Math 444/544 **Fall 2010**
Homework 6, due Friday 22 October.

1. (20 points)

   444 Students: I have started a template for your final project report at `http://en.wikiversity.org/wiki/User:Mjmohio/Final_Report_Template`. Edit the template to include:

   - Any other sections we should have.
   - Further instructions or snippets of text as guidance.
   - Other things you think would be useful for you or the 544 students.

   Print the template and highlight your changes. (Hurry, before the other students do all the easy ones.)

   544 Students: Your final project will be to add material to the wikiversity `Topic:Numerical_analysis`. You can add exercises, quizzes, further examples, links, etc. Try not to duplicate material already available on wikipedia or other web sources. Decide what topic you want to work on for your final project. Write a project proposal on your user page, and print that portion. Include a list of specific things you plan to do.

2. (40 points)

   (a) Complete the following MATLAB code:

```
function [T Y] = backwardeuler(f,tspan,y0,n,tol)
% Solves dy/dt = f(t,y) with initial condition y(a) = y0
% on the interval [a,b] using n steps of the backward Euler method.
% Each step is backward Euler: Y(i+1)=Y(i)+h*f(t(i+1),Y(i+1)).
% To solve for Y(i+1) uses z0=Y(i) and z1=Y(i)+h*f(t(i),Y(i))
% and then runs the secant method on g(z)=z-Y(i)-h*f(t(i+1),z).
% Inputs: f -- the (scalar) function, as an inline
%         tspan -- a vector [a,b] with the start and end times
%         y0 -- the starting value, y(a)=y0.
%         b -- the right end (ending point) of the interval
%         n -- the number of steps to use; defines the step size.
%         tol -- the secant step runs until |g(z)|<tol.
% Outputs: T -- a n+1 column vector containing the times
%          Y -- a n+1 column vector with the y values
```

   (b) Verify that your code works by running it on exercise 30.1 in the 344 notes. and comparing it to `myeuler`.

   (c) Run `myeuler` from the 344 notes on a stiff equation that clearly demonstrates instability.

   (d) Run your `backwardeuler` on the same example to demonstrate its stability.

3. (40 points) Do this problem as a Good Problem.

   (a) Derive the Adams-Bashforth two-step method.

   (b) For a particularly difficult ODE (with a vertical asymptote), you determine that the step size should decrease by a factor of 2 each time, so that $t_{i+1} - t_i = 2^{-i}$. Derive a two-step method similar to Adams-Bashforth for this situation.