Introduction to Computational Physics SS2019

Lecturers: Rainer Spurzem & Ralf Klessen
Tutors: Branislav Avramov, Raul Dominguez, Robin Tress, Yun Kiyun

Exercise 1 from April 17, 2019
Return by noon of May 3, 2019

1 General Comments

Notice: The first part of any tutorial sheet should normally be done during the tutorial hours with help and advice from your tutors. You can also go independently to any CIP pool and work out the tutorial sheets by yourself (but we recommend to work in a group of 2-3 students). Or you may use your own computer. Please note that our tutors can only help with the main programming languages. You are free to use any language, but you may be on your own. Please check with your tutors.

- Please hand in the computer programs, graphs or tabular values (if no graphs are required) by email to the tutor(s).
- This is best done by making a PDF document containing all these items using LaTeX. You can use the template provided by the tutors.
- One document per group is sufficient.
- Please write the names of the group members in the document.
- Tutorials are grouped together on Friday and Monday the following week.

2 Basic Exercises

- Get acquainted with the Unix/Linux operating system. For example the Unix commands ls, cd, ps, less, etc. Also try out a text editor which you can use for programming, e.g. emacs, vi, joe. Start the plotting program gnuplot and try to plot a few simple functions as in the lecture notes.
- Practice writing a simple computer program in a language of your choice (e.g. C, C++, Fortran, python). Compile the program, as in the lecture, from source code via object code to executable.
- The main course webpage is `http://wwwstaff.ari.uni-heidelberg.de/mitarbeiter/spurzem/lehre/SS19/compphys/compphys.php.en`. New tutorial sheets, messages and other up-to-date material you will find in the "elearning2" system of Heidelberg University: `https://elearning2.uni-heidelberg.de/course/view.php?id=20724`.

- Use the gnuplot tutorial (Section “More Info” on the course webpage) to do some exercises on how to plot data - you can use the example datafile “myfile.txt”).

- Also, try to get acquainted with python and or Mathematica.

3 Numerical Integration

In this exercise we will numerically evaluate the integral

$$y_n = y_n(a) = \int_0^1 \left(\frac{x^n}{x+a}\right) dx \; = \; \frac{1}{n} \; - \; a \, y_{n-1} \; .$$

(a) Plot the integrand for $a = 5$ and $n = 1, 5, 10, 20, 30, 50$ in the domain $0 \leq x \leq 1$.

(b) Write a compute program that reads the value of a, the starting values n_0 and y_0, and the final value n_1, and performs the iteration from n_0 to n_1 (either backward or forward, depending on whether $n_1 < n_0$ or $n_1 > n_0$).

(c) Experiment how this series behaves for iteration from $n_0 = 0$ to $n_1 = 30$ for $y_0 = \ln[(1 + a)/a]$ with $a = 5$. Also try starting with $n_0 = 50$ and iterate back to $n_1 = 30$ for any starting value y_0.

4 Numerical Simulation of the 2-Body Problem

In this second part of the first tutorial we will start integrating ordinary differential equations.

- Write a computer program that computes the relative motion of two point-like bodies under their mutual gravitational influence. Use a step-by-step Euler integration procedure. Set $G = 1$, $M_1 = M_2 = 1$.

- For which velocity v_0 can the two bodies rotate around each other in a circular fashion with a separation of 1?

- Now perform the numerical integration. Use, for the moment, a constant time step of $\Delta t = 0.01$. What happens with the numerical model if you choose $v_0/\sqrt{2}$ as the initial velocity? Make a plot of the orbits.

- Compute the eccentricity from the Runge-Lenz vector.

- What happens if you choose an initial velocity larger than $\sqrt{2}v_0$?

- Simulate another case, like e.g. the $v_0/3$ initial velocity case. Experiment with decreasing the time step (and simultaneously increasing the number of time steps) and see how the results change.

5 Error Analysis of Euler Scheme (HOMEWORK)

Continue with the analysis of the gravitational 2-body problem discussed above.

(a) Choose 3 different eccentricities by varying the initial velocity and different time steps (the latter spanning orders of magnitude!). Integrate the 2-body problem for one orbit. Plot, in a double-logarithmic fashion, the error in the energy at the end of this orbit as a function of Δt. Discuss the result, is it consistent with what one should expect?

(10 points)

(b) Do the same as above, but now using the leapfrog integrator scheme. How does the result change?

(10 points)

NOTE, for this and all following homework exercises, which require writing a computer program, it is necessary to submit all parts of the program written by you, together with the results. It must be comprehensible for your tutor how your program generates your results.