```
> restart;
  with( DEtools ): with( plots ):
```

Homework Solutions for Section 3.5 (Improved Euler's Method)

Prepared by Doug Meade
22 February 1995

Filename: soln3-5.ms

PROBLEM # 8

```
> f := (x,y) -> (y^2+y)/x;
  x[0] := 1:
  y[0] := 1:
  h := 0.2:
  N := 4:
```

$$f := (x, y) \rightarrow \frac{y^2 + y}{x}$$

Here's the Maple implementation of improved Euler's method.

```
> imp_euler_pts := [ x[0],  y[0] ]:
  for n from 0 to N-1 do
    F := f(x[n],y[n]);
    G := f(x[n]+h,y[n]+h*F);
    x[n+1] := x[n] + h;
    y[n+1] := y[n] + h/2*(F+G);
    new_pt := [ x[n+1], y[n+1] ];
    imp_euler_pts := imp_euler_pts, new_pt;
    print( new_pt );
  od:
```

$$[1.2, 1.480000000]$$

$$[1.4, 2.247801072]$$

$$[1.6, 3.651729891]$$

$$[1.8, 6.887115568]$$

PROBLEM # 9

```
> f := (x,y) -> x + 3*cos(x*y);
  x[0] := 0:
  y[0] := 0:
  h := 0.2:
  N := 10:
```

$$f := (x, y) \rightarrow x + 3\cos(x\,y)$$

```
> imp_euler_pts := [ x[0],  y[0] ]:
  for n from 0 to N-1 do
    F := f(x[n],y[n]);
    G := f(x[n]+h,y[n]+h*F);
    x[n+1] := x[n] + h;
    y[n+1] := y[n] + h/2*(F+G);
    new_pt := [ x[n+1], y[n+1] ];
    imp_euler_pts := imp_euler_pts, new_pt;
    print( new_pt );
  od:
```

$$[.2, .6178425910]$$

$$[.4, 1.238641679]$$

$$[.6, 1.736530812]$$

$$[.8, 1.981106282]$$

$$[1.0, 1.997052218]$$

$$[1.2, 1.884609508]$$

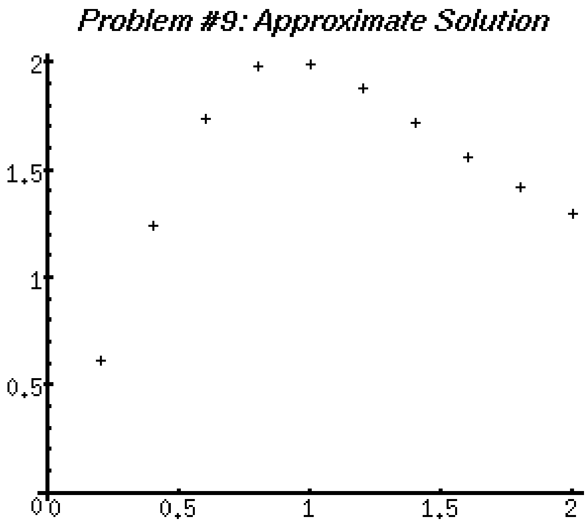
$$[1.4, 1.724472076]$$

$$[1.6, 1.561836096]$$

$$[1.8, 1.417317546]$$

$$[2.0, 1.297794262]$$

```
> plot( [ imp_euler_pts], style=POINT, title=‘Problem #9: Approximate Solution‘ );
```



PROBLEM # 13 and 15

```
> f := (x,y) -> 1-y+y^3;
  x[0] := 0:
  y[0] := 0:
  h := 0.125:
  N := 8:
```

$$f := (x, y) \to 1 - y + y^3$$

```
> imp_euler_pts := [ x[0],  y[0] ]:
  for n from 0 to N-1 do
    F := f(x[n],y[n]);
    G := f(x[n]+h,y[n]+h*F);
    x[n+1] := x[n] + h;
    y[n+1] := y[n] + h/2*(F+G);
    new_pt := [ x[n+1], y[n+1] ];
    imp_euler_pts := imp_euler_pts, new_pt;
    print( new_pt );
  od:
```

$$[.125, .1173095703]$$

$$[.250, .2215774286]$$

$$[.375, .3154463651]$$

$$[.500, .4015340894]$$

$$[.625, .4823125935]$$

$$[.750, .5601526793]$$

$$[.875, .6374717029]$$

$$[1.000, .7169838198]$$

Here’s a summary of my results (it’s much easier to write this on paper, but I think it will be more convenient for you if all work is kept in one place)

h	N	y(1)	abs error	rel error
1	1	1	---	---
0.5	2	0.7372	0.2628	0.3565
0.25	4	0.7194	0.0198	0.0275
0.125	8	0.7170	* 0.0024 *	0.0033
0.0625	16	0.7165	0.0005	* 0.0007 *

Note that the absolute error tolerance is met with 8 steps and the relative (or percent) error tolerance requires 16 steps.

YOUR ANSWERS MAY VARY, DEPENDING ON YOUR CHOICES OF h and N.

