Maple Demonstration - Year 12

4th March 2014 - David Wilson (D.J.Wilson@bath.ac.uk)„ research student of James Davenport (J.H. Davenport@bath.ac.uk)

Introduction to Maple

Before we start, let's make sure Maple is ready to go. Restart means all variables are cleared and we are starting 'fresh'. We have to enter "math mode" (where your cursor is surrounded by a dashed box) by pressing F5, and we use a semi-colon to finish the command. We then press enter to run the command:
restart;

What is Maple? To start, it's a calculator: $1 + 1 =$. We press ctrl+= to compute something inline (the answer appears in blue), or press enter to get the output on a new line.
$2^{1000} - 1$;

10715086071862673209484250490600018105614048117055336074437503883703510511249\
61224931983788156958581275946729175531468251871452856923140435984577574698\
57480393456777482423098542107460506237114187795418215304647498358194126739\
8767559165543946077062914571196477686542167660429831652624386837205668069375 **(1)**

But it can do a lot more than a calculator! Maple treats everything symbolically so it can deal with more than just numbers. It can do algebra:
$expand\left((x+1)^3 \cdot (x-1)^3 - (x-2)^2\right);$

$$x^6 - 3x^4 + 2x^2 + 4x - 5 \tag{2}$$

$factor\left(x^{10} - 1\right);$

$$(x-1)\ (x+1)\ \left(x^4 + x^3 + x^2 + x + 1\right)\ \left(x^4 - x^3 + x^2 - x + 1\right) \tag{3}$$

We can also solve equations, both with exact solutions and in terms of other variables:
$solve\left(x^4 - 10x^3 + 35x^2 - 50x + 24\right);$

$$1, 2, 3, 4 \tag{4}$$

$solve\left(a \cdot x^2 + b \cdot x + c, x\right);$

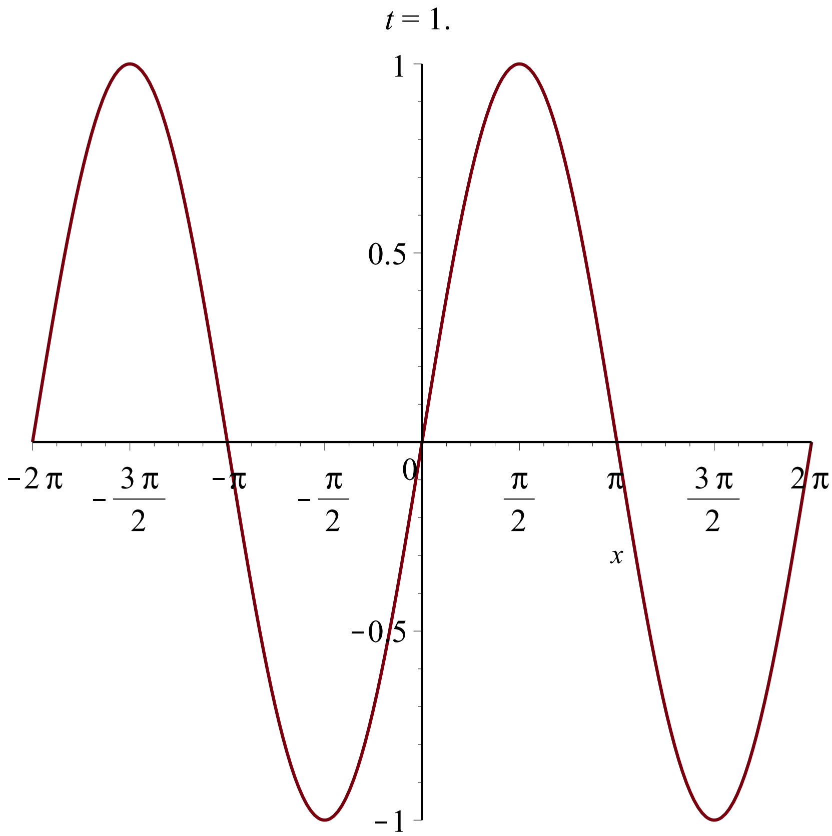
$$\frac{1}{2}\ \frac{-b + \sqrt{-4ac + b^2}}{a}, -\frac{1}{2}\ \frac{b + \sqrt{-4ac + b^2}}{a} \tag{5}$$

Be careful though! Maple will solve **exactly** what you ask it to. For example, if you write ab and not a*b, it will think ab is a new variable!
$is(ab = a \cdot b);$

$$false \tag{6}$$

$solve\left(ax^2 + bx + c, x\right);$
The lack of output means Maple cannot find any solutions - mainly because Maple thinks x doesn't occur as a variable.

