ESC103F Engineering Mathematics and Computation: Lab #2

Pre-Lab

1. For Exercise 1, develop a function called *IEMsolver* that will take the same inputs as *EMsolver* (see below), but the outputs in this case will correspond to the numerical solution to the problem by applying the improved Euler's method.
2. The time variable t is a local variable in the *EMsolver* function. However, you will need t in your main program for plotting purposes. Think about how to modify your functions accordingly.

Exercise 1

Consider the initial value problem:

$$\frac{dx(t)}{dt} = ex(t) + fy(t)$$
$$\frac{dy(t)}{dt} = gx(t) + hy(t)$$
$$x(0) = j$$
$$y(0) = k$$

with constants *e, f, g, h, j, k*.

The function provided below called *EMsolver* takes as inputs:

- the six constants *e, f, g, h, j, k*;
- an upper bound T on the simulation time t;
- the number of time steps to be used N (this implies that $\Delta t = T / N$).

The outputs of this function are two arrays, x and y, that contain the numerical solution to the problem by applying Euler's method.

Numerically solve the following IVP using the Euler method and the improved Euler method:

$$\frac{dx(t)}{dt} = -y(t)$$
$$\frac{dy(t)}{dt} = x(t)$$
$$x(0) = 1$$
$$y(0) = 0$$

For each method, develop a figure showing the resulting numerical solutions for $t \in [0,10]$, using different line types for different time steps N. Use the subplot command to plot *x(t)* versus *t* on the top plot and *y(t)* versus *t* on the bottom plot. Label the axes and add a legend showing the time step *N* associated with each line type.

The exact solution to this system is:

$$x(t) = \cos(t)$$
$$y(t) = \sin(t)$$

In the same figure, using again a different line type, plot the exact solution for $t \in [0,10]$. For this plot you will want to define a grid for *t* that is very fine so that your plot of the exact solution is smooth.

By studying the approximate solutions for different numbers of time steps:

1. Determine how many time steps are needed before the Euler approximation gives a close match to the exact solution.
2. Determine how many time steps are needed before the improved Euler approximation gives a close match to the exact solution.

```
function [x y] = EMsolver(e,f,g,h,j,k,T,N)

    A = [e f; g h];
    dt = T/N;
    t = 0:dt:T;

    SOL = NaN(2,length(t));
    SOL(1,1) = j;
    SOL(2,1) = k;

    for(k = 2:length(t))
        SOL(:,k) = SOL(:,k-1) + dt*A*SOL(:,k-1);
    end

    x = SOL(1,:);
    y = SOL(2,:);
end
```