COMPUTATIONAL FEATURES

Manual and Auto mode

To prevent continuous screen update, choose **Manual Mode** from the **Math** menu. Then press **F9** to whenever you want to update the screen.

How Mathcad scans a document

Mathcad evaluates left to right and top to bottom, with the exception of the $\equiv$ sign. Click on this topic for further details.

Turning equations on and off

Click on the equation and choose **Toggle Equation** from the **Math** menu.

Interrupting calculation

Press **Esc** to interrupt a calculation. Press **F9** to resume.

Manual and Auto mode

In automatic mode:

- Calculations are done immediately.
- The word 'auto' appears on the message line

To switch to manual mode, choose **Manual Mode** from the **Math** menu.

In manual mode:

- Calculations occur only when you press **F9**, or choose **Calculate Document** from the **Math** menu.
- The words 'calc F9' appear on the message line whenever the screen needs updating.

To switch to auto mode, choose **Automatic Mode** from the **Math** menu.

Mathcad performs only those calculations necessary to update what's on the screen. To perform calculations throughout the entire document, choose **Calculate Document** from the **Math** menu.

File I/O functions are not affected by automatic screen update. To update file I/O functions, either choose **Calculate Document** from the **Math** menu, or click in the equation and press **F9**.

If you prefer to start Mathcad in manual mode,

- Choose **Manual Mode** from the **Math** menu.
- Choose **Save Configuration File** from the **File** menu.
- Specify `mcad.mcc` as your configuration file.

The next time Mathcad starts, it will do so in manual mode.

How Mathcad scans a document

Mathcad makes two passes through your document. Each pass is left to right, top to bottom. The action taken in each pass is as follows:

- On the first pass, Mathcad performs all assignments involving the global assignment ''.
- On the second pass, Mathcad performs all assignments involving the local assignment ':='. Mathcad also displays results of all calculations.

In deciding what order to evaluate equations, Mathcad examines the midpoint of an invisible frame surrounding the equation.

Turning equations on or off

To prevent Mathcad from evaluating an equation:

- Click in the equation to select it.
- Choose **Toggle Equation** from the **Math** menu.

Mathcad draws a small box next to the equation to indicate that it is turned off.

To turn an equation back on:

- Click in the equation to select it.
- Choose **Toggle Equation** from the **Math** menu.

Mathcad removes the small box next to the equation.

Interrupting calculation

To interrupt a calculation, press **Esc**. Mathcad will mark the equation being processed with the error message '*interrupted*'

To resume calculation, press **F9**.

To insert ≡

- press **~** (the tilde key)

To insert :=

- press **:** the colon key

To insert =

- press **=**

To insert =

- press **Ctrl**+**=~**

BUILT-IN VARIABLES

MATH VARIABLES

Name	Keystroke
∞	**Ctrl+z**
π	**Ctrl+p**
e	**e**
i	**1i**
j	**1j**
%	**%**

To redefine...

SYSTEM VARIABLES

Name	Default Value
TOL	0.001
ORIGIN	0
PRNPRECISION	4
PRNCOLWIDTH	8

To redefine...

π Press **Ctrl**+**p**

The value of π to 15 decimal digits when used in numerical calculations.

The exact value of π when used in symbolic calculations.

i or **j** Press **1i** or **1j**

The imaginary unit. Note that you must type **1i** and **1j**, not **i** and **j**. In either case, Mathcad hides the '1' when you leave the expression.

e Press **e**

The value of **e** to 15 decimal digits when used in numerical calculations.

The exact value of **e** when used in symbolic calculations.

∞ Press **Ctrl+z**

When used in a numerical expression, this is Mathcad's largest number:

10^{307}

When used in symbolic calculations, this represents the usual mathematical definition of infinity

% Press **%**

This is set to 0.01. Use it as a convenient way to work with percent. For example:

`100·30·% = 30`

ORIGIN Default = 0

This represents the starting index of all arrays. To change it, choose **Built-In Variables** from the **Math** menu.

You can set ORIGIN to any integer, positive or negative. The value you set for ORIGIN will affect every array in your document.

PRNCOLWIDTH Default = 8

This represents the width of columns in ASCII files created by the WRITEPRN function.

To change it, choose **Built-in Variables** from the **Math** menu.

PRNCOLWIDTH can be set to any integer between 1 and 132.

PRNPRECISION Default = 4

This represents the number of significant digits Mathcad writes to a file with the WRITEPRN function.

To change it, choose **Built-in Variables** from the **Math** menu.

PRNPRECISION can be any integer between 1 and 16.

TOL Default = 0.001

Controls the precision to which integrals and derivatives are evaluated. Also controls how long Mathcad iterates in solve blocks and in the root function.

Very low values of TOL will increase computation time and may result in a '*not converging*' error message.

High values of TOL will decrease computation time at the expense of precision.

Redefining built-in math variables

You can change the value of the constants

- ∞
- π
- e
- i
- j
- %

by using '**:=**' in your document just as you would for any variable.

Redefining built-in system variables

You can change the value of the constants

ORIGIN
PRNPRECISION
TOL
PRNCOLWIDTH

in two ways:

- By redefining them with '**:=**' in your document
- By choosing **Built-In Variables** from the **Math** menu.

Pathname

A pathname is the complete DOS name for a file. For example, if a file named **tuba.mcd** is in a subdirectory called **\orchestra\brass** on your **A** drive, its full pathname is

A:\orchestra\brass\tuba.mcd.

Scalar

A scalar is a single number, for example 2 or 34+2i. Anything that's not a scalar is either a vector or a matrix.

When something is 'scalar valued,' it returns a scalar as a result. For example, **cos(0)** is scalar valued because it returns the single number 1. However **identity(3)** returns a 3x3 matrix. It is therefore *not* scalar valued, but matrix valued.

Variables tag

This is the tag Mathcad assigns by default to any variable or function name you type in the document.

Constants tag

This is the tag Mathcad assigns by default to any constant you type in the document.

Tag

A tag is consists of a font, size and style combination. Rather than change the font of a name or number directly, you change the font of a tag and assign the tag to a name or number. This lets you change the font of many names or numbers in your document all at once.

To change the font characteristics of a tag, choose **Modify Font Tag** from the **Math** menu.

To change the tag associated with a name or number, click in the name or number and choose **Apply Font Tag** from the **Math** menu.

Graphics region

A graphics region is a region containing either:

- A two dimensional plot
- A surface plot
- A graphics image

To create a graphics region, press:

@	a two dimensional plot
Ctrl+2	a surface plot
Ctrl+5	an empty region for a graphics image
Shift+Ins	to paste a bitmap from the clipboard

Windows bitmap image

A graphic image stored in standard Windows graphics format.

Many Windows applications, for example the Paintbrush accessory that ships with Windows, can create bitmap images.

Bitmap images stores in files usually end in the extension 'BMP.

Note that this demo version of Mathcad cannot import bitmaps via the clipboard.

Insertion point

The 'insertion point' is marked by a blue vertical line behind which characters you type appear. Use the and keys to move it. You can also click wherever you want to move it.

Characters behind the insertion point are deleted when you press **BkSp**.

Crosshair

The 'crosshair' appears when you click in empty space. It shows where the next character you type or the next paste from the clipboard will appear.

Move the crosshair either by clicking wherever you want to put it, with the arrow keys, the **Enter** key, the space bar, or the **PgUp** and **PgDn** keys.

Operand

An 'operand' is a number or an expression upon which an operator acts. For example, in the expression '5! + 3' the number '3' and the expression '5!' are operands of the operator 'plus.' The numbers '5' is an operand of the factorial (!) operator.

The following table gives examples of operands:

EXPRESSION	OPERANDS
10	10 and 307
3!	3
(3+x)!	3+x
$\sqrt{2}$	2
$\int 1$	J1(x), 3, 1 and dx

Clipboard

The 'clipboard' is a temporary storage location maintained by Windows. This demo version of Mathcad cannot use the clipboard to communicate with other Windows applications.

To place something on the clipboard, select it and press

- **Shift**+**Del** to remove it from the document and put it on the clipboard. This is called 'cutting' to the clipboard.
- **Ctrl**+**Ins** to put it on the clipboard without removing it from the document. This is called 'copying' to the clipboard.

When you cut or copy to the clipboard, the previous contents of the clipboard are discarded.

To put whatever is on the clipboard into your document, press **Shift**+**Ins**.

Operator palette

The 'operator palette' is a strip of buttons along the left edge of the application window. You can click on these buttons to insert operators, plots and Greek letters.

Empty space

This refers to the space between regions. Whenever you click in empty space, you see a crosshair.

The blank spaces that separate variables and operators in an equation are not considered empty space.

Placeholder

A small black box in a math region is a 'placeholder.' The placeholder indicates that the mathematical expression or plot is incomplete.

To fill the placeholder, click on it and begin typing.

Selection box

When you click on an operator or in a plot, you see a blue box with a notched corner. This is the 'selection box.' Use the arrow keys or click in different parts of the expression to change what's enclosed by the selection box.

As a rule. *whatever is enclosed in the selection box becomes the first operand of the next operator you type.*

To enclose an expression in a selection box:

- Click in the equation with the mouse.
- Press the key until the selection box encloses the expression.

Region

Anything you create in Mathcad is placed in a 'region.' Procedures for selecting, deleting and moving regions are the same regardless of what's in the region.

When you click in a region, you'll see either the insertion point or the selection box. When you click outside the region, you'll see a small crosshair.

Solve block

A 'solve block' consists of several equations and inequalities between the keyword 'Given' and either a **Find** or **Minerr** function.

Solve blocks should not contain range variables. They should contain only the equations necessary to solve your system of equations.

When typing an equals sign in a solve block, use **Ctrl**+**=**.

WMCADDIR

This is an environmental variable defined automatically during the installation procedure. If you see a dialog box indicating that WMCADDIR is not set:

- Exit Mathcad and Windows.
- Reboot the computer.

This will reexecute your `autoexec.bat` file.

OLE server

A Windows application capable of exporting OLE (Object Linking and Embedding) objects via the clipboard.

This feature allows you to edit the OLE object using whatever application created it. To do so, double-click on the object.

This feature is not available in this demo version of Mathcad.

Embedded object

An object which, when double-clicked on, opens the application that created it and places a copy of itself into the application window.

Unlike a linked object, any changes you make will affect only the Mathcad document.

This feature is not available in this demo version of Mathcad.

Linked object

An object which, when double-clicked on, opens the application that created it together with a file containing the object.

Unlike an embedded object, any changes you make will affect the file containing the object, not just the Mathcad document.

This feature is not available in this demo version of Mathcad.

DOCUMENT MANAGEMENT

Open a document.	**F5**
Save current document.	Disabled in demo
Rename document.	Disabled in demo
Insert a document	**File -> Insert Document**
Open a new document.	**F7**
Close a document	**Ctrl**+**F4**
Exit Mathcad	**Alt+F4**

Opening a document

To open a document:

- Press **F5** to display a scrolling list of mcd files in the current directory.
- Double click on the name of the file you want to open.

To change the current directory, click in the scrolling list of the Open Document dialog box. Clicking on **[..]** moves up one directory.

Creating a new document

Press **F7** to open a new empty document.

Exiting Mathcad

Alt+F4 exits Mathcad. If any changes have been made to any open document since it was last saved, Mathcad prompts you to save them before quitting.

If you just want to close a document, press **Ctrl+F4**.

Closing a Document

Pressing **Ctrl**+**F4** will close the active document.

If any changes have been made since the document was last saved, Mathcad prompts you to save them before closing the document.

Inserting one document into another

To insert one document into another:

- Choose **Insert Document** from the **File** menu to display a scrolling list of .mcd files in the current directory
- Double click on the name of the document you want to insert.
- Mathcad will insert the contents of this file wherever your cursor is located.

To change the current directory, click in the scrolling list of the Insert Document dialog box. Clicking on **[..]** moves up one directory.

CREATING AN EQUATION

Names

Click in an empty space or on a placeholder. Type variable and function names directly from the keyboard. Names are case-sensitive. All characters must be in the same font. Click on operator palette for most Greek letters.

Numbers

Click in an empty space or on a placeholder. Type numbers directly from the keyboard. Numbers can be any length. You can use scientific notation for convenience.

Operators

Click in an empty space or on a placeholder. Type operators directly from the keyboard. Mathcad uses +, *, -, / and ^ for the four arithmetic operations and exponent. For other operators, click in the operator palette.

Equals sign

To define a variable or function, press '**:**' (the colon). To evaluate a variable or function, press '**=**' In solve blocks, press **Ctrl+=**

Parentheses

Click the button marked '(x)' on the operator palette. You can also use the **(** and **)** keys.

EDITING AN EXPRESSION

Changing a name or number

Click in name or number to position insertion point. Then type additional letters and numbers or press **BkSp** to delete.

Deleting operators

Click on smallest expression containing operator. This encloses it in the selection box. Then press **BkSp** to delete. Mathcad leaves an empty operator placeholder.

Inserting operators

Click on first operand to enclose it in the selection box. Then type the operator.

Applying a function

Enclose the argument with parentheses. Then enclose everything, including parentheses with the selection box. Press **Ins** and type the function name.

Inserting or deleting parentheses

To enclose an expression with parentheses, first enclose it in the selection box. Then click on '(x)' on the operator palette.

To delete, enclose the expression, including parentheses, with the selection box.

Moving expressions

Enclose the expression in the selection box and cut it to the clipboard. Then click in an empty space or a placeholder and paste it.

Deleting expressions

Enclose the expression in the selection box and cut it to the clipboard.

Changing a name or number

- ≥ Click in the name or number. If necessary, press ⊞ to see the insertion point. Use or to move the insertion point.
- ≥ Type letters and numbers. To delete a character just behind the insertion point, press **BkSp**

Operators

Replacing an operator

- Click on operator to enclose expression in selection box.
- Press **BkSp** to delete the operator.
- Type in the new operator.

Inserting an operator

- Click on operand to enclose it in the selection box. Use ↑ if necessary to enclose the operand.
- Type keystroke for operator.
- To insert an operator *before* the selected expression, press **Ins** before typing the operator.

Inserting a minus sign

- Click in expression to enclose it in the selection box. Use ↑ if necessary.
- Press **Ins**.
- Press the minus sign.

Applying a function to an expression

- > Enclose the expression in selection box.
- > Click on '(x)' on the operator palette.
- > Click to enclose the parentheses in the selection box.
- > Press **Ins**.
- > Type the function name.

Parentheses

Inserting parentheses

- Enclose the expression in selection box
- Click on '(x)' on the operator palette to insert parentheses.

Deleting parentheses

- Enclose the expression, including the parentheses, in the selection box.
- Press **Del**.

Moving and deleting parts of an expression

Moving part of an expression

> Enclose the expression to be moved in the selection box.
> Press **Shift**+**Del** or **Ctrl**+**Del** to cut or copy it to clipboard.
> Click on an empty space or a placeholder into which you want to paste the selected expression.
> Press **Shift**+**Ins** to paste the expression.

Deleting part of an expression

> Enclose the expression to be deleted in the selection box.
> Press **Shift**+**Del**. Mathcad leaves behind an empty placeholder.

