Practical 1

Plotting the family of solutions of first order differential equations:

1 *Using the pre-defined function: ode2*

(solves an ODE of order upto 2)

1.1 y'= sin(x)*y, y(Π)=k

(%i18)
```
eqn: 'diff(y,x)= sin(x)*y;
sol: ode2(eqn,y,x);
sol1: ic1(sol,x=%pi/2,y=k);
p1: ev(sol1,k=-1);
p2: ev(sol1,k=-2);
p3: ev(sol1,k=1);
p4: ev(sol1,k=2);
wxplot2d([rhs(p1),rhs(p2),rhs(p3),rhs(p4)],[x,-10,10],
              [style,[lines,1],[lines,2],[lines,3],[lines,4]],
              [legend,"p1","p2","p3","p4"])
;
```

(%o11) $\frac{d}{dx} y = \sin(x)\, y$

(%o12) $y = \%c\, \%e^{-\cos(x)}$

(%o13) $y = k\, \%e^{-\cos(x)}$

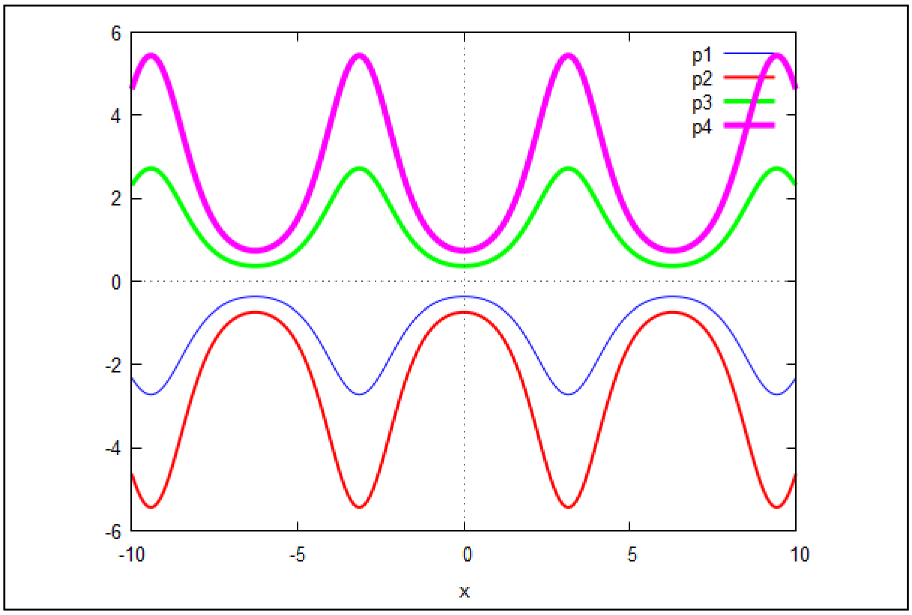
(%o14) $y = -\%e^{-\cos(x)}$

(%o15) $y = -2\, \%e^{-\cos(x)}$

(%o16) $y = \%e^{-\cos(x)}$

(%o17) $y = 2\, \%e^{-\cos(x)}$

(%t18)



(%o18)

1.2 (2xy)y'= y² -x², y(1)=k

→ `kill(all);`

(%i101)
```
eqn: (2*x*y)* 'diff(y,x)= y^2 -x^2;
sol: ode2(eqn,y,x);
sol1: ic1(sol,x=1,y=k);
p1: ev(sol1,k=1);
p2: ev(sol1,k=2);

load(draw);
wxdraw2d(implicit(p1,x,-1,6,y,-5,5),
         implicit(p2,x,-1,6,y,-5,5));
```

