UNIVERSITY OF CALIFORNIA BERKELEY

Numerical Algorithms Applied to Computational Quantum Chemistry
Class Project Guidelines and Suggestions.

Dr. Mayank Agrawal, with Austin Glover (CHEM279)

Final Project Guidelines - Fall 2025

1 GENERAL INSTRUCTIONS.

To finish the semester, we would like you to either further develop the SCF program you wrote for CNDO/2 or the non-iterative extended Huckel method that you developed, or write something entirely new. Here is your chance to go beyond structured homeworks and exercise your creativity in some scientific programming! This should be the equivalent of about 1-2 homework sets in terms of effort, but we know different people have different time constraints, so we'll work with you to try to make sure that you are choosing something achievable.

We do want each of you to produce something that you are proud of, that reflects what you've learned throughout the degree so far, and that takes you further. Also, since this class has had problem sets that build cumulatively on each other, by building further with the project, you can develop a toolset that is distinctively yours.

2 TEAMWORK

For the final project, you will work in a team of 2. Ideally, you should try to make teams in which one student with strong chemistry knowledge and another with programming expertise can leverage each other's strength. For the project, contribution of each team member must be specified at the end of the report in a separate "Contributions" section.

3 SCHEDULE

1. **Nov 17:** Select a topic, in consultation with your teaching assistant and myself – see some ideas below. Please submit a document that is half a page to one full page long,

including your topic title and a brief description of your goals. Please try to be specific with your objective rather than copy pasting one of the ideas below.

2. **Dec 3 - Dec 5:** Preliminary progress report. This will be a short discussion with your teaching assistant and/or me to describe your initial progress and plans. The objective of this exercise is to ensure that your progress is on schedule. Please prepare some slides to update your progress.

3. **Dec 10 - Dec 12:** Present your theory, design, code, tests, and results to the class in a short talk, including questions. We will do presentations during classes. The time slots of 15 mins (10 mins talk + 5 mins Q&A) will be posted and each team can pick a time slot based on their convenience. It is mandatory for both team members to be present during the presentations and Q&A, so pick your time slot after coordination.

4. **Dec 15:** Submission of the final code and report. You should submit a report that has proper instruction on how to compile and run your code, discusses your results in detail and provides conclusions. Also, please attach your slides that you used for your presentation. All of it can be submitted in a github repo format.

4 CHOOSING A PROJECT.

The main idea of the project is for you to do something different. The topic can be something drawn from the class topics, or an extension of them. Ideally it should be something that interests you, and makes you want to go further! A list of potentially suitable topics are given below. They are not all equally difficult, so a rough estimate of how many homework sets each one is equivalent to is given. Regardless of the level of difficulty, we will award bonus points (up to 3 marks) for outstanding achievements in the project! You can either choose one of the topics below, or use the list to get you thinking about your own project suggestion. Ideally we want everyone to have their own project, but it is possible some topics will be popular enough that two people choose the same one.

1. Extend your code to perform vibrational analysis to obtain the vibrational frequencies and the normal modes of a molecule at its optimized geometry: see ICC pp. 555-556. You will evaluate the hessian matrix by finite differences (unless you are *extremely* ambitious), transform to mass-weighted coordinates, and diagonalize. The resulting vibrational frequencies can be compared against experimental infrared results. ($1-2$ homeworks)

2. Create a Visualization Module for Normal Modes. In addition to calculating the normal modes / vibrational frequencies of the molecule, you can also create a tool to visualize them. The code would write a module that generates an output file (e.g., a multi-frame .xyz file) that animates the atomic motion for each mode, which can then be viewed in standard molecular visualization software (like VMD or Avogadro). ($2-3$ homework)

3. Improve the non-linear optimization aspects of your code. First implement the DIIS method to accelerate convergence of your SCF equations. Then improve your geometry optimizer by implementing a quasi-Newton method such as BFGS to search for stable structures, perhaps including a preconditioner also. (1 – 2 homeworks)

4. Implement your own replacement for Armadillo for the matrix operations that we need. The main challenge will be writing your own diagonalizer! There is a good discussion of methods that can yield all the eigenvalues and eigenvectors of a symmetric matrix in Numerical Recipes and elsewhere. In addition you must also build suitable code infrastructure. (1 – 2 homeworks)

5. Extend your Huckel code to include the pairwise atomic corrections needed to obtain high accuracy for hydrocarbons (see the paper by Voityuk posted under reading material), and implement the gradient. Get some results and compare to your CNDO/2 code for accuracy of structures and relative energies. (1 – 2 homeworks)

6. Experiment with extending the CNDO/2 model to include overlap. Make a new version of your code for energies (and possibly gradients) that explicitly accounts for overlap. See how results for relative energies (and structures if you do the gradient) compare. Maybe you can even try adjusting the parameters to improve accuracy for a given class of molecules such as the hydrocarbons (look at the Voityuk paper mentioned in the project above, for instance). (1 – 2 homeworks).

7. Make your CNDO/2 code efficient enough to treat very large molecules by eliminating overhead and optimizing the construction of **h** and **f** so that the computation time is dominated by the linear algebra of diagonalizing **f**. Demonstrate with timings and profiling on molecules that range up to 1000 atoms, hopefully exploiting your expertise in parallel computing from CHEM281! (1 – 2 homeworks)

8. Calculate Molecular Properties from the CNDO/2 Wavefunction. The self-consistent CNDO/2 density matrix contains a wealth of information. This project involves extending your code to calculate molecular properties from this density. A great starting point is the molecular dipole moment. (1–2 homework)

9. Periodic solids via extended Huckel theory. We have done the quantum chemistry of molecules, but the same method can be used to treat solids and surfaces. A challenging project is to build a new version of your code that can treat periodic boundary conditions. There is good existing literature on how to do this which we can point you to. (2 – 3 homeworks).

10. Implement a force field method. This is beginning to get open-ended, but you know that there are *many* force field methods out there that are much more sophisticated and broadly applicable than the Lennard-Jones example we did in homework 1. One example that is quite widely used is the MMFF94 force field, whose energy you could implement. If time and progress permit you could even consider doing its gradient. (2 – 3 homeworks).

11. Feel free to suggest anything else that interests you in scientific programming that connects to our class! We will try to stop you from doing something where you may not get results in time, or help you to rescope appropriately.