DOCUMENT LAYOUT

Selecting regions

To select regions, for moving, enclose them in a selection rectangle. To add regions to a selection, shift-click on the regions you want to add. To fill in a selection, select the end regions and control-click in any region between the endpoints.

Moving regions

Select regions you want to move. Then drag them to the new location.

Deleting regions

Select regions you want to delete. Then press **Shift+Del** to cut to the clipboard.

Blank lines between regions

Choose **Insert Blank Line** or **Delete Blank Line** from the **Edit** menu.

Separating regions

If regions overlap, you can choose **Separate Regions** from the **Edit** menu. This operation separates all overlapping regions in the document and cannot be undone.

Equals signs

Mathcad has four variants of the equals sign:

Evaluate (=)

Use to display numerical value of left hand side. Result can be formatted but not edited.

Definition (:=)

Use to define a variable name or function.

Global definition (≡)

Use to make Mathcad define a variable name or function before definitions involving :=.

Logical equals (=)

Use in solve blocks and as a boolean operator.

Separate regions

If you find that regions in your document are overlapping, this command separates them. Note that this command cannot be undone.

Be careful using this when you have a lot of error messages. Once the error messages disappear, you may find the regions too far apart.

Selecting regions

To select several regions visible in your window:

- Click the mouse just outside a corner of the area enclosing the equations and regions you want to select. This anchors one corner of the selection rectangle.

- Press and hold down the left mouse button. With the button still held, drag the mouse. A selection rectangle framed by dashed lines emerges from the anchor point.

- When the selection rectangle touches all the regions you want to select, let go of the left mouse button. All selected regions are enclosed by dashed selection rectangles.

To add regions to your selection, move the pointer over the region, press **Shift** and click the mouse.

To select all regions between two selected regions, move the pointer over any of the regions, press **Ctrl** and click the mouse.

Dragging regions

- Move the mouse pointer into one of the regions enclosed by a dashed selection rectangle. The pointer turns from an arrow into a cross.
- Press and hold down the left mouse button.
- With the mouse button still held down, move the mouse pointer. Outlines of all selected regions will follow your movements.
- When the outlines are on the desired spots, let go of the mouse button.

X := 140

X := 100

$x +$

$y = \sin($[illegible]$)$

$y = \sin(\blacksquare)$

$y = \sin(\boxed{\blacksquare})$

Filling in a selection

If you want to select everything *between* two regions:

- Select the first region by dragging a selection rectangle around it.
- Move the pointer over the last region.
- Press **Ctrl** and click the mouse.

Do not click the mouse without pressing **Ctrl**. Doing so will deselect the regions you've already selected.

Adding a region to a selection

If you want to add a region to a set of regions you've already selected:

- Move the pointer over the region.
- Press **Shift** and click the mouse.

Do not click the mouse without pressing **Shift**. Doing so will deselect the regions you've already selected.

ERROR MESSAGES

array size mismatch
cannot be defined
cannot take subscript
cannot take superscript
definition stack overflow
did not find solution
dimension to nonreal power
domain error
duplicate
end of file
equation too large
error in constant
error in list
error in solve block
file error
file not found
illegal array operation
illegal context
illegal factor
illegal function name
illegal ORIGINillegal range
illegal toleranceincompatible units
indeterminate dimension
index out of bounds
subscript too large
interrupted
list too long
misplaced comma
missing operand
missing operator
must be 3-vector
must be array
must be dimensionless
must be increasing
must be integer
must be nonzero
must be positive
must be range
must be real
must be scalar
must be square
must be vector
nested solve block
no matching Given
non-scalar value
not a name
not converging
overflow
range not allowed
significance lost
singularity
stack overflow

too few arguments
too few constraints
too few elements
too few subscripts
too large to compute
too large to display
too many arguments
too many constraints
too many points
too many subscripts
undefined
underflow
unmatched parenthesis
wrong size vector

array size mismatch

You tried to perform a vector or matrix operation on arrays whose sizes are inappropriate. For example, adding two unequal size matrices is illegal. Matrix multiplication requires that the number of columns in the first matrix match the number of rows in the second.

cannot be defined

Something other than a legally definable expression is on the left side of this definition. Mathcad accepts any of the following on the left side of a definition:

- A variable name
- A variable name with a subscript:
- A variable name with a superscript:
- An explicit vector or matrix generated by typing **Ctrl**+**V**. The vector or matrix can hold variable names or subscripted variable names only.
- A function name with arguments: **f(x,y)**.

Any other expression is illegal. If you want to compute a result instead of defining a variable, use an equals sign '**=**' instead of a colon.

cannot take subscript

You used a subscript on something other than a vector or a matrix

cannot take superscript

You used a superscript on something other than a vector or a matrix.

definition stack overflow

This function definition contains too many nested functions.

did not find a solution

In a solve block, Mathcad couldn't find a solution. Try raising the value of **TOL** or using the **minerr** function instead of the **find** function.

dimension to nonreal power

This expression contains units that have been raised to a complex power. An expression with units can only be raised to a real power.

domain error

You attempted to evaluate an operator or a function at an inappropriate value. For example, **ln(0)** will give this error. So will **-3!** because factorial is undefined for negative numbers.

duplicate

You tried to evaluate the same variable twice in the same definition. For example, if you create a vector on the left side of a **`:=`** and use the same name in more than one element, you will get this error.

equation too large

This expression is too complicated for Mathcad to evaluate. Break the equation down into two or more smaller expressions.

error in constant

This expression contains a digit followed by some letters. Mathcad tries to interpret anything beginning with a digit as a constant.

Sometimes this arises when you leave out the ***** operator as you would when writing equations on paper. For example, writing **6x** instead of **6*x** is often done on paper, but generates this error in Mathcad.

error in list

The indicated function definition contains an invalid list of arguments. A valid function definition begins like this:

functionname(argument list) := ...

The argument list must be a name or a list of names separated by commas. Any other expression is illegal.

You also see this error message if you create an invalid list in another context, for example, in the list of y-axis expressions for a plot.

error in solve block

You see this error if you evaluate a user function that is defined in terms of a solve block with an error in it.

To fix the error, fix the problem in the solve block. If you evaluate the solve block directly, instead of defining a function with it, you will see a more specific error message.

file error

The file you are trying to access with **READ** or **READPRN** does not have the required format.

The file must be ASCII with spaces or tabs separating the numbers.

file not found

This expression refers to a file Mathcad was unable to find the file specified in the argument to **READ**, **READPRN**, **APPEND**, or **APPENDPRN**.

illegal array operation

One of the functions or operators in this expression has received a vector or matrix when it was expecting a scalar.

For example, you see this message if you try to divide one vector by another.

If you want to apply a function or operator to every element in a vector or matrix, use the vectorize operator **Ctrl**+**-**.

illegal context

The most common causes of this error message are:

- You entered a semicolon somewhere other than in a legal range definition. (The semicolon appears as two dots.)
- You used a **WRITE** or **APPEND** function anywhere other than on the left side of a definition. These functions on the right side of a definition.
- You used an existing function name as a variable name or an existing variable name as a function name.
- You used a in a solve block.

illegal factor

You entered an illegal expression in the placeholder at the end of a calculation equation. This placeholder requires real, non-zero, scalar values.

illegal function name

You used an expression that Mathcad interprets as a function, but the function name is invalid. You see this error, for example, if you use a number as a function name: **6(x)**.

You may see this error if you omit an operator like *, causing Mathcad to interpret the parentheses in your equation as defining a function instead of as a way to group operations. This often happens because in algebra books, something like **3(x+2)** means **3x + 6**. In Mathcad, you must type **3*(x+2)**.

illegal ***ORIGIN***

You defined **ORIGIN** to be either non-integer, or a integer whose magnitude is greater than 16,000,000.

This error marks the first use of a subscript after the illegal definition of **ORIGIN**.

illegal range

You defined a range variable with a range Mathcad could not interpret. When you define a range, you must use one of the following forms:

- **Rvar := n1 ..n2** (type Rvar:n1;n2)
- **Rvar := n1,n2 ..n3** (type Rvar:n1,n2;n3)

You can use at most one comma and one semicolon in the definition of a range for a range variable. If you use the second form, the value of **n2** must lie between the values of **n1** and **n3**, but not equal **n1**.

illegal tolerance

The built-in constant **TOL** is outside the allowable range.

To fix the error, define **TOL** with a value between 0 and 1 above the indicated expression.

incompatible units

This error message has two likely causes:

- You are adding or subtracting two terms with different units.
- You have created a matrix in which the elements do not have the same units.

indeterminate dimension

You have raised an expression involving units to something other than a real power. Mathcad cannot determine the dimensions of the result. If an expression is defined with units, you can raise it only to a fixed real power.

index out of bounds

This expression uses a subscript or superscript that refers to a nonexistent array element.

Keep in mind that unless you specify otherwise, all Mathcad arrays begin with 0th element. If you are used to thinking of arrays as beginning at the 1st element, reset **ORIGIN** to 1.

subscript too large

You tried to use a subscript or superscript that exceeds Mathcad's limit.

interrupted

You interrupted a calculation pressing **Esc**.

To recalculate the marked equation, put the cursor in the equation and press **F9**.

list too long

You entered too many elements in a list separated by commas. This can occur if you try to plot more expressions than Mathcad's capacity.

misplaced comma

Commas can be used to separate

- the arguments of a function
- the first two elements of a range in the definition of a range variable
- expressions in a plot
- elements in an input table
- subscripts in a matrix

Any other use of commas is illegal in Mathcad expressions.

missing operand

You have not filled in the indicated placeholders. You should place numbers or expressions in these placeholders.

missing operator

This expression is missing an operator at the indicated location.

must be 3-vector

The vector cross product is defined only for vectors having three elements.

must be array

You performed an operation that requires an array on a scalar. For example, you see this error if you define a superscripted variable as a scalar. Since a superscripted variable represents a column of a matrix, you must define it as a vector.

For surface plots, the array plotted must have at least two rows or two columns.

must be dimensionless

This expression uses units somewhere units are not permitted.

Units are not permitted...

- in the arguments of most functions
- in exponents
- in subscripts and superscripts

must be increasing

The first argument of **lspline**, **pspline**, **cspline**, **linterp**, and **hist**, as well as the second argument of **interp**, must be a vector whose elements are strictly increasing.

Note that all vectors start at the 0th element unless you specifically set **ORIGIN** to 1. If you do not explicitly define the 0th element, Mathcad sets it to zero.

must be integer

This expression contains a non-integer expression where an integer is required.

Some possible places to look are:

- Superscripts and subscripts
- Arguments to certain functions such as identity, Bessel functions of integer order, **csort**, **rsort**, Kronecker delta, antisymmetric tensor function.

Note that although you can define range variables with fractional values, you cannot use these as subscripts or superscripts.

must be nonzero

This function is undefined at zero.

must be positive

This plot has a logarithmic axis in which either the limits, or some of the values are not positive.

Negative numbers and zero cannot be placed anywhere on a logarithmic axis.

must be range

This expression requires a range variable.

If the expression contains either a summation or an iterated product, check to see that the index is in fact a range variable. The index is directly under the sigma or pi. It should have been defined in an earlier statement.

must be real

This expression contains an imaginary or complex valued expression somewhere in which it is not allowed.

Examples are subscripts and superscripts, arguments to Bessel functions, the Heaviside step function, **mod**, and **angle**.

must be scalar

You used a vector or matrix expression where a scalar is required.

must be square

This error marks a non-square matrix in an operation or function that requires a square matrix, such as determinants, inverse, raising to a power, or the **eigenvals** and **eigenvec** functions.

must be vector

This error marks a matrix or scalar in an operation that requires a vector, for example, with the vector sum operator.

nested solve block

You used two "**Given**"s in a row with no intervening **find** or **minerr**. Mathcad does not allow nested solve blocks.

You can however, define functions with solve blocks and use those function in other solve blocks.

no matching Given

This error marks a **find** or **minerr** function with no matching **Given**. Each solve block that ends with **find** or **minerr** must begin with a region containing only the name **Given**.

non-scalar value

You used a vector or ranged expression where a scalar is required.

For example, you see this message if **i** is a range variable and you try to enter an equation like **x := i**. You cannot define one range variable in terms of another.

not a name

You used a number or other combination of symbols where Mathcad requires a name, for example, as the second argument of the root function.

not converging

Mathcad was unable to compute an answer for an integral, derivative, **root**, **find**, or **minerr** function within the required tolerance. Try setting **TOL** to be larger, or try a different guess value.

overflow

You evaluated an expression that exceeds the largest number that Mathcad can represent.

significance lost or ***significance reduced***

These error messages indicate that you tried to evaluate a function for a value beyond the accurate range for the function.

For example, you see this message if you try to evaluate sin(10^100). Since the value of sin(10^100) depends on the ones digit of 10^100, any value that Mathcad could return would have no significant digits. Instead of returning a value of dubious accuracy, Mathcad shows one of these error message.

singularity

You evaluated a function or performed an operation at an illegal value. For example, you see this error if you divide by zero or if you try to invert a singular matrix.

stack overflow

This expression overflows Mathcad's internal stack. Simplify the expression or divide it into two subexpressions.

too few arguments

This expression contains a function with fewer than the required number of arguments.

For built-in functions, click on the function name and press **F1** to see the correct number and types of arguments. For functions you defined, check the definition.

too few constraints

This error marks a **find** or **Given** with fewer constraints than variables to be solved. Add dummy constraints or decrease the number of variables to be solved for.

too few elements

This error message marks a Fourier transform, cubic spline, or linear interpolation function applied to a vector with too few elements.

Splines and linear interpolation require vectors with at least two elements. Fourier transform functions and their inverses require at least four elements.

too few subscripts

You used one subscript on a matrix. You must use two subscripts separated by a comma to specify a matrix element.

too large to display

You tried to display a vector or matrix bigger than Mathcad can display. Try breaking it down into pieces before displaying it.

too many arguments

The indicated expression contains a function with more than the required number of arguments.

For built-in functions, click on the function name and press **F1** to see the correct number and types of arguments. For functions you defined, check the definition.

too many constraints

You included more than fifty constraints in a solve block.

too many subscripts

You used two or more subscripts on a vector.

too many points

You tried to plot more points than Mathcad can handle in one plot.

undefined (reverse video)

This variable or function is undefined. To define it, enter the variable or function name, followed by a colon (**:**) and an expression or number for its definition.

This error often means you have typed an equals sign (**=**) instead of a colon to define a variable. To define a variable, you must use a colon. The equals sign tells Mathcad to display a result.

You also see this message if you use a variable incorrectly in a global definition. If you use a variable on the right side of a global definition, the variable must be globally defined above the definition in which it is used. If you use a locally defined variable or a variable whose global definition is below the place where it is used, Mathcad marks the variable as undefined.

NOTE: An undefined error is often an indication that another equation further up in the document is in error. If this is the case, all expressions that use the expression in error will be displayed in inverse video.

underflow

You evaluated an expression smaller in magnitude than the smallest positive number that Mathcad can represent.

unmatched parenthesis

This expression does not contain equal numbers of **(**'s and **)**'s.

wrong size vector

This error marks a Fourier transform function whose argument has the wrong number of elements.

- **fft** requires a vector with 2^n elements.
- **ifft** requires an argument vector with 2^n+1 elements, where n is a whole number greater than 0.

Use **cfft** and **icfft** for arrays and vectors of any size.