(%o95) $2\,x\,y\left(\frac{d}{d\,x}\,y\right)=y^2-x^2$

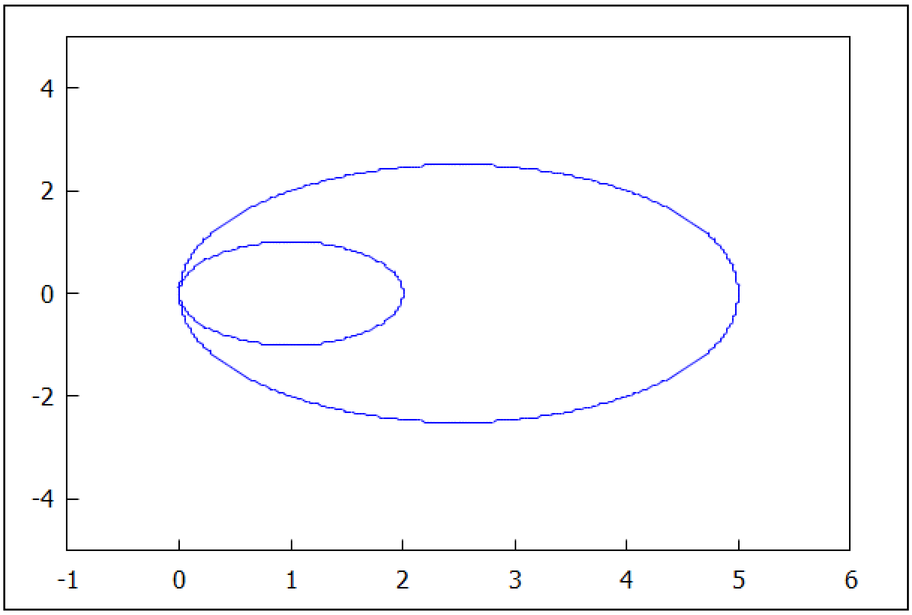
(%o96) $-\frac{x}{y^2+x^2}=\%c$

(%o97) $-\frac{x}{y^2+x^2}=-\frac{1}{k^2+1}$

(%o98) $-\frac{x}{y^2+x^2}=-\frac{1}{2}$

(%o99) $-\frac{x}{y^2+x^2}=-\frac{1}{5}$

(%o100)
C:/maxima-5.44.0/share/maxima/5.44.0/share/draw/draw.lisp

(%t101)


(%o101)

1.3 y'= y+ e^(-x), y(2)= -0.1 (Exercise !)

```
kill (all);
eqn: 'diff(y,x)= y+ exp(-x);    /* or write as %e^(-x)*/
```

(%o0) done

(%o1) $\frac{d}{d\,x}\,y=\frac{1}{exp^{x}}+y$

2 Using the pre-defined function: desolve (solves a system of linear ODEs of any order)

2.1 y' =x^2, y(0)=k

```
eqn: diff(y(x),x)= x^2;
sol: desolve(eqn,y(x)); /*desolve doesn't give arbitrary constants
                          but their value(s) in terms of y */
p1: ev(sol,y(0)=-1);    /*can't use "ic1" because no constant
                         %c in the solution*/
p2: ev(sol,y(0)=1);
wxplot2d([rhs(p1),rhs(p2)],[x,-1,1],
         [style,[lines,1],[lines,2]],
         [legend,"p1","p2"])


;
```

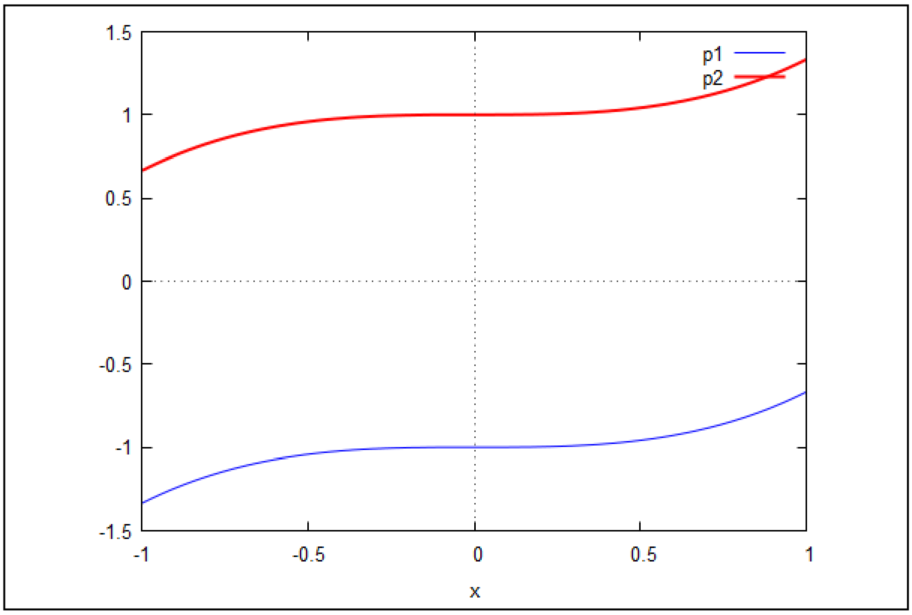
(%o62) $\frac{d}{d\,x}\,y(x)=x^2$

(%o63) $y(x)=\frac{x^3}{3}+y(0)$

(%o64) $y(x)=\frac{x^3}{3}-1$

(%o65) $y(x)=\frac{x^3}{3}+1$

(%t66)



