In [41]:
```
%autosave 0
```

Autosave disabled

(Custom CSS files are not reliable for controlling Jupyter font style. To establish the same appearance as the original notebook, depend on the browser to control the font, by setting the desired font faces in the browser settings. For example, Chrome 135 or Firefox 134 can do this. In this notebook series, Bookerly font is for markdown and Monaco is for code.)

Chapter 26: Solutions of Linear Differential Equations w Constant Coefficients by Matrix Methods. Learning to handle differential equations according to matrix principles, and to transmute equations into that format.

Note: When transcribing text for entry into Wolfram Alpha the "fenceposts" !| ... |! are excluded from the text.

Starting with the example equation: $t^4\,\frac{d^2x}{dt^2} + (\sin t)\,\frac{dx}{dt} - 4\,x = \log t$

It is not difficult to rearrange to isolate the second order derivative on lhs:

$$\frac{d^2x}{dt^2} = -\frac{(\sin t)}{t^4}\,\frac{dx}{dt} + \frac{4\,x}{t^4} + \frac{\log t}{t^4}$$

Then suppose $v = x'$ and $v' = x''$. A matrix multiplication could be arranged whereby, as shown in the sympy cell below,

In [38]:
```
from sympy import *

t, v, x, xprime, vprime = symbols('t v x xprime vprime')
pr = Matrix(2, 1, (xprime, vprime))
asy = Matrix(2, 2, [0, 1, 4/t**4, -sin(t)/t**4])
vp = Matrix(2, 1, [x, v])
ex = Matrix(2, 1, [0, log(t)/t**4])
res = asy*vp + ex

pr
```

Out[38]:

$$\begin{bmatrix} x' \\ v' \end{bmatrix}$$

In [77]:
```
asy
```

Out[77]:

$$\begin{bmatrix} 0 & 1 \\ \frac{4}{t^4} & -\frac{\sin(t)}{t^4} \end{bmatrix}$$

In [78]:
```
ex
```

Out[78]:

$$\begin{bmatrix} 0 \\ \frac{\log(t)}{t^4} \end{bmatrix}$$

In [39]:
```
pr = asy*vp + ex
pr
```

Out[39]:

$$\begin{bmatrix} v \\ -\frac{v\sin(t)}{t^4} + \frac{4x}{t^4} + \frac{\log(t)}{t^4} \end{bmatrix}$$

The above two cells show a basic matrix representation of an ODE and its solution. And the paradigm of solving ODEs via linear algebra methods could be developed, which is exactly what this chapter is intended to do. However, since most system equation goals can be achieved through Laplace methods, as shown in the last chapter, this, at least for now, will be another derailed chapter. Only if some equation cannot be operated on by routine plugging, or by trial of a Laplace procedure, will the prospect of matrices be revisited.

26.1 Solve $\ddot{x} + 2\dot{x} - 8x = 0; \quad x(1) = 2, \dot{x}(1) = 3.$

This problem can be entered in Wolfram Alpha. The entry line is:

!| [x'' + 2 x' - 8 x = 0; {x(1) = 2, x'(1) = 3}] |!

and the result is:

$$x(t) = \frac{1}{6}e^{-4t-2}\left(11e^{6t} + e^{6}\right)$$

26.2 Solve $\ddot{x} + 2\dot{x} - 8x = e^{t}; \quad x(0) = 1, \dot{x}(0) = -4.$

This problem can be entered in Wolfram Alpha. The entry line is:

!| [x'' +2 x' - 8 x = e^t, {x(0) = 1, x'(0) = -4}] |!

and the result is:

$$x(t) = \frac{1}{30}e^{-4t}\left(-6e^{5t} + 5e^{6t} + 31\right)$$

26.4 Solve $\ddot{x} + x = 3; \quad x(\pi) = 1, \dot{x}(\pi) = 2.$

This problem can be entered in Wolfram Alpha. The entry line is:

!| [x" + x = 3, {x(pi) = 1, x'(pi) = 2}] |!

and the result is:

$$x(t) = -2\sin(t) + 2\cos(t) + 3$$

26.5 Solve the differential equation $\ddot{x} - 6\dot{x} + 9x = t$.

This problem can be entered in Wolfram Alpha. The entry line is:

!| [x" - 6 x' + 9 x = t] |!

and the result is:

$$x(t) = c_2 e^{3t} t + c_1 e^{3t} + \frac{t}{9} + \frac{2}{27}$$

26.6 Solve the differential equation $\frac{d^3x}{dt^3} - 2\frac{d^2v}{dt^2} + \frac{dx}{dt} = 0.$

This problem can be entered in Wolfram Alpha. The entry line is:

!| d^3x/dt^3 -2 x" + x' = 0 |!

and the result is:

$$x(t) = e^t (c_2 t + (c_1 - c_2)) + c_3$$

26.7 Solve the system

$$\ddot{x} = -2\dot{x} - 5y + 3$$

$$\dot{y} = \dot{x} + 2y;$$

$$x(0) = 0 \quad \dot{x}(0) = 0 \quad y(0) = 1$$

Wolfram Alpha will not process this problem, either as a direct ODE or as an input to a Laplace transform. Therefore Maxima will have to do the initial work. The two cells below show the individual equations for both parts of the system of equations. The lines corresponding to entry into Laplaceland are numbers 6 and 15.

