Numerical Methods & .m Files

In order to use MATLAB routines for the Euler, Improved Euler and Runge-Kutta Methods, you will need the files eul.m, rk2.m and rk4.m, respectively. These files are already present on all PUCC machines as standard software. If you are using your own copy of MATLAB you may need to download these files. Here is a link:  

http://math.rice.edu/~dfield/

• You must first create a function file in the same directory (or folder) as your MATLAB. Here is one way. After MATLAB has been opened, pull down the File menu and select “New M-File”. A window will pop up for you to create your function file. For example, to create a function file for the function

\[ f(t, y) = 6t^3 - e^{2y} + \frac{\sqrt{t}}{y}, \]

type:

\[
function \ W=fcn1(t,y) \n\]
\[
W=6*t^3-exp(2*y)+sqrt(t)/y; \n\]

(Don’t forget the “;” at the end.)

• Save this file as a .m file with the SAME name as your function. The above example would be saved as “fcn1.m”. You can check if your function has been saved by typing something like the following at a MATLAB prompt:

```matlab
>> fcn1(1,3)
```

You should get the value of \( f(1,3) \).

• Your initial value problem will have the form:

\[
\begin{cases}
y' = f(t,y) \\
y(t_0) = y_0
\end{cases}
\]

Assuming \( f(t,y) \) was saved as the file fcn1.m, the syntax for eul (as well as rk2 and rk4, just replace eul) will be:

```matlab
eul(’fcn1’, [t0,tf],y0,h)
```

where \( t_0 \) and \( t_f \) are the initial and final values of \( t \) and \( h = \text{step size} \).

(Your version of MATLAB may not utilize brackets. Type help eul to find out.) To approximate the actual solution to the IVP at \( t_f \), with given \( h \), using eul, just type the following at a MATLAB prompt:

```matlab
>> [t,y]=eul(’fcn1’, [t0,tf],y0,h);
```

The approximations \( y_0, y_1, y_2, \ldots, y_n \) are stored in the matrix \( y \). To print them out type: 

\([t,y] \n\]
To plot them, type: 

plot(t,y)