Math 561 Homework 5 due Thursday, Dec 11, 2014
Fritz Keinert

For each problem that uses Matlab or some other tool, you should hand in a printout of the relevant script or function file(s), or a transcript of your interactive session, plus whatever outputs or plots are requested. Put the problems in the proper order, and label all printouts clearly. The final output should have full accuracy (`format long`); intermediate results can be shorter, if you want.

1. For the implicit multistep scheme

$$y_{n+2} = 3y_{n+1} - 2y_n + \frac{h}{2}\left[f_{n+2} - f_{n+1} - 2f_n\right],$$

determine the leading term of the truncation error, and find out whether this method is stable for small $h\lambda$, or not.

Remark: It is not clear whether you should be expanding around t_{n+1}, or rather t_n or t_{n+2}. I tried all three, and got the same leading term (except that the y-derivative is evaluated at different points). However, the following other terms are different. (10)

Solution: The Taylor series expansion of the formula on the right around the point t_{n+1} starts with

$$y_{n+1} + hy'_{n+1} + \frac{h^2}{2}y''_{n+1} + \frac{h^3}{12}y'''_{n+1}.$$

The first three terms are correct, the fourth one is supposed to have a 6 in the denominator, not a 12. The truncation error is the local error divided by h, so

$$T \approx \frac{h^2}{12}y'''_{n+1}.$$

The reduced characteristic equation is

$$r^2 = 3r - 2$$

with solutions $r = 1$ and $r = 2$. This method is unstable.

2. In file `co2.dat` on the web site, you will find 216 numbers which represent monthly measurements of the CO_2 concentration near Mauna Loa observatory in Hawaii, from January 1958 to December 1975. We want to look for periodicities.

Plot the data (you don't need to hand that in, just look at it). You should see a lot of oscillation, and a general upward trend.

First, we have to remove the upward trend. We did not cover this in class, so I will tell you what the Matlab commands are:

```
load co2.dat
t = (1:216)';
p = polyfit(t,co2,1)
co2 = co2 - polyval(p,t);
```

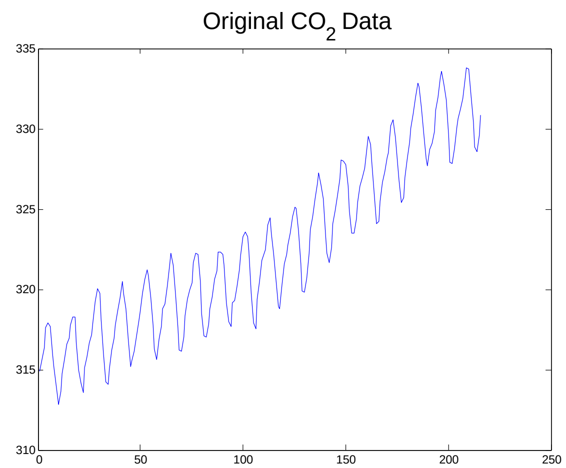
We are fitting a polynomial of degree 1 (straight line) to the data, and subtracting it.

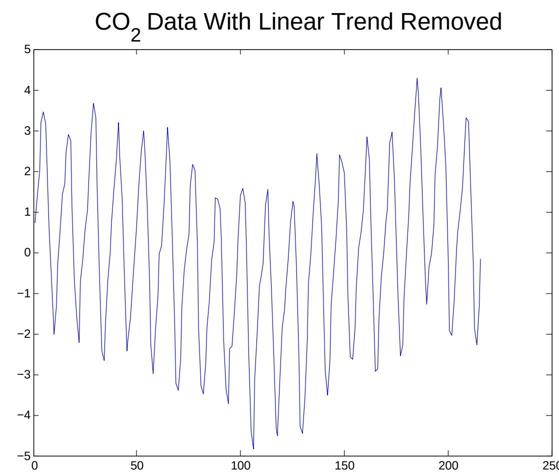
(a) Plot the data with linear trend removed. You don't need to print the numbers, just plot them.