NOTE: Unless you specify otherwise, all Mathcad arrays begin with 0th element. If you are used to thinking of arrays as beginning at the 1st element, reset **ORIGIN** to 1.

end of file

You have asked Mathcad to read more data values from a datafile than there are values in the datafile. For example, if your datafile has ten numbers in it, and you type...

i:1;100

x:READ(file)[j

Mathcad places the ten numbers in the **x** vector and displays this message.

too large to compute

Mathcad displays this when somewhere in the course of a calculation, it is forced to generate a matrix having more than 64 elements.

In the full version of Mathcad 3.1, the maximum matrix size is 8000 elements.

range not allowed

Mathcad displays this message when you use a range variable in an inappropriate place, for example a solve block.

If you want to iterate with a solve block, try defining the a function in terms of the solve block and using that function for iteration.

FOURIER TRANSFORMS

Click on function and argument names for more detail:

fft(**v**)	cfft(**A**)
ifft(**u**)	icfft(**B**)
FFT(**v**)	CFFT(**A**)
IFFT(**u**)	ICFFT(**B**)

Additional Topics:

- Frequency axis
- Aliasing
- Two dimensional fft's
- Power and phase spectrum
- fft/ifft vs. cfft/icfft

REAL FOURIER TRANSFORM

fft(v) Returns the Fourier transform of a vector. The result is a $1+2^{m-1}$ element vector whose jth element is given by:

$$c_j = \frac{1}{\sqrt{n}} \sum_k A_k e^{i(2\pi j/n)k}$$

Arguments:

- **v** must have elements (m>2).
- All elements in **v** are real.

For vectors with complex values or with any number of elements, use **cfft** instead.

For 2 dimensional Fourier transforms, use **cfft** instead.

v is a vector having 2^m elements (m>2), all of which are real.

For vectors not satisfying these criteria, use **cfft** instead.

v is a vector having 2^m elements (m>2), all of which are real.

For vectors not satisfying these criteria, use **CFFT** instead.

INVERSE FOURIER TRANSFORM OF REAL DATA

ifft(u) Returns the inverse Fourier transform corresponding to **fft**. The result is a 2^m element vector whose jth element is given by:

$$c_j = \frac{1}{\sqrt{n}} \sum_k u_k e^{-i(2\pi j/n)k}$$

Arguments:

- **u** is a vector with elements (m>2).

Use this only when the original data was purely real. If the original data was complex, use **icfft** instead.

u is a vector having $1+2^{m-1}$ elements where m>2.

FOURIER TRANSFORM

cfft(A) Returns the Fourier transform of a vector or matrix. The result has the same number of rows and columns as **A**. If **A** is a vector, the result is given by:

$$c_j = \frac{1}{\sqrt{n}} \sum_k A_k \, e^{i(2\pi j/n)k}$$

Arguments:

- **A** can be a vector or a matrix.

If...

- **A** is a vector with elements (m>2), and...

- All elements in the vector are real.

...use **fft** instead. For such vectors, the second half of the spectrum is the mirror image of the first and need not be computed.

For matrix arguments, **cfft** returns the two dimensional Fourier transform.

A is a vector or matrix.

INVERSE FOURIER TRANSFORM

icfft(B) Returns the inverse Fourier transform corresponding to **cfft**.

Arguments:

- **B** can be a vector or a matrix.

B is a vector or a matrix.

Frequencies corresponding to coefficients

Both **fft** and **cfft** return vectors whose elements are the complex amplitudes of the various frequencies in the original signal.

To recover the actual frequencies, you must know:

- The sampling frequency of the original signal.
- The number of samples in the original signal.

Given these parameters, the frequency associated with the jth element is given by:

f_s = sampling frequency

N = number of samples

Aliasing

Because the fft is a discrete approximation to a continuous transform, you may encounter misplaced harmonics. This is called aliasing. To avoid aliasing:

- Make sure the signal has finite bandwidth.
- Make sure the sampling frequency is at least twice the bandwidth (the Nyquist frequency.)

Power and phase spectrum

The **fft** and **cfft** functions return the real and imaginary parts of the Fourier transform. You can recover the power spectrum and phase spectrum from these by using the vectorize operator:

$$\overrightarrow{\mathrm{arg}(\mathrm{fft}(v))} \qquad \text{phase spectrum}$$

To apply the vectorize operator,

- Enclose the entire expression in a selection box.
- Press **Ctrl+-**

The vectorize operator causes the magnitude operator to return the magnitude of every element in **v** instead of the magnitude of the vector **v** itself.

fft/ifft vs. cfft/icfft

Use the **fft**/**ifft** pair if both of the following statements are true:

- The time domain data is real.
- The data vector has elements (m>2).

Use the **cfft**/**icfft** pair if any of the following statements are true:

- The time domain data is complex
- The data vector does not have elements
- The data is in a matrix, not a vector.

For real data in the time domain, the Fourier transform has conjugate symmetry. Because of this, **fft** will drop the redundant second half of the result. This is why the vector returned by **fft** is half the size of the original vector.

The **ifft** function reconstructs this redundant half. For this reason, you should use **fft** and **ifft** together.

The **cfft**/**icfft** pair makes no such assumptions about symmetry. Although these functions are not as efficient for real data, they can nevertheless be used.

REAL FOURIER TRANSFORM

FFT(v) Returns the Fourier transform of a vector. The result is a $1+2^{m-1}$ element vector whose jth element is given by:

$$c_j = \frac{1}{n}\sum_k A_k e^{-i(2\pi j/n)k}$$

Arguments:

- **v** must have elements (m>2).
- All elements in **v** are real.

For vectors with complex values or with any number of elements, use **CFFT** instead.

For 2 dimensional Fourier transforms, use **CFFT** instead.

INVERSE FOURIER TRANSFORM OF REAL DATA

IFFT(u) Returns the inverse Fourier transform corresponding to FFT. The result is a 2^m element vector whose jth element is given by:

$$c_j = \sum_k u_k e^{i(2\pi j/n)k}$$

Arguments:

- **u** is a vector with elements (m>2).

Use this only when the original data was purely real. If the original data was complex, use ICFFT instead.

FOURIER TRANSFORM

CFFT(A) Returns the Fourier transform of a vector or matrix. The result has the same number of rows and columns as **A**. If **A** is a vector, the result is given by:

$$c_j = \frac{1}{n}\sum_k A_k e^{-i(2\pi j/n)k}$$

Arguments:

- **A** can be a vector or a matrix.

If...

- **A** is a vector with elements (m>2), and...
- All elements in the vector are real.

...use FFT instead. For such vectors, the second half of the spectrum is the mirror image of the first and need not be computed.

For matrix arguments, **CFFT** returns the two dimensional Fourier transform.

INVERSE FOURIER TRANSFORM

ICFFT(B) Returns the inverse Fourier transform corresponding to **CFFT**.

Arguments:

- **B** can be a vector or a matrix.

MATH FONTS

In equations, unlike text, each name or number is assigned a tag, and each tag is assigned a font, size and style. This lets you change the font characteristics of many names at once. It also means that all characters in a name must have the same font characteristics.

Changing the font for variables

Choose **Modify Font Tag** from the **Math** menu. In the dialog box, make sure the Tag Name is 'Variables.' Then choose the font family, size, and style. This affects all variables in the document.

If you want to change the font of a particular name without affecting anything else, you must give it its own tag.

Changing the font for constants

Choose **Modify Font Tag** from the **Math** menu. In the dialog box, make sure the Tag Name is 'Constants.' Then choose the font family, size, and style. This affects all constants in the document.

Font tags

Choose **Modify Font Tag** from the **Math** menu. In the dialog box, Choose a tag from the drop-down list at the top of the dialog box. Then choose the font family, size, and style you want to assign to that tag. This affects every name or number tagged with the selected tag.

Applying a font tag to a name or number

Click in the name or number and choose **Apply Font Tag** from the **Math** menu. Mathcad brings up a scrolling tag list with your selection's tag highlighted. Choose a tag from the scrolling list to apply it to your selection.

Greek variables

Type the appropriate character from the Symbol font. Then press **Ctrl**+**G**. Alternatively, click twice on the top button of the operator palette to display the Greek letter palette. If you don't find the letter you're looking for, click again on the top button to display additional Greek letters.

FUNCTIONS

Click on a function name or its arguments to get more info...

Trigonometric

sin(z)	cos(z)
tan(z)	csc(z)
sec(z)	cot(z)

Hyperbolic

sinh(z)	cosh(z)
tanh(z)	csch(z)
sech(z)	coth(z)

Inverse trig	**Inverse hyperbolic**
asin(z)	asinh(z)
acos(z)	acosh(z)
atan(z)	atanh(z)

Exponential and logarithmic

exp(z)	e to the z
ln(z)	Natural log
log(z)	Log base 10

Bessel

J0(x)	Y0(x)
I0(x)	K0(x)
J1(x)	Y1(x)
I1(x)	K1(x)
Jn(m, x)	Yn(m, x)
In(m, x)	Kn(m, x)

Complex functions

Re(z)	Real part of z
Im(z)	Imaginary part of z
arg(z)	Argument of z

Conditional functions

if(cond, tval, fval)

until(expr1, expr2)

δ(m,n)	1 if m=n, 0 otherwise
ε(i,j,k)	Antisymmetric tensor of rank 3
Φ(x)	Unit step function

Array Functions

augment(A,B)	Join two arrays
cols(M)	Number of colmns
eigenvals(M)	Vector of eigenvalues
eigenvec(M,z)	Eigenvector of M corresponding to z
identity(n)	n x n identity matrix
last(v)	Index of last element in v.
length(v)	Number of elements in v
max(M)	Largest element in M
min(M)	Smallest element in M

rows(M)	Number of rows
tr(M)	Sum of elements on diagonal

Miscellaneous

ceil(x)	Smallest integer $\geq$ x
floor(x)	Greatest integer $\leq$ x
mod(x,y)	Remainder of x/y with sign of x
angle(x,y)	Positive angle from x axis to the point (x,y)

TRIGONOMETRIC FUNCTIONS

sin(z) **cos(z)**

tan(z) **csc(z)**

sec(z) **cot(z)**

Arguments:

- **z** must be in **radians**
- **z** must be a scalar.
- **z** must be dimensionless

Related Topics:

Arrays as arguments
Numerical issues
Working with degrees

Working with degrees

Mathcad's trigonometric functions assume the argument is in radians. You can define your own trig functions to accept degrees as arguments.

Numerical issues

Because Mathcad evaluates functions using double precision floating point numbers, calculations are inevitably affected by roundoff errors, particularly when the function's argument takes on extreme values.

Trigonometric functions are subject to roundoff error in the following cases:

Large arguments

For real arguments approaching 10**9 in magnitude, these functions begin to lose precision. Mathcad signals the loss of precision with an error message.

Singularities

The tangent function is undefined at integer multiples of π/2. Arguments near this singularity are subject to precision errors.

Multiples of π

Since Mathcad's value of π can only be an approximation of , sin() is very close to, but not exactly 0. Similarly, tan(/2) is not undefined, but returns a very large number.

INVERSE TRIG FUNCTIONS

asin(z)

acos(z)

atan(z)

Arguments:

- **z** must be a scalar.
- **z** must be dimensionless

Values returned are from the **principal branch** of these functions.

Related Topics:

Arrays as arguments
Numerical issues
Converting to degrees

Converting to degrees

Mathcad's inverse trigonometric functions return angles in radians. To convert a result to degrees, type 'deg' in the units placeholder at the end of an equation.

Numerical issues

Because Mathcad evaluates functions using double precision floating point numbers, calculations are inevitably affected by roundoff errors, particularly when the function's argument takes on extreme values.

HYPERBOLIC FUNCTIONS

sinh(z)	**cosh(z)**
tanh(z)	**csch(z)**
sech(z)	**coth(z)**

Arguments:

- **z** must be in **radians**
- **z** must be a scalar.
- **z** must be dimensionless

INVERSE HYPERBOLIC FUNCTIONS

asinh(z)

acosh(z)

atanh(z)

Arguments:

- **z** must be a scalar.
- **z** must be dimensionless

Values returned are from the **principal branch** of these functions.

EXPONENTIAL AND LOGARITHMIC FUNCTIONS

exp(z) The number **e** raised to the power **z**.

log(z) Base 10 log of **z**.

ln(z) Natural log (base **e**) of **z**.

Arguments:

- **z** must be a scalar.
- **z** must be dimensionless
- For **log** and **ln** functions, **z** cannot be zero.

For complex z, the log functions return values from the **principal branch** of these functions. In other words:

ln(z) = ln(|z|) + i arg(z)

Related Topics:

Logs to other bases

Logs to other bases

To evaluate logs to an arbitrary base n, define

BESSEL FUNCTIONS

J0(x)	**Y0(x)**
J1(x)	**Y1(x)**
Jn(m,x)	**Yn(m,x)**
I0(x)	**K0(x)**
I1(x)	**K1(x)**
In(m,x)	**Kn(m,x)**

Arguments:

- **x** must be a real scalar for I and J.
- **x** must be a positive real scalar for Y and K.
- **x** cannot have units
- **m** must be an integer between 1 and 100 inclusive.

COMPLEX FUNCTIONS

Re(Z) Real part of **Z**

Im(Z) Imaginary part of **Z**

arg(z) Argument of **z**, between -π and .

Arguments:

- **Z** can be a real or complex scalar or matrix.
- **z** can be a real or complex scalar.

FUNCTIONS FOR ITERATION

until(expr1, expr2) This function allows you to halt an iterative process based on a condition.

Arguments:

- **expr1** is a test expression. Mathcad halts iteration when this becomes negative.
- **expr2** is the value returned by the **until** function at each iteration.

CONDITIONAL BRANCHING

if(cond, tval, fval) This function lets Mathcad return one of two values depending on the value of a logical condition.

Arguments:

- **cond** is usually an expression involving a logical operator. For example (i<2).
- **tval** is the value returned when **cond** is true.
- **fval** is the value returned when **cond** is false.

HEAVISIDE STEP FUNCTION

Φ(x) This returns 0 if **x** is negative, and 1 otherwise

Arguments:

- **x** must be a real scalar.

Since this function is discontinuous at 0, you should not integrate expressions involving this function over an interval containing 0. Instead, divide the region of integration into two parts, integrate over each separately and add the result.

For the same reason, the derivative of this function is undefined at 0.

TRUNCATION FUNCTIONS

floor(x) greatest integer ≤ **x**

ceil(x) smallest integer ≥ **x**

Arguments:

- **x** must be real.

Related Topics:

Mantissa of a number
Roundoff

Mantissa of a number

To get just the decimal part of a positive number, define:

mantissa(x) := x - floor(x)

Rounding a number

To round a number to the nearest integer, define:

```
round(x) := if(x - floor(x)<.5,floor(x),ceil(x))
```

z can be anything except a vector or a matrix.

z can be anything except a vector or a matrix.

z cannot have units. If you want to pass a number with units, divide them out first. For example

z := 0*ft   sin(z/ft) = 0

z can be anything except a vector or a matrix.

z cannot be 0.

z cannot have units.

If you want to pass a number with units, divide them out first. For example

z := 10*ft   log(z/ft) = 1

z can be any valid Mathcad expression.

x must be a real scalar for I and J.
x must be a positive real scalar for Y and K.

x cannot have units.