PROBLEM # 20

```
> f := (x,y) -> 9.81-2*y;
  x[0] := 0:
  y[0] := 0:
  h := 0.2:
  N := 25:
```

$$f := (x, y) \rightarrow 9.81 - 2\, y$$

```
> imp_euler_pts := [ x[0],  y[0] ]:
  for n from 0 to N-1 do
    F := f(x[n],y[n]);
    G := f(x[n]+h,y[n]+h*F);
    x[n+1] := x[n] + h;
    y[n+1] := y[n] + h/2*(F+G);
    new_pt := [ x[n+1], y[n+1] ];
    imp_euler_pts := imp_euler_pts, new_pt;
    print( new_pt );
  od:
```

We will need a plot of these points, so let’s assemble it now.

```
> p1 := plot( [imp_euler_pts], style=POINT, color=RED, symbol=BOX ):
```

Now, repeat this with a different exponent.

```
> f := (x,y) -> 9.81-2*y^1.5;
  x[0] := 0:
  y[0] := 0:
  h := 0.2:
  N := 25:
```

$$f := (x, y) \rightarrow 9.81 - 2\, y^{1.5}$$

```
> imp_euler_pts := [ x[0],  y[0] ]:
  for n from 0 to N-1 do
    F := f(x[n],y[n]);
    G := f(x[n]+h,y[n]+h*F);
    x[n+1] := x[n] + h;
    y[n+1] := y[n] + h/2*(F+G);
    new_pt := [ x[n+1], y[n+1] ];
    imp_euler_pts := imp_euler_pts, new_pt;
    print( new_pt );
  od:
```

Now, the plot of the points.

```
> p2 := plot( [imp_euler_pts], style=POINT, color=BLUE, symbol=DIAMOND ):
```

And, for a third exponent

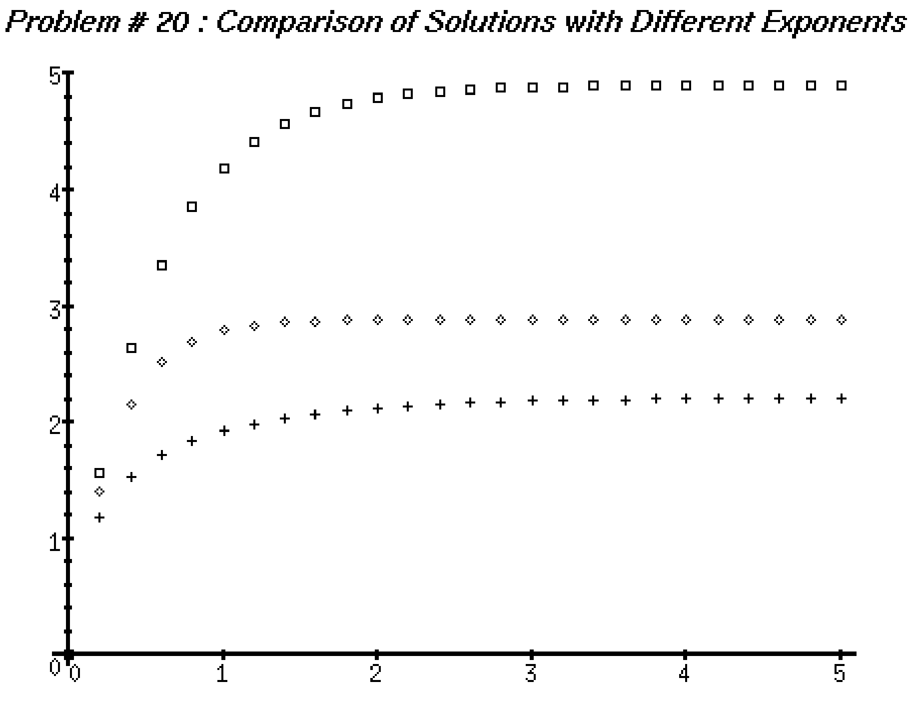
```
> f := (x,y) -> 9.81-2*y^2;
  x[0] := 0:
  y[0] := 0:
  h := 0.2:
  N := 25:
```

$$f := (x, y) \rightarrow 9.81 - 2\,y^2$$

```
> imp_euler_pts := [ x[0],  y[0] ]:
  for n from 0 to N-1 do
    F := f(x[n],y[n]);
    G := f(x[n]+h,y[n]+h*F);
    x[n+1] := x[n] + h;
    y[n+1] := y[n] + h/2*(F+G);
    new_pt := [ x[n+1], y[n+1] ];
    imp_euler_pts := imp_euler_pts, new_pt;
    print( new_pt );
  od:
> p3 := plot( [imp_euler_pts], style=POINT, color=GREEN, symbol=CROSS ):
```

And, finally, here’s the plot of all three solutions. Note that each solution has the same general shape, but a different limiting behavior as t -> infinity.

```
> display( { p1, p2, p3}, title=‘Problem # 20 : Comparison of Solutions with Different Exponen
  ts‘ );
```



END OF WORKSHEET

```
>
```