Maple is also really useful for visualising mathematics. It can do cool things like animated graphs to see how a parameter changes a function:
plots:-animate(*plot*, [sin(x·t), x =−2·Pi ..2·Pi], t = 1 ..10);



More on that later.

To define a variable we use the := symbol

$x := 5;$

$$5 \tag{7}$$

And we can check what the value of x is:

$x;$

$$5 \tag{8}$$

Careful! Now x will be 5 unless you undefine it:

$f := x^3 + \mathrm{x} - 7;$

$$123 \tag{9}$$

So we have to redefine x to 'reset' it

$x :=\text{'}x\text{'};$

$$x \tag{10}$$

And now we can define f as we want

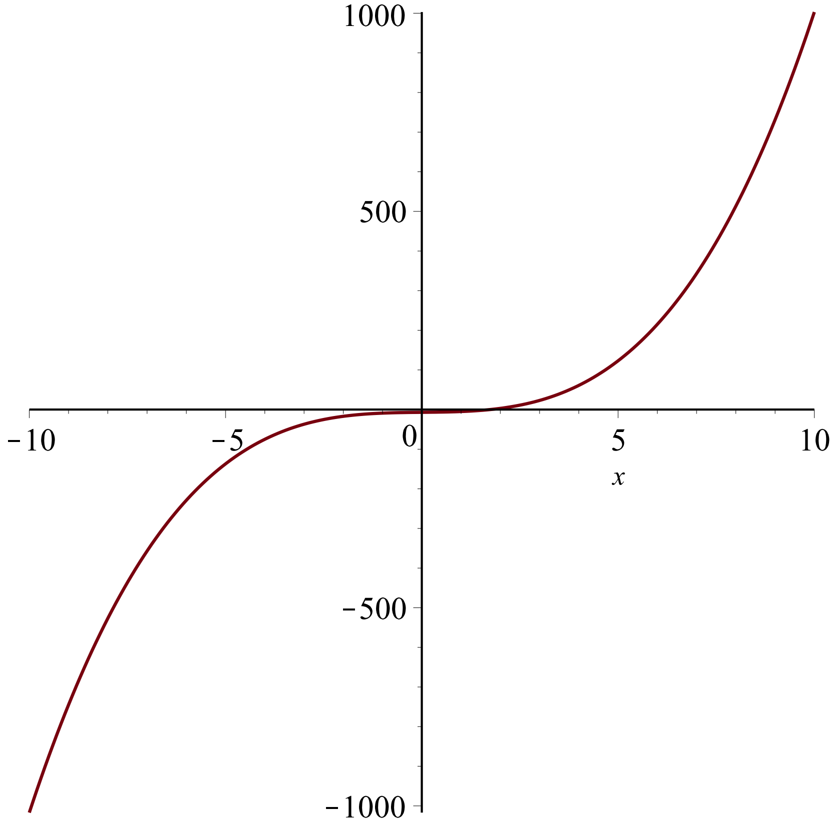
