**: Your repository should align with GitHub's best practices. This includes effective organization, use of branches for development, and regular commits to track changes.
- **Detailed README File**: A critical aspect of your repository is the README file. This file should:
  - Clearly describe the purpose and contents of your repository.
  - Include instructions on how to navigate your repository and run the codes.
  - Summarize your projects and assignments, providing context and insights into your work.
- **Reflective of Coursework**: The repository should comprehensively cover all your work throughout the course, including codes from homework, projects, and any additional exercises or explorations you have undertaken.

This repository will be a part of your grading criteria and a valuable asset in your professional portfolio, demonstrating your skills and approach to computational chemistry. It is an opportunity to showcase your technical abilities, organizational skills, and commitment to best software development practices.

4.5 Late Policy

This policy applies to all graded work in the course (homework, projects, and the GitHub repository):

- All work must be submitted as a link to your GitHub repository by the posted deadline.
- If you anticipate missing a deadline, you should request an extension *before* the deadline. In most cases, Prof. Wexler will grant the extension without penalty.
- If no extension is requested in advance, late penalties will apply as follows:
  - ≤ 1 day late: -10%

- ≤ 2 days late: -30%
- ≤ 3 days late: -60%
- > 3 days late: -100% (no credit can be earned)

- If you request an extension within one day after the deadline, Prof. Wexler may freeze the penalty at -10%; requests within two days may be considered on a similar basis.

4.6 Undergraduate vs. Graduate Student Requirements

This course caters to undergraduate and graduate students, with assignments and expectations tailored to each group's expertise level.

4.6.1 Undergraduate Students

- **Focus**: Building a solid foundation in computational chemistry and Python programming.
- **Assignments**: Aimed to develop basic to intermediate coding skills and understand simple chemical phenomena.
- **Projects**: Scoped to ensure achievability, focusing on applying computational methods to model and analyze defined chemical problems.
- **GitHub Portfolio**: Should demonstrate foundational coding skills, comprehension of computational chemistry concepts, and effective documentation.

4.6.2 Graduate Students

- **Focus**: Tackling complex problems and demonstrating advanced proficiency in computational chemistry.
- **Assignments**: Developing sophisticated code for advanced chemical problems and simulations.
- **Projects**: More demanding, integrating multiple computational methods and extensive data analysis to draw conclusions from complex datasets.
- **GitHub Portfolio**: Showcasing advanced coding skills, a deeper understanding of computational chemistry, and critical analysis of results.

5 Attendance, Participation, and Classroom Climate

While class participation and attendance do not factor into course grades, attendance is mandatory as these are vital components of this course. The structure and content of our lectures are designed to impart knowledge and provide a dynamic, interactive learning environment.

5.1 Interactive Lectures with Python Demonstrations

Each lecture will feature live Python demonstrations, offering you a practical view of how theoretical concepts are applied in computational chemistry. These sessions are an excellent opportunity to clarify doubts in real-time.

5.2 Hands-on Sessions

Regular hands-on sessions will be integrated into the lectures. The instructor and AI(s) will assist you with in-class exercises, homework assignments, and projects during these sessions. This is a perfect time to work collaboratively, receive personalized guidance, and enhance your understanding of course material.

5.3 Weekly Office Hours

I, along with the AI, will host weekly office hours (day and time TBD by poll of students on Day 1) in Jolley 431. These opportunities allow you to seek additional help, discuss course material in depth, or get feedback on your assignments and projects.

6 Technical Requirements and Support Available

This course is designed to provide students with ample technological support and resources to facilitate a comprehensive learning experience in computational chemistry. Understanding and accessing these resources is essential for maximizing your success in this course. **There will be no additional course fees and all required software will be open-source**.

6.1 Personal Computing Requirements

Ideally, each student should have access to a laptop or personal computer. This will be their main tool for coding, accessing course materials, and participating in hands-on activities. For students who may not have a personal computer, the Department of Chemistry offers access to spare laptops. These resources are available to ensure all students have the necessary technology to participate fully in the course.

6.2 Open-Source Software Installation and Access

Instructions for installing Python will be provided on the course website before the first day of class. Familiarizing yourself with Python installation and setup is crucial for a smooth start to the course. Students are encouraged to gain access to Google Colab, a powerful online platform for Python programming. The university's information technology staff will assist in accessing Google Colab and other Google products.

