```
the rate of radioactive decay is directly proportional to the amount of the radi
oactive material, such that

dR/dt = k*N
        where R is the rate of decay, k is a constant probability, and N is mols
 remaining

d(theta)/dt = -g/l*sin(theta)
        where g is the gravitational constant, l is the length of the pendulum,
and theta is the angle from the center

dV/dt = V/(RC)
        where V is the voltage across the capacitor and R and C are the constant
s in ohms and farads, respectively
```

```
import sympy as s

e = s.exp(1)
(x, t) = s.var('x t')


## PART A
fx_t = e**(4*t)
dxdt = 4*x
print(fx_t.diff())
# >>> 4*exp(4*t)
print(fx_t.diff().subs(fx_t, x), dxdt)
# >>> 4*x 4*x

# we can see that the provided value for x(t) is a solution of dxdt = 4x by
# * differentiating the provided equation
# * substituting the provided value into our expression, simplifying it
# * comparing to the expected value


ANSWER_PART_B = """
this is a little more awkward than the above, so let's do it by hand

fx_t = −1/4

constant function, derivative is zero

dxdt = 4*x+1

substitute provided x(t) value of −1/4 into expression reduces to zero

0 == 0, checks out. (and probably why CAS failed)
"""

fx_t = 3*e**(4*t)−1/4
fx_t.diff()
# >>> 12*exp(4*t)
dxdt = 4*x + 1
dxdt.subs(x, fx_t)
# >>> 12*exp(4*t)

ANSWER_PART_C = """
here, we differentiate our provided x(t) giving the same answer as provided in the prompt.

we next substitute our provided x(t) into the provided formulation of dxdt and compare.

the values are equal.
"""


ddxdt = −4*x
fx_t = s.sin(2*t)
fx_t.diff().diff()
# >>> −4*sin(2*t)
fx_t.diff().diff().subs(fx_t, x)
# >>> −4*x

ANSWER_PART_D = """
```

we differentiate our provided x(t) twice and substitute our provided x(t) into the doublederivative

this equals our provided expression for the double derivative
"""

```
fx_t = 3*s.sin(2*t)+2*s.cos(2*t)
fx_t.diff().diff()
# >>> -12*sin(2*t) - 8*cos(2*t)
fx_t.diff().diff()/fx_t
# >>> (-12*sin(2*t) - 8*cos(2*t))*x/(3*sin(2*t) + 2*cos(2*t))
_.trigsimp()
# >>> -4


ANSWER_PART_E = """
```

using a similar hybrid approach as above...

we take the double derivative of our expression twice, do algebra, simplify, and compare −4 with the coefficient of our expression
"""

```
two parts... first, educated guesses:

a) -6
b) +3, -3
c) -1, +1, possibly \pm i?

these guesses are derived by basic algebra and some extrapolation / half-remembe
red algorithm for dealing with double and triple derivs

the second half of this would be actually working through the algebra in sympy a
nd validating my guesses... maybe later!
```

```
4a)

This one is harder to conceptualise - we're tasked with relating two sets of equ
ations:

a homogenous system:

dxdt = 4x
x = exp(4t)

an inhomogenous system:

dxdt = 4x + 1
x = -1/4

The stated relation we're exploring is that

x = a*x_h + x_p
for the equation:

dxdt = 4x + 1

such that:

x = a * exp(4t) - 1/4

we're comparing this formulation with the provided value of dxdt, so let's integ
rate the provided value:

dxdt = 4x + 1
dx = (4x+1)*dt
x = (4x+1)*t

....

picking this up later.
```