If you want to pass a number with units, divide them out first. For example

z := 0 ft   J0(z/ft) = 1

m

- must be a nonnegative integer.
- cannot have units.
- must be less than or equal to 100.

x must be real.

x and **y** must be real.

Both **m** and **n** must be integers without units.

i, **j** and **k** must be real integers with no units.

cond is any expression that returns a scalar.

If the scalar is 0, Mathcad considers the condition 'false'
If the scalar is anything else, Mathcad considers it 'true'.

For example,

(1**=**2) = 0 **<-- False**

(2>1) = 1 **<-- True**

tval can be anything at all.

It is the value the 'if' function returns if **cond** is not equal to 0.

fval can be anything at all.

It is the value the 'if' function returns if
cond = 0

expr1 must be a real scalar.

When this expression becomes negative, the 'until' function will terminate execution.

You should arrange for this value to eventually be negative. Failure to do so will result in an endless loop.

expr2 can be anything at all.

This is the value returned by the 'until' function at each iteration.

MODULO FUNCTION

mod(x,y) This function returns the remainder on dividing **x** by **y**. Result has same sign as **x**.

Arguments:

- **x** and **y** should both be real scalars.
- **y** must be nonzero.

POLAR ANGLE

angle(x,y) This function returns an angle (in radians) between the positive x axis, and the point (x,y). The value returned is always between 0 and 2π.

Arguments:

- Both **x** and **y** must be real.

KRONECKER DELTA FUNCTION

δ(m,n) This function returns 1 if m and n are equal. It returns 0 otherwise

Arguments:

- Both **m** and **n** must be integers without units.

Since this function is discontinuous at 0, you should not integrate expressions involving this function over an interval containing 0. Instead, divide the region of integration into two parts, integrate over each separately and add the result.

For the same reason, the derivative of this function is undefined at 0.

ANTISYMMETRIC TENSOR FUNCTION

ε(i,j,k) This function returns the completely antisymmetric tensor of rank 3. Result is 0 if any two are the same, 1 for even permutations, -1 for odd permutations.

Arguments:

- i, j, and k must be integers between 0 and 2 (or ORIGIN and ORIGIN+2)

DIMENSIONS OF A MATRIX

rows(M) Returns the number of rows in **M**.

cols(M) Returns the number of columns in **M**.

Arguments:

- **M** can be a vector or a matrix.

LENGTH OF A VECTOR

length(v) Returns the number of elements in the vector **v**.

last(v) Returns the index of the last element in the vector **v**.

Arguments:

- **v** must be a vector.

To get the Euclidean length of **v**, use the magnitude operator **|**.

MAXIMA AND MINIMA OF ARRAYS

max(M) Returns the largest element in **M**.

min(M) Returns the smallest element in **M**.

Arguments:

- **M** can be a vector or a matrix.

When **M** has complex elements:

- **max(M)** returns the largest real part plus i times the largest imaginary part.

- **min(M)** returns the smallest real part plus i times the smallest imaginary part.

IDENTITY MATRIX

identity(n) Returns an n x n identity matrix (a matrix of 0's with 1's along the diagonal).

Arguments:

- n must be a positive integer.

EIGENVALUES AND EIGENVECTORS

eigenvec(M,z) Returns an eigenvector associated with eigenvalue **z** of the matrix **M**. The eigenvector is normalized to unit length.

eigenvals(M) Returns a vector whose elements are the eigenvalues of **M**.

Arguments:

- **M** must be a square matrix.
- The elements of **M** should be real.
- **z** should be an eigenvalue of **M**.
- **z** can be complex.

TRACE OF A MATRIX

tr(M) Returns the sum of the elements along the diagonal of **M**.

Arguments:

- **M** must be a square matrix.

JOINING ARRAYS

augment(A,B) Returns an array formed by placing **B** to the right of **A**.

Arguments:

- **A** and **B** can be vectors or matrices.
- **A** and **B** must have the same number of rows.

A and **B** are arrays having the same number of rows.

M is either a matrix or a vector.

v is a vector

n

- is an integer
- must be dimensionless

A is a matrix or vector having the same number of rows as **B**.

B is a matrix or vector having the same number of rows as **A**.

M is a square matrix.

M

- must be a square matrix.
- must have real elements.

z is a real or complex eigenvalue of the matrix **M**.

M

- must be a square matrix.
- must have real elements.

RECTANGULAR GRAPHS

Creating a graph

Press **@** and place expressions to be plotted in the middle placeholder of each axis. Then press F9 to see graph.

Deleting a graph

Click on graph to select it. Then press Shift+Del to delete it.

Moving a graph

Enclose graph in dashed selection rectangle. Then drag it or cut and paste it to another spot.

Changing the size of a graph

Enclose graph in dashed selection rectangle. Move pointer to right or bottom edge where it turns into a double headed arrow. Hold down the mouse button and move the pointer to resize.

Graph formatting

Double-click on the plot. A dialog box presents options for logarithmic axes, gridlines, legends, trace types, markers, colors, and axis limits.

Graph format

Double-click on a plot. The options on the Graph format dialog box are shown below. Click on an option to learn more about it.

Axis settings

- Log scale
- Grid lines
- Numbered
- Clip to markers
- Auto grid

Trace settings

- Label
- Symbol
- Line
- Color
- Trace type

Legend settings

- Hide Arguments
- Hide Legend
- Use for defaults

Selecting trace characteristics

To select the characteristics for each trace:

- Click on the trace's name under the 'Label' column of the scrolling list in the middle of the dialog box. The characteristics assigned to that trace appear in the five text boxes under the corresponding columns.

- To change the name of the trace, type the new name in the text box under the 'Label' column. Then click on the small 'OK' next to the label. This changes the name that appears on the legend.

- Assign a color, line type, marker and trace type to this trace. To do so, click on the arrow beside each text box to see a drop-down list of options.

- To select a graph type for the second or subsequent traces, choose the appropriate label from the scrolling list of labels. Then assign characteristics to the selected trace the same way you assigned them to the first.

- When you're done, click 'OK.'

When you click outside the graph, Mathcad redraws it using the formatting information you have set. If you are in manual mode, press F9 to redraw the graph.

Creating a rectangular graph

To create a rectangular graph,

- Define the expressions to be graphed. These will generally involve range variables.
- Press **@** to create a graph region.
- Fill in the six placeholders

Placeholders for axis limits are optional.

SURFACE PLOTS

Creating a surface plot

Press **Ctrl+2** and place the matrix to be plotted in the placeholder. Then press F9 to see the surface plot.

Deleting a surface plot

Click on surface plot to select it. Then press Shift+Del to delete it.

Moving a surface plot

Enclose the surface plot in a dashed selection rectangle. Then drag it or cut and paste it to another spot.

Changing the size of a surface plot

Enclose the surface plot in a dashed selection rectangle. Move the pointer to the right or bottom edge where it turns into a double headed arrow. Hold down the mouse button and move the mouse to resize the surface plot.

Graph formatting

Double-click on the surface plot. A dialog box presents options for changing vertical scale, perspective, hidden lines, shading, and displaying axes and borders.

Creating a surface plot

To create a surface plot:

- Define the matrix of elements to plot.
- Press **Ctrl+2** to a create a surface plot region.
- Place the name of the matrix in the placeholder.

The matrix you plot must have at least two rows or two columns.

Surface plot format

Double-click on a surface plot. The options on the Surface Plot Format dialog box are shown below. Click on an option to learn more about it.

Perspective settings

- Rotation
- Tilt
- Vertical scale

Surface and axis options

- Hide lines
- Patch Plot
- Show axes
- Show border

Shading options

- Color spectrum
- White
- Grayscale

IMPORTING BITMAPS

Importing a bitmap from the clipboard

Mathcad 3.1 can import bitmaps from other applications via the Windows clipboard and act as an OLE server. However, this feature has been disabled for this demo version of Mathcad.

Deleting a bitmap

Click on the bitmap to select it. Then press Shift+Del to delete it.

Moving a bitmap

Enclose the bitmap in a dashed selection rectangle. Then drag it or cut and paste it to another spot.

Changing the size of a bitmap

Enclose the bitmap in a dashed selection rectangle. Move the pointer to the right or bottom edge where it turns into a double headed arrow. Hold down the mouse button and move the mouse to resize the image.

Importing graphics from a file

Press **Ctrl+5** and type the name of the file in the placeholder. The file should contain a Windows bitmap.

Framing a bitmap

Double-click on the picture to bring up a dialog box.

Importing graphics from plotter files

Programs which can drive a pen plotter can usually send the data to a file instead of directly to the pen plotter. You can then import this file into your Mathcad document. This feature has been disabled for this demo version of Mathcad.

Importing graphics from another directory

By default, the name you put in the placeholder is the filename of the graphics file in the current directory. To import a graphics file from a another directory:

- Click on the filename in the placeholder of the graphics region.
- Choose **Associate Filename** from the **File** menu.
- Enter the full pathname to the file of interest.

Framing a graphics image

To place a box around an imported graphics image:

- Click on the image with the left mouse button. This encloses it in a selection box.
- Double-click on the picture to display a dialog box.
- Click on the box labelled 'Show border.'

Importing graphics images from a file

You can insert graphics into Mathcad by referring to a file containing the graphics instead of by pasting the image directly from the clipboard. The advantages are:

If several Mathcad documents refer to the same picture, you only need one copy of that picture.

If you want to change a picture in several Mathcad documents, you only have to change one picture file.You need not change every single Mathcad document.

To import graphics by referring to a file, do the following:

- Click the mouse wherever you want the graphics image plot
- Press **Ctrl**+**5** to create a graphics region.
- Type the name of the file into the placeholder. If the file is in a directory other than the default directory, use the Associate Filename command from the **File** menu to specify its complete pathname.
- Press F9 or, in automatic mode, click outside the plot.
- If you change the file on disk, you must click in the graphics region and press **F9** to update your document.

Deleting a graphics region

To **delete** a graphics region:

- Click on the graphics region. This encloses it in a selection box.
- Press **Shift**+**Del**. This removes the graphics region from the document and places it on the clipboard.

To **delete several** graphics regions:

- Click on the first region as described above.
- Hold down the **Shift** key and click on the remaining regions.
- Press **Shift**+**Del**. This removes the graphics regions from the document and places them on the clipboard.

Selecting a graphics region:

- Click the mouse just outside the graphics region. This anchors one corner of the selection rectangle.
- Press and hold down the left mouse button. With the button still held, drag the mouse. A selection rectangle framed by dashed lines emerges from the anchor point.
- When the selection rectangle just encloses the graphics region, let go of the left mouse button.

Resizing graphics

To change the length and width of a graphics region:

- Select the graphics region.
- Move the cursor to the edge of the region. It should change to a double headed arrow.
- With the mouse button still pressed, move the mouse. The graphics region will be stretched in the direction of motion.
- Once the graphics region is the right size, let go of the mouse button.

Moving a graphics region

There are two ways to move a graphics region:

Cutting and pasting

- Select the region you want to move by clicking on it.
- Press **Shift**+**Del**.
- Click where you want to move the region.
- Press **Shift+Ins**.

Dragging

- Select the region by dragging the mouse around it.
- Click the cursor inside the region.
- Press and hold down the mouse button while dragging the mouse. An outline of the region moves with the mouse.
- When the outline of the region is where you want it,let go of the mouse button.

Selecting a graphics regions by dragging

- Click the mouse just outside the graphics region. This anchors one corner of the selection rectangle.
- Press and hold down the left mouse button. With the button still held, drag the mouse. A selection rectangle framed by dashed lines emerges from the anchor point.
- When the selection rectangle touches all the regions you want to select, let go of the left mouse button.

Rotation

Rotation must be between 0 and 90 degrees.

Rotation = 0
You look directly down first column of matrix.First row points right.

Rotation = 90
You look directly down first row of matrix.First column points left.

Tilt

Tilt must be between 0 and 90 degrees.

Tilt = 0
: You look edge on the plane of the matrix. Effect is like standing at sea level and looking at mountains.

Tilt = 90
: You look from directly above. Effect is like flying directly over those same mountains.

Vertical Scale

The vertical scale controls how 'bumpy' a plot looks. A large number magnifies variations in the surface, making it look bumpy. A small number makes a surface appear flatter.

Hidden lines

When 'White' shading is selected, this controls whether a surface is transparent or opaque.

When either 'Color spectrum' or 'Grayscale' is selected, this controls whether Mathcad draws the frame around each patch.

Patch Plot

When checked, patches making up surface are constrained to be flat. Result is a discontinuous surface.

When left unchecked, patches making up surface are allowed to tilt in whatever direction is required to form a continuous surface.

Show axes

When checked, Mathcad displays numbered three-dimensional axes.

Show border

When checked, the surface plot is enclosed in a rectangle.

Color spectrum

Lowest points on surface are blue. Colors progress through green, yellow, and finally, red.

White

Surface patches are white. When this is selected, checking 'Hide lines' makes the surface opaque.

Gray scale

Highest points on surface are white. Lowest are black. Those in between are appropriate shades of gray.

Log scale
When this box is checked, the selected axis is logarithmic. The axis limits must be positive.

Grid lines
When this box is checked, the tickmarks on the selected axis are replaced by gridlines.

Numbered
When this box is checked, the tickmarks on the selected axis are numbered.

Auto grid
When this box is checked, Mathcad automatically chooses the number of tick marks.

To choose the number of tick marks yourself:

- Click in the box to remove the checkmark.
- Type an integer from 1 to 99 in the text box labelled "No. of grids." Make sure the number you type isn't so large that the labels on the tickmarks crowd together too much.

Clip to markers
This lets you override Mathcad's choice of axis limits. To do so, check this box and:

- Click in the plot to reveal placeholders for the axis limits.
- Type in the new axis limits in the corresponding placeholders.
- Press **F9** to redisplay the plot.

Label
This is the name of the trace as it appears on the legend. Feel free to change Mathcad's default names

Symbol
This controls whether each point on the curve is marked with a symbol. If you have a lot of points packed closely together, you should probably select 'none.'

Line
This controls whether the line is solid, dotted, dashed, or whether it consists of alternating dashes and dots. This provides a useful way to distinguish unmarked curves in black and white printouts.

Color
This controls whether the selected trace is red, blue, green, magenta, cyan (a light blue), or brown. Mathcad ignores this on monochrome terminals.

Trace type
This controls whether the selected trace is shown as a bar chart, as a curve, as a stepped curve, or error bars.

Hide Arguments

When this box is checked, Mathcad hides the math expressions you typed in the middle placeholder of each axis.

Hide Legend
When this box is checked, Mathcad hides the legend that appears under the plot.

Use for defaults
When you check this box, Mathcad uses the current settings for all other plots in your document.

MATHCAD ELECTRONIC HANDBOOKS

The Mathcad series of Electronic Handbooks contains mathematical and engineering formulas, physical constants, properties of various materials and other useful information you would ordinarily look up in a handbook.

The following Handbooks ship with Mathcad.

- Getting Started
- Standard

To open a handbook:

- Choose **Open Handbook** from the **Help** menu.
- Double-click on the Handbook's filename in the dialog box. This name always ends in '.hbk'. Clicking once on the name shows the complete Handbook title in the dialog box.