6.3 Additional, Optional Computational Resources

During the semester, you will gain access to high-performance computing resources essential for running complex simulations and analyses. These resources include:

- Research Infrastructure Services (RIS) at Washington University.
- National Science Foundation (NSF)-supported computing facilities.

Access to specialized computer programming courses on DataCamp will be provided. These courses offer additional learning opportunities to enhance your programming skills and understanding of data analysis techniques.

These resources are intended to support your learning journey in this course by providing access to high-quality computational tools and educational materials. Proper utilization of these resources will enable you to engage effectively with the course content and develop your skills in computational chemistry.

7 Course Calendar

1. **Week 1: Introduction to Python Computations** (*August 25–29, 2025*)

   Homework 1 assigned on Friday, *August 29, 2025*

2. **Week 2: Numerical Methods in Python** (*September 3–5, 2025*) (there will be no lecture on Monday, *September 1, 2025* because of Labor Day)

   (a) Chemical Reaction Equilibria and Roots of Equations

   (b) Chemical Bonding and Numerical Integration

   (c) Balancing Chemical Equations and Systems of Linear Algebraic Equations

Homework 1 due on Friday, *September 5, 2025* at 11:59 PM Central

Homework 2 assigned on Friday, *September 5, 2025*

3. **Week 3: Statistics – Regression and Correlation** (*September 8–12, 2025*)
   (a) Orders of Reaction and Linear Regression Analysis
   (b) Calibration Data, Confidence Intervals, and Correlation Analysis

   Homework 2 due on Friday, *September 12, 2025* at 11:59 PM Central

   Homework 3 assigned on Friday, *September 12, 2025*

4. **Week 4: Introduction to Molecular Simulation** (*September 15–19, 2025*)
   (a) Classical Thermodynamics
   (b) Statistical Thermodynamics
   (c) Ensembles
   (d) Ergodicity

   Homework 3 due on Friday, *September 19, 2025* at 11:59 PM Central

   Homework 4 assigned on Friday, *September 19, 2025*

5. **Weeks 5-8: Monte Carlo Simulations** (*September 22–October 17, 2025*)

   Week 5 (*September 22–26, 2025*) (there will be no lecture on Wednesday, *September 24, 2025* because Prof. Wexler will be observing the holiday of Rosh Hashanah; this lecture will be made up the following day, Thursday, *September 25, 2025*, replacing the first hour of the previously scheduled recitation)

   (a) The Monte Carlo Method
   (b) Chemical Bonding and Monte Carlo Integration

   Homework 4 due on Friday, *September 26, 2025* at 11:59 PM Central

   Homework 5 assigned on Friday, *September 26, 2025*

   Week 6 (*September 29–October 3, 2025*) (Prof. Wexler will be absent from recitation on Thursday, *October 2, 2025* because he will be observing the holiday of Yom Kippur; this recitation will have a TBA guest instructor)

   (a) A Basic Monte Carlo Algorithm
   (b) Nanoparticle Shape and Simulated Annealing

   Homework 5 due on Friday, *October 3, 2025* at 11:59 PM Central

Week 7 (*October 9–10, 2025*) (there will be no lecture on Monday, *October 6, 2025* because of Fall break; there will also be no lecture on Wednesday, *October 8, 2025* because Prof. Wexler will be observing the holiday of Sukkot; the lecture on Wednesday, *October 8, 2025* will be made up the following day, Thursday, *October 9, 2025*, replacing the first hour of the previously scheduled recitation)

(a) Technical Details: Boundary Conditions, Truncation of Interactions, *Etc.*

(b) Lipid Interactions in Membranes and Monte Carlo Simulations

Week 8 (*October 13–17, 2025*) (there will be no lecture on Wednesday, *October 15, 2025* because Prof. Wexler will again be observing the holiday of Sukkot; this lecture will be made up the following day, Thursday, *October 16, 2025*, replacing the first hour of the previously scheduled recitation)

(a) Introduction to Project 1: Monte Carlo Simulations of Adsorption on Surfaces

Project 1 assigned on Friday, *October 17, 2025*

6. **Weeks 9-12: Molecular Dynamics Simulations** (*October 20–November 14, 2025*)

Week 9 (*October 20–24, 2025*)

(a) Molecular Dynamics: The Idea

Week 10 (*October 27–31, 2025*)

(a) Molecular Dynamics: A Program

Project 1 due on Friday, *October 31, 2025* at 11:59 PM Central (Advice: Don't wait until the last minute and ruin your Halloween!)

