• We are starting with the regular subject
• Since we need it for the tp, let’s start with equation solving
• A point $x$ is a root, zero, or $x$-intercept of the function

$$f : \mathbb{R} \rightarrow \mathbb{R}$$

if

$$f(x) = 0$$

• Social Rule of Mathematics ....
• A function may have several roots, we’ll focus on finding one of them.
• Right now we are looking at a single equation, will get to several later.
Bisection

• Suppose \( f(a) \) and \( f(b) \) have opposite signs, and \( f \) is continuous.

• We always assume there is sufficient smoothness.

• Then there must be a root in \((a, b)\).

• Evaluate at 

\[
  c = \frac{a + b}{2}.
\]

Throw away the point \( a \) or \( b \) where \( f(c) \) has the same sign. Repeat. (Stop if you happen to find a \( c \) such that \( f(c) = 0 \).) Stop when \( b - a \) is sufficiently small and take \((a + b)/2\) as your final approximation.

• The error (absolute value of the difference between true and approximate solution) is smaller than \(|(b - a)/2|\) and you know beforehand how many steps you need to go.
Newton’s Method

• Covered in Calculus, quick review.
• $x_0$ given, depends on context
• Define sequence by

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}.$$ 

• Why:

• Stop when $|x_{n+1} - x_n|$ is sufficiently small.
Example: $f(x) = x^2 - 2$.

```maple
> restart:
> Digits:=50:
> f:=x^2-2:
> g:=x-f/diff(f,x):
> xn:=1:
> lprint(sqrt(2.0)):
> 1.4142135623730950488016887242096980785696718753769
> for i from 1 to 8 do
> xn:=evalf(subs(x=xn,g)):
> lprint(i,xn):
> end do:
> 1, 1.500000000000000000000000000000000000000000000000
> 2, 1.4166666666666666666666666666666666666666666666667
> 3, 1.4142156862745098039215686274509803921568627450981
> 4, 1.4142135623746899106262955788901349101165596221157
> 5, 1.4142135623730950488016896235025302436149819257762
> 6, 1.4142135623730950488016887242096980785696718753772
> 7, 1.4142135623730950488016887242096980785696718753770
> 8, 1.4142135623730950488016887242096980785696718753770
> quit
```
Example: \( f(x) = x^3 - 2.\)

```maple
restart:
Digits:=50:
f:=x^3-2:
g:=x-f/diff(f,x):
xn:=1:
lprint(evalf(2.0**(1/3))):
1.2599210498948731647672106072782283505702514647015
for i from 1 to 8 do
  xn:=evalf(subs(x=xn,g)):
lprint(i,xn):
end do:
1, 1.33333333333333333333333333333333333333333333333333333
2, 1.263888888888888888888888888888888888888888888888888888889
3, 1.259934949976966460482943994275159110323945489
4, 1.259921050177697737293010979898432536537097958626
5, 1.2599210498948731647791983238845005280761096256985
6, 1.259921049894873164767210607278228350570365237129
7, 1.2599210498948731647672106072782283505702514647015
8, 1.2599210498948731647672106072782283505702514647015
```

Math 5600 Spring 2021  Notes of the date  page 5
The Secant Method

• Biggest Drawback of Newton’s Method: it requires derivatives

• The secant method uses the secant through the last two points instead of the tangent.

• Approximate:

\[ f'(x_n) \approx \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}} \]

This gives

\[ x_{n+1} = x_n - \frac{f(x_n)}{\frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}} = \frac{x_{n-1}f(x_n) - x_nf(x_{n-1})}{f(x_n) - f(x_{n-1})} \]
Comparison

