MAPLE Procedures

Below is a listing of the MAPLE procedures that are needed to solve the numerical differential equations for the Take Home Exam. These procedures can be obtained electronically in the MacLab (BAM 120).

Euler's Method for a differential equation
  y' = f(t,y)
  y(t0) = y0

f   function of 2 variables defined using ->
t0   initial time
y0   initial condition
h   step size
N   number of iterations
output   data storage

```
> Euler:=proc(f,t0,y0,h,N,output)       # Euler's Method
local n:
t(0):=t0:
y(0):=y0:
for n from 0 to N do
   y(n+1):=y(n)+h*f(t(n),y(n)):
  t(n+1):=t(n)+h:
od:
output:=[seq([t(n),y(n)],n=0..N)]:
end:
```

Here is an example

```
> f:=(t,y)->t*(2-y);                    # y'=t*(2-y)
> t0:=0;y0:=1;                  # initial conditions
> h:=.01;N:=100;        # stepsize and number of iterations
> output:='output':             # clear the output variable
> Euler(f,t0,y0,h,N,output):
> output;                       # to view results
> plot(output);                   # to graph the solution
```

Runge Kutta Method for a differential equation
  y' = f(t,y)
  y(t0) = y0

f   function of 2 variables defined using ->
t0   initial time

y0 initial condition
h step size
N number of iterations
output data storage

```
> RK:=proc(f,t0,y0,h,N,output)      # Runge Kutta Method
local n, k1, k2, k3, k4:
t(0):=t0:
y(0):=y0:
for n from 0 to N do
   k1:=h*f(t(n),y(n)):
    k2:=h*f(t(n)+h/2,y(n)+k1/2):
    k3:=h*f(t(n)+h/2,y(n)+k2/2):
   k4:=h*f(t(n)+h,y(n)+k3):
    y(n+1):=y(n)+(1/6)*(k1+2*k2+2*k3+k4):
  t(n+1):=t(n)+h:
od:
output:=[seq([t(n),y(n)],n=0..N)]:
end:

> f:=(t,y)->t*(2-y);                    # y'=t*(2-y)
> t0:=0;y0:=4;                  # initial conditions
> h:=.1;N:=10;          # stepsize and number of iterations
> output1:='output1':           # clear the output variable
> RK(f,t0,y0,h,N,output1):
> output1;
```

The lab manual shows you how to solve differential equations using MAPLE's `dsolve`. It is important to know that this routine does not allow the use of decimal coefficients in the equation. You need to use either fractions or parameters that you assign after using `dsolve`. The other routine you'll need is `DEplot1`. Below shows how to obtain the direction field with the solution to the initial value problem. (Note the solution from this plot is not necessarily very accurate as is apparent in the example below.)

$$y' = t(2-y), \quad y(0) = 1.$$

```
> with(DEtools):          # clear the output variable
> de := diff(y(t),t) = t*(2-y(t));
> dsolve({de, y(0)=1},y(t));                 #Solves the differential equation
> DEplot1(de, [t,y], t=0..10, {[0,1]}, y=0..5);   # Gives Direction Field
```