Math 128a, Homework 12

due December 4.

1. Exercise 7.13. Note that you don't need to find the precise order of the method.

2. Exercise 7.17.

3. Consider the equation $y' = -20y$, $x \in [0,1]$, $y(0) = 1$. Show that Euler's method with step h leads to the approximations $y_i = (1-20h)^i$ whereas with the implicit Euler's method we get $y_i = \frac{1}{(1+20h)^i}$. Compute the approximations as $x = 1$ for both methods and the respective relative errors with $h = 0.1$ and $h = 0.01$. *Hint:* the exact solution is $y(x) = e^{-20x}$.

4. In this problem you will compare a number of methods for solving initial-value problems.

(a) Using `help`, read the descriptions of Matlab functions `ode45`, `ode113`, `ode23s`, `odefile`, and `odeset`. Some hints on the use of these functions. Suppose `func.m` contains the lines

```
function yprime = func(x,y,flag,const)
yprime = const*x;
```

Then the sequence of commands

```
reltol = 1e-4; abstol = 1e-4;
odeoptions = odeset('RelTol', reltol, 'AbsTol', abstol);
xinit = 0; xfinal = 1; yinit = 1; const = 10;
[X,Y,S] = ode45('func',[xinit,xfinal],yinit,odeoptions,const)
plot(X,Y)
```

will solve the ODE $y' = f(x,y) = \text{const} \cdot y = 10y$, $y(xinit) = yinit$ on the interval $[xinit, xfinal]$ and plot the answer. On return, X will be the array of values of x at which the solution has been evaluated, and Y will be the array of values of $y(x)$. `ode45` will attempt to make the error $|Y(i) - \text{true}Y(i)|$ be less than $\max(reltol * |Y(i)|, abstol)$. S will contain some performance information:

$S(1)$ = number of successful steps taken by ODE solver
$S(2)$ = number of failed attempts (because error estimate was too large)
$S(3)$ = number of times `func` called
$S(3)$ is the best measure of cost.

(b) Write a Matlab function to implement the Euler-modified Euler pair (that is, use the Euler method for solving the equation and the modified Euler method for estimating the error and choosing the step size). If you have time, also write a Matlab function to implement the extrapolation method, based on the midpoint or the modified midpoint method (the second one is better). Make sure that in both methods, the step size is not allowed to vary by more than a factor of 5 up or down. For the parameters that the functions take, you should use the Matlab functions as a model.

(c) Test the three functions from part (a) and (one or) two functions that you wrote on the following equations. Take the initial step size to be 0.1, and the desired error tolerance, both relative and absolute, to be 10^{-4}.

- $y' = 10y$, $y(0) = 1$, on $[0,1]$.
- $y' = -100y$, $y(0) = 1$, on $[0,1]$.
- $y' = 1 + (x-y)^2$, $y(2) = 1$, on $[2,3]$.

- $y' = y(4-y) + 2\sin^5(t)(2-y)$, $y(0) = 2$, on $[0, 2\pi]$.

For all of these, make a printout of the mesh points x_i, the time steps h_i, and the approximations y_i. Make sure to clearly list, for each method, the value of your approximation y at the other endpoint of the interval, and some performance informations $(S(1), S(3)$ or their analogs).