Calculate the FFT and plot the absolute value of the first half. (The second half is just a mirror image of the first). You should see 3 distinct peaks. Remember that a peak at 0 does not count. (Actually, there is no peak at 0 in our case; since we subtracted a linear trend, the average signal is now 0).

(b) Figure out where the peaks occur (get an estimate from the plot and look at the FFT values themselves to localize them exactly). Remember that Fourier coefficients are numbered starting with 0, while Matlab vectors start with 1. Figure out what periodicity (in months) these peaks correspond to.

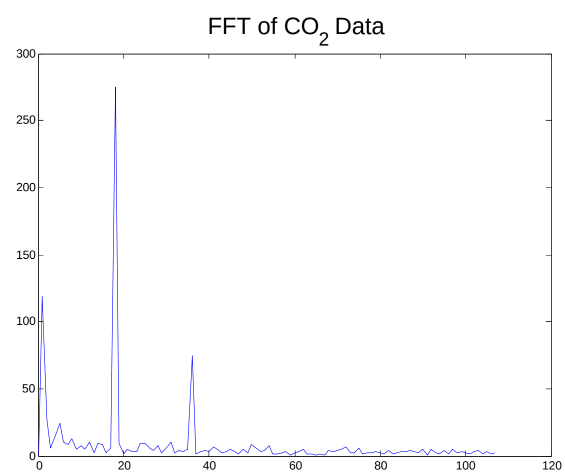
Solution: **(a)** Graphs of the original data (the first one was not required):





Take the DFT, and graph it.

```
co2hat = fft(co2);
plot(0:107,abs(co2hat(1:108)))
title('FFT of CO_2 Data','FontSize',20)
```



There seem to be peaks near 0, 20 and 40.

(b) Look at the absolute value of the FFT to locate the peaks:

```
 0   9.9476e-12
 1        118.4
 2       26.999
 3       5.6336
 ...
17       5.8438
18       274.99
19       9.1348
...
35       4.5599
```

```
36        74.699
37        1.5683
...
```

The precise locations are 1, 18 and 36.

The Fourier coefficient with index j corresponds to a sin or cos with j periods over the span of the data, so the periods are $216/1 = 216$ months, $216/18 = 12$ months and $216/36 = 6$ months.

According to the book where I got the data, the actual trend is exponential, not linear. If you do that, the first peak occurs at index 2, corresponding to a period of 108 months. That is supposedly an actual cycle.

The 12 month cycle obviously corresponds to yearly seasonal variations. The 6 month cycle is also related to the yearly cycle.

3. These are the same two equations as on the last homework, but now we apply a multistep method to them.

(a) Solve the ODE

$$\begin{aligned} y'(t) &= \frac{y(t)}{t} - \frac{t^2}{y^2(t)}, \\ y(1) &= 1 \end{aligned}$$

on the interval $[1, 1.7]$, using 4th order Adams-Bashforth-Moulton (PECE) with Runge-Kutta startup, stepsize $h = 0.05$.

Print out the results for this method, along with the exact values, and the difference between the two. Plot the ABM solution and the true solution (in the same picture).

(b) Solve the ODE

$$\begin{aligned} t^2 y''(t) + t y'(t) + (t^2 - 1) y(t) &= 0 \\ y(1) &= 1 \\ y'(1) &= 0 \end{aligned}$$

on the interval $[1, 2]$, using 4th order Adams-Bashforth-Moulton (PECE) with Runge-Kutta startup, stepsize $h = 0.05$.

Print out the results for y and y', along with the exact values and the difference between the two.

You don't need to turn in plots, they are very boring-looking. (10)

