University of California San Diego
Department of Physics
Physics 142/242, Winter 2025
Computational Physics II: PDE and Matrix Models (4 units)

Instructor: Javier Duarte, jduarte@ucsd.edu, OH after lecture M and by appt., Zoom: 95142068599
Teaching assistant: Anthony Aportela, aaportel@ucsd.edu, OH during/after lab TuTh, Zoom: 99809876934

Course webpage: https://jduarte.physics.ucsd.edu/phys142.
Canvas: https://canvas.ucsd.edu/courses/62113.
Gradescope: https://www.gradescope.com/courses/935676.

Schedule:

Lecture	MWF	10:00a-10:50a	PCYNH 121, Zoom: 95142068599
Lab	TuTh	2:00p-3:20p	CENTR 222, Zoom: 99809876934

First lecture: Monday, January 6, 2025.
First lab: Tuesday, January 7, 2025.

Textbook: There is no required textbook for this course. At the end of the syllabus, we list a bibliography of textbooks and online resources we will draw from.

Course information: This course is an upper-division undergraduate course and introductory graduate course on computational physics, focusing on solving select physics problems in quantum mechanics using Feynman's path integral approach, combined with Markov chain Monte Carlo methods. The course will explore both the theoretical foundations and computer implementations. Students will develop their own code to solve the physics applications. Basic knowledge of calculus, quantum mechanics, Linux, and programming in some language is expected.

The course structure will consist of weekly lectures on conceptual topics, e.g, quantum mechanics, and lab sections on computational tools, e.g., programming in Python and C/C++. Students will learn how to apply physical reasoning to programming, optimize and debug code, create simulations of physical systems. We will focus primarily on numerically solving quantum mechanics problems using Feynman's path integral approach, combined with Markov chain Monte Carlo methods. Students will also learn how to use modern tools to efficiently solve scientific computing problems interpreted (Python) vs. compiled (C/C++) languages and how to link the two. There will be 2 individual homework assignments and an individual midterm project. There will also be a final project in which students will work in groups.

Student learning outcomes: Upon successful completion of Physics 142/242, students will be able to:

- Design computer programs to numerically solve physics problems, like the harmonic oscillator using the Feynman path integral approach.
- Consider multiple approaches and compare their computational performance, accuracy, and fidelity to physical laws.
- Find and choose the best tool or programming language for the task.
- Visualize the solutions.
- Collaborate with peers to tackle complex, realistic problems.
- Present findings.

Grading policy: Your final course grade will be determined according to the following:

- 30% Homework.
- 15% Quizzes.
- 10% Participation and attendance.
- 20% Midterm project.
- 25% Final project.

Attendance (lectures and labs): In-person lecture and lab attendance is strongly recommended and worth 10% of your grade (see below). Lecture will be mostly conceptual while labs will include hands-on portions, with interactive problem-solving and pair programming throughout. These sessions will be recorded.

Exit tickets: At the end of each class, you will be invited to fill out an exit ticket.

- **Lecture attendance (5%)**:
  - To record your attendance, write your name on the whiteboard or chalkboard either at the beginning or end of lecture.
  - The full 5% will be awarded for attending 80% of the lectures.
- **Lab attendance (5%)**:
  - Similar to lecture, write your name on the whiteboard or chalkboard either at the beginning or end of lab.
  - The full 5% will be awarded for attending 80% of the labs.

Discussion board: We will use Discord: https://discord.gg/WnDCU6xsGk

Homework: Homework assignments will be submitted as code and a report on Gradescope.

There will be a first deadline to submit the homework, which will be graded based on effort and completeness.

There will be a second deadline to submit corrections to the homework, which will be graded based on effort and correctness. You are required to submit corrections for all assignments even if everything is correct. For each problem, you can indicate that you've checked the solution and your solution is equivalent.

Final project: For the final project, students will work in groups of ∼4. The project deliverables are: (1) code provided as a public GitHub repository, (2) a 20-minute presentation by all members of the group, and (3) self and peer evaluations for group contributions.

