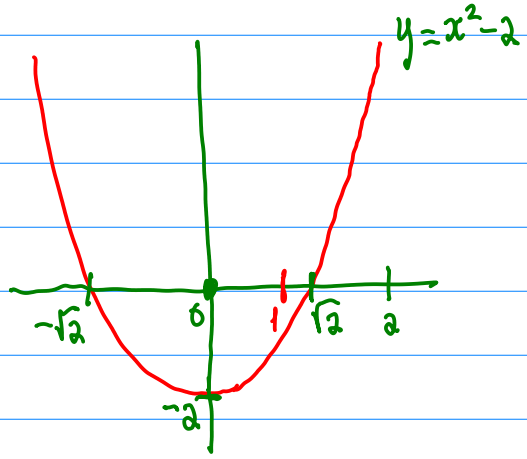
Example: Find $\sqrt{2}$. See this as the root of $x^2 - 2 = 0$

Set $f(x) = x^2 - 2$ and find x's so $f(x) = 0$.



Guess and check method.

Bisection method.

Start with 2 guesses:

a is too small

b is too big

There is $x \in [a,b]$ such that $f(x) = 0$.

$a = 0$ $\quad b = 2$ $\quad x \in [0,2]$ $\quad c = \frac{a+b}{2} = \frac{0+2}{2} = 1$

Check $f(1) = 1^2 - 2 = -1$ negative so c is also too small

$a = 1$ $\quad b = 2$ $\quad x \in [1,2]$ $\quad c = \frac{1+2}{2} = \frac{3}{2}$

check $f\left(\frac{3}{2}\right) = \frac{9}{4} - \frac{8}{4} = \frac{1}{4}$ positive so c is too big

$a = 1$ $\quad b = \frac{3}{2}$ $\quad x \in [1, \frac{3}{2}]$ $\quad c = \frac{1+3/2}{2} = \frac{5}{4}$

check $f\left(\frac{5}{4}\right) = \frac{25}{16} - \frac{32}{16} = \frac{-7}{16}$ too small

$a = \frac{5}{4}$ $\quad b = \frac{3}{2}$ $\quad x \in \left[\frac{5}{4}, \frac{3}{2}\right]$ $\quad c = \frac{5/4 + 6/4}{2} = \frac{11}{8}$

check $f\left(\frac{10}{8}\right)$

```
julia> a=0; b=2;
       for n=1:15
           c=(a+b)/2
           print("[$a,$b] c=$c ")
           if f(c)<0
               println("too small")
               a=c
           else
               println("too big")
               b=c
           end
       end
```

```
[0,2] c=1.0 too small
[1.0,2] c=1.5 too big
[1.0,1.5] c=1.25 too small
[1.25,1.5] c=1.375 too small
[1.375,1.5] c=1.4375 too big
[1.375,1.4375] c=1.40625 too small
[1.40625,1.4375] c=1.421875 too big
[1.40625,1.421875] c=1.4140625 too small
[1.4140625,1.421875] c=1.41796875 too big
[1.4140625,1.41796875] c=1.416015625 too big
[1.4140625,1.416015625] c=1.4150390625 too big
[1.4140625,1.4150390625] c=1.41455078125 too big
[1.4140625,1.41455078125] c=1.414306640625 too big
[1.4140625,1.414306640625] c=1.4141845703125 too small
[1.4141845703125,1.414306640625] c=1.41424560546875 too big
```

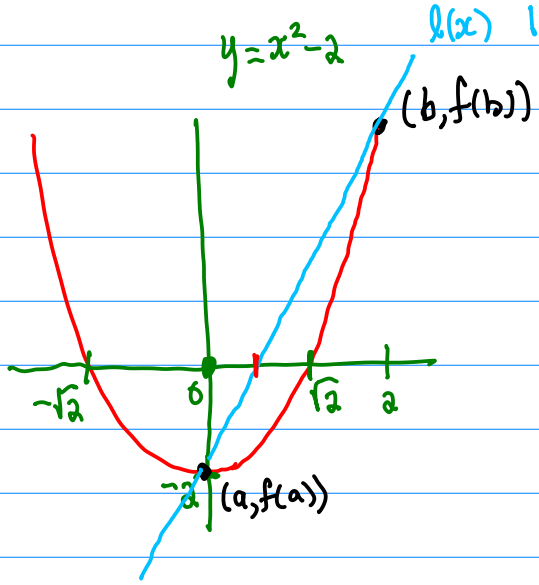
at every stage $x \in [a,b]$ so a, b are upper and lower bounds on x and

$$|c-x| < \frac{1}{2}|b-a|$$

and the distance between a and b halves at each iteration...