(%i1) ode: 'diff(x(t), t, 2) + 2·'diff(x(t), t) + 5·y(t) = 3;

(%o1) $\frac{d^2}{d t^2} x(t) + 2\left(\frac{d}{d t} x(t)\right) + 5 y(t) = 3$

(%i2) atvalue(x(t), t=0, 0);

(%o2) 0

(%i3) atvalue('diff(x(t), t) , t=0, 0);

(%o3) 0

(%i5) atvalue(y(t), t=0, 1);

(%o5) 1

(%i6) lap_ode: laplace(ode, t, s);

(%o6) $5\,\mathrm{laplace}(y(t), t, s) + s^2\,\mathrm{laplace}(x(t), t, s) + 2\,s\,\mathrm{laplace}(x(t), t, s) = \frac{3}{s}$

(%i11) ode2: 'diff(y(t), t) = 'diff(x(t), t) + 2·y(t);

(%o11) $\frac{d}{d t} y(t) = \frac{d}{d t} x(t) + 2\,y(t)$

(%i12) atvalue(x(t), t=0, 0);

(%o12) 0

(%i13) atvalue('diff(x(t), t) , t=0, 0);

(%o13) 0

(%i14) atvalue(y(t), t=0, 1);

(%o14) 1

(%i15) lap_ode2: laplace(ode2, t, s);

(%o15) $s\,\mathrm{laplace}(y(t), t, s) - 1 = 2\,\mathrm{laplace}(y(t), t, s) + s\,\mathrm{laplace}(x(t), t, s)$

The s-algebraic equations are the cleaned-up versions of the ones shown in the raw Maxima cells. They are displayed below.

$$5\,Y(s) \;+\; s^2\,X(s) \;+\; 2sX(s) \;=\; \frac{3}{s}$$

$$s\,Y(s) \;-\; 1 \;=\; 2\,Y(s) \;+\; s\,X(s)$$

The algebraic step of the Laplace method is to get the two above equations solved simultaneously. This will be done in Wolfram Alpha, with the entry line:

!| solve[5 y + s^2 x + 2 s x = 3/s, s y -1 =2 y + s x], x, y |!

and having the result:

$$x \;=\; -\frac{2\,(s \;+\; 3)}{s^4 \;+\; s^2}$$

$$y \;=\; \frac{s^2 \;+\; 2\,s \;+\; 3}{s^3 \;+\; s}$$

The two equations above are now staged and ready to return to Regularland, via Wolfram Alpha. The entry line for the first is:

!| inverse laplace transform[-(2s + 6)/(s^4 + s^2), s, t] |!

and the result is:

$$-6\,t \;+\; 2\,(3\,\sin\,(t) \;+\; \cos\,(t)) \;-\; 2$$

and the second entry line is:

!| inverse laplace transform[(s^2 + 2s + 3)/(s^3 + s), s, t] |!

and the result is:

$$2\,\sin\,(t) \;-\; 2\,\cos\,(t) \;+\; 3$$

In Regularland, x symbols will be inserted to replace the s symbols, and the problem will be complete. The problem admitted application of the Laplace method, resulting in a much shorter workspace. Of course this is not guaranteed to always be the case.

26.8 Solve the system of differential equations

$$\dot{x} = x + y$$

$$\dot{y} = 9x + y$$

Wolfram Alpha has no difficulty solving this problem, with the input line:

!| solve[{x' = x + y, y' = 9 x + y}] |!