Academic integrity (including AI tools, e.g. ChatGPT): Please read the College Policies section of the UCSD's Policy on Integrity of Scholarship. These rules will be enforced. Cheating includes, but is not limited to: submitting another person's work as your own, copying from any person/source, and using any unauthorized materials or aids during exams.

We understand that many of you may utilize AI tools such as ChatGPT to assist with your coursework. While these tools can be valuable resources for learning and exploring concepts, it is important to use them responsibly. Copying and pasting AI-generated answers without making an effort to understand the material not only undermines your learning but also violates the principles of academic integrity. Although we can often identify the use of such tools, we recognize that some instances may go undetected. Relying solely on these tools without comprehension will hinder your long-term academic and professional development. We encourage you to use AI assistance to enhance your understanding and support your problem-solving processes, but ensure that all submitted work reflects your own effort and comprehension.

Copying from an online solution, a peer's solution, a Chegg solution, or shared work (on Discord, for example) is considered cheating. Collaboration is encouraged, but by the time you start writing your own solution to turn in, you should not be looking at any other source. You should know the rough outline of the solution well enough that you do not need to reference something line-by-line. Plagiarizing a solution but changing variable names is considered cheating. Soliciting help online via Chegg, Quora, etc. is considered cheating. If suspected, you might be asked to rework similar problems in a Zoom one-on-one meeting with the instructor and/or TA.

Any questions on what constitutes an academic integrity violation should be addressed to the instructor; any violation of academic integrity will result in immediate reporting to the UCSD Office of Academic Integrity, and can result in an automatic "F" for the course at the discretion of the instructor.

Counseling and Psychological Services (CAPS): The mission of CAPS is to promote the personal, social, and emotional growth of students. Many services are available to UCSD students including individual, couples, and family counseling, groups, workshops, and forums, consultations and outreach, psychiatry, and peer education. To make an appointment, call (858) 534-755. For more information, visit https://wellness.ucsd.edu/caps/.

Schedule (Subject to change):

Week 1
Monday 1/6: Lecture 01: Course overview, preview of double-slit experiments
Tuesday 1/7: Lab 01: Python, Jupyter, and DataHub
Wednesday 1/8: Lecture 02: Feynman path integral
Thursday 1/9: Lab 02: NumPy, SciPy, Numba, and Matplotlib
Friday 1/10: Lecture 03: Feynman path integral (continued); **Quiz 1**
Week 2
Monday 1/13: Lecture 04: Free particle
Tuesday 1/14: Lab 03: Introduction to Julia
Wednesday 1/15: Lecture 05: Harmonic oscillator
Thursday 1/16: Lab 04: Assignment 1 tutorial
Friday 1/17: Lecture 06: Schrödinger equation; **Quiz 2**
Week 3
Monday 1/20: **Martin Luther King Jr. Day: No class**
Tuesday 1/21: Lab 05: Git/GitHub tutorial
Wednesday 1/22: Lecture 07: Unitarity and propagator trace
Thursday 1/23: Lab 06: Assignment 1 help
Friday 1/24: Lecture 08: Unitarity and propagator trace (continued); **Assignment 1 Due**
Week 4
Monday 1/27: Lecture 09: Double well potential
Tuesday 1/28: Lab 07: Assignment 1 review
Wednesday 1/29: Lecture 10: Eigenvalue problem; **Assignment 1 Corrections Due**
Thursday 1/30: Lab 08: Assignment 2 tutorial
Friday 1/31: Lecture 11: Statistical mechanics recap
Week 5
Monday 2/3: Lecture 12: Density matrix and path integral
Tuesday 2/4: Lab 09: Eigenvalue problem demo
Wednesday 2/5: Lecture 13: Markov chain Monte Carlo, Metropolis algorithm
Thursday 2/6: Lab 10: Assignment 2 help
Friday 2/7: Lecture 14: MCMC (continued); **Assignment 2 Due**
Week 6
Monday 2/10: Lecture 15: MCMC for 2D Ising model

