# COMPUTATIONAL PHYSICS

SBU, FALL 2018

## Course objectives

The main objective of this course is to extend and deepen theoretical and practical computational skills of graduate students of physics who have taken a related introductory course (e.g. computer applications in Physics). During lectures and hands-on sessions, the students would learn how to employ numerical methods to attack physics problems, in particular when they face a sophisticated problem in their current and future research activity.

## Course theme

The course consists of lectures and hands-on sessions and addresses both methodological aspects and applications. The first part introduces/reviews **basic and selected advanced computational methods, and is** followed by a second part where a number of related physics **simulation experiments** are presented to show how computers can be served as a virtual lab for studying physical phenomena.

## Syllabus

The main topics will be covered is as the following: computational physicist's view of computers, concepts of scientific programming, Fortran 90 quick review, numeric arithmetic and errors, interpolation, differentiation, integration, data fitting, direct & iterative linear algebra, root finding, function minimization, genetic algorithm, structure prediction, ODE & PDE, randomness, Monte Carlo methods, classical force-fields, MD, Langevin Eq., Nonlinear dynamics methods.

## Evaluation policy:

**Exam(s):** 25%, **Exercises+Projects:** 75%, **Bonus Project:** +10%

## Exercises:

You will be asked to answer several exercises during the course. Sometimes you should derive or prove an expression while in some cases you are asked to write up a program to implement an algorithm. If your solution contains several files (program body, numerical results, plots, etc. ) Please hand in only a single tar/zip file. Solutions should be sent by email to ali.sadeqi@gmail.com, with a subject congaing the exercise-number and your-name. Please note that late solutions will not be accepted.

## Projects:

Students must complete two projects during the course. The projects may be presented orally in the class or submitted as a paper-like report (in Farsi or English, with sections such as Introduction, Methods, Results, Discussion and Conclusions).

* Students wishing to get an extra credit (up to 10%) may propose, solve and present a project-like problem based on what they will learn in the course.

## Prerequisites:

**Physics background:** You are expected to have passed successfully an undergraduate-level course on computational physics. You should be familiar with classical mechanics, electrostatics, quantum mechanics, thermodynamics and statistical physics and mathematical physics.

**Programming skills:** For doing exercises and projects, you need to know a programming/scripting language. You may use your favourite high-level language/packege which can be Fortran, c/c++, python, MATLAB, Maple and so on. We will use Fortran 90 in the lectures and hands-on sessions.

**Computer workstation:** For the practical part of the course, you may use either your own laptop/PC or the computer workstations at the Physics Department Computer Lab. As operating system, a distribution of Linux is recommended and will be used in the hands-on sessions. Still, you may use your favourite operating system, but then you have to learn on your own what you would need.

**Softwares:** You need to install and know how to some softwares to do the following tasks. Those mentioned in parentheses are installed on the workstations of the Computer Lab:

- compiler (gfortran, or Intel's non-commercial fortran compiler)
- text editor for writing/editing your programs (vim)
- plotting software (gnuplot)
- file handling tools (awk, sed)

**Recommended books:**

I cannot find a textbook that cover all the topics we will address. We therefore use multiple references, supported by additional materials (handouts, book chapters, etc.). Many of the topics are well described in **Refs. 1,2,3 and 11.** Other references are excellent (undergraduate) textbooks for those who need a review.

1. *Philipp O.J. Scherer, Computational Physics: Simulation of Classical and Quantum Systems*
2. *Rubin H. Landau, Manuel José Páez, Cristian C. Bordeianu, A Survey of Computational Physics*
3. *Morten Hjorth-Jensen, Computational Physics*
4. *Joel Franklin, Computational Methods for Physics*
5. *Franz J. Vesely, Computational Physics - An Introduction*
6. *Tao Pang, An Introduction to Computational Physics*
7. *J.M. Thijssen, Computational Physics*
8. *Nicholas J. Giordano, Computational Physics*
9. *Harvey Gould and Jan Tobochnik, An Introduction to Computer Simulation Methods*
10. *Dieter W. Heermann, Computer Simulation Methods in Theoretical Physics*

*Books in persian:*

11. مهدی نیک عمل و امین الله واعظ و امیر لهراسبی، آشنایی با روشهای شبیه سازی در فیزیک
12. بهمن مهری و رضا نخعی، محاسبات عددی

