

An Introduction to *Maple*:

Help, Expressions, Functions, Substitutions, Solving Equations

Bob Bradshaw, Ohlone College

- Maple or Mathematica?

Both *Maple* and *Mathematica* are excellent software packages, with strengths and weaknesses. Learning one package will help you understand both packages. Developing a working knowledge of either program is well within the abilities of the interested student. Becoming a master at either package will require both time and effort.

- Help

Before starting to use *Maple*, the first thing is to know how to get help. There is a HELP menu on your screen but there is an easier way.

If you are looking for help with a particular command, simply type "*?command name*" For example

```
> ?solve
```

- Restart

It sometimes becomes necessary to clear the *Maple* memory. Do this by typing **restart**; Note that this does not completely release allocated memory. To do so, quite *Maple* and reboot the program.

- Syntax

Maple uses lower case letters for most, but not all, commands and uses parentheses for the arguments of functions. Note it is required to capitalize **Pi**

```
> sin(Pi/4);
```

Parentheses are used for both arguments of functions and for grouping symbols. Square brackets are used for vectors and curly braces are used for sets.

A semicolon is required at the end of each statement unless you want to suppress the output. In that case, use a colon.

```
> 3+4;  
3*4;
```

Multiplication requires a multiplication symbol in all instances and forgetting it is a frequent error.

```
> 3x;
```

- Expressions

Expressions are the types of equations you normally see in a math course, such as the following. Notice that *Maple* requires a **:=** in the assignment statement.

```
> y:=x^2+3;
```

Substituting a value for x gives

```
> x:=3;y;
```

However, trying to use this syntax with standard math notation gives the incorrect answer.

```
> y(2);
```

```
[ > y(7);
```

You must explicitly define the new value of x.

```
[ > x:=1;y;
```

If you ask for the value of y, the program returns the last value rather than the equation itself.

```
[ > y;
```

To return to the equation for y rather than the last numerical value, use

```
[ > x:='x';y;
```

You can also use the subs command to perform a substitution.

```
[ > subs(x=r^2,y);
```

[-] Functions

Functions are declared using the notation *name:=variable -> equation*. For example

```
[ > z:=x->x^2+7;
```

This now works using standard mathematical notation

```
[ > z(2);
```

To label a certain value, use

```
[ > z(5):=z(5);
```

[-] Numerical Values

Since *Maple* performs exact arithmetic, you must tell it to give a decimal result if that is what you are looking for. There are a couple of ways. The usual way is to use the *evalf* command.

```
[ > myfractions:=1/3+7/17;
```

```
[ > evalf(myfractions);
```

The *evalf* command can be used to get the first 50 decimal places of Pi.

```
[ > evalf(Pi,50);
```

Another way is to include a decimal in the original problem. However this does not always work.

```
[ > 1./3+7/17;
```

```
[ > Pi/4;
```

```
[ > Pi/4.;
```

```
[ >
```

[-] Solving Equations

Maple will solve equations with one or more variables and, if possible, return an exact answer.

```
[ answer1:=solve(a*x+b=c,x);
```

Maple will also return multiple answers where appropriate.

```
[ answer2:=solve(x^2-5*x-7=0,x);
```

To verify that these are solutions, we can use the *subs* command to substitute the values back into the equation. The solutions need to be checked one at a time.

```
[ > checkthis1:=subs(x=answer2[1],x^2-5*x-7);
```

```
[ checkthis2:=subs(x=answer2[2],x^2-5*x-7);
```

```
[ > simplify(checkthis1);
```

```
[ simplify(checkthis2);
```

If you want a decimal value, use the "fsolve" command.

```
[ > answer:=fsolve(x^2-5*x-7=0,x);
```

It is also possible to specify the interval containing the answer.

```
> eqn:=x^3-8*x^2-35*x+150;  
answer:=fsolve(eqn=0,x);  
posanswer:=fsolve(eqn=0,x=0..infinity);  
neganswer:=fsolve(eqn=0,x=-infinity..0);
```

When there are multiple answers, you can access the one you want by using *name*[*answer number*]

```
> answer[2];
```

In some cases, *Maple* is unable to give the exact solution. Instead, it leaves the answer in the form "RootOf"

```
> first:=solve(x^5-x^3+2*x-9=0,x);
```

Use the *evalf* command to get the decimal values.

```
> evalf(first);
```

In this instance, using the "fsolve" does not give all the solutions.

```
> answer:=fsolve(x^5-x^3+2*x-9=0,x);
```

It is also possible solve a system of equations in *x* and *y*. First, reset the variables *x* and *y* and then write the equations.

```
> x:='x';y:='y';  
e1:=3*x+5*y=1;  
e2:=2*x-8*y=11;
```

Solve the equations with the solve command. Be sure to enclose both the equations and the variables in set notation { }.

```
> answer:=solve({e1,e2},{x,y});
```

To use the value of *x* should be straight forward but, unfortunately, simply typing "*x*" will not give the desired value.

```
x;
```

Instead, you must use the "assign" command on your solution.

```
> assign(answer);x;
```

```
>
```

You Try It!

1. Create an expression $y = x^2 + 6x - 81$. Set $y = 0$ and solve for both exact and decimal values of x .

2. Determine the values of x in the interval $1 < x < 12$ that are solutions to $2x^5 - 29x^4 - 108x^3 + 1611x^2 - 3240x = 0$

3. Create the following function and find the value of $f(x+h) - f(x)$
 $f(x) = x^2 + 6x + 9$

4. Solve the system of equations
 $3x + 5y - 8z = 10$

$$\begin{cases} 2x - 8y + z = 12 \\ x - 3y + 7z = 13 \\ > \end{cases}$$