Additional Topics

- Browsing through a Handbook
- Searching Through a Handbook
- Pasting From Handbooks
- Additional Handbooks

Additional Handbooks

Additional Electronic Handbooks currently available as of March 1992 are:

Mathcad Treasury of Methods and Formulas
by Paul R. Lorczak
CRC Materials Science and Engineering Handbook
Edited by James Shackelford and William Alexander
Machine Design and Analysis
from Hicks' Standard Handbook of Engineering Calculations

For more information, contact MathSoft at 1-800-MATHCAD. In Massachusetts, or outside the USA, call 1-617-577-1017.

Browsing Through a Handbook

There are two ways to move around a Handbook:

- Use the buttons on the Handbook control palette along the left side of the window.
- Double-click on certain regions with the right mouse button. Consult the specific Handbook for conventions on how to identify such a region.

Note that when you click on a region that goes someplace else, an appropriate message appears on the message line.

Because this demo version of Mathcad is limited to one screen, you cannot scroll beyond the end of a page.

Searching Through a Handbook

You can find information on a topic by looking for it in either the Table of Contents or the Index. You can also search for every occurrence of a text string as follows:

- Open a Handbook and choose **Search Handbook** from the **Help** menu.
- Type the text string in the Choose word text box. Alternatively, scroll through the list of topics. An [eq] after a string means Mathcad will search only equations for that string. Otherwise, Mathcad searches only text.
- Click the Search Handbook button. Mathcad shows how many occurrences of that word are in the Handbook, what sections they are in, and how many there are in each section.
- To open a section, click on its title in the scrolling list and click on the Goto Section button.
- Click the Next Occurrence button. Mathcad scrolls down until the next occurrence of the string is visible in the Handbook window. This button grays out when you reach the last occurrence.
- Clicking the Previous Occurrence button scrolls up until the previous occurrence of the string is visible in the Handbook window. This button grays out when you reach the first occurrence.

Since this demo version of Mathcad cannot scroll beyond one page, only occurrences on the first page of a Handbook section can be displayed.

Pasting from a Handbook

Most regions in a Handbook can be pasted directly into your document. To paste a region:

- Place the cursor wherever you want to paste the region.
- Using the *right* mouse button, double-click on the Handbook region you want to paste.

Note that not all Handbook regions can be pasted when double-clicked on. Some will do nothing, others will jump to a different section of the Handbook.

Getting Started Handbook

A guided tour of Mathcad illustrating some features and techniques. You can access numerous sample documents through this Handbook.

Standard Handbook

A smorgasbord of formulas from planar and solid geometry, antenna engineering, oscillators, and mechanics. Also contains tables of properties for various materials.

Mathcad Treasury of Methods and Formulas
by Paul R. Lorczak

This Handbook uses Mathcad to explain widely used mathematical methods. It demonstrates how to combine Mathcad's features to solve a wide range of problems.

CRC Materials Science and Engineering Handbook
Edited by James Shackelford and William Alexander

This provides interactive on-line access to over 140 tables of data on various materials. Double-clicking on one of these live numbers makes it appear in your document, complete with units and ready for any calculations.

Machine Design and Analysis
from Hicks' Standard Handbook of Engineering Calculations

This Electronic Handbook provides easy access to over 125 practical calculations from the widely used *Hicks Handbook*.Double-clicking on these formulas copies them directly to your document where you can use them for your own calculations.

ABOUT THIS DEMO

Many features of the full version of Mathcad 3.1 have been disabled in this demo version. These include:

- The ability to save files.
- Creating documents larger than one 8.5 x 11 page.
- Search and Replace commands.
- Creating matrices larger than 64 elements.
- All symbolic mathematics features.
- Printing.
- Importing graphics.
- Use of the Windows clipboard to transfer data between applications.

ABOUT HELP

This Help system is a smaller version of that which comes with the full version of Mathcad. However, it retains the context-sensitive feature of Mathcad 3.1's Help system. To access this feature:

- Press **Shift**+**F1**. The pointer changes into a question mark.
- Click on operators, menu choices,and error messages. Mathcad displays the relevant section of Help in the Help window.

READING AND WRITING DATA FROM DISK

Mathcad can read and write ASCII data to your disk. This data can be either structured or unstructured

Reading and writing files in other directories:

Describes how to tell Mathcad what file to read from and what file to write to. How to use the **Associate Filename** command to read and write files in other directories or drives.

Reading and writing structured data:

Describes how to use the READPRN(file) and WRITEPRN(file) functions to read and write numbers arranged in rows and columns.

Reading and writing unstructured data:

Describes how to use the READ(file) and WRITE(file) functions to read and write streams of numbers.

Appending existing data to a datafile:

Describes how to use the APPEND(file) and APPENDPRN(file) functions to add data to existing data.

Complex data:

Describes how to read and write data with real and imaginary parts.

Controlling how Mathcad writes data:

Describes how to control the precision with which Mathcad writes to disk, as well as the layout of the ASCII datafile Mathcad creates.

Structured data file

A file in which the data is arranged in rows and columns separated by spaces and tabs. For example:

```
2  2  3
2  8  6
3  6  9
```

A text header is allowed in the beginning of a file. However, once Mathcad encounters a number, it assumes the data has begun. Because of this, the text header should contain no numbers, i.e. dates etc.

Data files can contain numbers in `E` format. For example:

```
16E-2  -16.125E-15  200  50
```

are all valid in a Mathcad datafile.

Data may not be hex, octal or binary.

Unstructured data file

A file consisting of a stream of numbers separated by spaces and/or tabs.

Data files can contain numbers in E format. For example:

```
16E-2   -16.125E-15   200 50
```

are all valid in a Mathcad datafile.

Data may not be hex, octal or binary.

Identifying the file

By default, Mathcad performs all file I/O on files located in the default directory. The argument to a file I/O function is the filename minus the default extension.

You can manually override these defaults by doing the following:

- Click on the **filename** to select it.
- Choose **Associate Filename** from the **File** menu.
- Enter the full pathname to the file of interest.

File I/O Functions

These are

READPRN	READ
WRITEPRN	WRITE
APPENDPRN	APPEND

Note that the names are in UPPERCASE.

Default extensions

The default extension is 'PRN' for

READPRN
WRITEPRN
APPENDPRN

The default extension is 'DAT' for

READ
WRITE
APPEND

Appending existing data to a datafile

Structured data:

To append structured data to an existing structured data file:

- Make sure the number of columns of data to append matches the number of columns in the datafile.
- Use the APPENDPRN function as follows:

APPENDPRN(file) := **A**

The APPENDPRN function will append the contents of the array A to the end of the file identified by **file**.

Unstructured data:

To append a list of numbers to an existing list of numbers stored as an ASCII file, use the APPEND function with a range variable as follows:

i := 1 to N

APPEND(file) := v_i

Reading and writing structured data

Reading a file from disk

To read an ASCII file consisting of rows and columns, use the READPRN function as follows:

A := READPRN(file)

Writing a file to disk

To write an ASCII file consisting of rows and columns, use the WRITEPRN function as follows:

WRITEPRN(file) := A

Note that the names of the functions must be in UPPERCASE.

Reading and writing unstructured data

Reading a file from disk

To read an ASCII file consisting of a simple list of numbers, use the READ function with a range variable as follows:

i := 1 to N

v_i := READ(file)

Writing a file to disk

To write an ASCII file consisting of a simple list of numbers, use the WRITE function with a range variable as follows:

i := 1 to N

WRITE(file) := **v_i**

Reading and writing complex data

Mathcad does not keep track of complex numbers stored in ASCII files. It is good practice to write the real and imaginary parts to separate files.

For example, let **Z** be a complex matrix. Then

> **WRITEPRN(z_real) := Re(Z)**
>
> **WRITEPRN(z_imag) := Im(Z)**

To read the data stored this way, do the following:

> **Z := READPRN(z_real) + 1i * READPRN(z_imag)**

Controlling how Mathcad writes data

Significant digits:

By default, WRITEPRN writes 4 significant digits to disk. You can change this by changing the value of PRNPRECISION. To do so, choose **Built-In Variables** from the **Math** menu.

Column width:

By default, WRITEPRN writes data in columns that are 8 digits wide. You can change this by changing the value of PRNCOLWIDTH. To do so, choose **Built-In Variables** from the **Math** menu.

file

This is the name of an ASCII file from which Mathcad will read data. The default extension is 'PRN'.

Mathcad assumes that this file is in the default directory. If you want to read from a file located elsewhere, you must

- Click on the **filename** to select it.
- Choose **Associate Filename** from the **File** menu.
- Enter the full pathname to the file of interest.

file

This is the name of an ASCII file into which Mathcad will place the data. The default extension is 'PRN'. **If such a file already exists, Mathcad will overwrite it.**

Mathcad assumes that this file is in the default directory. If you want to write to a file located elsewhere, you must

- Click on the **filename** to select it.
- Choose **Associate Filename** from the **File** menu.
- Enter the full pathname to the file of interest.

file

This is the name of an ASCII file to which Mathcad will append the data. The default extension is 'PRN'. This file should have the same number of rows as the matrix you are appending to it.

Mathcad assumes that this file is in the default directory. If you want to read from a file located elsewhere, you must

- Click on the **filename** to select it.
- Choose **Associate Filename** from the **File** menu.
- Enter the full pathname to the file of interest.

Note that the file must already exist for APPENDPRN to work. You cannot use APPENDPRN to create a file. It can only append to an existing file.

file

This is the name of an ASCII file from which Mathcad will read data. The default extension is 'DAT'.

Mathcad assumes that this file is in the default directory. If you want to read from a file located elsewhere, you must

- Click on the **filename** to select it.
- Choose **Associate Filename** from the **File** menu.
- Enter the full pathname to the file of interest.

file

This is the name of an ASCII file into which Mathcad will place the data. The default extension is 'DAT'. **If such a file already exists, Mathcad will overwrite it.**

Mathcad assumes that this file is in the default directory. If you want to write to a file located elsewhere, you must

- Click on the **filename** to select it.
- Choose **Associate Filename** from the **File** menu.
- Enter the full pathname to the file of interest.

---

file

This is the name of an ASCII file to which Mathcad will append the data. The default extension is 'DAT'.

Mathcad assumes that this file is in the default directory. If you want to read from a file located elsewhere, you must

- Click on **filename** to select it.
- Choose **Associate Filename** from the **File** menu.
- Enter the full pathname to the file of interest.

READING DATA FROM DISK

READPRN(file) Returns an array containing whatever is in the file **file.prn** in the current directory.

Arguments:

- **file** is either the name of a file or a variable assigned to a file by choosing **Associate Filename** from the **File** menu. The file being read must be a structured ASCII file.

WRITEPRN**(file)** Writes an array to **file.prn** in the current directory.

Arguments:

- **file** is either the name of a file or a variable assigned to a file by choosing **Associate Filename** from the **File** menu.

WRITEPRN arranges the ASCII file in rows and columns like the original matrix. By default, WRITEPRN will write four significant digits in columns eight digits wide.To change this, choose **Built-In Variables** from the **Math** menu and change PRNPRECISION and PRNCOLWIDTH.

APPENDPRN(file) Appends an array to **file.prn** in the current directory. This file must already exist for APPENDPRN to work. Unlike WRITEPRN, APPENDPRN can not create a file, it can only append to an existing file.

Arguments:

- **file** is either the name of a file or a variable assigned to a file by choosing **Associate Filename** from the **File** menu. The file being appended to must be a structured ASCII file. The number of columns in the file must match the number of columns in the array.

APPENDPRN arranges the ASCII file in rows and columns like the original matrix. By default, APPENDPRN will append four significant digits in columns eight digits wide.To change this, choose **Built-In Variables from** the **Math** menu and change PRNPRECISION and PRNCOLWIDTH.

READING FROM A FILE

READ**(file)** Returns a single entry from the file **file.dat** in the current directory.

Arguments:

- **file** is either the name of a file or a variable assigned to a file by choosing **Associate Filename** from the **File** menu. The file being read must be an ASCII file containing numbers arranged in columns and separated by tabs or spaces.

READ is usually used in conjunction with a range variables as in:

i := 0 .. N-1

v_i **:=** READ**(file)**

This creates an array **v** containing whatever numbers are in **file.dat**.

The file being read must be an ASCII file containing a list of decimal numbers separated by tabs or spaces. Exponential notation such as

```
16E-2-16.125E-15
```

is permitted.

WRITING FILES TO DISK

WRITE(file) Writes to a number to the file **file.dat** in the current directory.

Arguments:

- **file** is either the name of a file or a variable assigned to a file by choosing **Associate Filename** from the **File** menu.

WRITE is usually used in conjunction with a range variables as in:

i := 0..N-1

WRITE(file) := v_i

The above example writes the list of numbers in the vector **v** into **file.dat**.

APPENDING ONE FILE TO ANOTHER

APPEND**(file)** Appends a number to the file **file.dat** in the current directory.

Arguments:

- **file** is either the name of a file or a variable assigned to a file by choosing **Associate Filename** from the **File** menu.

APPEND is usually used in conjunction with a range variable as in:

i := 0 to N-1

APPEND**(file) :=** v_i

The above example appends the list of numbers in the vector **v** to those already in **file.dat**.

Note that the file must already exist for APPEND to work. You cannot use APPEND to create a file. It can only append to an existing file.

Array to be appended to **file.prn**.

This array should have as many columns as there are in the file **file.prn**.

The number of elements in the vector **v**.

Array into which contents of **file** are placed.

Array whose contents are to be placed in **file**.

LOGICAL OPERATORS

Greater than	**>**
Less than	**<**
Greater than or equal	**Ctrl**+0
Less than or equal	**Ctrl**+9
Not equal	**Ctrl**+3
Equal	**Ctrl**+=

Related Topics:

Scalar operators

Array operators

Greater than Press **>**

x **>** y Returns 1 if x > y, 0 otherwise.

Operands:

- x and y must be real scalars.

Less than Press **<**

$x < y$ Returns 1 if $x < y$, 0 otherwise.

Operands:

- x and y must be real scalars.

Greater than or = Press **Ctrl**+0

x ≥ y Returns 1 if x y, 0 otherwise.

Operands:

- x and y must be real scalars.

Less than or = Press **Ctrl+9**

$x \le y$ Returns 1 if x y, 0 otherwise.

Operands:

- x and y must be real scalars.

Equal to Press **Ctrl+=**

$x = y$ Returns 1 if $x = y$, 0 otherwise.

Operands:

- x and y can be real or complex scalars or arrays.

Mathcad compares both real and imaginary parts of x and y.

Not equal to Press **Ctrl+3**

$x \neq y$ Returns 1 if x y, 0 otherwise.

Operands:

- x and y can be real or complex scalars or arrays.

Mathcad compares both real and imaginary parts of x and y.

File -> New Document

This opens a new, untitled document.

Keyboard shortcut	**F7**

File -> Open Document...

This command displays a scrolling list of MCD files in the current directory. Highlight a name and press **Enter** to open that document.

Keyboard shortcut **F5**

File -> Save Document...

Disabled in demo version

File -> Save Document As...

Disabled in demo version

File -> Insert Document...

This command displays a scrolling list of MCD files in the current directory. Highlight a name and press **Enter** to insert that document into the current document.