<table>
<thead>
<tr>
<th>Property</th>
<th>Bisection</th>
<th>Secant</th>
<th>Newton</th>
</tr>
</thead>
<tbody>
<tr>
<td>Starting</td>
<td>$f(a)f(b) &lt; 0$</td>
<td>$x_0, x_1$</td>
<td>$x_0$ close to root</td>
</tr>
<tr>
<td>speed</td>
<td>slow</td>
<td>faster</td>
<td>fastest</td>
</tr>
<tr>
<td>Convergence</td>
<td>yes</td>
<td>usually</td>
<td>usually</td>
</tr>
<tr>
<td>complex numbers</td>
<td>complicated</td>
<td>works</td>
<td>works</td>
</tr>
<tr>
<td>Systems</td>
<td>complicated</td>
<td>very subtle</td>
<td>easy</td>
</tr>
<tr>
<td>data</td>
<td>$f$</td>
<td>$f$</td>
<td>$f, f'$</td>
</tr>
</tbody>
</table>

- All three methods have a starting problem. For the bisection method we need to have two points of opposite signs. Both Newton’s Method and the Secant Method may not converge if we don’t start sufficiently close to the solution.
• Here is a different view of Newton’s Method. It’s given by

\[ x_{n+1} = g(x_n) \quad \text{where} \quad g(x) = x - \frac{f(x)}{f'(x)}. \]

• Clearly

\[ f(x) = 0 \quad \iff \quad x = g(x). \]

• \( x \) is a **fixed point** of \( g \) if \( x = g(x) \). Finding \( x \) such that \( x = g(x) \) is a **Fixed Point Problem**.

• Defining a sequence by

\[ x_{n+1} = g(x_n) \]

is a **fixed point iteration**.

• Newton’s Method is a special case of a fixed point iteration.
• There are many ways to convert a root finding problem to a fixed point problem. Depending on the context you may find an effective way without using derivatives.

• Example: Exercise 9 in hw 1. How does the satellite program compute the satellite data?

• Notation:
  \( t_V \) is the (known) time the vehicle receives the data.
  \( x_V \) is the (known) location of the vehicle.
  \( t_S \) is the (unknown) time the satellite broadcasts.
  \( x_S \) is the location from which the satellite broadcasts. It is a function of \( t_S \):

  \[
x_S(t) = (R + h) \left[ u \cos \left( \frac{2\pi t}{p} + \theta \right) + v \sin \left( \frac{2\pi t}{p} + \theta \right) \right]
  \]

  where the unit vectors \( u \) and \( v \), the period \( p \), and the phase shift \( \theta \) are given in the data file data.dat.

• We need to solve the equation

  \[ F(t_S) = t_V - t_S - \frac{\|x_S(t_S) - x_V\|}{c} = 0 \] (1)

  where \( c \) is the speed of light (also given in the data file).
• The equation (1) can be solved by Newton’s Method (and my satellite uses it).

• A natural starting value is $t_S \approx t_V$.

• Note the problem dependence of this choice!

• Newton’s method works well, but here is a much simpler approach that is just as effective:

• Write $t = t_S$ and rewrite (1) as

$$t = t_V - \frac{\|x_S(t) - x_V\|}{c}$$

• Then iterate

$$t_0 = t_V$$

$$t_{n+1} = t_V - \frac{\|x_S(t_n) - x_V\|}{c}$$

• We want 1 cm accuracy. So stop when

$$c|t_{n+1} - t_n| < 10^{-2}$$

• This is typical. The starting point, the termination criterion, and the actual iteration, all depend on the underlying problem.

• We do not just apply a formula!
Convergence of Fixed Point Iteration

- So when does $x_{n+1} = g(x_n)$ converge?
- The following five drawings illustrate different possible behaviors of fixed point iterations.

![Figure 1. Fixed Point Iteration 1.](image1)

![Figure 2. Fixed Point Iteration 2.](image2)
Figure 3. Fixed Point Iteration 3.

Figure 4. Fixed Point Iteration 4.
Figure 5. $g(x) = 4x(1-x)$. 

• The last example is an instance of what in mathematics is called **chaos**. If you are interested you can take Math 5470, Chaos and Nonlinear Systems, for more info.

• Looking at those pictures, can you figure out what is going on?

• Think about it before Monday’s class!