

Section 4: Solving Equations

In this section you will learn how to apply Maple's **solve** command to find the exact solutions of equations (when this is possible). You may recall from Precalculus that we are not able in many cases to find exact solutions to equations and so we rely on numerical solvers to find approximate solutions. Later in this section you will use Maple's **fsolve** command to find decimal approximations for solutions. The solution to linear systems of equations will also be discussed.

```
> restart;  
> with(plots):  
>
```

Entering and Manipulating Equations: The *lhs* and *rhs* commands

Example 1

Recall that we can give a name to an entire equation just as we have done for expressions.

On the next line we enter the equation $x^3 - 5x^2 + 23 = 2x^2 + 4x - 8$ and give it the name "eqn1".

```
> eqn1 := x^3-5*x^2+23=2*x^2+4*x-8;
```

Example 2

We can isolate the left-hand and right-hand sides of the equation by using the **lhs** and **rhs** commands.

```
> lhs(eqn1);
```

```
> rhs(eqn1);
```

Example 3

Use the **lhs** and **rhs** commands to find an equation that is equivalent to the original equation eqn1 but has zero on the right-hand side. Label the new equation eqn2.

```
> eqn2 := lhs(eqn1)-rhs(eqn1)=0;
```

Finding Exact Solutions: The *solve* command

We first consider polynomial equations. Algorithms exist for calculating the exact solutions for polynomial equations up to degree 4. Maple's **solve** command implements these algorithms.

Example 1

To find the exact solutions to the polynomial equation $3x^3 - 4x^2 - 43x + 84 = 0$ use the **solve** command.

Note that the second argument of the command tells Maple that x is the unknown variable that we are solving for.

```
> solve(3*x^3-4*x^2-43*x+84=0, x);
```

```
>
```

Here Maple has found all three solutions and listed them for you.

```
>
```

Example 2

Sometimes you will want to select one solution from the list of solutions and use it in another computation.

You can do this by first assigning a name (we use the letter N in this case) to the output of the **solve** command.

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

```
>
```

Then N[1] is the first number in the expression sequence, N[2] is the second number and so on. Note the square brackets.

```
> N[1];
```

```
>
```

The order of the solutions in such a sequence is not necessarily unique. It might change if the command is re-executed. Hence, it is not a good strategy to reference solutions by their position in the sequence output by **solve**. A more effective strategy will be demonstrated in the section "Solving a Linear System of Equations Using the **solve** Command."

```
>
```

Example 3

When working with the **solve** command it is often convenient to begin by giving a name to the equation.

Note we use **:=** to assign the name and just "=" for the equation itself.

```
> eqn1 := 7*x^3-11*x^2-27*x-9=0;
```

```
>
```

Next we solve the equation using the **solve** command assigning the name H to the output.

```
> H := solve(eqn1, x);
```

```
>
```

For practice let's check that each of these values satisfies the equation. This is easy to do using the **eval** command.

```
> eval(eqn1, x=H[1]);
```

```
eval(eqn1, x=H[2]);
```

```
eval(eqn1, x=H[3]);
```

```
>
```

Warning: the solutions obtained from the **solve** command may not be in a unique order. A more effective strategy will be demonstrated in the section "Solving a Linear System of Equations Using the **solve** Command."

>
▼ **Example 4**

Sometimes the "exact" solutions are too cumbersome to be of much use. In the next two lines we solve the equation $x^3 - 34x^2 + 4 = 0$.

```
> eqn1 := x^3-34*x^2+4=0;  
H := solve(eqn1,x);
```

>
As you can see, reading these exact solutions is quite a challenge! Note that the I stands for $\sqrt{-1}$. When a solution is this complicated it is more useful to look at the approximate solutions using **evalf** command.

```
> sol := evalf(H);
```

>
The small imaginary part attached to each real root can be removed with the **fnormal** command.

```
> fnormal([sol]);
```

>
Finally, the imaginary unit, $i = \sqrt{-1}$, can be removed with

```
> simplify(fnormal([sol]));
```

>
A good alternative to the **solve** command in a situation like this is the **fsolve** command which will be discussed in the next section.

The **solve** command can also be used to find the exact solutions for non-polynomial equations. Some simple examples are listed below. However if the equations are at all complicated, for example combining exponential, polynomial and trigonometric expressions, then an exact solution will typically not be available. Again the **fsolve** command is an alternative.

