PROBLEM SET 5 (SUPPLEMENT)

Section 1: Damped driven pendulum

For the equation of motion, we can tranform this into three linear first order ODEs:

$$\begin{aligned} \dot{y_1} &= 1 && (1) \\ \dot{y_2} &= y_3 && (2) \\ \dot{y_3} &= -2\gamma y_3 - \omega_0^2(1 + h\cos(2(\omega_0+\epsilon)y_1))\sin y_2 && (3) \end{aligned}$$

We use one of the simplest integration method, the Euler's method.

$$\begin{aligned} y_3^{n+1} &= y_3^n - (2\gamma y_3^n + \omega_o^2(1 + h\cos 2(\omega_0+\epsilon)y_1^n)\sin y_2^n)\Delta t && (4) \\ y_2^{n+1} &= y_2^n + y_3^{n+1}\Delta t && (5) \\ y_1^{n+1} &= y_1^n + \Delta t && (6) \end{aligned}$$

where y_1^n stands for y_1 at nth iteration. This is implemented in drvpend.c and drvpend.m, copy this from the class homepage into your working directory. To compile it, type

```
% cc drvpend.c -O -lm -o drvpend
```

and start Matlab. Note that drvpend.m is a Matlab script which invokes the compiled program drvpend.

Type "help drvpend" to see how to use it to compute the trajectory with given parameters. For example,

```
>> gamma = 0.0;
>> omega0 = 1.0;
>> epsilon = 0.0;
>> h = 0.0;
>> ic = [0.01 0.0];
>> finaltime = 1000;
>> dt = 0.001*2*pi;
>> [t,theta,thetadot] = drvpend(gamma, omega0, epsilon, h, ic, finaltime, dt);
```

so $\gamma = 0$, $\omega_o = \omega = 1.0$, $h = 0.0$ with the initial conditions $\theta_o = 0.01$, $\dot{\theta}_o = 0.0$ (note that the elements of `ic` are ordered `[theta thetadot]`), the duration of integration is from $t = 0$ to $t = 1000$, and $dt = 0.001 \times 2 \times \pi$. The output are 3 vectors: t, theta and thetadot.

Notice that somewhere above, dt is specified to be $0.001 \times 2\pi$, so that $t = 0, 2\pi, 4\pi$...etc are conveniently located at $t(1) = 0$, $t(1001) = 2\pi$, $t(2001) = 4\pi$...etc. This will make it easy to generate Poincare section with $\omega_o = 1.0, T_o = 2\pi$. If you should change the natural frequency of the system ω_o, you may want to adjust dt accordingly. For example, to plot a Poincare section,

```
>> thetamax = max(theta);
>> thetamin = min(theta);
>> thetadotmax = max(thetadot);
>> thetadotmin = min(thetadot);
>> axis([thetamin thetamax thetadotmin thetadotmax]);
>> hold on;
>> for i=1:1000:length(t),
      plot(theta(i), thetadot(i), '.');
   end
>> hold off
```

Also, most of the time, you are interested in the steady state behavior rather than the transient. You may have to chop off the beginning part of your data set. Please refer to Matlab documentation for the details.

Please let us know if you experience any problems.