Stop after c starts repeating, because that means the interval is not changing due to rounding effects. Expect this to happen after about 53 iterations because Float64 has 53 bits of precision...

Method of false position



$l(x)$ is the line between $(a,f(a))$ and $(b,f(b))$

$$l(x) = \frac{x-b}{a-b} f(a) + \frac{x-a}{b-a} f(b)$$

$$l(x) = 0$$

$$\frac{x}{a-b}(f(a)-f(b)) = \frac{b}{a-b} f(a) - \frac{a}{a-b} f(b)$$

$$x = \frac{b f(a) - a f(b)}{f(a) - f(b)}$$

$a=0 \quad b=2 \quad x \in [0,2] \qquad c = \frac{b f(a) - a f(b)}{f(a) - f(b)} = 1$

$f(1) = 1^2 - 2 = -1 < 0$ too small

$a=1 \quad b=2 \quad x \in [1,2) \qquad c = 1.\overline{3}$

$f(1.\overline{3}) = -.2\overline{2} < 0$ too small

$a = 1.\overline{3} \quad b=2 \quad x \in [1.\overline{3}, 2] \qquad c = 1.4$

$f(1.4) = -.04$ too small

$a = 1.4 \quad b=2 \quad x \in [1.4, 2] \qquad c =$

```
julia> c=findc(a,b)
1.4117647058823533
```

```
julia> findc(a,b)=(b*f(a)-a*f(b))/(f(a)-f(b))
findc (generic function with 1 method)
```

$f(1.4117\ldots) \approx -.0062$ too small...

Only the left endpoint moves, so the error bound has 2 as the upper bound at each iteration. This means that the interval $[a,b]$ that brackets where the root is does not get converge to a length of 0.

Solution if b doesn't move then substitute $\frac{1}{2}f(b)$ into the interpolation formula...

$$\text{findcb}(a,b) = \frac{b f(a) - a\frac{1}{2}f(b)}{f(a) - \frac{1}{2}f(b)}$$ ← moves b (hopefully)

Solution if a doesn't move then substitute $\frac{1}{2}f(a)$ into the interpolation formula...

$$\text{findca}(a,b) = \frac{b\frac{1}{2}f(a) - af(b)}{\frac{1}{2}f(a) - f(b)}$$ ← moves a (hopefully).

This is called the Illinois algorithm...

```
julia> findcb(a,b)=(b*f(a)-0.5*a*f(b))/(f(a)-0.5*f(b))
findcb (generic function with 1 method)

julia> findca(a,b)=(0.5*b*f(a)-a*f(b))/(0.5*f(a)-f(b))
findca (generic function with 1 method)
```

$a=0 \quad b=2 \quad x \in [0,2] \quad c = \frac{b f(a) - a f(b)}{f(a) - f(b)} = 1$

$f(1) = 1^2 - 2 = -1 < 0$ too small a moved

$a=1 \quad b=2 \quad x \in [1,2) \quad c = 1.\overline{33}$

$f(1.\overline{33}) = -.2\overline{2} < 0$ too small a moved

$a = 1.\overline{33} \quad b=2 \quad x \in [1.\overline{33}, 2] \quad c = \text{findcb}(a,b)$

```
julia> a=findc(1,2)
1.3333333333333333

julia> b=2
2

julia> c=findcb(a,b)
1.4545454545454546
```

```
julia> f(c)
0.11570247933884303
```

too big...

$a = 1.\overline{33} \quad b = 1.\overline{45} \quad x \in [1.\overline{33}, 1.\overline{45}] \quad c =$

```
julia> c=findc(a,b)
1.4130434782608694
```

```
julia> f(c)
-0.0033081285444238606
```

too small

$a = 1.413\ldots \quad b = 1.\overline{45} \quad x \in [1.413\ldots, 1.\overline{45}]$

```
julia> c=findc(a,b)
1.4141971054445208
```

```
julia> f(c)
-4.654695233896611e-5
```

too small

$a = 1.414197\ldots$ $b = 1.\overline{45}$ $x \in [1.414197\ldots, 1.\overline{45}]$

$c =$

```
julia> c=findcb(a,b)
1.4142295435184464
```

```
julia> f(c)
4.520176039335411e-5
```

too big

$a = 1.414197\ldots$ $b = 1.414229\ldots$ $x \in [1.414197\ldots, 1.414229\ldots]$,

This is converging much faster than bisection and much faster than false position without the Illinois modification.