



Maxima 5.27.0 <http://maxima.sourceforge.net>  
 using Lisp GNU Common Lisp (GCL) GCL 2.6.7 (a.k.a. GCL)  
 Distributed under the GNU Public License. See the file COPYING.  
 Dedicated to the memory of William Schelter.

## Équations différentielles linéaires d'ordre 2

Résoudre  $y'' + 4y' - 5y = 0$  avec les conditions initiales  $y''(0) = 6$  et  $y'(0) = 0$ .

▷ `s:ode2('diff(y,x,2)+4*'diff(y,x)-5*y=0,y,x);`

1:  $y = \%k_1 e^x + \%k_2 e^{-5x}$

**Maxima** propose une procédure (`ic2`) pour déterminer les constantes en fonction des conditions initiales si celles-ci se présentent sous la forme  $y(a) = \alpha$  et  $y'(a) = \beta$ . Ce n'est pas tout à fait le cas ici où c'est  $y''(0)$  qui est connu et non  $y(0)$ . On peut s'en sortir en calculant  $y(0)$  à l'aide de l'équation elle-même.

▷ `ic2(s,x=0,y=6/5,'diff(y,x)=0);`

2:  $y = e^x + \frac{e^{-5x}}{5}$

Résoudre  $y'' + 4y' + 5y = 0$  avec  $y(0) = 1$  et  $y'(0) = -1$ .

▷ `s:ode2('diff(y,x,2)+4*'diff(y,x)+5*y=0,y,x);`

3:  $y = e^{-2x} (\%k_1 \sin x + \%k_2 \cos x)$

▷ `ic2(s,x=0,y=1,'diff(y,x)=-1);`

4:  $y = e^{-2x} (\sin x + \cos x)$

Résoudre  $y'' + 2y' + y = 0$  avec  $y(0) = 1$  et  $y'(0) = 2$ .

▷ `s:ode2('diff(y,x,2)+2*'diff(y,x)+y=0,y,x);`

5:  $y = (\%k_2 x + \%k_1) e^{-x}$

▷ `ic2(s,x=0,y=1,'diff(y,x)=2);`

6:  $y = (3x + 1) e^{-x}$