Mathcad inserts the document wherever the cursor is located.

File -> Close Document

This command clears the screen. If any changes have been made since the document was last saved, Mathcad asks if you want to save the current document.

Keyboard shortcut **Ctrl+F4**

File -> Save Configuration...

This command saves all your current formatting information, built-in constants, in a file called `mcad.mcc` in the current directory. These will become the default formatting parameters the next time you start Mathcad from this directory.

Mathcad saves the following defaults:

- Numerical result format
- Graphics format
- Default text font
- All math font tags
- Values of ORIGIN, PRNCOLWIDTH, PRNPRECISION, and TOL.
- Filename associations

File -> Execute Configuration file...

This command displays a scrolling list of MCC files in the current directory. Highlight a name and press **Enter** to execute the commands found in that file.

File -> Associate Filename...

This menu command associates the name of a datafile on disk with the argument of one of the file I/O functions listed below:

READ	**READPRN**
WRITE	**WRITEPRN**
APPEND	**APPENDPRN**

To associate a file name with a variable:

- Click on the argument to one of the above functions
- Choose this from the menu. Mathcad displays a scrolling list of PRN and DAT Files.
- Select from this list or specify a name in the space provided in the dialog box.

Whenever you use this file variable in an I/O function, Mathcad will read from or write to the associated file.

File -> Print Document

Disabled in demo version

File -> Page SetUp...

Disabled in demo version

File -> Exit

This will close the Mathcad application window and return you to the Program Manager.

To close just the document you are working on without quitting Mathcad, choose **Close Document** from the **File** menu.

Keyboard shortcut **Alt+F4**

Edit -> Undo Last Edit

In equations, plots and sketches, this undoes your most recent edit command or keystroke.

In text, this command is unavailable.

Keyboard shortcut **Alt+BkSp**

Edit -> Cut

This deletes whatever has been selected and places it on the clipboard.

Note that only the most recently cut selection is on the clipboard.

Keyboard shortcuts	**Shift**+**Del** or **F3**

Edit -> Copy

This copies whatever has been selected to the clipboard. This command is usually followed by **Edit -> Paste**.

In the demo version, the clipboard cannot be used by other Windows applications.

Keyboard shortcuts **Ctrl**+**Ins** or **F2**

Edit -> Paste

This places whatever is on the clipboard into the document.

This operation does not change the contents of the clipboard.

In the demo version, the clipboard cannot be used by other Windows applications.

Keyboard shortcuts **Shift**+**Ins** or **F4**

Edit -> Paste Special

Use for OLE objects. Disabled in this demo version.

Edit -> Cut All Regions

This command will remove all regions in your document and place them on the clipboard. You can then paste these regions elsewhere.

In the demo version, the clipboard cannot be used by other Windows applications.

Edit -> Separate Regions

If you find that regions in your document are overlapping, this command separates them. Note that this command cannot be undone.

Be careful using this when you have a lot of error messages. Once the error messages disappear, you may find the regions too far apart.

Keyboard shortcut **Ctrl+S**

Edit -> Insert Blank Line

This command inserts a blank line below the current location of the cursor.

Keyboard shortcut	**Ctrl+F9**

Edit -> Delete Blank Line

This command deletes a blank line below the current location of the cursor.

Keyboard shortcut	**Ctrl+F10**

Edit -> Insert Pagebreak

Disabled in demo version.

Choose this to insert a pagebreak wherever the cursor is located.

To delete a page break, click where the line indicating the pagebreak touches the left edge of your document. Then choose **Cut** from the **Edit** menu.

Edit -> Set Right Margin

Disabled in demo version.

To set the right margin, click where you want the right margin to go and choose this command.

Edit -> Clear Right Margin

Disabled in demo version.

Choose this command to remove the right margin.

Edit -> Find...

This prompts you for a text string. Mathcad then moves the cursor to the next (or previous) occurrence of this string.

Mathcad will search for strings in both text and variable and function names.

Keyboard shortcut **Ctrl+F5**

Edit -> Replace...

This prompts you for two text strings. You can choose from four Find and Change options, each of which will work in both math and text regions.

Keyboard shortcut **Shift+F5**

Text -> Create Text Region

Choose this to create a region of text wherever the cursor is. The cross hair cursor will change into a thin vertical line.

You can change the width of a text region by selecting it and dragging on the right edge of the selection rectangle with the mouse.

Keyboard shortcut **"**

Text -> Create Text Band

Choose this to create a band of text on th line occupied by the cursor. The cross hair cursor will change into a thin vertical line.

A text band always stretches across the page. If a text band grows big enough to overlap regions below it, Mathcad will push the regions down when you click outside the text band.

Keyboard shortcut **Ctrl+T**

Text -> Change Font

Before choosing this, you should have selected some text.This menu command displays a dialog box for changing the font, size and style for the selected text.The text must be in a text band or region. To change fonts in equations, choose **Apply Font Tag** or **Modify Font Tag** from the **Math** menu.

Text -> Change Default Font

Choose this to change the default font of all text regions and text bands.

Text in fonts other than the default font is unaffected.

This command affects only text regions and text bands. To change math fonts, choose either **Apply Font Tag** or **Modify Font Tag** from the **Math** menu.

Math -> Matrices...

Choose this to create a vector or matrix.

To change the size of an existing vector or matrix, select it with the cursor and choose this from the menu.

Keyboard shortcut **Ctrl+V**

Math -> Built-In Variables...

Choose this to change the values of the built-in variables:

- **TOL**
- **PRNPRECISION**
- **ORIGIN**
- **PRNCOLWIDTH**

Math -> Units...

Choose this to change the names of the built-in dimensions: mass, length, time and charge.

You can also select a default system of units for your document.

Math -> Randomize...

Mathcad's **rnd(x)** function uses a seed to generate uniformly distributed random numbers between 0 and x.

Choose this to change the value of this seed.

If you choose this without changing the seed, Mathcad restarts the random number sequence using this seed value.

Math -> Calculate

Choose this if you are in manual mode and wish to update all plots and numerical results on the screen. Mathcad place the words "calc F9" in the message line to let you know that the screen needs to be updated.

If you want to update all plots and numerical results throughout the document, choose **Calculate Document** from the **Math** menu instead.

Keyboard shortcut **F9**

Math -> Calculate Document

Choose this if you are in manual mode and wish to update all plots and numerical results throughout the document.

If you want to update only the plots and numerical results currently displayed, choose **Calculate** from the **Math** menu instead.

Math -> Toggle Equation

Use this to force Mathcad to ignore a selected equation. Disabled equations are marked with a small box.

If a selected equation is disabled, this command re-enables it.

This menu command allows you to use Mathcad's equation typesetting ability independently of its calculation features.

Math -> Automatic Mode

Choose this to force Mathcad to update the screen whenever you make a change in the document.

When editing a document that involves lengthy calculations, you may find it more efficient to work in manual mode.

Math -> Manual Mode

Choose this to prevent Mathcad from updating the screen until you press **F9** to force recalculation.

When editing a document that involves lengthy calculations, you may find it more efficient to work in manual mode.

Math -> Numerical Format...

To change the way a particular numerical result is displayed, click on the result, choose this, and click on Local in the dialog box.

To change the way all numbers are displayed, click in an empty region, choose this, and click on Global in the dialog box.

You can control

- The number of decimal places displayed.
- How big or small a result must be before it's displayed in exponential form.
- How small a result must be before it's displayed as zero. (Global only)
- How small the real or imaginary part must be before it's suppressed.
- Whether Mathcad uses i or j for imaginary numbers. (Global only)
- Whether the number is hex, octal or decimal.

Mouse shortcut **Double-click on result**

Math -> Apply Font Tag...

Click on a variable name and choose this from the menu to change its font tag.

Mathcad will bring up a list of available font tags. To tag the selected variable name, click on the button beside the desired math font.

To see what font, size and style go with a given tag, choose **Modify Font Tag** from the **Math** menu.

Math -> Modify Font Tags...

This brings up a list of currently defined font tags. You can add additional user font tags or change existing ones.

Note that if you change the font associated with an existing font tag, you change the font of every variable sharing that tag.

Math -> Change to Greek Variable

To type a variable name in Greek letters,

- Type the appropriate roman letters from the table below.
- Leave the insertion point in the variable name.
- Choose this menu command.

Note that the entire variable name will be in this font. It is not possible to mix fonts in a variable name.

If you click outside the name, Mathcad reverts to the default math font.

Keyboard shortcut **Ctrl+G**

Graphics -> Create Graph

Choose this to create graphs with either linear, log, or semilog axes. The plot will contain placeholders for x and y values as well as endpoints of the axes.

To change the axes, legends, plot style and other plot characteristics, choose **Graph Format** from this same menu.

Keyboard shortcut **@**

Graphics -> Create Surface Plot

Choose this to create a surface plot with a placeholder for an array to be plotted. The rows and columns will correspond to the x and y axes. The value in the array will represent the height above the xy plane.

To change the size and other plot characteristics, choose **Surface Plot Format** from this same menu.

Keyboard shortcut **Ctrl+2**

Graphics -> Create Picture

Since access to the Windows clipboard is disabled in this demo version, you won't be able to import a picture via the clipboard.

You can still import a Windows BMP file by choosing this from the menu to create a picture region and typing the name of the file in the placeholder.

With the full version of Mathcad, you could paste a bitmap on the clipboard into the region as follows:

- Click in it to select it
- Choose **Paste** from the **Edit** menu.

Keyboard shortcut **Ctrl+5**

Graphics-> Graph Format...

To change the characteristics of a particular graph, click on the graph and choose this.

To change the characteristics of all graphs in the document, either

- Place the cursor in an empty region and choose this, or
- Click in the Use for defaults check box.

Mouse shortcut **Double-click on graph**

Graphics-> Surface Plot Format...

To change the characteristics of a particular surface plot, click on the graph and choose this.

Mouse shortcut	**Double-click on surface plot**

Graphics-> Picture Format...

This command lets you

- Insert or remove a border around the selected picture
- Restore the original size of the picture.

Mouse shortcut	**Double-click on picture**

Window -> Cascade

Choose this to hide inactive document windows behind the active window without obscuring any title bars.

Window -> Tile

Use this to arrange all document windows so that they:

- all have the same area
- completely cover the Mathcad window.

Window -> Arrange Icons

When you close a document, an icon appears in the lower left corner of the application window. Choose this to arrange all these icons neatly.

Window -> Refresh

Choose this to make Mathcad redraw the screen.

Symbolic -> Evaluate Symbolically

This menu choice is disabled in this demo version of Mathcad.

This carries out symbolic evaluation of definite integrals, derivatives, summations, products, functions and other algebraic and matrix expressions.

Keyboard shortcut **Shift+F9**

Symbolic -> Simplify

This menu choice is disabled in this demo version of Mathcad.

This simplifies the selected expression, performing arithmetic, canceling common factors and using basic trigonometric and inverse function identities.

Symbolic -> Expand Expression

This menu choice is disabled in this demo version of Mathcad.

This expands all powers and products of sums in the selected expression.

Symbolic -> Factor Expression

This menu choice is disabled in this demo version of Mathcad.

This factors the selected expression into a product, if the entire expression can be written as a product. To factor a subexpression of a larger expression, select the subexpression.

Use this command to combine a sum of fractions into a single fraction.

Symbolic -> Collect on Subexpression

This menu choice is disabled in this demo version of Mathcad.

This collects terms containing like powers of the selected subexpression, which may be a single variable or a function together with its argument. The result is a polynomial in the selected expression.

Symbolic -> Differentiate on Variable

This menu choice is disabled in this demo version of Mathcad.

This differentiates the entire expression containing the selected variable with respect to that variable. Other variables are treated as constants.

Symbolic -> Integrate on Variable

This menu choice is disabled in this demo version of Mathcad.

This integrates the entire expression containing the selected variable with respect to that variable.

Symbolic -> Solve for Variable

This menu choice is disabled in this demo version of Mathcad.

This finds the value of the selected variable that makes the expression containing the variable equal to zero. If you select a variable in an equation or inequality, this command solves the equation or inequality for the selected variable.

Symbolic -> Substitute for Variable

This menu choice is disabled in this demo version of Mathcad.

This substitutes the contents of the clipboard for each occurrence of a selected variable in an expression. To use this menu command, first put the expression being substituted into the clipboard by selecting it and choosing **Copy** or **Cut**. Then select an occurrence of the variable you are substituting for and choose this menu command.

Symbolic -> Expand to Series...

This menu choice is disabled in this demo version of Mathcad.

This derives a power series for an expression with respect to the variable you have selected. A dialog box allows you to choose the order of the series.

Symbolic -> Convert to Partial Fraction

This menu choice is disabled in this demo version of Mathcad.

This generates a partial fraction expansion for an expression. If such an expansion is found, the denominators are linear or quadratic in the selected variable.

Symbolic -> Transpose Matrix

This menu choice is disabled in this demo version of Mathcad.

This gives the transpose of the selected matrix.

Symbolic -> Invert Matrix

This menu choice is disabled in this demo version of Mathcad.

This gives the symbolic inverse of the selected square matrix.

Symbolic -> Determinant of Matrix

This menu choice is disabled in this demo version of Mathcad.

This gives the symbolic determinant of the selected square matrix.

Symbolic -> Derivation Format...

This menu choice is disabled in this demo version of Mathcad.

This command allows you to choose the format for symbolic results. A dialog box presents the options, which include vertically stacked display of results and comments, horizontal display, and display without derivation comments.

Symbolic -> Derive in Place

This menu choice is disabled in this demo version of Mathcad.

This command causes symbolic results to be substituted directly for the original expression rather than displayed in derivation format.

Symbolic -> Load Symbolic Processor

This menu choice is disabled in this demo version of Mathcad.

This loads the symbolic processor into memory. You must choose this option before using any of the other Symbolic menu commands.

Help -> Index

This opens a window containing index for on-line help.

Keyboard shortcut	**F1**

Help -> Keyboard

This opens a window containing help on all keystrokes.

Help -> Using Help

This opens a window containing instructions for using the Mathcad Help system.

Help -> Open Handbook

This shows a list of installed Electronic Handbooks.

Help -> Search Handbook

Opens dialog box into which you type a Handbook string you want to search for. Use only when a Handbook is already open.

Help -> Show Controls

This enlarges the handbook control buttons and places them on a movable palette.

Help -> Getting Started

This opens a window that briefly demonstrates some of Mathcad's features.

Help -> About Mathcad

This shows the version number and memory information for this copy of Mathcad.

VARIABLE AND FUNCTION NAMES

Variable and function names can be any length. They can consist of:

- Upper and lowercase letters
- Numbers 0 through 9
- The infinity symbol
- The prime symbol `
- Greek letters
- Underscores (_)
- Percent symbols (%)
- Subscripts

Alphabetic characters

Names are case and font sensitive. Names are also size and style sensitive. For example, a bold **f** and italic *f* refer to different variables.

A name can be in any font; however, every character in the name must be in the same font.

Numbers in variable names

The numbers 0 through 9 can appear anywhere in a name except as the leading character. For example, the name 'x2' is legal, but not the name '2x'.

Infinity symbol

To insert the infinity symbol, type **Ctrl**+**z**.

Greek letters

To insert a Greek letter,

- Type the appropriate letter from the Windows Symbol font.