$f := x^3 + \mathrm{x} - 7;$

$$x^3 + x - 7 \tag{11}$$

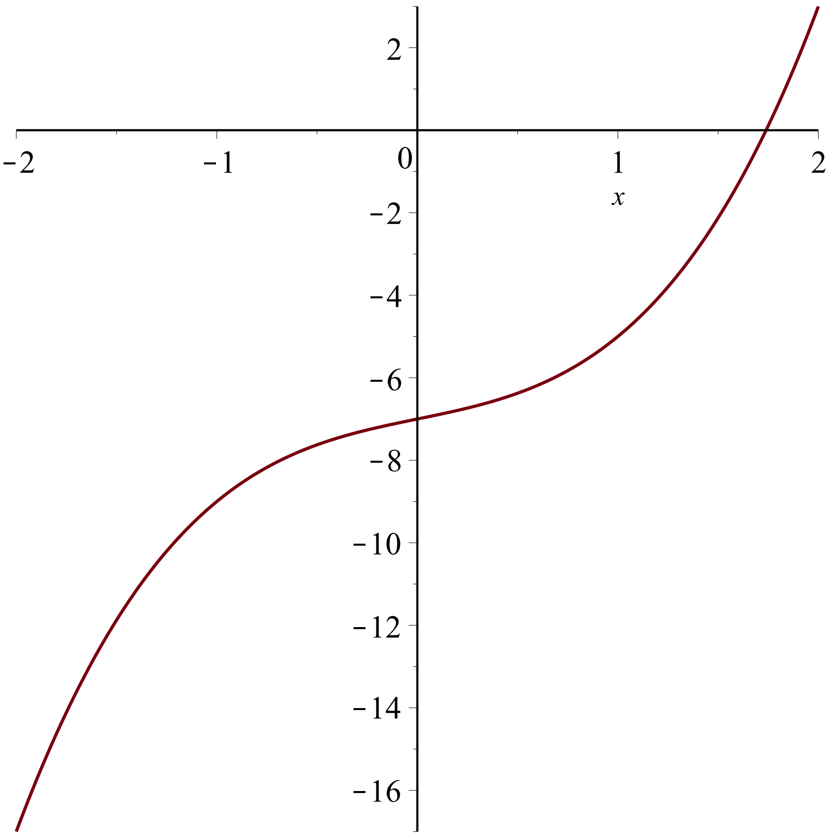
Notice that we have had a semi-colon at the end of each line. If we use a full colon then the command runs, but we don't see any output:

$f := x^3 + x - 7 :$

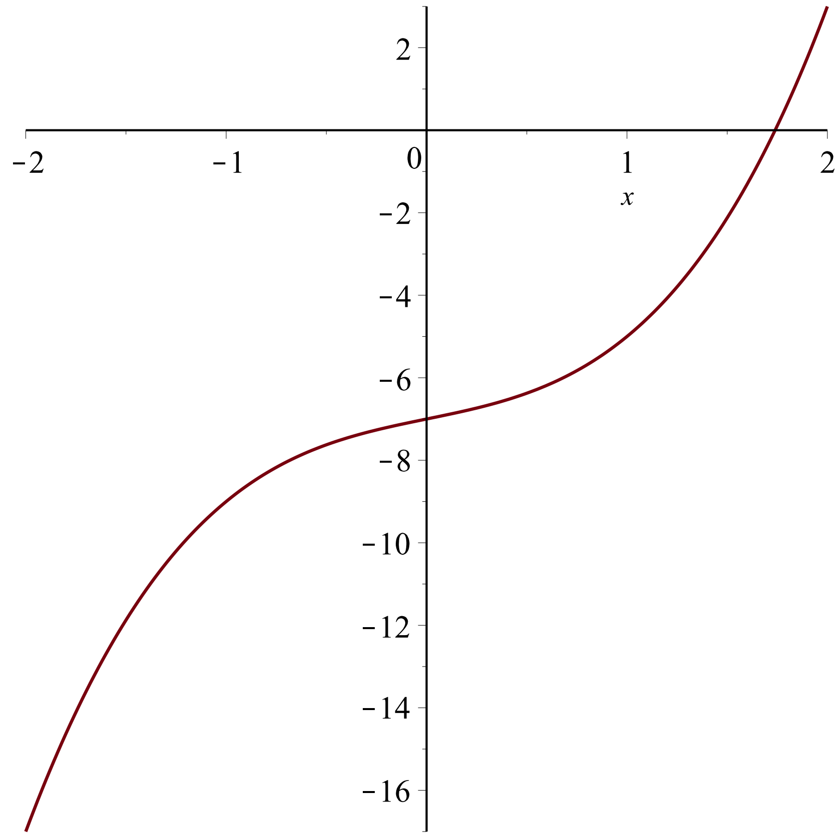
Now let us see what f looks like
$plot(f);$



This is a bit of a wide zoom - we can't see where it intersects the x-axis. Let's check by specifying what we want to see.
$plot(f, x=-2..2)$



2
−2
−4
−6
−8
−10
−12
−14
−16
−2
−1
0
1
2
x



We can also click on the zoom tool to zoom in by dragging an area, or using the scroll button on the mouse.