Week 11 (*November 3–7, 2025*)

(a) Equations of Motion

Week 12 (*November 10–14, 2025*)

(a) Introduction to Project 2: Molecular Dynamics Simulations of Linear Polymer Melts

Project 2 assigned on Monday, *November 10, 2025*

7. **Weeks 13-15: Advanced and Emerging Topics in Computational Chemistry** (*November 17–December 5, 2025*)

Week 13 (*November 17–21, 2025*)

(a) Students' choice

Week 14 (*November 24, 2025*) (there will be no lectures on Wednesday, *November 26, 2025* and Friday, *November 28, 2025* and no recitation on Thursday, *November, 27, 2025* because of Thanksgiving break)

(a) Students' choice

Optional Final Paper assigned on Monday, *November 24, 2025*

Project 2 due on Monday, *November 24, 2025* at 11:59 PM Central

Week 15 (*December 2–5, 2025*)

(a) Students' choice

Optional Final Paper due on Friday, *December 12, 2025* at 11:59 PM Central

8 University-Wide Policies & Guidelines

8.1 Academic Integrity

In all academic work, the ideas and contributions of others (including generative artificial intelligence) must be appropriately acknowledged and work that is presented as original must be, in fact, original. You should familiarize yourself with the appropriate academic integrity policies of your academic program(s). I will not utilize TurnItIn functionality in this course. Please see the Navigating Artificial Intelligence Resource webpage for examples.

8.2 Unauthorized Recording and Distribution of Classroom Activities & Materials

The following applies to all students in my class: "Except as otherwise expressly authorized by the instructor or the university, students may not record, stream, reproduce, display, publish or further distribute any classroom activities or course materials. This includes lectures, class discussions, advising meetings, office hours, assessments, problems, answers, presentations, slides, screenshots or other materials presented as part of the course. If a student with a disability wishes to request the use of assistive technology as a reasonable accommodation, the student must first contact the Office of Disability Resources to seek approval. If recording is permitted, unauthorized use or distribution of recordings is also prohibited."

8.3 Disability Resources (DR)

WashU supports the right of all enrolled students to an equitable educational opportunity and strives to create an inclusive learning environment. In the event a physical or online environment, learning activity, or learning interaction results in barriers to your inclusion due to a disability, please contact WashU's Disability Resources (DR) to engage in a process for determining and communicating approved accommodations. As soon as possible after receiving an accommodation from DR, send me your WashU Accommodation Letter. Because accommodations are not applied retroactively, initiate your request to DR prior to, or at the beginning of, the academic term to avoid delays in accessing accommodations once classes begin. https://disability.washu.edu/

8.4 Sexual Harassment and Assault

If you have experienced sex discrimination, sexual harassment or violence, we encourage you to speak with someone as soon as possible. If you choose to share this information with me, as an instructor I am required to report your disclosure to my department chair, dean, or the Gender Equity and Title IX Compliance Office. You may also reach out to the Relationship & Sexual Violence Prevention (RSVP) Center to receive confidential support and discuss your options. https://titleix.wustl.edu/students/confidentiality-resources-support/

8.5 Religious Holidays

To ensure that accommodations may be made for students who miss class, assignments, or exams to observe a religious holiday, you must inform me in writing before the end of the third week of class, or as soon as possible if the holiday occurs during the first three weeks of the semester. For more information, please see the university's Religious Holiday Class Absence Policy.

8.6 Emergency Preparedness

Before an emergency affects our class, students can take steps to be prepared by downloading the WashU SAFE App. In addition, each classroom contains a "Quick Guide for Emergencies" near the door.

8.7 Resources for Students

WashU provides a wealth of support services that address academic, personal, and professional needs. To start exploring resources that can help you along the way, please visit: Resources for Students.