- Type **Ctrl**+**G**

Literal subscripts

To type a literal subscript like
v_{init}

- Type the first part of the name, the part that comes before 'sub' when you pronounce it out loud (**v**).
- Type a period (**v.**) The insertion point extends down half a line
- Type the second half of the name, the part you want in the subscript (**v.init**)

NUMBERS

Entering numbers:

Place an 'o', 'h', or 'i' as appropriate

Examples:

octal	36o
hexadecimal	0af1h
imaginary	1i

Displaying numbers:

To control the way Mathcad displays the result of a particular calculation, double-click on it. Mathcad displays a dialog box.

NUMERICAL FORMAT

Double-click on a result to change:

Radix

Whether a number is displayed as octal, hexadecimal or decimal.

Displayed Precision

How many digits to the right of the decimal point are displayed.

Exponential Threshold

How large or small does a number have to be before Mathcad displays it in scientific notation.

Complex Tolerance

How small the real or imaginary part of number must be before Mathcad displays only the larger part.

Zero Tolerance

How close to zero must a number be before Mathcad displays it as zero.

Changes you make here apply only to the selected result. To change the way results are displayed throughout the document, click Global Result Format in this dialog box.

To view a scalar result in full precision without changing its local format,

- Click in it.
- Press **Ctrl**+**F**.

Mathcad displays the value in the message line.

Octal numbers

These are numbers in base 8. For example

10o = 8
613o = 395

Only integers can be represented in octal form. Non-integers are truncated. For example:

8 = 10o
8.99 = 10o

Complex octal numbers can also be displayed. If either the real or imaginary part is non-integer, Mathcad truncates it before display.

Hexadecimal numbers

These are numbers in base 16. For example

10h = 16
0a3h = 163

Hexadecimal numbers beginning with the letters 'a' through 'f' must be preceded by a 0 as shown above.

Only integers can be represented in hexadecimal form. Non-integers are truncated. For example:

16 = 10h
16.99 = 10h

Complex hexadecimal numbers can also be displayed. If either the real or imaginary part is non-integer, Mathcad truncates it before display.

Decimal numbers

These are numbers in base 10. Mathcad displays results in base 10 by default.

Imaginary numbers

To make a number imaginary, type i or j right next to it. For example,

2i	-33.4j
1i	1j

Note that you must type 1i or 1j, not just i or j. The digit '1' disappears when you move the cursor away.

Radix

Mathcad displays results as either:

<u>Decimal</u>	base 10
<u>Octal</u>	base 8
<u>Hexadecimal</u>	base 16

Mathcad truncates hexadecimal and octal numbers before displaying them.

Displayed Precision

This setting places an upper limit on the number of displayed decimal digits. This affects only the display. Mathcad maintains full precision for all calculations.

For example, the number π is displayed as follows:

Displayed Precision	**Displayed result**
0	3
1	3.1
4	3.1416
9	3.141592654

The displayed precision must be between 0 and 15 inclusive. By default, it is set to 3.

Exponential Threshold

This setting controls how large a number's absolute value must be before Mathcad begins displaying it in exponential notation.

To see how Mathcad displays various numbers using different exponential thresholds, click on the number below:

Examples:

1123581
0.1235813

The exponential threshold must be between 0 and 15 inclusive.

By default the threshold is set to 3. This means that all numbers outside the intervals [.001, 1000] and [-1000,-.001] are shown in exponential notation.

1,123,581 displayed with different exponential threshold values.

0.1235813 displayed with different exponential threshold values.

Complex Tolerance

This number controls how much larger the real or imaginary part of a number must be before Mathcad stops displaying the smaller part.

To see how Mathcad displays various numbers using different complex tolerances, click on the number below:

Examples:

<u>1+0.002i</u>
<u>0.002+1i</u>

Complex tolerance must be an integer between 0 and 63. By default, it is set to 10. This means that

z is shown as pure real if

Im(z) ` 10

z is shown as pure imaginary if

Re(z) ` 10

1+0.002i displayed with different complex tolerance values.

Complex Tolerance	Displayed result
2	1
3	1+0.002i
4	1+0.002i

0.002+1i displayed with different complex tolerance values.

Complex Tolerance	Displayed result
2	1i
3	0.002+1i
4	0.002+1i

Zero Tolerance

This number controls how close a result must be to zero before Mathcad displays it as such.

To see how Mathcad displays various numbers using different zero tolerances, click on the number below:

Examples:

0.0001215
0.0121556

The zero tolerance must be between 0 and 307. By default it is set to 15. This means that numbers smaller than 10^{-15} will be displayed as zero.

You can set this only when the Global option button is clicked in the dialog box.

0.0001215 displayed with different zero tolerance values.

Zero Tolerance	Displayed result
2	0
3	0
4	.0001215
5	.0001215

0.0121556 displayed with different zero tolerance values.

Zero Tolerance	**Displayed result**
1	0
2	0.0121556
3	0.0121556

Imaginary unit

This controls whether Mathcad uses 'i' or 'j' to display an imaginary number.

Mathcad will continue to understand both 'i' and 'j' when you input an imaginary number. This setting controls only how Mathcad displays numbers.

You can set this only when the Global option button is clicked in the dialog box.

Global result format

When you click this button, the settings in the dialog box will be the default settings for your document.

This will affect everything to the right of an equals sign in your document, except for those results that have been locally formatted.

SCALAR OPERATORS

addition with linebreak	**Ctrl+Enter**
addition	**+**
complex conjugate	**"**
differentiation	**?**
division	**/**
exponentiation	**^**
factorial	**!**
integration	**&**
iterated product	**#**
magnitude	**|**
multiplication	*****
negation	**–**
square root	****
subtraction	**–**
summation	**$**

Related Topics:

Array operators
Logical operators

Scalar

A number as opposed to a vector or a matrix.

Addition Press **+**

x + y Returns the sum of the <u>scalars</u> x and y.

If you find an expression so long that it scrolls off the right of the screen, you may prefer to use **Ctrl+Enter** instead. This inserts

```
 …
+ y
```

Subtraction Press −

$x - y$ Subtracts y from x.

If you find an expression so long that it scrolls off the right of the screen, you may prefer to negate the y and use **Ctrl+Enter** instead. This inserts

$$\ldots + -y$$

Multiplication Press *

$x \cdot y$

This returns the product of y and x.

Division Press **/**

$\frac{y}{x}$

Returns y divided by x. If x=0, returns error message singularity

Operands:

- x and y are scalars.
- x cannot be 0.

Negation Press –

–x This returns the negative of its operand.

To insert a minus sign in front of an existing expression:

- Surround the expression with a selection box
- Press **Ins**
- Type the minus sign.

To delete a minus sign in front of an expression:

- Surround the expression and the minus sign with a selection box.
- Press **BkSp**

Exponentiation Press **^**

y^x

This returns y raised to the x power. Use this for cube roots, fourth roots and so on.

When the result is multivalued, Mathcad returns

exp(x ln(y))

To calculate the real cube root of a real number y, use

$$\mathrm{cubert}(y) := \mathrm{if}\left[y < 0, -\left|y^3\right|, y^3 \right]$$

Factorial Press **!**

n! Returns n(n-1)(n-2)...1

Operands:

- n must be a nonnegative integer.

Factorials for n>170 will overflow when evaluated numerically. Large factorials can nevertheless be evaluated by the symbolic processor.

Complex conjugate Press "

$\overline{x}$

Returns the conjugate of x:

Re(x) - i Im(x)

Square root Press ****

$\sqrt{x}$

Returns the positive square root of the operand.

When the operand is negative or complex, returns the square root having the smallest argument.

Summation Press **$**

$\sum_i$

Returns the sum over i of the expression X.

Operands:

- i must be a range of integers.
- X is any expression.

Define the range variable i before using this operator. For example:

To make a sum with a variable upper limit of summation, use the following example:

If the summand has several terms, be sure to enclose them in parentheses to keep them inside the summation operator.

Iterated product Press #

$\prod_i$

Returns the product over i of the expression X.

Operands:

- i must be a range of integers.
- X is any expression.

If X consists of several terms, be sure to enclose them in parentheses to keep them inside the iterated product operator.

You must define the range variable i before using this operator. For example:

i

Integration Press &

$\int_a$

Returns the definite integral of f(x) from a to b.

Operands:

- f is any scalar valued function.
- x is the dummy variable of integration.
- a and b must be real, but f may be complex valued.

For example:

Although the example shows a function of one variable, f can be a function of any number of variables.

Differentiation Press **?**

$\frac{d}{dt} f(t)$

Returns the derivative of f(x) evaluated at the point x.

Operands:

- f(x) is a <u>scalar</u> valued function.
- x is the point at which the derivative is to be evaluated.

You must first define the point at which the derivative is to be evaluated. For example:

$\frac{d}{dx} x \cdot \ln(x) = 75.005$

Note in the above example that the value returned is the derivative of f with respect to x evaluated at whatever point(s) x is defined at.

Although the example shows a function of one variable, f can be a function of any number of variables.

If f has several terms, be sure to enclose them in parentheses to keep them inside the derivative operator.

Addition with linebreak Ctrl+Enter

$\ldots$
$+\ y$

Same as addition, except the two terms are on different lines.

Use this to avoid expressions that scroll off the edge of the screen.

Magnitude Press **|**

|x| Returns the magnitude of x.

MATHCAD APPLICATIONS PACKS

The Mathcad series of applications packs contains thoroughly documented and tested Mathcad implementations of common mathematical techniques.

For more information, contact MathSoft at 1-800-MATHCAD. In Massachusetts, or outside the USA, call 1-617-577-1017.

Applications Packs currently available are:

- Advanced Math
- Chemical Engineering
- Civil Engineering I:
  - Structures and Materials
- Civil Engineering II:
  - Soil Mechanics and Hydraulics
- Electrical Engineering
- Mechanical Engineering I
  - Structural Design
- Mechanical Engineering II
  - Machine Design
  - Fluid Flow
  - Heat Transfer
- Numerical Methods
- Statistics I:
  - Tests and Estimation
- Statistics II:
  - Modeling and Simulation

Advanced Math Applications Pack

Solution of a First Order Differential Equation
Solution of a Second Order Differential Equation
Solving a System of Differential Equations
Roots of Polynomials
Matrix Functions
Convolution and Correlation of Sequences
Convolution Using FFT's
Discrete Fourier Transform in Two Dimensions
Digital Filtering
Conformal Mapping
Diffusion
The Laplace Equation
Coordinate Transformations
Static Equilibrium: A Nonlinear System
Polynomial Least Squares Fit
Plotting a Surface

Statistics I: Tests and Estimation

Chi-square Test for Goodness of Fit
A Two-Way Contingency Table
One-Way Analysis of Variance
Choosing the Sample Size for a *t* Test on Means
A *t* Test on Means
The Wilcoxon Signed-Rank Test
The Rank Sum Test
Kendall's Rank Correlation τ
Spearman's Rank Correlation Coefficient
The Kolmogorov-Smirnov Test
Counting Runs
Estimating the Mean of a Normal Population
Jacknife Tests
Bootstrap Estimates for Standard Error
The Median, Interquartile Range, and Biweight
Probability Distributions

Statistics II: Modeling and Simulation

Frequency Distribution of Data
Cumulative Distribution of Data
Smoothing with Running Medians
Multiple Regression
Forecasting be Exponential Smoothing
Simulating a Multinomial Experiment
Operating Characteristic Curves
Simulating a Single Server Queue
Shuffling Elements of an Array
Generating Random Deviates
Random Deviates with Normal Distribution
Random Deviates with Weibull Distribution
Random Deviates with Exponential Distribution
Random Deviates with Poisson Distribution
Uniformly Distributed Random Deviates
Combinatorial Formulas
Probability Distributions

Electrical Engineering

Field Patterns for a Uniform Linear Array
Waveguides, Striplines, Coaxial Lines
Two-Port Parameter Conversions
Network Analysis Using an Admittance Matrix
American Wire Gauge Table
Transmission Line Impedance
The Smith Chart
Transmission Line Calculations
FIR Filter Design by Windowing
Design of an IIR Filter
Elliptic IIR Filter Design
Chebyshev Polynomials
Transfer Function Calculations
Polar Plots and Nyquist Plots
Convolution and Deconvolution
Algebraic Codes
Quantizing a Signal
Delta Modulation
Z-Transform Applications
Unit Definitions

Mechanical Engineering I: Structural Design

Design of Pressure Vessels
Local Vessel Stresses
Vertical Vessel Support
Sizing Nozzle Reinforcement in Pressure Vessels
Integral Flange Stresses in Pressure Vessels
Loads and Welds in Beams and Bars
Composite Walls
Maximum Bending Moment in Beams
Formulas for Combined Stress
Stresses in Oblique Planes
Sizing of Vertical Cylindrical Tanks
Stresses in Cylinders
Euler-Johnson Column Buckling Analysis
Sizing of Rectangular Tanks under Hydrostatic Pressure
Arbitrary Shape Disc Weight
Wind Load in Vertical Equipment

Mechanical Engineering II: Machine Design, Fluid Flow and Heat Transfer

Chemical Engineering

- **Thermodynamics**
  - Determination of Equilibrium Composition
  - Heat of Reaction
  - Heat of Formation
- **Steam and Furnaces**
  - Steam Properties
  - Furnace Performance
- **Phase Equilibrium and Distillation**
  - Binary Phase Diagrams for VLE
  - Isothermal Flashing
  - Sizing of Distillation Column Internals
- **Transient Heat Transfer**
  - Transient Heat Conduction
- **Heat Exchangers**
  - Building Larger Applications
  - Outside Film Heat-Transfer Coefficient
  - Inside Film Heat-Transfer Coefficient
  - Agitated Vessel Film Heat-Transfer Coefficient
  - Clean Overall Heat-Transfer Coefficient
  - Design Overall Heat-Transfer Coefficient
  - Fouling Factor
- **Reactor Design and Kinetics**
  - Chemical Reactor Design
  - Rate Expression from Differential Method
  - Reaction Rate Equation: Integral Method
  - Isothermal Batch Reactors
  - Sizing of CSTR
  - Non-Adiabatic Plug Flow Reactor
  - Effectiveness Factors
- **Fluid Flow**
  - Flow Pattern Identification
  - Two-Phase Flow Systems

Civil Engineering I: Structures and Materials

- Backfill Load in Concrete Pipes
- Design of a Continuous Beam
- Moment and Shear in Beams
- Statically Determinate Bents
- Cement Storage Silos
- Brittle and Ductile Failures in Concrete
- Stresses and Capacity of Concrete Beams
- Counterfort Wall - Approximate Design
- Design of Concrete Collars
- Buried Structures: Design of Flexible Conduits
- Structural Design of Footings
- Capacity of Foundations
- Stresses Due to Prestressing in Concrete Girders
- Moments of Inertia of Bodies
- Loads for Steel Columns
- Reinforced Masonry Columns
- Design of Retaining Walls
- Determination of Settlement of Foundations
- Loads on a Fixed-End Steel Beam
- Design of Steel Rigid Frames
- Welded Joint Design
- Design of Circular Concrete Tanks
- Capacity of T-Beams
- Deflection of Trusses
- Capacity and Design of Wooden Columns

Civil Engineering II: Soil Mechanics and Hydraulics

Bearing Capacity of Deep Foundations
Boussinesq Method for Evaluating Soil Pressure
Flow Classification and Critical Depth
Pipes in Series: Equivalent Length and Total Discharge
Design of Flocculators
Prediction of Friction Loss in Pipes
Discharge Rate and Total Depth
Groundwater Investigation
Pipes in Parallel: Equivalent Length and Friction Loss
Quicksand - Head and Critical Hydraulic Gradient
Rheological Behavior of Soils
Seepage Through Non-Uniform Soil
Slope Analysis
Soil Samples
Allowable Soil Pressure
Consolidation Ratio by the Terzaghi Theory
Contracted and Suppressed Weirs
Well Yield

Numerical Methods

Ordinary Differential Equations
- The Runge-Kutta Method
- The Runge-Kutta Method - Fast Version
- Higher Order Equations
- Approximating Eigenvalues
- The Predictor-Corrector Method
- An Implicit Method for Stiff Systems
- Boundary-Value Problems I: Linear Shooting
- Boundary-Value Problems II: The Rayleigh-Ritz Method

Partial Differential Equations
- The Wave Equation
- Normal Modes of a Circular Membrane
- The Poisson Equation
- The Heat Equation
- Sine and Cosine Transforms

Integral Equations
- The Fredholm Equation
- The Volterra Equation

Special Functions
- Spherical Harmonics
- Elliptic Integrals of the First and Second Kinds
- Jacobian Elliptic Functions
- Roots of Bessel Functions

Polynomials
- Orthogonal Polynomials
- Finding the Roots of a Polynomial

Applications of Contour Integration
- Higher Order Derivatives
- Counting Zeros
- Iterated Mappings
- Space Curves
- Pade Approximation

VECTORS AND MATRICES

Creating vectors and matrices:
Shows how to define a new vector or matrix and how to change the elements in an existing vector or matrix.