For this equation, the appropriate syntax would be

```
> fsolve(eqn1,x);
```

>
The **fsolve** command will be discussed in greater detail in the next section.

>
▼ **Example 5**

Solve the equation: $5 e^{\left(\frac{x}{4}\right)} = 43$

```
> solve(5*exp(x/4)=43,x);
```

>
▼ **Example 6**

Sometimes Maple does not display all of the solutions. How would you use the result below

to write down the entire set of solutions to the equation?

```
> solve(sin(x)=1/2,x);
```

```
>
```

The *environment variable* `_EnvAllSolutions`, if set to true, will force all inverse transcendental functions to return the entire set of solutions. This usually requires additional, system created, variables, which take integer values. Normally such variables are named with prefix `_Z` for integer values, `_NN` for non-negative integer values and `_B` for binary values (0 and 1).

For example:

```
> _EnvAllSolutions := true:
> solve(sin(x)=1/2,x);
```

```
>
```

The output is $\frac{\pi}{6} + \frac{2\pi}{3}BI + 2\pi_ZI$. This shows that $\sin(x) = \frac{1}{2}$ whenever $x = \frac{\pi}{6} + 2\pi n$ or $\frac{5\pi}{6} + 2\pi n$ where n is an integer.

```
>
```

Exercise 4.1

Solve the equation $x^3 - 11x^2 + 7x + 147 = 0$. Why does Maple produce only two distinct solutions for this cubic equation? Why is one of them written twice?

(HINT: Factor the left hand side of the equation.)

Student Workspace 4.1

```
>
>
>
>
>
```

Answer 4.1

```
> solve(x^3-11*x^2+7*x+147=0,x);
```

```
>
```

```
> factor(x^3-11*x^2+7*x+147);
```

```
>
```

Because $x - 7$ is a repeated factor, the cubic equation has only two distinct roots, namely, -3 and 7. We say that root 7 has a multiplicity of 2, meaning that the factor $(x - 7)$ appears *twice* in the factored polynomial.

```
>
```

Finding Approximate Solutions: The *fsolve* command

Maple's **fsolve** command can be used to find approximate solutions for any equation. For polynomial equations **fsolve** produces a complete list of all of the real solutions in one step (see

Example 1). For other equations **fsolve** can be used to get one solution at a time (see Examples 2 and 3).

Example 1

Maple's **fsolve** command will compute a numerical approximation for each solution of a polynomial equation.

To obtain approximate solutions for the equation $x^4 - x^3 - 17x^2 - 6x + 2 = 0$, that is, for

```
> eqn := x^4-x^3-17*x^2-6*x+2=0;
```

```
>
```

```
use
```

```
> fsolve(eqn, x);
```

```
>
```

The four solutions listed above provide us with a complete list of the solutions to the given polynomial equation.

```
>
```

Example 2

For the equation $x^3 + 1 - e^x = 0$, that is, for

```
> eqn := x^3+1-exp(x)=0;
```

```
>
```

the **fsolve** command will yield just the solution

```
> fsolve(eqn, x);
```

```
>
```

Maple returns one real solution. This time Maple has not given us the whole story. Are there any other solutions? How do we find them? A systematic procedure for finding the remaining solutions is presented in Example 3.

```
>
```

Example 3

Find the other real solutions for the equation $x^3 + 1 - e^x = 0$.

The first step in finding the other solutions is to plot a graph of the left-hand side of the equation.

Key Concept: Recall that the *x-intercepts* of $y = x^3 + 1 - e^x$ correspond exactly to the solutions of the equation $x^3 + 1 - e^x = 0$.

```
> plot(x^3+1-exp(x), x=-3..5, y=-5..15);
```

```
>
```

The graph shows four *x*-intercepts. One of these corresponds to the solution we found in Example 2. Which one?

The $x = 0$ solution is also easy to spot. How do we find the other three?