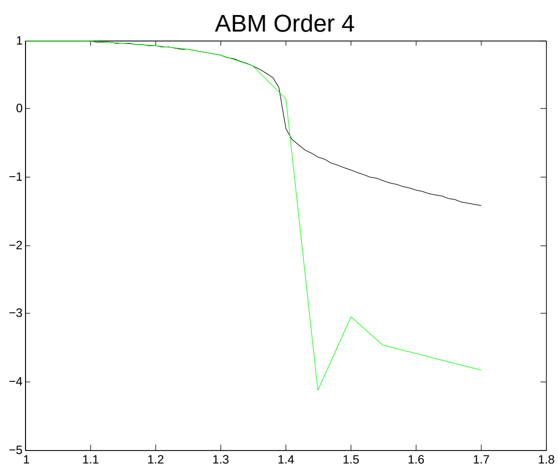
Solution: The following code works for both parts.

```
clear all

% part (a)
f = @(t,y) y/t - (t./y).^2;
y0 = 1;
true = @(t) t .* sign(1 - 3*log(t)) .* abs(1 - 3 * log(t)).^(1/3);
h = 0.05;
t = 1:h:1.7;
tt = 1:0.01:1.7;
y_true = true(t);
yy_true = true(tt);
```

```
% % part (b)
% f = @(t,y) [y(2); - (t.*y(2)+ (t.^2-1)*y(1))./t.^2];
% y0 = [1;0];
% d = besselj(1,1)*(bessely(0,1)-bessely(2,1)) ...
%     - bessely(1,1)*(besselj(0,1)-besselj(2,1));
% c1 = (bessely(0,1)-bessely(2,1))/d; %    1.36575994534763
% c2 = -(besselj(0,1)-besselj(2,1))/d; %   -0.51073987162512
% true = @(t) [c1*besselj(1,t) + c2*bessely(1,t);
%        c1*(besselj(0,t)-besselj(2,t))/2 ...
%             + c2*(bessely(0,t)-bessely(2,t))/2];
% h = 0.05;
% t = 1:h:2;
% tt = 1:0.01:2;
% y_true = true(t);
% yy_true = true(tt);

y_abm = zeros(length(y0),length(t));
f_abm = y_abm;
y_abm(:,1) = y0;
f_abm(:,1) = f(t(1),y_abm(:,1));

% Runge-Kutta startup
for i = 2:4
       k1 = f(t(i-1),y_abm(:,i-1));
       k2 = f(t(i-1)+0.5*h,y_abm(:,i-1)+0.5*h*k1);
       k3 = f(t(i-1)+0.5*h,y_abm(:,i-1)+0.5*h*k2);
       k4 = f(t(i-1)+h,y_abm(:,i-1)+h*k3);
       y_abm(:,i) = y_abm(:,i-1) + h*(k1 + 2*k2 + 2*k3 + k4)/6;
       f_abm(:,i) = f(t(i),y_abm(:,i));
end

% Predictor-Corrector
for i = 5:length(t)
    y_abm(:,i) = y_abm(:,i-1) + h/24*(55*f_abm(:,i-1)...
                 -59*f_abm(:,i-2)+37*f_abm(:,i-3)-9*f_abm(:,i-4));
    f_abm(:,i) = f(t(i),y_abm(:,i));
    y_abm(:,i) = y_abm(:,i-1) + h/24*(9*f_abm(:,i)...
                 +19*f_abm(:,i-1)-5*f_abm(:,i-2)+f_abm(:,i-3));
    f_abm(:,i) = f(t(i),y_abm(:,i));
end

% print answers

figure(1);
plot(tt,yy_true(1,:),'k',t,y_abm(1,:),'g')
title('ABM Order 4','FontSize',20)

disp(' '); disp('Solution Values'); disp(' ');
disp('     t       ABM            exact          error');
```

```
disp('    ================================================');
for i = 1:length(t)
    disp(sprintf('    %4.2f%15.10f%15.10f%15.10f',t(i),y_abm(1,i),...
                 y_true(1,i),y_abm(1,i)-y_true(1,i)));
end

if (length(y0) > 1)
    figure(2);
    plot(tt,yy_true(2,:),'k',t,y_abm(2,:),'g')
    title('ABM Order 4','FontSize',20)

    disp(' '); disp('Derivative'); disp(' ');
    disp('      t       ABM            exact          error');
    disp('    ================================================');
    for i = 1:length(t)
        disp(sprintf('    %4.2f%15.10f%15.10f%15.10f',t(i),y_abm(2,i),...
                     y_true(2,i),y_abm(2,i)-y_true(2,i)));
    end
end
```

(a)