We can do a lot more with a function than just plot it or solve it. We can differentiate it:
diff (*f*, *x*);

$$3\,x^2 + 1 \tag{12}$$

Or we can integrate it.
int(*f*, *x*);

$$\frac{1}{4}\,x^4 + \frac{1}{2}\,x^2 - 7\,x \tag{13}$$

$intvalue := int\left(f, x = \frac{\text{sqrt}(2)}{3} \,..5\cdot\text{Pi}\right);$

$$\frac{625}{4}\,\pi^4 - \frac{10}{81} + \frac{25}{2}\,\pi^2 - 35\,\pi + \frac{7}{3}\,\sqrt{2} \tag{14}$$

Notice that Maple gave us an exact answer. It will always try to give an exact answer when it can. We can see the value of this by using the evalf command.
evalf (*intvalue*)

$$15236.76117 \tag{15}$$

And we can get a numerical answer to as many digits as we want:
evalf (Pi, 1000)

```
3.1415926535897932384626433832795028841971693993751058209749445923078164062862\
   0899862803482534211706798214808651328230664709384460955058223172535940812848\
   1117450284102701938521105559644622948954930381964428810975665933446128475648\
   2337867831652712019091456485669234603486104543266482133936072602491412737245\
   8700660631558817488152092096282925409171536436789259036001133053054882046652\
   1384146951941511609433057270365759591953092186117381932611793105118548074462\
   3799627495673518857527248912279381830119491298336733624406566430860213949463\
   9522473719070217986094370277053921717629317675238467481846766940513200056812\
   7145263560827785771342757789609173637178721468440901224953430146549585371050\
   7922796892589235420199561121290219608640344181598136297747713099605187072113\
   4999999837297804995105973173281609631859502445945534690830264252230825334468\
   5035261931188171010003137838752886587533208381420617177669147303598253490428\
   7554687311595628638823537875937519577818577805321712268066130019278766111959\
   09216420199
```
(16)

There are lots of useful assistants for Maple which can be found under the Tools option in the menu bar. Assistants help you create things in Maple (plots, equations etc), Math Apps show you different concepts, Tutors help you understand how to do certain things in Maple, and Tasks are templates to do certain things.

The best place to find help is **Maple Help.** You can find it under Help in the toolbar.

Some 'Programming'

You can also 'program' with Maple. For example we can have a look at the first 100 Fibonacci numbers. You can create a list using [] and access elements by using [] too. We fill it with a 'for' loop:

```
FibNums := [0$100] :
FibNums[1] := 1 :
 FibNums[2] := 1 :
for i from 3 to 100 do
  FibNums[i] := FibNums[i - 2] + FibNums[i - 1] :
 end do:
```

This has filled the list with the Fibonacci numbers and we can look at the final element to see the 100th Fibonacci Number

$FibNums[100]$

$$354224848179261915075 \quad \textbf{(17)}$$

We can also create 'procedures' which are functions to do things in Maple. If you are doing something over and over then creating a procedure can be helpful.

```
myProc := proc(x)
   local i, runningtotal :
   runningtotal := 0 :
   for i from 1 to 10 do:
     runningtotal := runningtotal·x + x :
   end do:
   return runningtotal :
 end proc:
myProc(9)
```

$$3922632450 \quad \textbf{(18)}$$

Numerical Analysis

For your coursework you will be looking at a variety of Numerical Analysis methods to find roots. This means we need the Numerical Analysis package - this is a collection of functions all related to Numerical Analysis and it is in the Student package. We load a Maple package using 'with':

$with(Student)$

$[Basics, Calculus1, LinearAlgebra, MultivariateCalculus, NumericalAnalysis, Precalculus, SetColors, SetDefault, SetDefaults, Statistics, VectorCalculus]$ **(19)**

$with(NumericalAnalysis);$

$[AbsoluteError, AdamsBashforth, AdamsBashforthMoulton, AdamsMoulton, AdaptiveQuadrature, AddPoint, ApproximateExactUpperBound, ApproximateValue, BackSubstitution, BasisFunctions, Bisection, CubicSpline, DataPoints, Distance, DividedDifferenceTable, Draw, Euler, EulerTutor, ExactValue, FalsePosition, FixedPointIteration, ForwardSubstitution, Function, InitialValueProblem, InitialValueProblemTutor, Interpolant, InterpolantRemainderTerm, IsConvergent, IsMatrixShape, IterativeApproximate, IterativeFormula, IterativeFormulaTutor, LeadingPrincipalSubmatrix, LinearSolve, LinearSystem, MatrixConvergence, MatrixDecomposition, MatrixDecompositionTutor, ModifiedNewton, NevilleTable, Newton, NumberOfSignificantDigits, PolynomialInterpolation, Quadrature, RateOfConvergence, RelativeError, RemainderTerm, Roots, RungeKutta, Secant, SpectralRadius, Steffensen, Taylor, TaylorPolynomial, UpperBoundOfRemainderTerm, VectorLimit]$ **(20)**

There's lots of functions! We want to have a look at the Bisection method so let's open up the help page. We can do this by clicking on Help in the menu bar, or by using the question mark

$?Bisection$

There's a lot of information on the help page, but most importantly it tells us how to call the function and gives us examples.