We can extend the **fsolve** command to look for a solution in a particular interval. For example to find the negative solution we ask Maple to search on the interval $[-1, -0.2]$ since we can see from the graph that there definitely is one (and only one) solution on that interval.

```
> fsolve(eqn, x=-1..-.2);
```

```
>
```

To find the other two solutions we use **fsolve** again, this time with search interval $[1, 2]$ and then with interval $[4, 5]$.

```
> fsolve(eqn, x=1..2);  
> fsolve(eqn, x=4..5);
```

```
>
```

What happens if you ask Maple to search for a solution on an interval where no solution exists ?

Let's try it out. From the graph it is clear that there are no x -intercepts (and therefore no solutions) between 2 and 4.

```
> fsolve(eqn, x=2..4);
```

```
>
```

Notice that Maple simply returns the original input line unchanged when it cannot find a solution on the given interval.

Are there any other solutions? For example, are there any solutions larger than 5 ? We can check this out by expanding the interval over which the graph is plotted. On the next line we expand the interval to $[-3, 50]$. No other x -intercepts appear. The graph confirms what we should expect by looking at the terms of the expression, namely the exponential term dominates and causes the graph to go down in the long run.

```
> plot(x^3+1-exp(x), x=-3..50, y=-10..15);
```

```
>
```

Alternatively we can use the **fsolve** command, now searching over this larger interval.

```
> fsolve(eqn, x=5..50);
```

```
>
```

As expected no solutions are found by Maple.

In a similar way, we can check for solutions to the left. Here we search for solutions over the interval $[-50, -1]$.

```
> fsolve(eqn, x=-50..-1);
```

```
>
```

None there either !

We now have a complete list of the four solutions of our original equation $x^3 + 1 - e^x = 0$. They are: -0.8251554597 , 0 , 1.545007279 and 4.567036837

>
▼ **Example 4**

Use **fsolve** to find the approximate solutions of the equation $\frac{x^2}{20} - 10x = 15 \cos(x + 15)$, that is, of the equation

```
> eqn := x^2/20-10*x-15*cos(x+15)=0;
```

```
>
```

Just as in the last example we will use a graph to help us determine the number and approximate location of the solutions. Our task is simplified if we start by converting the given equation to an equivalent one that has zero on the right-hand side.

So we will solve the equivalent equation $\frac{x^2}{20} - 10x - 15 \cos(x + 15) = 0$

If we now graph the left-hand side of this equation, we once again will find solutions at each of the x -intercepts.

```
> plot(lhs(eqn), x=-10..10);
```

```
>
```

From the graph it appears that there is a solution on the interval $[1, 2]$.

We now direct Maple to search for a solution on this interval.

```
> fsolve(eqn, x=1..2);
```

```
>
```

Have we found all of the solutions to this equation? In fact there is another solution! To find it, expand the interval over which the graph is drawn, then use **fsolve** to find a numerical approximation for this second solution. (This is left as an activity for the reader.)

```
>
```

▼ **Exercise 4.2**

Find all the solutions to the equation $x^5 - 4x^3 + 3x^2 + 7x - 1 = 0$. Begin by looking at a relevant graph.

▼ **Student Workspace 4.2**

```
>  
>  
>  
>  
>
```

▼ **Answer 4.2**

We begin by graphing the left-hand side of the equation.

```
> eqn := x^5-4*x^3+3*x^2+7*x-1=0;
```

```
>  
> plot(lhs(eqn), x=-5..5, y=-5..5);
```

>
This picture indicates that there are solutions near -2, -1.5, and 0. We next try the unrestricted **fsolve** command to see which solution(s) Maple finds.

> **fsolve(eqn, x);**

>
Since this is a polynomial equation the **fsolve** command gives us a complete list of the real solutions.

>

▼ *Exercise 4.3*

Find all the solutions to the equation $x^2 - 2 = \ln(x + 5)$. Use the graph of one expression to locate the solutions. Check each solution by substituting it back in the original equation.

▼ *Student Workspace 4.3*

>

>

>

>

>

>

>

▼ *Answer 4.3*

First we put the equation in "standard form", i.e. with zero on one side.

> **eqn := x^2 - 2 - ln(x+5) = 0;**

>

Now we can graph the right-hand side equation.

> **plot(lhs(eqn), x=-10..10);**

>

There appear to be two solutions. One near -2 and the other near 2. Use **fsolve** with a restricted domain to find the two solutions you've located more precisely.

