Error of Taylor Polynomials: (to be done with a partner!)

In this project you will experimentally investigate what happens to the error of Taylor approximations. For some of the questions I have included examples of the input/output that you should get.

1. Getting *Mathematica* to define Taylor Polynomials

a) Read about "Series" in the online documentation.

b) Have *Mathematica* calculate the 15 Taylor polynomial for f(x) = ArcSin(x) based at 0.

What does the $O(x^{16})$ mean?

```
Series[ArcSin[x], {x, 0, 15}]
```

$$x + \frac{x^3}{6} + \frac{3 x^5}{40} + \frac{5 x^7}{112} + \frac{35 x^9}{1152} + \frac{63 x^{11}}{2816} + \frac{231 x^{13}}{13\,312} + \frac{143 x^{15}}{10\,240} + O(x^{16})$$

c) Have *Mathematica* calculate the 5th Taylor polynomial for g(x) = e^x based at 0. (Note that you need   to use 'E' instead of 'e' in *Mathematica*.

d) Use *Mathematica* to add the answer from b) to the answer from c). Explain the $O(x^5)$ term.

e) To plug numbers in for x, we need to have *Mathematica* get rid of the $O(x^5)$ term. Use the Normal   command to do this for your answers from b) and c).

```
Normal[Series[ArcSin[x], {x, 0, 15}]]
```

$$\frac{143 x^{15}}{10\,240} + \frac{231 x^{13}}{13\,312} + \frac{63 x^{11}}{2816} + \frac{35 x^9}{1152} + \frac{5 x^7}{112} + \frac{3 x^5}{40} + \frac{x^3}{6} + x$$

According to *Mathematica* we still don't have a function. In other words, we still can't plug in for x.   Use the following command to do this for the ArcSin example

```
f[x_] := ArcSin[x]
p[x_] := Normal[Series[ArcSin[y], {y, 0, 15}]] /. y → x
```

Define similar functions for g(x) = e^x and the polynomials from 1 c.

f) Use *Mathematica* to comput the result of plugging 3 into ArcSin[x], e^x, and the Taylor polynomials.

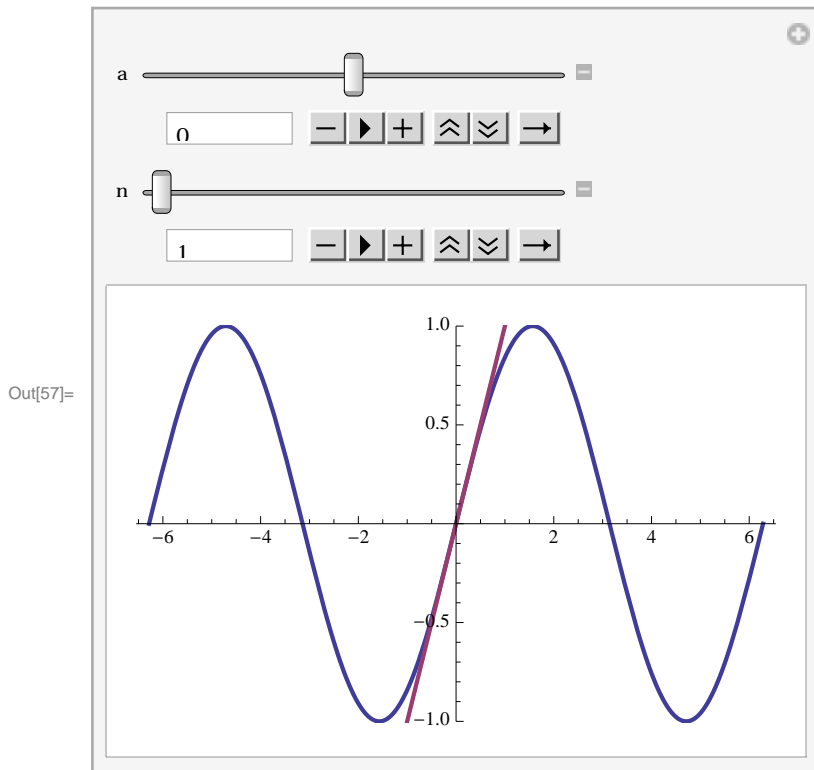## 2. Graphing the Taylor polynomials and analyzing error.
How good a Taylor approximation is, depends on the function being approximated, how many terms the Taylor polynomial has, and the point where the polynomial is based.

Read about "Manipulate" in the documentation and study/play around with this example:

In[55]:= ```
f[x_] := Sin[x]
p[x_, n_, a_] := Normal[Series[f[y], {y, a, n}]] /. y → x
Manipulate[Plot[{f[x], p[x, n, a]}, {x, -2 Pi, 2 Pi},
  PlotRange → {-1, 1}, PlotStyle → Thick], {{a, 0}, -Pi / 2, Pi / 2}, {n, 1, 15, 1}]
```

Out[57]=



To have the result display the values of *a* and *n* as pictured, click the little plus sign next to the slider.

We can have *Mathematica* plot the error of the approximation below the graphs if we add another plot command to "Manipulate".