We try it with the most basic inputs: the function and the range we want to search. We will use $f = x^3 + x - 7$:

$Bisection(f, x = [1.5, 2])$

$$1.739135742 \quad \textbf{(21)}$$

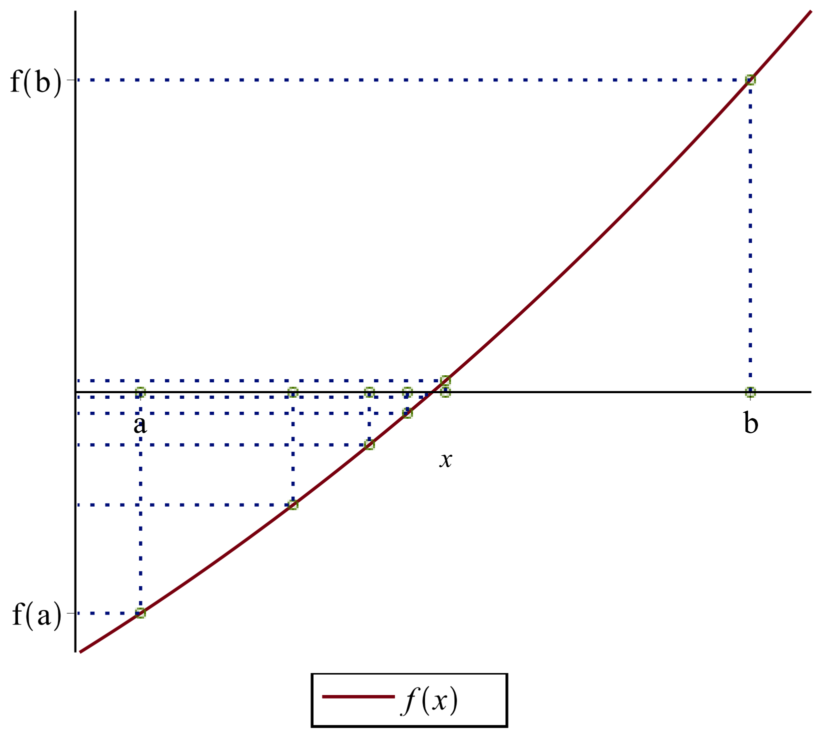
That's useful, but we could do with more information. Let's add some options, asking for it to gives us the sequence of intervals it used

$Bisection(f, x = [1.5, 2], output = sequence)$

$[1.5, 2.], [1.5, 1.750000000], [1.625000000, 1.750000000], [1.687500000, 1.750000000], [1.718750000, 1.750000000], [1.734375000, 1.750000000], [1.734375000, 1.742187500], [1.738281250, 1.742187500], [1.738281250, 1.740234375], [1.738281250, 1.739257812], [1.738769531, 1.739257812]$ **(22)**