> **soln1 := fsolve(eqn, x=-5..0);**

>

> **soln2 := fsolve(eqn, x=1..3);**

>

Check by substituting back into the original equation.

> **evalf(eval(eqn, x=soln1));**
evalf(eval(eqn, x=soln2));

>

Notice that the original equation is only "approximately" satisfied by each of our solutions. The slight discrepancy is a result of round-off error in the approximate solutions.

Check that there are no additional solutions.

>

Exercise 4.4

The graphs of $y = 10 - x^2$ and $y = 4 \sin(2x) + 5$ intersect twice on the interval $[-5, 5]$.

- Graph the two equations together and estimate the intersection points using the mouse.
- Write an equation that can be solved to find the x-coordinates of the intersection points.
- Use **fsolve** to solve this equation.
- Use the results from part c) to estimate the y-coordinates of the intersection points.
- It appears that the curves may intersect at a third point near (1, 9). Use **fsolve** and/or a relevant graph to demonstrate that there is no intersection point at that location.

Student Workspace 4.4

>
>
>
>
>
>
>
>
>

Answer 4.4

First enter in the two equations.

```
> y1 := 10-x^2;  
y2 := 4*sin(2*x)+5;
```

Here is a plot of the two equations.

```
> plot([y1,y2],x=-5..5);
```

Intersection points are located approximately at: (-1.8, 6.6) and (2.75, 2) .

b) The equation to solve is $y_1 = y_2$.

```
> eqn := y1=y2;
```

c) We now find the two solutions using **fsolve**.

```
> x_soln1 := fsolve(y1=y2,x=-4..0);  
x_soln2 := fsolve(y1=y2,x=0..4);
```

d) We can use the **eval** command to find the corresponding y-coordinates.

```
> y_soln1 := eval(y1,x=x_soln1);  
> y_soln2 := eval(y1,x=x_soln2);
```

So the points of intersection are : (-1.800,6.763) and (2.773,2.311)

e) Here is a closer look at what is happening near $x = 1$.

```
> plot([y1,y2],x=.5..1.5);
```

Alternatively we can use **fsolve** to confirm that there is no solution near $x = 1$.

```
> fsolve(y1=y2,x=.5..1.5);
```

▼ Solving Literal Equations

```
> restart;
```

Often Maple can solve literal equations for any one of the variables.

Suppose we want to solve for the variable g in the equation $4 - v = 2T - kg$.

The **solve** command works well here.

```
> solve(4-v=2*T-k*g,g);
```

Here is a little nicer way of displaying the same result:

```
> g = solve(4-v=2*T-k*g,g);
```

▼ Exercise 4.5

Edit the last command to solve for each of the other letters T , k and v .

▼ *Student Workspace 4.5*

```
>  
>  
>  
>  
>
```

▼ *Answer 4.5*

```
> T = solve(4-v=2*T-k*g,T);  
>  
> k = solve(4-v=2*T-k*g,k);  
>  
> v = solve(4-v=2*T-k*g,v);  
>
```

▼ Exercise 4.6

Solve the equation $x^2 + y^2 = 9$ for y . Assign the set of solutions to a variable named S . How are the two solutions $S[1]$ and $S[2]$ related ?

Student Workspace 4.6

```
>  
>  
>  
>  
>
```

Answer 4.6

```
> S := solve(x^2+y^2=25,y);  
>  
> S[1];  
>  
> S[2];  
>
```

The solution $S[1]$ is the negative of solution $S[2]$.

```
>
```

Solving a Linear System of Equations Using the *solve* Command

Please execute the next two lines before proceeding:

```
> restart;  
> with(plots):  
>
```

The **solve** command can also be used to solve a system of m linear equations in n variables. We call these m by n linear systems for short.

Example 1

The 2 by 2 linear system whose equations are

```
> eqn1 := 3*x+2*y=3;  
eqn2 := x-y=-4;
```

```
>
```

has solution

```
> q := solve({eqn1,eqn2});  
>
```

We can use the **eval** command to directly select the x - and y -coordinates from the return of the **solve** command.

For example, for the x -coordinate, type:

```
> eval(x, q);  
>
```

and for the y -coordinate, type

```
> eval(y, q);
```

```
>
```

In the first of these usages, we are asking Maple to evaluate the symbol x using the information it finds in the set

```
> q;
```

```
>
```

Similarly, by evaluating the symbol y with the information in this set, the value of y is selected.

The solution can be expressed as a point in Maple via the usage

```
> soln := eval([x,y],q);
```

```
>
```

Again, note the use of the **eval** construct. The symbol $[x,y]$ is evaluated with the information in the set q , and results in the point $[-1, 3]$.

A graph of the two lines represented by the two equations can be generated if first, each equation is solved explicitly for y .

```
> y1 := solve(3*x+2*y=3,y);  
y2 := solve(x-y=-4,y);
```

```
>
```

Now we construct a picture made up of two parts: "part1" contains the graphs the two equations and "part2" plots the solution point that we found. This point should be the intersection point of the two lines. Is it?

```
> part1 := plot([y1,y2],x=-5..5, color=[black,red]):  
part2 := plot([-1,3],style=point,color=blue,symbol=  
circle):  
display([part1,part2]);
```

```
>
```

Alternatively, since the equations of the lines were given implicitly, the desired graph could be obtained with Maple's **implicitplot** command. The following figure results.

```
> p1 := implicitplot([eqn1,eqn2], x=-5..5, y=-6..8, color=  
[black,red]):  
p2 := plot([soln], style=point, symbol=circle, color=blue)  
:  
display([p1,p2]);
```

```
>
```

Example 2

Here is an example of the solution of a 3 by 3 system with variables x , y , and z .

Solve the 3 by 3 system: $\{x + y + z = 1, 3x + y = 3, x - 2y - z = 0\}$

```
> q := solve({x+y+z=1, 3*x+y=3, x-2*y-z=0});
```

```
>
```

The solution can be expressed as a point via the syntax

```
> eval([x,y,z],q);  
>
```

Exercise 4.7

Find the solution to the system:

$$4x + 3y = 12 \text{ and } 5x - 7y = 35$$

Check by evaluating both equations in the system at the solution pair

Student Workspace 4.7

```
>  
>  
>  
>  
>  
>  
>
```

Answer 4.7

```
> eqns := {4*x+3*y=12, 5*x-7*y=35};  
>
```

Solve the system using the `solve` command.

```
> ans := solve(eqns);  
>
```

And to check if the solution $x = \frac{189}{43}$, $y = -\frac{80}{43}$ is a correct solution, use the `eval` command.

```
> eval(eqns, ans);  
>
```

Linear Systems with an Infinite Number of Solutions (Optional)

When a system has more variables than equations we often get not one, but an infinite number of solutions.

Here is an example.

Example 1

Solve the system : $\{x + y + z = 1, 3x + y = 3\}$.

```
> solns := solve({x+y+z=1, 3*x+y=3});  
>
```

Notice this time we do not get a single set of numerical values for x , y and z . Instead Maple tells us how the values of x , y and z must be related to construct a typical solution.

In particular the expression $x = x$ in the output above indicates that x can be any number. We refer to it as the "free" variable in the solution. To find any particular solution (among the infinite number possible) pick any value for x and use this to calculate the corresponding

values for y and z . For example let $x = 4$.

```
> eval(solns, x=4);
```

```
>
```

So one solution is: $x = 4, y = -9, z = 6$.

Take a minute and check by hand that these three numbers do in fact satisfy our original equations: $x + y + z = 1$ and $3x + y = 3$.

Now let's look at the solution that is generated when we take $x = 2$.

```
> eval(solns, x=2);
```

```
>
```

So two of the infinitely many solutions are: $(x, y, z) = (4, -9, 6)$ and $(2, -3, 2)$.

```
>
```

Exercise 4.8

Solve the system: $\{x + 2y + z = 2, 3x + y = 1\}$ and find at least three particular solutions.

Student Workspace 4.8

```
>  
>  
>  
>  
>  
>
```

Answer 4.8

```
> eqns := { x+2*y+z=2 , 3*x+y=1 };
```

```
>
```

```
> soln := solve(eqns);
```

```
>
```

Here are the solutions for $x = 1, 2, 3,$ and 4 :

```
> eval(soln, x=1);
```

```
>
```

```
> eval(soln, x=2);
```

```
>
```

```
> eval(soln, x=3);
```

```
>
```

```
> eval(soln, x=4);
```

```
>
```