Take-Home Quiz 6: Intro to differential equations (§3.1-3.3)

Directions: This quiz is due on March 1, 2018 at the beginning of lecture. You may use whatever resources you like – e.g., other textbooks, websites, collaboration with classmates – to complete it **but YOU MUST DOCUMENT YOUR SOURCES**. Acceptable documentation is enough information for me to find the source myself. Rote copying another's work is unacceptable, regardless of whether you document it.

For each of the following problems (see next page):

- Determine whether the 1st order ODE is separable, exact, or neither. If it is separable or exact, find the general solution by hand.
- If necessary, rewrite the equation so that it is of the form $y' = f(x, y)$. Then use the Sage code

```
y = function('y')(x)
theEquation = diff(y,x) == f # in place of f, type in the right-hand side of the equation y'
    = f(x,y)
#show(theEquation)
yGeneral = desolve( theEquation, y )
show(yGeneral.expand())
```

  to find the solution. (It should be consistent with what you computed.)
- Verify Sage's solution by hand, by plugging it into the original differential equation.
- Add the Sage code

```
grWin = ( -8, 8, -8, 8 ) # these are the dimensions for the graphing window and you can
    change them if you want
y = var('y')
P0 = plot_slope_field( f, (x, grWin[0], grWin[1]), (y, grWin[2], grWin[3]) ) # instead of f,
    type in what you put for f earlier in the code
show(P0)
```

  to generate a slope field.
- Choose three initial values. Plot them and their solutions in Sage by adding the following code:

```
# Comment out the previous three lines of code first.
var( "a1 a2 a3 b1 b2 b3" )
(a1, b1) = # enter the first point
(a2, b2) = # enter the second point
(a3, b3) = # enter the third point
Y1 = desolve( theEquation, y, [a1, b1] )
#show(Y1.expand())
Y2 = desolve( theEquation, y, [a2, b2] )
#show(Y2.expand())
Y3 = desolve( theEquation, y, [a3, b3] )
#show(Y3.expand())
y = var('y')
P0 = plot_slope_field( f, (x, grWin[0], grWin[1]), (y, grWin[2], grWin[3]) ) # instead of f,
    type in what you put for f earlier in the code
P1 = plot( Y1, (x, grWin[0], grWin[1]), ymax = grWin[3], ymin = grWin[2] )
P2 = plot( Y2, (x, grWin[0], grWin[1]), ymax = grWin[3], ymin = grWin[2] )
P3 = plot( Y3, (x, grWin[0], grWin[1]), ymax = grWin[3], ymin = grWin[2] )
dotSize = 60 # size of the points -- you can change it if you wish
IV1 = point( (a1, b1), size = dotSize )
IV2 = point( (a2, b2), size = dotSize )
```

```
IV3 = point( (a3, b3), size = dotSize )
# Chapter 3 of the Sage manual has more information about how to change the attributes of the
    graph, such as colors, axes, etc., if you wish to do that.
P = P0 + P1 + P2 + P3 + IV1 + IV2 + IV3
show(P)
```

- Print the graph. Label it with the general solution. Label your initial values and the equations for their respective solution curves.

---

1. **§3.1 #14** $y' = xy + 4x$
2. **§3.1 #22** $y' - 2xy = x^3$
3. **§3.1 #24** $(x^2 - y^2)\ dx + xy\ dy = 0$

   Hint on graphing this one: Uncomment `show(Y1.expand())` *so you can see the solution to the initial value problem. Move everything to one side of the equation. Then use the code*

   ```
   P1 = implicit_plot( Y1, (x, grWin[0], grWin[1]), (y, grWin[2], grWin[3]) )
   ```

   where instead of putting `Y1`*, type in what you got when you moved everything to one side of the equation.*
4. **§3.1 #26** $2y\sin(xy)\ dx + (2x\sin(xy) + 3y^2)\ dy$
5. **§3.2 #16** $ye^x\ dy - \sec y\ dx = 0$
6. **§3.3 #4** $(2xe^{xy} + x^2ye^{xy} - 2)\ dx + x^3e^{xy}\ dy = 0$
7. **§3.3 #12** $y' = \dfrac{2xe^y - 3x^2y}{x^3 - x^2e^y}$