And we can ask for a graph of the intervals

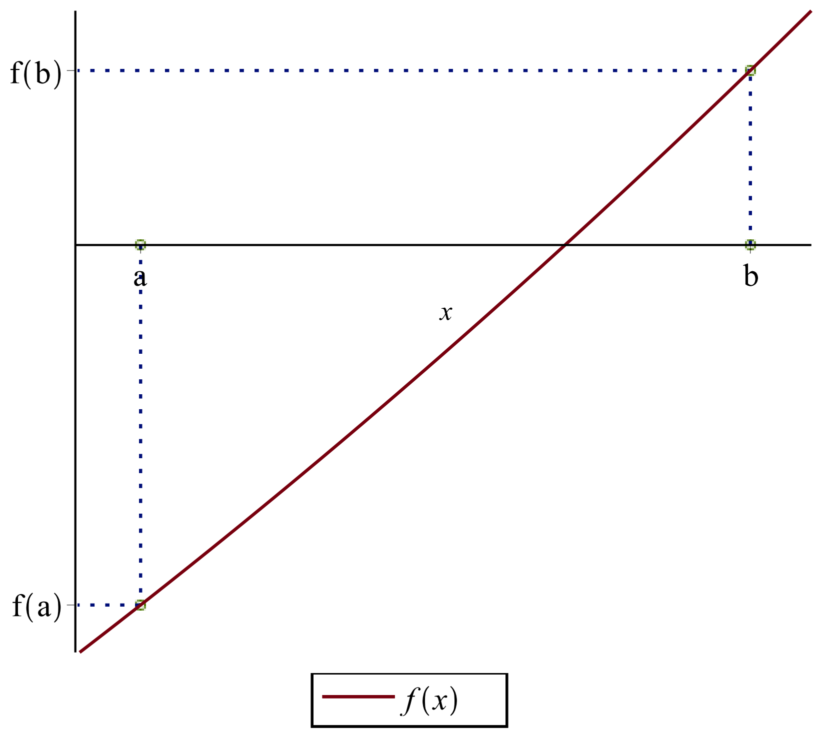
$Bisection(f, x = [1.5, 2], output = plot)$



5 iteration(s) of the bisection method applied to $f(x) = x^3 + x - 7$ with initial points $a = 1.5$ and $b = 2.$. The stopping criteria is not met.

We can even get it to be animated and show us each step!

$Bisection(f, x = [1.6, 1.8], output = animation)$



10 iteration(s) of the bisection method applied to
$f(x) = x^3 + x - 7$ with initial points $a = 1.6$ and $b = 1.8$. The stopping criteria is not met.

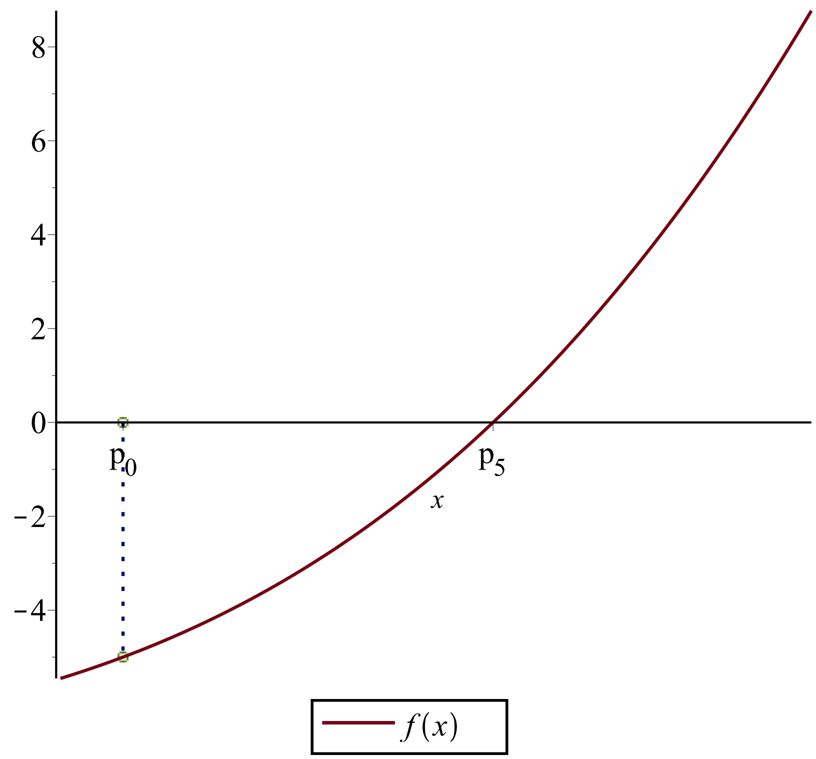
There are other methods we can use to approximate roots: let's check out the Newton-Raphson method, which is called Newton in Maple. We can search for it in Help but we need to make sure
?*Newton*
This has a similar input to Bisection, but instead of giving it an interval we now give it an approximate location for the root of the function.
$Newton(f, x = 1.5)$

$$1.739203861 \tag{23}$$

We can also ask for a sequence, this time of numbers:
$Newton(f, x = 1, output = sequence)$

$$1., 2.250000000, 1.839768340, 1.744116869, 1.739216322, 1.739203861 \tag{24}$$

And we can plot or animate the output
$Newton(f, x = 1, output = animation)$



5 iterations of Newton's method applied to $f(x) = x^3 + x - 7$
with initial point $p_0 = 1.$.