```
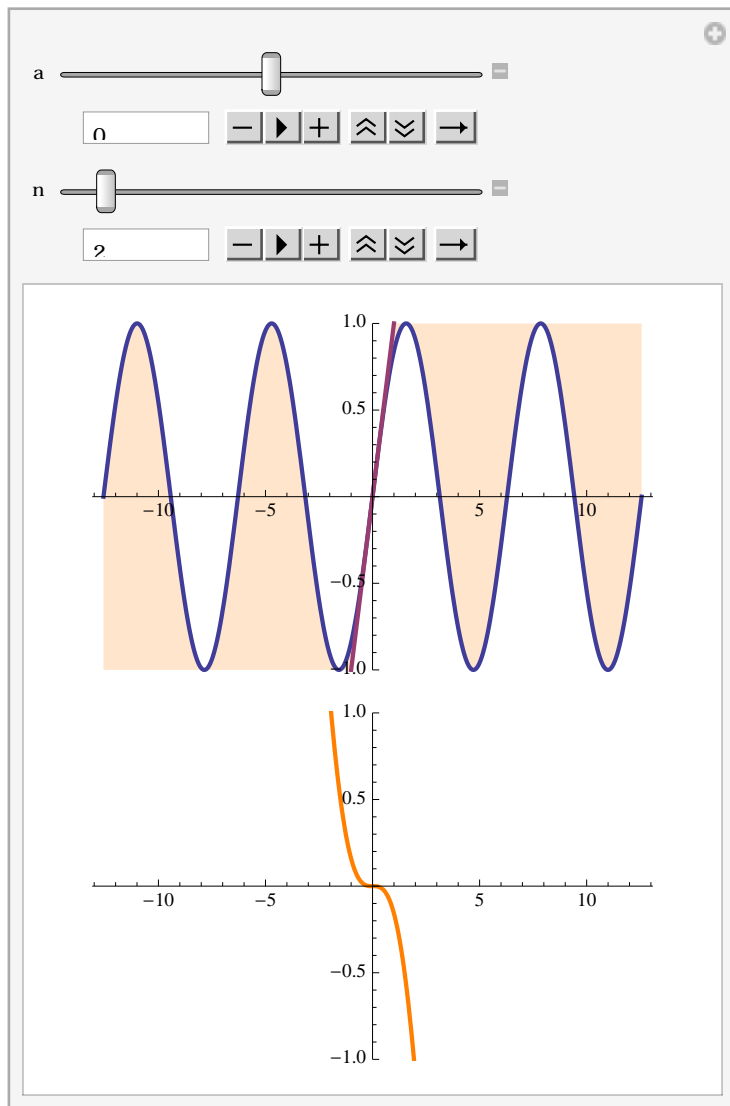In[207]:= f[x_] := Sin[x]
        p[x_, n_, a_] := Normal[Series[f[y], {y, a, n}]] /. y → x
        xRange = 4 Pi;
        Manipulate[
         GraphicsColumn[{
           Plot[
             {f[x], p[x, n, a]},
             {x, -xRange, xRange},
             PlotRange → {-1, 1},
             PlotStyle → Thick,
             Filling → {1 → {2}},
             FillingStyle → Directive[Opacity[.2], Orange]
            ],
           Plot[
             f[x] - p[x, n, a],
             {x, -xRange, xRange},
             PlotRange → {-1, 1},
             PlotStyle → {Thick, Orange}
            ]
          }],
         {{a, 0}, -Pi / 2, Pi / 2}, {n, 1, 15, 1}]
```

Out[210]=

a) Spend some time playing with the plots above and analyze the example(s) above for the Taylor polynomials of f(x) = sin(x). Write 1 or 2 paragraphs describing the effects of changing the degree of the Taylor polynomial and changing the basepoint of the Taylor polynomial. How does the error behave with respect to n, x, and a?

b) Choose two functions for which you'll do a similar analysis. You may choose any two functions as long as:

       i) Neither function is sin(x), cos(x), ln(1 + x), or e^x

       ii) The two functions are significantly different from each other.

   Remember that you can create functions by combining well known functions using basic arithmetic options, and function composition. For example, you could use the function $e^{\sin(x)}$.

c) For each of your two functions, use Manipulate to create plots like those for sin(x) above. You will need to figure out what good choices for the x-Range and y-Range of your plots are. If you make your functions too complicated, it may take *Mathematica* a long time to compute the Taylor polynomials. Make sure that the results look good. You may find it helpful to refer frequently to the documentation.

Your output does not have to look exactly like what I have. For example, perhaps you want to add a 2-D slider to Manipulate so that the user can adjust the dimensions of the plot for him/herself. Your output should allow for n to go up to at least 15, but depending on your function it may be useful to have n go considerably higher.

d) For each of your two functions, use the output from your Manipulate command to analyze the dependence of the error on n and a. For each of your functions you should have one or two paragraphs summarizing your findings. Be sure to discuss any places where the approximation gets worse, the larger n gets as well as values of a that always result in better or worse approximations than other values of a.

e) Be sure that your names are in the body of the *Mathematica* notebook. Name your file using the last names of both people. For example, if Scott Taylor and Stephanie Arnold work together, the file would be called: TaylorArnold_Proj2.nb

f) Email your file to sataylor@colby.edu