CS4210 Assignment 6 Due: 11/22/14 (Sat) at 6pm

You must work either on your own or with one partner. You may discuss background issues and general solution strategies with others, but the solutions you submit must be the work of just you (and your partner). If you work with a partner, you and your partner must first register as a group in CMS and then submit your work as a group. Points may be deducted for poor style and reckless inefficiency.

Topics: Stiff IVPs, Lorenz system

1 2nd Order Adams-Moulton (AM2)

Consider the IVP

$$y'(t) \ = \ f(t, y(t)) \qquad y(t_0) = y_0$$

We are assuming that $y(t)$ is a scalar.

Let $h > 0$ be given and define $t_k = t_0 + kh$, $k = 0{:}n$. Our goal is to approximate $y(t_k)$ with y_k defined by the AM2 method:

$$y_{k+1} \ = \ y_k \ + \ \frac{h}{2}\,(\,f(t_{k+1}, y_{k+1}) + f(y_k, t_k))$$

Note that y_{k+1} is defined implicitly. We need a nonlinear equation solver strategy in order to compute y_{k+1}.

If $G(z)$ is defined by

$$G(z) = z + \frac{h}{2}f(t_{k+1}, z) + c \qquad c = y_k + \frac{h}{2}f(t_k, y_k)$$

then y_{k+1} is a *fixed point* for G meaning that

$$y_{k+1} = G(y_{k+1})$$

If $|G'(z)| \leq \tau < 1$ then it can be shown that the iteration

z = initial guess
Repeat:
  $z = G(z)$
end

converges to y_{k+1}. We also see that y_{k+1} is a zero of the function

$$F(z) \ = \ z - \frac{h}{2}f(t_{k+1}, z) - c \qquad c = y_k + \frac{h}{2}f(t_k, y_k)$$

This sets us up for a Newton approach:

z = initial guess
Repeat:
  $z = z - F(z)/F'(z)$
end

Note that

$$F'(z) \ = \ \frac{d}{dz}(z - \frac{h}{2}f(t_{k+1}, z) - c) \ = \ 1 - \frac{h}{2}\frac{\partial f}{\partial z}(t_{k+1}, z)$$

Using these ideas, implement the following functions:

```
  function yOut = AM2FixPt(f,t0,y0,n,h)
% f is a handle to a function of the form yp = f(t,y) where t and y are scalars.
% yOut is a column (n+1)-vector with the property that
% yOut(k) is an approximate solution to the IVP
%
%                 y'(t) = f(t,y)     y(t0) = y0
%
% at t = t0+(k-1)h, k=1:n+1. The 2nd order Adams-Moulton method (AM2)
% with step length h is used to generate the approximate solutions. In
% particular, z = y_{k+1} approximately satisfies
%
%                z = y_k  + (h/2)*(f(t_{k+1},z) + f(t_{k},y_{k})
%
% where t_{k}=t0+kh and  y_{k} \approx y(t_[k}).
% Fixed point iteration is used to do this.
```

```
  function yOut = AM2Newton(f,t0,y0,n,h)
% f is a handle to a function of the form [yp,dfdt] = f(t,y) where t and y are scalars.
% yp is the rate of change of f(t,y(t)) with respect to t and dfdy is the
% rate of change of f with respect to y.
%
% yOut is a column (n+1)-vector with the property that
% yOut(k) is an approximate solution to the IVP
%
%                 y'(t) = f(t,y)     y(t0) = y0
%
% at t = t0+(k-1)h, k=1:n+1. The 2nd order Adams-Moulton method (AM2)
% with step length h is used to generate the approximate solutions. In
% particular, z = y_{k+1} approximately satisfies
%
%                z = y_k  + (h/2)*(f(t_{k+1},z) + f(t_{k},y_{k})
%
% where t_{k}=t0+kh and  y_{k} \approx y(t_[k}).
% A Newton iteration is used to do this.
```

Each implementation should oversee the iteration with a `while loop`. In each case, you must formulate a reasonable termination criteria (it will involve h) and a reasonable starting value. Include comments that justify your choices. A test script `P1` is provided on the website. Submit `AM2FixPt` and `AM2Newton`

2 Lorenz

Download the lecture function `ShowLorenz`. Study it and play with it. Suppose

$$\text{Attractor 1} \;=\; u \;=\; \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \qquad \text{Attractor 2} \;=\; v \;=\; \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}$$

We say that trajectory point $y \in \mathbb{R}^3$ is in Zone 1 if

$$\sqrt{(y_1-u_1)^2+(y_2-u_2)^2+(y_3-u_3)^2} \;\le\; \sqrt{(y_1-v_1)^2+(y_2-v_2)^2+(y_3-v_3)^2}.$$

This means that the trajectory point y is closer to Attractor 1 than Attractor 2. We say that a trajectory point y is in Zone 2 if $(y-u)^T(v-u) \le 0$ (this means angle that yuv is greater than 90 degrees). Complete the following function so that it performs as specified:

```
  function [Enter1,Exit1,Enter2,Exit2,yFinal] = LorenzZones(y0,tFinal)
% Integrates the IVP
%
%      [ y1'(t) ]      [  beta     0        y2(t) ]  [ y1(t) ]
%      [ y2'(t) ]  =   [   0     -sigma    sigma  ]  [ y2(t) ]    y(0) = y0
%      [ y3'(t) ]      [ -y2(t)  rho         -1   ]  [ y3(t) ]
%
% from t = 0 to t= tFinal. The parameter values are...

  sigma = 10;
  rho = 28;
  beta = 8/3;

% Think of y(t) as the location of a comet at time t.
% Uses ode45 with Relerr = .000001.
%
% Enter1 a column vector (possibly empty) indicating when the comet enters Zone 1
% Exit1  a column vector (possibly empty) indicating when the comet leaves Zone 1
% Enter2 a column vector (possibly empty) indicating when the comet enters Zone 2
% Exit2  a column vector (possibly empty) indicating when the comet leaves Zone 2
%
% yFinal is the location of the comet at t = tFinal.
```

If a comet enters Zone 1, can Zone 2 events be used to determine how many times the comet "orbits Attractor 1" before it leaves Zone 1? Formulate a definition of "orbits Attractor 1" and write a demo script `P2` that checks it out. Make effective use of `LorenzZones`. Your script should run a single simulation with a starting value that is NOT in Zone 1. It should print a table indicating how many orbits are completed each time the comet enters Zone 1. Submit `LorenzZones` and `P2` to CMS.

3 Global Warming

Write a function `P3()` that solves the IVP described in Problem 7.21 of the NCM online text. (Pages 45–46 of Chapter 7.). Your script should produce two plots just like Figure 7.10. One produced using `ode23` and one produced using `ode23s` (or any other stiff solver). In both cases, report the number of required f-evaluations in the title of each plot. Submit `P3` to CMS.