If we wanted to do the Newton-Raphson method by hand we would need to compute the tangent line. We can find the slope of this by differentiating f with respect to x
df := *diff*(*f*, *x*)

$$3x^2 + 1 \tag{25}$$

Then we can evaluate this derivative f at whatever value we need - say 1.75
eval(*df*, *x* = 1.5)

$$7.75 \tag{26}$$

Another function we might want to look at is Fixed Point Iteration
?*FixedPointIteration*

Visualising Functions

How can we easily visualise functions? There's a large plots package:
with(*plots*);

[*animate*, *animate3d*, *animatecurve*, *arrow*, *changecoords*, *complexplot*, *complexplot3d*, *conformal*, *conformal3d*, *contourplot*, *contourplot3d*, *coordplot*, *coordplot3d*, *densityplot*, *display*, *dualaxisplot*, *fieldplot*, *fieldplot3d*, *gradplot*, *gradplot3d*, *implicitplot*, *implicitplot3d*, **(27)**

$inequal, interactive, interactiveparams, intersectplot, listcontplot, listcontplot3d, listdensityplot, listplot, listplot3d, loglogplot, logplot, matrixplot, multiple, odeplot, pareto, plotcompare, pointplot, pointplot3d, polarplot, polygonplot, polygonplot3d, polyhedra\_supported, polyhedraplot, rootlocus, semilogplot, setcolors, setoptions, setoptions3d, spacecurve, sparsematrixplot, surfdata, textplot, textplot3d, tubeplot]$

Let's create a new cubic f, where we don't know what a,b,c and d are.

$f := a \cdot x^3 + b \cdot x^2 + c \cdot x + d$

$$a\,x^3 + b\,x^2 + c\,x + d \tag{28}$$

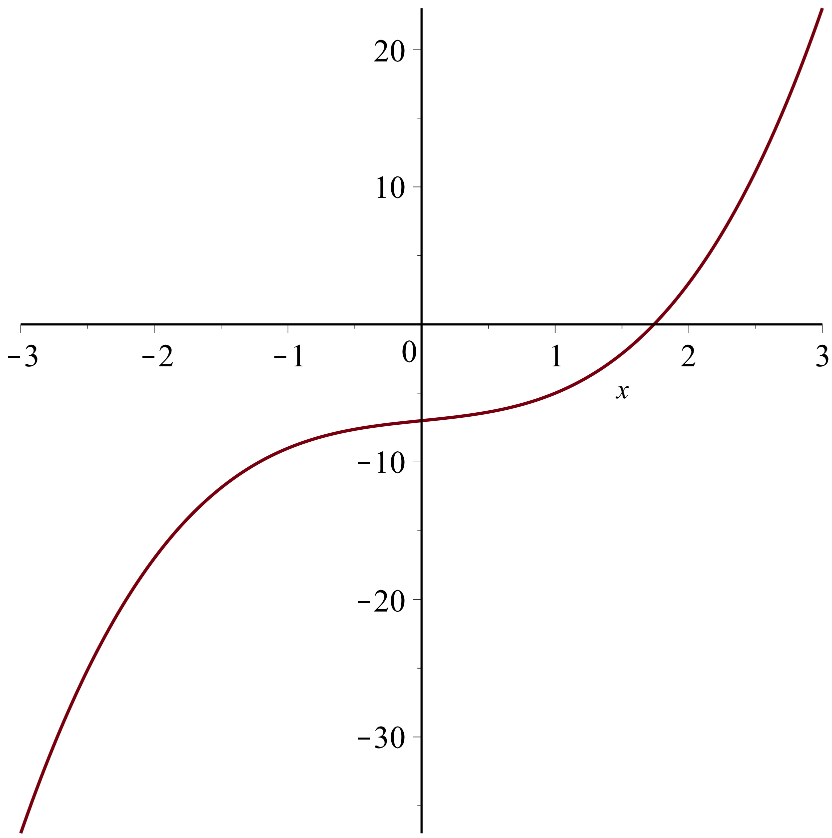
We can substitute in values for this using subs. For example to get x^3+x-7 again:

