Exercise 1: Leapfrog-integrator
(see Lecture 02: ODEs)

date: 23.10.2007 date for handing over: 06.11.2007

Problem 1: Precision of integrators
Show that the numerical integration of a ODE of type $df(t)/dt = G(f,t)$ through a leapfrog-integrator corresponds to a precision of 2^{nd} order.
Hint: It should be demostrated that the global error of the obtained functional values $f(t)$ is of order $\mathcal{O}((\Delta t)^2)$ where Δt is the timestep.

(4 points)

Problem 2: Stars and planets
Write a program for calculation of stellar or planetary orbits, i.e. numerically solving the gravitational 2-body problem:

a) The program should be written in such a way that the bodies can have arbitrary coordinates (the centre-of-mass should NOT be in the coordinate origin). Introduce a parameter N to denote the number of particles - in our case it should be 2 (N=2), but it allows a generalization of the program for N bodies.

To solve the ODEs, use the leapfrog-integrator that has been discussed in the lecture. Choose appropriate internal units for your program! Remember the "jumpstart" and synchronization of the initial and final conditions!

(10 points)

b) Use one of the following sets of initial conditions (ICs):

Quantity	Sun - Earth	51 Pegasi with a planet	Binary star
m_1 $[M_\odot]$	1.0	1.04	1.0
m_2 $[M_\odot]$	3.005×10^{-6}	4.49×10^{-4}	1.0
r_0 $[AU]$	1.0	0.052	1.0
v_0 [km/s]	29.8	133.7	30.0

$m1$ and $m2$ are the masses of the bodies, r_0 is the distance between them and v_0 - the relative velocity of both of them. Keep in mind that the units should be converted to your internal units!

Choose an appropriate timestep and create a plot (e.g. with Gnuplot) that presents the relative motion of the bodies in the trajectory plane and with the centre-of-mass at rest. Don't forget to label the axes and specify the units used! (To simplify the task, you may consider the motion of the bodies in the XY-plane - then you have to describe only Y vs. X coordinates.)

(2 points)

c) The leapfrog integrator is symmetric. Test this in your program launching it "backwards": integrate 10 timesteps forward and use the obtained positions and the NEGATIVE velocities as new initial conditions. After the new integration the primeval state should be reconstructed. Is this true? Are there any deviations? If yes, how large are they and what is their origin?

Create a graphics that demonstrates that the "forward" and "backward" integrated trajectory points coincide.

(2 points)

d) Vary the initial conditions (ICs) and the timestep: make - in the example Sun - Earth, - the trajectory of the Earth more eccentric and fit the timestep so that the Earth remains gravitationally bound after minimum 3 circulations!

What initial velocity of the Earth is required in order to achieve such an eccentricity that the planet touches the Sun's surface during its orbital motion? ($r < (R_E + R_\odot) = 4.695 \times 10^{-3}\ AU$) (An accuracy of 0.1 km/s is enough.) How large should the timestep be?

(2 points)