(%o66)

2.2 y'= sin(x)*y, y(Π)=k

(%i68)
```
eqn: diff(y(x),x)= sin(x)*y(x);
sol: desolve(eqn,y(x));
```

(%o67) $\frac{d}{d\,x}\,y(x)=y(x)\,\sin(x)$

(%o68) $y(x)=ilt($

$(\%i\ laplace(y(x),x,g1315+\%i)-\%i\ laplace(y(x),x,g1315-\%i)+2\,y(0))$

$/(2\,g1315),g1315,x)$

```
/*"ilt" means that Maxima has failed to determine the inverse
Laplace transform. So desolve can't be used to solve the ODE*/
```

2.3 (2xy)y'= y² -x², y(1)=k

```
(%i70) eqn: (2*x*y(x))*diff(y(x),x)= (y(x))^2- x^2;
sol: desolve(eqn,y(x));
```

(%o69) $$2\,x\,y(x)\left(\frac{d}{d\,x}\,y(x)\right)=y(x)^2-x^2$$

desolve: can't handle this case.
-- an error. To debug this try: debugmode(true);

```
/* desolve works for linear ODE, but this is non-linear */
```

2.4 y'= 9.8- 0.196 y

```
eqn: diff(y(x),x)= 9.8- 0.196*y(x);
ratprint: false $ /*When Maxima gets float input, the simplifier
                    tries to convert this to rational representation
                    and informs the user about this conversion which
                    is normally irrelevant.To suppress this, we use
                    this command */

sol: desolve(eqn,y(x));
p1: ev(sol,y(0)=1);
p2: ev(sol,y(0)=0);
wxplot2d([rhs(p1),rhs(p2)],[x,-1,1],
         [style,[lines,1],[lines,2]],
         [legend,"p1","p2"]);
```

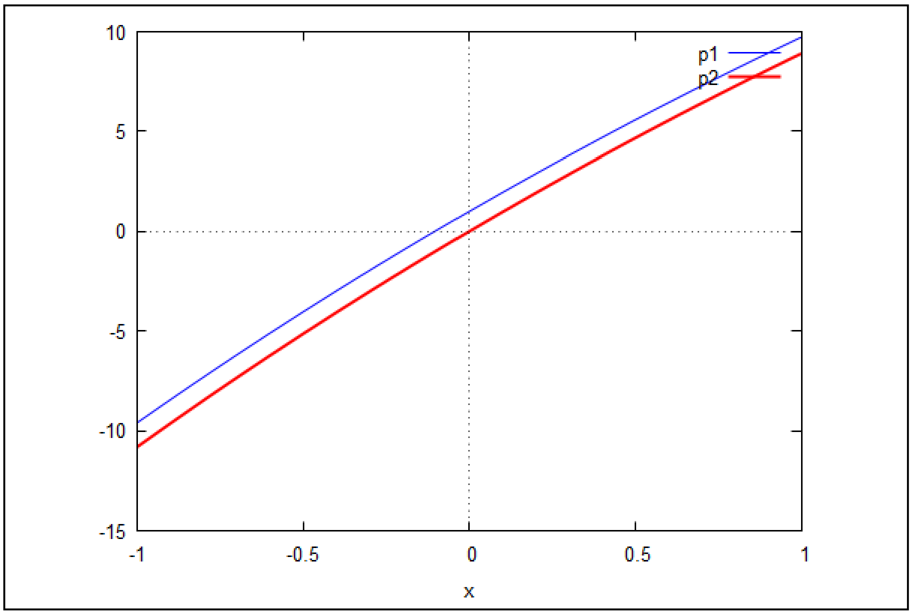
(%o89) $$\frac{d}{d\,x}\,y(x)=9.8-0.196\,y(x)$$

(%o91) $$y(x)=\frac{(250\,y(0)-12500)\,\%e^{-\frac{49\,x}{250}}}{250}+50$$

(%o92) $$y(x)=50-49\,\%e^{-\frac{49\,x}{250}}$$

(%o93) $$y(x)=50-50\,\%e^{-\frac{49\,x}{250}}$$

(%t94)



(%o94)