Example A: Approximate the value of $10^{3/2}$ using a linear approximation.  
\[ \text{answer: } \approx 31.5 \]

Can we get a better approximation?

The text explores quadratic and cubic approximations, which you don’t need to know for this class.

We’re going to go to the Newton-Raphson Algorithm, which uses linear approximations in an iterated recursive process, which will give us an approximation to any desired accuracy, i.e. within $\varepsilon$.

Example B: Approximate (5 decimal places) the zero/root of $f(x) = 2x^3 + x^2 - x + 1$.
\[ \text{answer: } \approx -1.23375 \]

There are some limitations of the Newton-Raphson Algorithm:

a) If $f'(x_n) = 0$, then we get division by 0 and the process stops without an approximation.

b) While just about any approximation for $x_0$ will do, it is better to be as close as possible to reduce the number of iterations necessary.

c) The rounding involved creates an error which can sometimes grow with successive steps, thus limiting the overall accuracy. Reducing the number of iterations will help reduce the possible rounding error.
Example C: Approximate the negative-value intersection of \( y = 2^x \) and \( y = x^2 \).

*answer:* \( x \) is approximately \(-0.766664696\)

Example A revisited: Remember that the Newton-Raphson Algorithm will find approximations for the roots of a function, i.e. \( x \)-values for which \( y = 0 \). As a result, we’ll always need to rearrange our information to a convenient form. For this Example, to approximate \( 10^{\frac{3}{2}} \), we’d go through the process below.

\[
x = 10^{\frac{3}{2}} \Rightarrow x^{\frac{2}{3}} = 10 \Rightarrow x^{\frac{2}{3}} - 10 = 0 \Rightarrow \text{Let } f(x) = x^{\frac{2}{3}} - 10.
\]

Example D: Approximate (5 decimal places) a solution to \( x e^x - x^2 = 0 \).

*answer:* approximately 0.65292

Even with a close “first approximation” of 0.6, ten iterations are needed, which is a lot of calculating to enter by hand, and could result in entry errors, which would thoroughly smeg the process. We’ll use a spreadsheet to do the work for us.

[The text does the same thing with a handheld calculator.]

**caveat:** If you are not already somewhat familiar with spreadsheets, this example isn’t going to teach you how to enter the formulae correctly.

In the grid below, column A contains the numbers of iterations. Column B will have \( x_0 \) in row 1, then have the successive \( x \)-values transported from the end of the previous row. Columns C through E have the formulae for calculating \( f(x_n) \), \( f'(x_n) \), and \( x_{\text{new}} \) respectively. In the formulae, “B1”, “C1”, “D2”, etc. refer to the number values from the specified column-row cells. “EXP( )”, like in WebAssign, calculates \( e \) raised to the value in the parentheses. The formulae in columns C and D will change according to the equation being solved, while columns B and E will stay the same for every

\[
y = x - e^{-x^2}
\]

\[
y = x
\]

\[
y = 2^x
\]

\[
y = x^2
\]
Newton-Raphson process.

\[ n \quad c_n \quad f(x) = x - e^{-x^2} \quad f'(x) = 1 + 2xe^{-x^2} \quad c_{n+1} = c_n - \frac{f(c_n)}{f'(c_n)} \]

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>( c_1 ) = B1 - EXP(-1*B1^2)</td>
<td>( =1+2<em>B1</em>EXP(-B1^2) )</td>
<td>=B1-C1/D1</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>( =E1 ) = B2 - EXP(-1*B2^2)</td>
<td>( =1+2<em>B2</em>EXP(-B2^2) )</td>
<td>=B2-C2/D2</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Once the formulae are in row 2, they can be copied down as many rows as needed, and the spreadsheet software will automatically adjust the column-row cell references.

When \( c_0 \) is entered into cell B1, the software will do all the calculating for us.

We can enter various values for \( x_0 \) into cell B1 and observe the changes in the Newton-Raphson process. You won’t have time to copy every number into the chart below during the Lecture. Try it yourself using some spreadsheet software to see if you can get the correct result.

Example E: Approximate (7 decimal places) the intersections of \( y = \cos x \) and \( y = x^2 \).

**answer:** \( x \) is approximately \(-0.8241323\) and \(0.8241323\).

**This one is for you to do on your own for practice. Try both the “by hand” and the spreadsheet methods.**

**Note that if you find the positive-value answer, symmetry gives you the negative-value answer.**

**Starting with an \( x_0 \) between 0 and 1, five iterations should be enough to get you to the correct approximation.**

**Remember that you’ll need to construct a function to set equal to 0.**