The result is

```
Solution Values

  t       ABM            exact          error
================================================
1.00   1.0000000000   1.0000000000   0.0000000000
1.05   0.9960453968   0.9960453341   0.0000000627
1.10   0.9831947607   0.9831946462   0.0000001146
1.15   0.9594419461   0.9594418296   0.0000001165
1.20   0.9216211746   0.9216342446  -0.0000130700
1.25   0.8642496388   0.8642993888  -0.0000497500
1.30   0.7760928601   0.7762592821  -0.0001664220
1.35   0.6253779328   0.6259584144  -0.0005804816
1.40   0.1367114136  -0.2956385741   0.4323499876
1.45  -4.1203275783  -0.7044941070  -3.4158334714
```

1.50	-3.0540246326	-0.9005487269	-2.1534759057
1.55	-3.4643589156	-1.0543717178	-2.4099871978
1.60	-3.5795386225	-1.1886439361	-2.3908946864
1.65	-3.7009624248	-1.3116333705	-2.3893290543
1.70	-3.8232061250	-1.4273417603	-2.3958643647

(b) The output is

```
Solution Values

  t       ABM            exact          error
=================================================
1.00   1.0000000000   1.0000000000   0.0000000000
1.05   0.9999603350   0.9999603225   0.0000000126
1.10   0.9996971457   0.9996971236   0.0000000221
1.15   0.9990228941   0.9990228644   0.0000000297
1.20   0.9977819869   0.9977823509  -0.0000003640
1.25   0.9958462079   0.9958468092  -0.0000006013
1.30   0.9931086017   0.9931093383  -0.0000007366
1.35   0.9894805817   0.9894813795  -0.0000007978
1.40   0.9848891391   0.9848899443  -0.0000008052
1.45   0.9792746428   0.9792754161  -0.0000007733
1.50   0.9725890768   0.9725897898  -0.0000007129
1.55   0.9647946145   0.9647952462  -0.0000006317
1.60   0.9558624534   0.9558629887  -0.0000005353
1.65   0.9457718549   0.9457722830  -0.0000004281
1.70   0.9345093435   0.9345096569  -0.0000003134
1.75   0.9220680336   0.9220682270  -0.0000001934
1.80   0.9084470559   0.9084471263  -0.0000000703
1.85   0.8936510658   0.8936510114   0.0000000545
1.90   0.8776898147   0.8776896349   0.0000001798
1.95   0.8605777740   0.8605774693   0.0000003047
2.00   0.8423338006   0.8423333723   0.0000004283

Derivative

  t       ABM            exact          error
=================================================
1.00   0.0000000000   0.0000000000   0.0000000000
1.05  -0.0023426519  -0.0023426635   0.0000000115
1.10  -0.0088078321  -0.0088078516   0.0000000196
1.15  -0.0186778519  -0.0186778770   0.0000000251
1.20  -0.0313694121  -0.0313704114   0.0000009993
1.25  -0.0464047616  -0.0464064292   0.0000016677
1.30  -0.0633845515  -0.0633866846   0.0000021330
1.35  -0.0819717806  -0.0819742357   0.0000024551
1.40  -0.1018786387  -0.1018813143   0.0000026755
1.45  -0.1228565351  -0.1228593581   0.0000028229
1.50  -0.1446884573  -0.1446913744   0.0000029171
1.55  -0.1671830680  -0.1671860401   0.0000029720
```

1.60	-0.1901701100	-0.1901731075	0.0000029975
1.65	-0.2134968040	-0.2134998046	0.0000030005
1.70	-0.2370250088	-0.2370279950	0.0000029863
1.75	-0.2606289685	-0.2606319269	0.0000029584
1.80	-0.2841935190	-0.2841964385	0.0000029196
1.85	-0.3076126523	-0.3076155241	0.0000028718
1.90	-0.3307883645	-0.3307911812	0.0000028167
1.95	-0.3536297284	-0.3536324838	0.0000027554
2.00	-0.3760521447	-0.3760548334	0.0000026886

4. Investigate the implicit 2-stage Runge-Kutta method with Butcher tableau

$$\begin{array}{c|cc} 0 & \frac{1}{4} & -\frac{1}{4} \\ \frac{2}{3} & \frac{1}{4} & \frac{5}{12} \\ \hline & \frac{1}{4} & \frac{3}{4} \end{array}$$