Tuesday 2/11: Lab 11: Assignment 2 review
Wednesday 2/12: Lecture 16: MCMC for Feynman path integral; **Assignment 2 Corrections Due**
Thursday 2/13: Lab 12: MCMC tutorial
Friday 2/14: Lecture 17: MCMC for Feynman path integral (continued)

Week 7
Monday 2/17: **Presidents' Day: No class**
Tuesday 2/18: TBD
Wednesday 2/19: Lecture 17: MCMC for Feynman path integral (continued)
Thursday 2/20: Lab 13: Midterm help
Friday 2/21: Lecture 18: Particle physics MC and the VEGAS algorithm; **Midterm due**

Week 8
Monday 2/24: Lecture 19: Particle physics MC and the VEGAS algorithm (continued)
Tuesday 2/25: Lab 14: Midterm review
Wednesday 2/26: Lecture 20: Particle physics MC and the VEGAS algorithm (part 3); **Midterm corrections due**
Thursday 2/27: Lab 15: MCMC for Feynman Path Integral Tutorial
Friday 2/28: Lecture 21: Final project overview

Week 9
Monday 3/3: TBD
Tuesday 3/4: Lab 16: Final project help
Wednesday 3/5: TBD
Thursday 3/6: Lab 17: VEGAS Tutorial
Friday 3/7: Lecture 23: Preview of PHYS 141/241

Week 10
Monday 3/10: Lecture 26: Final presentations
Tuesday 3/11: Lab 19: Final presentations
Wednesday 3/12: Lecture 27: Final presentations
Thursday 3/13: Lab 20: Final presentations
Friday 3/14: Lecture 28: Final presentations

Finals Week
Friday 3/21: **Final paper and code due**

Bibliography:

[1] R. P. Feynman, "Space-time approach to nonrelativistic quantum mechanics", Rev. Mod. Phys. **20**, 367 (1948).

[2] G. P. Lepage, "A New Algorithm for Adaptive Multidimensional Integration", J. Comput. Phys. **27**, 192 (1978).

[3] K. Banerjee and S. P. Bhatnagar, "Two Well Oscillator", Phys. Rev. D **18**, 4767 (1978).

[4] R. P. Feynman, A. R. Hibbs, and D. F. Styer, *Quantum Mechanics and Path Integrals* (Dover Publications, 2010).

[5] R. Feynman, R. Leighton, and M. Sands, *The Feynman Lectures on Physics, Vol. III: The New Millennium Edition: Quantum Mechanics* (Basic Books, 2011).

[6] P. Eastman, *Introduction to statistical mechanics*, `https://web.stanford.edu/~peastman/statmech/phasetransitions.html`, 2014.

[7] R. Rodgers and L. Raes, *Monte Carlo simulations of harmonic and anharmonic oscillators in discrete Euclidean time*, DESY Summer Student Programme (2014).

[8] R. H. Landau, M. J. Páez, and C. C. Bordeianu, *Computational Physics: Problem Solving with Python*, 3rd ed. (Wiley, 2015).

[9] F. Krauss and D. Maitre, *Phase transitions*, `https://www.ippp.dur.ac.uk/~krauss/Lectures/NumericalMethods/PhaseTransitions/Lecture/pt1.html`, 2018.

[10] A. Papaefstathiou, "How-to: write a parton-level Monte Carlo particle physics event generator", Eur. Phys. J. Plus **135**, 497 (2020).

[11] S. Mittal, M. J. Westbroek, P. R. King, and D. D. Vvedensky, "Path integral Monte Carlo method for the quantum anharmonic oscillator", Eur. J. Phys. **41**, 055401 (2020).

[12] G. P. Lepage, "Adaptive multidimensional integration: VEGAS enhanced", J. Comput. Phys. **439**, 110386 (2021).

[13] Y. D. Chong, *Computational physics*, `https://phys.libretexts.org/@go/page/34507` (Nanyang Technological University, 2021).