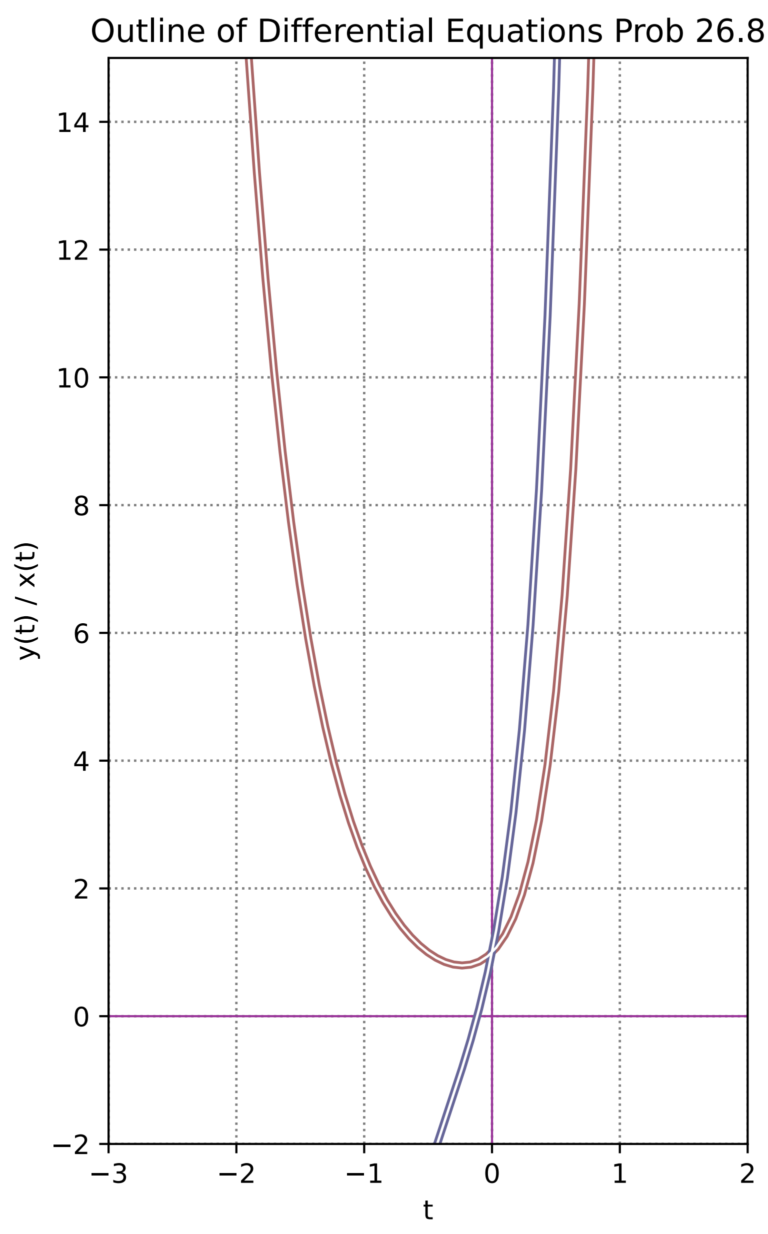
and the results:

$$x(t) = \frac{1}{2}\,c_1\,e^{-2t}\,(e^{6t} + 1) + \frac{1}{6}\,c_2\,e^{-2t}\,(e^{6t} - 1)$$

$$y(t) = \frac{3}{2}\,c_1\,e^{-2t}\,(e^{6t} - 1) + \frac{1}{2}\,c_2\,e^{-2t}\,(e^{6t} + 1)$$

Because of the complicated nature of these answers, it may be best to reconcile them with the text answer by drawing a plot. (In the plots, all arbitrary constants are assigned the value of 1.)

In [76]:
```
import numpy as np
import matplotlib.pyplot as plt

%config InlineBackend.figure_formats = ['svg']

x = np.linspace(-10,10,300)
y3 = (1/2)*np.exp(-2*x)*(np.exp(6*x) + 1) + (1/6)*np.exp(-2*x)*(np.exp(6*x) - 1)
y4 = (3/2)*np.exp(-2*x)*(np.exp(6*x) - 1) + (1/2)*np.exp(-2*x)*(np.exp(6*x) + 1)
# the two functions above were calculated by Wolfram Alpha
y5 = (1/6)*(3 + 1)*np.exp(4*x) + (1/6)*(3 - 1)*np.exp(-2*x)
y6 = (3/6)*(3 + 1)*np.exp(4*x) - (3/6)*(3 - 1)*np.exp(-2*x)
# the two functions above were calculated by the text


plt.grid(True, linestyle='dotted', color='gray', linewidth=0.9)
plt.xlabel("t")
plt.ylabel("y(t) / x(t)")
plt.title("Outline of Differential Equations Prob 26.8")
plt.rcParams['figure.figsize'] = [9, 7.5]


ax = plt.gca()
ax.axhline(y=0, color='#993399', linewidth=0.8)
ax.axvline(x=0, color='#993399', linewidth=0.8)
ratio = 0.5
xleft, xright = ax.get_xlim()
ybottom, ytop = ax.get_ylim()
ax.set_aspect(abs((xright-xleft)/(ybottom-ytop))*ratio)


#plt.text(965, -2.5e43, "-np.log(abs((np.cos(x/2))**2 - (np.sin(x/2))**2)))
#plt.text(965, 2.25e43, "SOLN: y = 3*np.exp(x**2) + 1/2", size=10,\
       # bbox=dict(boxstyle="square", ec=('#8C564B'),fc=(1., 1., 1),))
plt.ylim(-2,15)
plt.xlim(-3, 2)
plt.plot(x, y3, linewidth = 3, color = '#AA6666')
plt.plot(x, y4, linewidth = 3, color = '#666699')
plt.plot(x, y5, linewidth = 0.9, color = 'w')
plt.plot(x, y6, linewidth = 0.9, color = 'w')
plt.show()
```



In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: