

Introduction to *Maple*

Above all, remember that *Maple* is a program with a *rigid* structure. You must tell it what to do *in precise terms*—commas, colons, semicolons, spaces, parentheses, brackets must conform to an exacting syntax. Success requires **patience** and **persistence**.

Firing up *Maple* This will open a *worksheet*—the interface through which you communicate with the computational *kernel*. Opening a saved worksheet works in the standard way. What happens now?

- 1) Type a command or program to be evaluated. This material appears in an “evaluation cell.”
- 2) To evaluate the cell—that is, send it to the kernel, press < Enter >. *Note:* *Maple* will evaluate everything in the cell that you *activate*. Other cells will be left alone.

You can also enter text into the worksheet. An important part of your written lab work will be the descriptions of the computations and program that you create. You should *comment* on everything in your worksheet submission.

Getting help and getting started On the menu bar pull down Help and select Topic index. This will start a browser which should be your primary tool for finding information. You can browse items by selecting Contents or use Search to type the name of something in which you’re interested. Many of *Maple*’s built-in commands have names similar to the ones with which we’re familiar. For more direction, select Getting Started under Contents.

Once you’ve found a command that looks promising you can read the descriptions of its functions and, most usefully, see examples of how it’s used. Poke around here. Look at some examples—you should actually run what’s in the examples (you can cut-and-paste from the browser) into a worksheet. At the bottom of the page there are links to related commands that might be useful.

Naming things The way to assign a name to something is with ‘:=.’ Examples:

<i>Maple</i> expression	Description
$a := 7$	Assigns the number 7 to the symbol a
$v := [1, 2]$	Assigns the “list” (point in the plane) $(1, 2)$ to v
$v := \langle 1, 2 \rangle$	Assigns the vector $\begin{pmatrix} 1 \\ 2 \end{pmatrix}$ to v
$M := \langle\langle 1, 3 \rangle \mid \langle 2, 4 \rangle\rangle$	Assigns the list of lists (2×2 matrix/array) $\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$ to the symbol M

Note:

- Use `unassign(X)` to *unname* something that you called X .
- If things get hopelessly fouled up, you can restart the kernel—losing all assignments and definitions—by running `restart`.

Functions and maps The way to define a function is with ‘ \longrightarrow .’ Examples:

<i>Maple</i> expression	Description
$f := x \longrightarrow x^2$	Defines the map $f(x) = x^2$
$g := (x,y) \longrightarrow (x^2 + y, y^2 - 1)$	Defines the map $g(x,y) = (x^2 + y, y^2 - 1)$

Try defining these functions and compute their values at two or three points—use $f(2)$, etc. Now define the functions

$$f(x) = e^{\sin x} \quad g(x) = \sin e^x$$

and compute a few values. Use the help facility to find *Maple*’s built-in trig and exponential functions.

Lists, vectors, and matrices *Maple* thinks of a matrix as a vector of vectors. Look up `vectors` and `matrices` to learn more. Now, create the matrices

$$A = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} \quad B = \begin{pmatrix} 5 & 6 \\ 7 & 8 \end{pmatrix}.$$

Compute the matrix products AB and BA —use `A.B`. Now create the *symbolic* matrices

$$U = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \quad V = \begin{pmatrix} e & f \\ g & h \end{pmatrix}.$$

and compute the matrix products UV and VU .

Derivatives Use the help facility to learn how to take the derivative of a function of one variable. (Check out `diff`.) Compute the derivative with respect to x of the functions

$$f(x) = e^{\sin x} \quad g(x) = \sin e^x.$$

Plotting Use the help facility to learn how to plot a function of one variable. (Look at `plot` under `Graphics` and `2-D`.) Now, plot the graphs of the functions

$$f(x) = e^{\sin x} \quad \text{for } -1 \leq x \leq 10 \quad g(x) = \sin e^x \quad \text{for } -2 \leq x \leq 4.$$

Halting a computation in progress If you decide that you want to stop a calculation, select the interrupt button (a hand icon) on the toolbar. For “small” computations this usually produces the desired effect. However, sometimes this will have unfavorable results—like killing the program. It’s something of a last resort.

Save your work Every time you enter a new item for calculation, save what you’ve done before starting the evaluation. This can be done by pulling down the `File` menu and selecting `Save`. If it’s the first time, you must specify a file name and location. Look at the menu item for a quick keyboard command for saving your work.