(a) Determine the order of the method (see separate write-up).
(b) Find the reduced characteristic equation and show that this method is stable.
Hint: Apply the method to the test problem. You will have to solve a 2×2 linear system for k_1 and k_2.
(c) (2 points extra credit) Show that this method is A-stable. (20)

SOLUTION: **(a)** Check the order conditions:

$$\begin{aligned} \sum \alpha_i &= \frac{1}{4} + \frac{3}{4} = 1, \\ \sum \alpha_i \mu_i &= \frac{1}{4} \cdot 0 + \frac{3}{4} \cdot \frac{2}{3} = \frac{1}{2} \\ \sum \alpha_i \mu_i^2 &= \cdots = \frac{1}{3} \\ \sum \alpha_i \lambda_{ij} \mu_j &= \cdots = \frac{1}{6} \\ \sum \alpha_i \mu_i^3 &= \frac{2}{9} \neq \frac{1}{4} \end{aligned}$$

The conditions for order 3 check out, the first one for order 4 does not. The method has order 3.
(b) The method is

$$\begin{aligned} k_1 &= f(t_n, y_n + \frac{1}{4} h k_1 - \frac{1}{4} h k_2) \\ k_2 &= f(t_n + \frac{2}{3} h, y_n + \frac{1}{4} h k_1 + \frac{5}{12} h k_2) \\ y_{n+1} &= y_n + h \left[\frac{1}{4} k_1 + \frac{3}{4} k_2 \right]. \end{aligned}$$

For the test problem $y' = \lambda y$ we get

$$k_1 = \lambda\left(y_n + \frac{1}{4}hk_1 - \frac{1}{4}hk_2\right),$$
$$k_2 = \lambda\left(y_n + \frac{1}{4}hk_1 + \frac{5}{12}hk_2\right)$$

which leads to

$$\begin{pmatrix} 1-\frac{h\lambda}{4} & \frac{h\lambda}{4} \\ -\frac{h\lambda}{4} & 1-\frac{5h\lambda}{12} \end{pmatrix}\begin{pmatrix} k_1 \\ k_2 \end{pmatrix} = \lambda y_n \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

We find

$$k_1 = \frac{6-4h\lambda}{(h\lambda)^2 - 4h\lambda + 6}\lambda y_n, \qquad k_2 = \frac{6}{(h\lambda)^2 - 4h\lambda + 6}\lambda y_n,$$

which leads to

$$y_{n+1} = y_n \frac{2h\lambda + 6}{(h\lambda)^2 - 4h\lambda + 6}.$$

The characteristic equation is

$$r(h\lambda) = \frac{2h\lambda + 6}{(h\lambda)^2 - 4h\lambda + 6}.$$

The reduced characteristic equation is

$$r = 1$$

so this is stable.

(c) We want to show that

$$\left|\frac{2z+6}{z^2-4z+6}\right| \le 1$$

for all z in the left half plane. Let $z = x + iy$, x, y real. Use $|r(z)|^2 = r(z)\cdot\overline{r(z)}$ to get

$$\frac{4x^2 + 4y^2 + 24x + 36}{x^4 - 8x^3 + 2x^2y^2 + 28x^2 - 8xy^2 - 48x + y^4 + 4y^2 + 36} \le 1.$$

Computer algebra systems are a wonderful thing.

We can write the denominator as

$$(x^2+y^2)^2 + 28x^2 - 8x\left(x^2 + y^2 + 6\right) + 4y^2 + 36,$$

which is positive as long as $x < 0$. So, we can multiply by the denominator and simplify the problem to

$$4x^2 + 4y^2 + 24x + 36 \le x^4 - 8x^3 + 2x^2y^2 + 28x^2 - 8xy^2 - 48x + y^4 + 4y^2 + 36,$$

which simplifies to

$$(x^2+y^2)^2 - 8x\left(x^2+y^2+9\right) + 24x^2 \ge 0,$$

which is obviously true for $x < 0$.