PY 421 - Introduction to Computational Physics

Homework # 5. March 8, 2013.

Due at discussion time (3PM) on Friday March 22.

The assignment:

For this assignment you must write a program in C or Fortran which calculates the trajectory of a particle of mass m which moves in the $x-y$ plane subject to the force deriving from a potential $V(x,y)$ which is the sum of two repulsive Gaussian potentials:

$$V(x,y) = a_1 e^{-[(x-x_1)^2+(y-y_1)^2]/w^2} + a_2 e^{-[(x-x_2)^2+(y-y_2)^2]/w^2} \tag{1}$$

At time $t = 0$ the particle is located on the y-axis with coordinates $x = 0$ and $y = y_0$, where y_0 is a parameter which should be input by the user. The initial velocity of the particle has components $v_x = v_{x,0}$ and $v_y = 0$. The program should use the one-step (or Euler) method to calculate the values of x and y at subsequent times with time increments dt, until the particle reaches a final coordinate $x_{\max}$.

At the beginning of execution the program should ask the user to input the value of y_0. Then the program should print out the initial value of the total energy of the particle

$$E = \frac{m(v_x^2 + v_y^2)}{2} + V(x,y), \tag{2}$$

integrate the equations of motion, and print out the values of the particle's coordinates when x reaches $x_{\max}$ and the corresponding total energy. The last time step should be the one where x has just reached or passed $x_{\max}$. You do not need to interpolate the final values of the coordinates back to $x_{\max}$. In the course of the evolution the program should write to a file the values of x in increments of (approximately) Δ_x and the corresponding values of y. These data should be written in a form that makes it possible to use the file to plot the trajectory with the gnuplot program.

In the program, the variables introduced above must be given the following values

$$\begin{aligned} m &= 1 \\ a_1 &= 0.25 \\ a_2 &= 0.5 \\ x_1 &= 3 \\ y_1 &= 1 \\ x_2 &= 6 \\ y_2 &= -1 \\ w &= 2 \\ v_{x,0} &= 2 \\ dt &= 0.001 \\ x_{\max} &= 10 \\ \Delta_x &= 0.05 \end{aligned} \tag{3}$$

Returning the assignment:

Once you have completed your program, you should copy it using one of the CAS 327 workstations onto the file

~rebbi/courseware/asgn/asgn5.xxyyyy.

Hints:

1) You should take advantage of the program shoot.c or its Fortran version to complete this assignment. Most of the structure of the program you have to write is already present in shoot. Of course you will have to introduce some new variables and remove variables which are no longer used (leaving them in the code does not hurt, but it would be bad style.) Also, you will have to replace the expression of the force in shoot with the one appropriate to this problem. (Note that you do not have to build your program on shoot. If you prefer you may start from scratch. But taking advantage of the code in shoot will almost certainly save a lot of time.)

2) You can take advantage of the possibility of introducing auxiliary variables. For example you might introduce a statement

v1=a1*exp(-((x-x1)*(x-x1)+(y-y1)*(y-y1))/(w*w))

and then calculate the x component of the force due to this term in the potential simply by

fx=2*(x-x1)*v1/(w*w);

By cutting and pasting and changing x to y or 1 to 2 as needed, then you can rapidly take care of all the other required statements.

3) Compile your code often, after you inserted some new statements, to catch errors right away. You can compile without linking by cc -c yourcodename.c or gfortran -c yourcodename.f90

Grading:

A program which compiles and executes correctly will be given a score of 100, with points deducted for mistakes and poor program presentation, according to the severity of the error.