*Useful links:*

**Computational Physics:**

http://homepage.univie.ac.at/franz.vesely/cp_tut/nol2h/new/index.html

https://eee.uci.edu/04s/47520/p131spring04/index.html

http://stp.clarku.edu/simulations/

http://web.mit.edu/redingtn/www/netadv/Xmontecarl.html

**Fortran:**

http://www.tutorialspoint.com/fortran/index.htm

http://physics.bu.edu/py502/lectures1/f90.pdf

http://www.chem.helsinki.fi/~manninen/fortran2014

http://fortranwiki.org/fortran/show/HomePage

http://www.mie.uth.gr/ekp_yliko/fortran_quick_reference_cheat_crib_sheet.pdf

**Linux:**

http://physics.sharif.ir/~jafari/doc/chap1-linux.pdf

http://www.comptechdoc.org/os/linux/usersguide/linux_ugbasics.html

http://www.linuxdevcenter.com/excerpt/LinuxPG_quickref/linux.pdf

http://tldp.org/HOWTO/Bash-Prog-Intro-HOWTO.html

http://ss64.com/bash/

**Gnuplot:**

http://www.gnuplotting.org/about/

http://www.manpagez.com/info/gnuplot/gnuplot-4.2.3/gnuplot_toc.php#SEC_Contents

http://www.phyast.pitt.edu/~zov1/gnuplot/html/gallery.html

**vim:**

http://rayninfo.co.uk/vimtips.html

**awk:**

http://www.grymoire.com/Unix/Awk.html

**Python:**

https://scipy-lectures.github.io/

https://wiki.python.org/moin/BeginnersGuide

## Tentative Course Outline

Note: Based on the class needs, this outline is subject to small changes.

| Time line | Lecture | Hands-on | Theme |
|---|---|---|---|
| Week 1<br>6/25 | Introduction and demonstrative examples<br>Algorithm complexity | | Programming |
| Week 2 | Scientific programming<br>Computer architecture<br>Variable types in computer memory | | |
| Week 3<br>7/8 | Numeric precision & errors<br>Numeric differentiation<br>High-order & high precision differentiation | | Basic Numeric Methods |
| Week 4 | Numeric integration<br>Iterative root finding<br>Function minimization | f90 programming<br>compiling optimization<br>f90-gnuplot combination for plotting | |
| Week 5<br>7/22 | Multidimensional  minimization<br>Inter-atomic potentials<br>Energy minimization of structures | | Atomistic simulations |
| Week 6 | Geometry relaxation by SD, CG, FIRE<br>Atomic vibrations from Hessian<br>Matrix diagonalizing | Visualizing atomic structures<br>Efficient matrix operations in f90<br>linking to BLAS & LAPACK libraries | |

# Outline

| Time line | Lecture | Hands-on | Theme |
|---|---|---|---|
| Week 7 | Integrating equations of motion<br>Principles of molecular dynamics<br>PBC<br>Simulated annealing & structure prediction | | Classical Molecular Mechanics |
| Week 8<br>8/14 | Langevin Eq.<br>Thermodynamics from MD<br>Nonlinear dynamics and choas | | |
| | **Midterm Exam** | | |
| week 9 | Thermodynamics from MC<br>(Pseudo-) randomness<br>Ising model | | Statistical physics |
| week 10 | Monte Carlo integration: high-dimensional integration<br>Parallel programming & HPC & GPU<br>Ising with OMP | | |
| week 11<br>9/5 | Quantum Monte Carlo  for He and H2<br>Introduction to electronic structure methods<br>Schrodinger Eq.  in 1D with basis set<br>Large system of linear equations ***Ax*=*b*** | | Quantum mechanics |
| Week 12 | Electrostatics: iterative methods vs ***Ax*=*b***<br>Singular value decomposition<br>Principal Component Analysis | | Linear Algebra |

| Time line | Lecture | Hands-on | Theme |
|---|---|---|---|
| week 13 | Probability Distribution Functions | | |
| week 14<br>9/26 | Kernel methods<br>Regression for prediction<br>Principles of artificial neural networks | | Machine Learning |
| week 15 | Project presentations | | |
| | Unsupervised learning<br>Clustering | | |
| week 16 | Project presentations | | |
| | Concluding remarks | | |