Bisection Method

We begin by using the class maple library.

```maple
restart;
luname:="c:/nalib",libname;
libname := "/nalib", "/Library/Frameworks/Maple.framework/Versions/15/lib",
"/Library/Frameworks/Maple.framework/Versions/15/toolbox/NAG/lib"
with(numanal);
[SOR, SOR_dir, adaptq, adaptq_dir, bezier, bezier_dir, bisection, bisection_dir, chop, chop_dir,
clamped_spline, clamped_spline_dir, divided_diff, divided_diff_dir, extrap, extrap_dir,
falseposition, falseposition_dir, fixedpoint, fixedpoint_dir, gaussseidel, gaussseidel_dir, hermite,
hermite_dd, hermite_dd_dir, hermite_dir, horner, horner_dir, jacobi, jacobi_dir, muller,
muller_dir, natural_spline, natural_spline_dir, newton, newton_dir, romberg, romberg_dir,
secant, secant_dir, steffensen, steffensen_dir]
```

Suppose we wish to find all the roots of the following function accurate to within $10^{-6}$.

```maple
> f:=4*x^3-20*x^2+3*x+2+ln(x); 
f := 4 x^3 - 20 x^2 + 3 x + 2 + \ln(x)
```

We plot the graph.

```maple
> plot(f,x=0..5);
```
We see we have a root near 4.8. On the other hand, it is difficult to see what is happening near 0, so we change our plot window.

> plot(f, x=0..1);
Now we see we also have roots near .12 and .3. We use the procedure \textbf{bisection} to find these roots. First, let's look at the directions for \textbf{bisection}.

\texttt{bisection} returns a root of the given function.

The arguments for \texttt{bisection} are:

1. function expression in \texttt{x}
2. left end point
3. right end point
4. tolerance
5. maximum number of iterations
6. variable for returning root.

If assigning the result to a variable, have the variable and the 6th argument the same.

If \texttt{r} is the variable for returning the root and has already been given a value, the procedure should be preceded by the statement: \texttt{r:='r'}

We first find the root near .12. It is the only root between .1 and .14.

\texttt{> r1:=bisection(f,.1,.14,.000001,100,r1);}
The approximate solution is \( r_1 = 0.12649353 \) with \( f(r_1) = 0.00000011 \)

The second root is the only one between .2 and .4.

The approximate solution is \( r_2 = 0.30087509 \) with \( f(r_2) = -0.00000336 \)

The final root is the only one between 4 and 5.
The approximate solution is \( r_3 = 4.80527210 \) with \( f(r_3) = -0.00003702 \)

Let's see what happens if we enter our bracketing numbers in the opposite order.

```
> r3:=bisection(f,5,4,.000001,100,r3);
Error, invalid input: bisection expects its 6th argument, root, to be of type name, but received 4.805272102
```

Looks like we forgot our directions.

```
> r3:='r3';
r3 := r3
> r3:=bisection(f,5,4,.000001,100,r3);
```

```
The approximate solution is \( r_3 = 4.80527210 \) with \( f(r_3) = -0.00003702 \)
```

```
r3 := 4.805272102
```

Looks like the procedure can take the endpoints in any order. Let's see what happens if we look for a root in the interval \([.2, 5]\).

```
> r:=bisection(f,.2,5,.000001,100,r);
Error, (in bisection) The function values at a and b must have opposite signs
```

The procedure watches out for us. It is important to choose starting intervals that contain a single root.
There is also a Bisection command in the Student[NumericalAnalysis] package.

```maple
> with(Student): with(NumericalAnalysis):
We use the Bisection command on the above problem, looking for the root between 4 and 5. For just the root, we use

```maple
> Bisection(f,x=[4,5],stoppingcriterion=absolute,tolerance=10^(-6));
```  
For the sequence of intervals along with the solution:

```maple
> Bisection(f,x=[4,5],stoppingcriterion=absolute,tolerance=10^(-6), output=sequence,maxiterations=100);
```  
For even more information:

```maple
> Bisection(f,x=[4,5],stoppingcriterion=absolute,tolerance=10^(-6), output=information,maxiterations=100);
```  
For a plot of what is going on:

```maple
> Bisection(f,x=[4,5],stoppingcriterion=absolute,tolerance=10^(-6), output=plot,maxiterations=100);
```  
For a plot that can be animated:

```maple
> Bisection(f,x=[4,5],stoppingcriterion=absolute,tolerance=10^(-6), output=animation,maxiterations=100);