$subs(a = 1, b = 0, c = 1, d = -7, f)$

$$x^3 + x - 7 \tag{29}$$

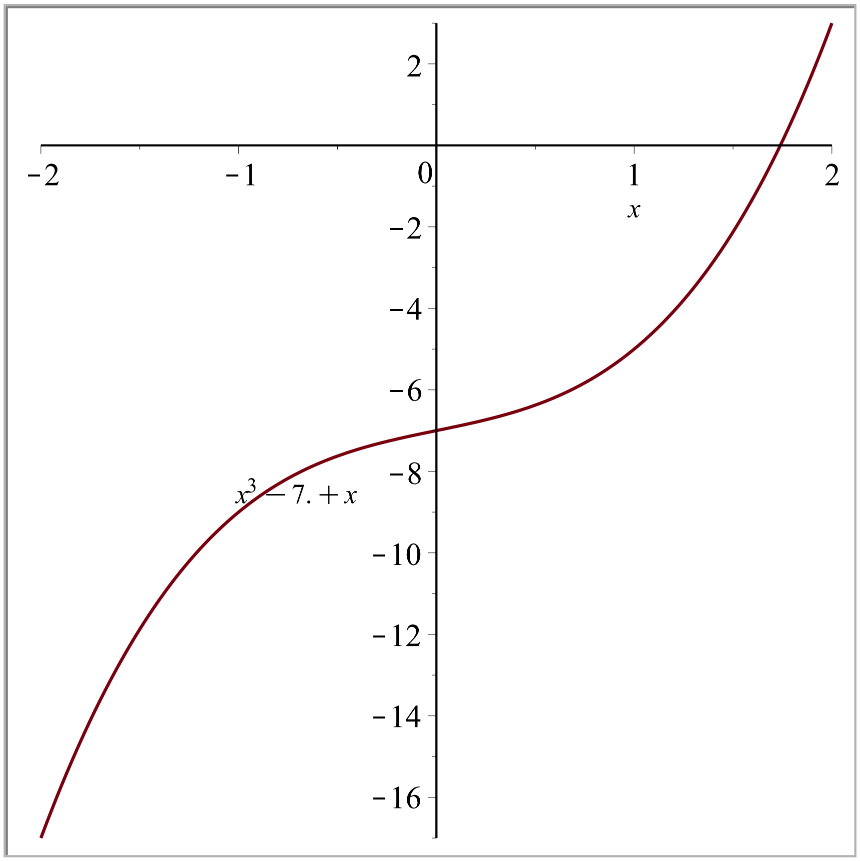
And we can plot this substitution using the basic plot command, with options to make it clearer:

$plot(subs(a = 1, b = 0, c = 1, d = -7, f), x = -3 ..3)$

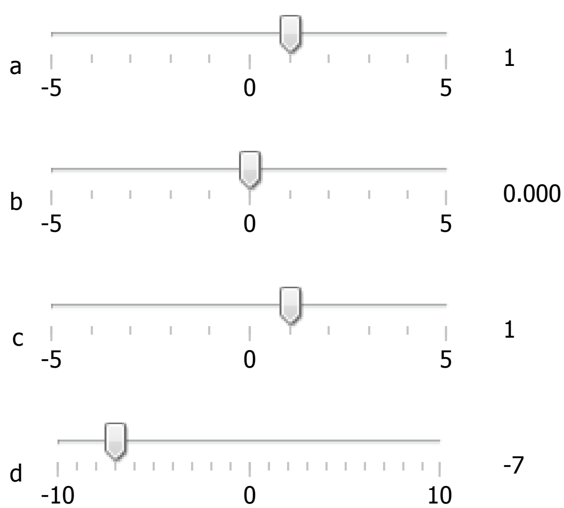


We can use the Plot Builder under Tools->Assistants to help explore what cubics look like. Here's one I made earlier

$Explore\left(plot\left(a\,x^3 + b\,x^2 + c\,x + d, x = -2 ..2, labels = \left[x, a\,x^3 + b\,x^2 + c\,x + d\right]\right), 'parameters' = [a = -5 ..5, b = -5 ..5, c = -5 ..5, d = -10 ..10], 'initialvalues' = [a = 1, b = 0., c = 1, d = -7]\right)$



2
−2
−1
0
1
2
x
−2
−4
−6
−8
x3 − 7. + x
−10
−12
−14
−16



a
-5
0
5
1
b
-5
0
5
0.000
c
-5
0
5
1
d
-10
0
10
-7