Data file I/O:
Shows how to read an external ASCII file into a matrix or vector, and how to write a matrix or vector out to an external ASCII file.

Changing the size of an array:
Shows how to add and delete rows and columns and how to combine two arrays.

Extracting elements, rows and columns:
Shows how to refer to individual elements,rows and columns of an array.

Vector and matrix functions:
Functions designed to return information about an array.

Vector and matrix operators:
Operations on vectors and matrices such as inner product, transpose, cross product etc...

Creating vectors and matrices:

The easiest way to create an array is to create an array of empty placeholders and just fill them in.

To do so:

- Press **Ctrl+V** to bring up a dialog box.
- Specify number of rows and column in the dialog box.
- Click "OK" to see an array of empty placeholders.
- Click on each placeholder to select it, then enter a number or an expression, or use **Tab** to move through the placeholders.

You can change the elements of a matrix or vector created this way by repeating step 4 above.

For larger arrays in which a formula relating the value of an array element to its indices exists, it is easier to use range variables. The example below illustrates the technique:

i := 0 ..9 j := 0 ..9

X[i,j := i^2+j/2

You can also create an array by reading directly from a datafile.

Data file I/O

Reading a file from disk

To read an ASCII file consisting of rows and columns, use the READPRN function as follows:

A := READPRN(filename)

To read an ASCII file consisting of a simple list of numbers, use the READ function with a range variable as follows:

i := 0 .. N-1
A[i := READ(filename)

Writing a file to disk

To write an ASCII file consisting of rows and columns, use the WRITEPRN function as follows:

WRITEPRN(filename) := A

To write an ASCII file consisting of a simple list of numbers, use the WRITE function with a range variable as follows:

i := 0 .. N-1
WRITE(filename) **:=** A[i

N
Number of elements in the data vector.

filename
This is the name of an ASCII file in the current default directory. The default extension is "PRN".

If you want to perform file I/O operations on a file not in the current default directory, you must

- Place any legal Mathcad variable name as argument to the file I/O function.
- Choose **Associate Filename** from the **File** menu.
- Enter the variable name and the pathname to the file.

Changing the size of an array:

To join two existing arrays together:

Use the augment function to combine an m x n array with an m x p array to form an m x (n+p) array.
For example:

To insert or delete rows and columns:

Matrices from Math menu.

Additional details on:

Inserting rows
Deleting rows
Inserting columns
Deleting columns

To insert one or more rows:

Move the cursor to where the array was defined and click on an element in the row below which you want to add additional rows.

- Press **Ctrl+V** to bring up a dialog box.
- Click the Insert button.
- Specify the number of rows to insert.

Mathcad inserts rows below the row containing the selected element.

To delete one or more rows:

Move the cursor to where the array was defined and click on an element in the first row you want to delete.

- Press **Ctrl+V** to bring up a dialog box.
- Click the delete button.
- Specify the number of rows to delete.

Mathcad deletes the row containing the selected element and the appropriate number of rows below it.

To insert one or more columns:

Move the cursor to where the array was defined and click on an element in the first column you want to delete.

- Press **Ctrl+V** to bring up a dialog box.
- Click the insert button.
- Specify the number of columns to insert.

Mathcad inserts columns to the right of the column containing the selected element.

To delete one or more columns:

Move the cursor to where the array was defined and click on an element on the right hand side.

- Press **Ctrl+V** to bring up a dialog box.
- Click the delete button.
- Specify the number of columns to delete.

Mathcad deletes the column containing the selected element and the appropriate number of columns to the right.

Extracting elements, rows and columns:

To access a particular array element, use the subscript operator and type an equals sign. For example:

Press A [(1,1)

To access an entire column at once, use the superscript operator as shown below:

Then extract with Ctrl+6

To access an entire row, use both the transpose and the superscript operator as shown below:

Transpose with Ctrl+1
Then extract with Ctrl+6

ITERATION AND LOOPING

Range variables

t := <u>firstval</u>, <u>nextval</u> <u>.. lastval</u>

Examples:

Stepsize = 1
t := 1 ..4 t =<u>?</u>

t := 4 ..-1 t =<u>?</u>

Stepsize ≠ 1
t := 1, 1.2 ..2 t =<u>?</u>

t := 2, 0 ..-6 t =<u>?</u>

firstval

This is the first value taken by the range variable.

This number cannot be complex.

nextval [optional]

This is the second value taken by the range variable. Note that this value is *not* the stepsize. The stepsize is given by

nextval - **firstval**

If you omit this value, the stepsize is

1 if **lastval** > **firstval**
-1 if **lastval** < **firstval**

This number cannot be complex.

lastval

If **lastval** - **firstval** is an integer multiple of the stepsize, lastval is the last value taken by the range variable.

If not, iteration stops just short of **lastval**.

For example:

t := 1, 1.1 ..π

ends with 3.1

The Iteration operator
This operator must always appear in a range variable definition.

Type the semicolon key

`;`

to insert it.

1<--- firstval
2<--- nextval
3
4<--- lastval

1<--- firstval
1.2<--- nextval
1.4
1.6
1.8
2<--- lastval

4 <--- firstval
3 <--- nextval
2
1
0
-1 <--- lastval

2 <--- firstval
0 <--- nextval
-2
-4
-6 <--- lastval

Repeatedly evaluating an expression

Use range variables to repeatedly evaluate an expression over a range of values. For example, to see the first twenty odd numbers, create the following:

i := 0..19

2*i + 1 =

Entering a table of values

To place the numbers 0, 3 and 1 into an array, type:

i:0;2 <-- Define range variable

x[i:0,3,1 <-- Fill array

Graphing a function

To graph a function:

- Define a range variable for the range of values over which to plot the function.
- Press @ to create a plot.
- Place the range variable on one axis and the function on the other.

Graphing a vector

To graph the elements of a N element vector:

- Define a range variable from 0 to N-1.
- Press **@** to create a plot.
- Place the subscripted vector on one axis and the range variable on the other.

ARRAY OPERATORS

addition	+
cross product	**Ctrl**+8
determinant	|
inverse	^-1
negation	-
powers of matrix	^
magnitude	|
multiplication	*
subscript (vector)	[
subscript (matrix)	[
subtraction	-
sum elements	**Ctrl**+4
superscript	**Ctrl**+6
transpose	**Ctrl**+1
vectorize	**Ctrl**+-

Related Topics:

Scalar operators
Logical operators

Matrix inverse Press `^-1`

$\mathbf{M}^{-1}$

This returns the inverse of a matrix. If **M** does not have an inverse, Mathcad displays the message *singularity*.

Operands:

- **M** is a square matrix with an inverse.

Vector subscript Press]

V_n

This returns the *n*th element of **V**. If **V** has no *n*th element, Mathcad displays the message *index out of bounds*.

Operands:

- **V** is a vector (an array with one column).
- **n** is an integer.

Keep in mind that all arrays in Mathcad start at 0 by default, not 1. For a vector with 10 elements, **n** must run between 0 and 9. To change this, choose **Built-In Variables** from the **Math** menu and change **ORIGIN**.

Matrix subscript Press]

$\mathbf{M}_{\mathbf{m},\mathbf{n}}$

This returns the *mn*th element of **M**. If **M** has no *mn*th element, Mathcad displays the message *index out of bounds*.

Operands:

- **M** is an array.
- **m** and **n** must be integers.

Keep in mind that all arrays in Mathcad start at 0 by default, not 1. For a matrix with 10 rows and 5 columns,

- m must run between 0 and 9
- n must run between 0 and 4

To change this, choose **Built-in Variables** from the **Math** menu and change **ORIGIN**.

Addition Press **+**

A + B Adds the elements of **A** to the corresponding elements of **B**.

A + x Adds **x** to every element in **A**.

Operands:

- **A** and **B** are arrays with the same number of rows and columns.
- **x** is any scalar, real or complex.

Vectorize operator Press **Ctrl**+ –

$\vec{\mathbf{v}}$

Applies a function or operator to every element of a matrix.

Operands:

- **A** and **B** are arrays with the same number of rows and columns.

EXAMPLES

To perform matrix multiplication of A and B, you would see

$\mathbf{A} \cdot \mathbf{B}$

To multiply each element of **A** by the corresponding element in **B**.

$\overrightarrow{(\mathbf{A} \cdot \mathbf{B})}$

To vectorize an expression,

- Enclose the expression in a selection box.
- Press **Ctrl**+–.

Vector cross product Press **Ctrl**+8

u × **v**

Returns vector cross product of **u** and **v**.

Operands:

- Both **u** and **v** are three element column vectors.

u × **v** is a vector with the following properties:

- Its direction is orthogonal to **u** and **v** in the direction determined by the right hand rule
- Its magnitude is **| u || v |**sin() where is the angle between **u** and **v**.

Negate an array Press −

−A This returns an array the same size as **A** containing the negative of each element of **A**. It is equivalent to multiplying **A** by **-1**.

Operands:

- **A** is an array

Matrix powers Press **^**

M^n

Depending on **n**, this returns one of the following:

- If **n = 0** this returns the identity matrix whose dimensions match those of **M**.
- If **n = 1** this just returns **M**.
- If **n = −1** this returns inverse of **M** if it exists.
- If **n** is a positive integer this multiplies **M** by itself **n** times.
- If **n** is a negative integer this multiplies the inverse of **M** by itself **n** times.

Operands:

- **M** is square.
- **n** is an integer.

Matrix powers can be evaluated symbolically. To do so:

- Select the expression to evaluate.
- Choose **Evaluate Symbolically** from the **Symbolic** menu.

Matrix multiplication Press *

A·B

Returns the matrix product of **A** and **B**.

Operands:

- If **A** is an *m* x *n* matrix
- **B** is an *n* x *p* matrix

SPECIAL CASES

If **A** is a matrix and **x** is scalar, then the product of **A** and **x** is a matrix in which each element of **A** has been multiplied by **x**.

To get the term by term product of two equal size arrays **A** and **B**, select the entire expression and press **Ctrl+−** to insert the vectorize operator.

Subtraction Press –

A – B Subtracts the elements of **B** from the corresponding elements of **A**. **A** and **B** must have the same number of rows and columns.

A – x If **A** is an array and **x** is a scalar then **A – x** is an array formed by subtracting **x** from every element in **A**.

Operands:

- **A** and **B** are matrices having the same number of rows and columns.
- **x** is a scalar.

Sum elements of a vector   Press **Ctrl**+**4**

Σ**v**   Returns the sum of all the elements in the vector **v**.

Operands:

- **v** is a vector.

Superscript operator Press **Ctrl+6**

$\mathbf{M}^{\langle \mathbf{n} \rangle}$

This returns the **n**th column of the array **M**. If **M** has no **n**th column, Mathcad displays the message *index out of bounds*.

Operands:

- **M** must be an array
- **n** is a scalar. It may not be complex

Keep in mind that all arrays in Mathcad start at 0 by default, not 1. For a matrix with 10 columns, **n** must run between 0 and 9. To change this, choose **Built-in Variables** from the **Math** menu and change **ORIGIN**.

EXAMPLE

To get a row from a matrix, use this in conjunction with the transpose operator. For example, to get the zeroth row from a matrix **M**, type **M Ctrl+1 Ctrl+6 0**

Transpose Press **Ctrl**+1

$\mathbf{M}^{\mathbf{T}}$

This returns the **n x m** array formed by interchanging rows and columns of an **m x n** array.

Operands:

- **M** is an **m x n** array.

Determinant Press **|**

| M | Returns the determinant of a square matrix.

Operands:

- M is a non-singular square matrix.

To take the magnitude of each element in a matrix,

- Place the selection box around the **| M |**
- Apply the vectorize operator, **Ctrl+−**.

Magnitude Press **|**

| v | Returns the magnitude of the vector **v**. This is the square root of the sum of the squared magnitude of the elements in **v**.

Operands:

- **v** is a vector

To take the magnitude of each element in a vector,

- Place the selection box around the **| v |**
- Apply the vectorize operator, **Ctrl+–**.

SOLVING EQUATIONS

One equation in one unknown:

Use the root function as in root(f(x),x)=

N linear equations in N unknowns:

Place the coefficients in a matrix. Then invert it.

N nonlinear equations in N unknowns:

Use a solve block and the Find(x, y, ..z) function.

Solving one equation in one unknown

Any such equation can be written as

    f(x) = g(x)

which can be simplified to

    f(x) - g(x) = 0

The solution is given by Mathcad's root function:

    root(f(x),x)

For example, consider the equation:

    cos(x) = x + .2

To solve this in Mathcad you would write:

    x := 1<--- Guess value

    root(cos(x)-x-.2,x) =

Solving a system of linear equations

The example below illustrates how to solve three linear equations in three unknowns.

$$2 \cdot x + 7 \cdot y + 9 \cdot z = 9$$

Define the matrix **A** and the vector **b** as

$$\lfloor 2 \ 7 \ 9 \rfloor \qquad \lfloor 9 \rfloor$$

The original system can be written as the matrix equation:

$$\mathbf{b} := \mathbf{A} \cdot \mathbf{x}$$

The solution vector **x** is therefore

$$\mathbf{x} := \mathbf{A}^{-1} \cdot \mathbf{b}$$

Solving a system of nonlinear equations

The example below illustrates how to solve two nonlinear equations in two unknowns.

Roots of an expression

root(f(x),x) Returns the value of x that makes the function f equal to zero.

Arguments:

- **f** is a scalar valued function of any number of variables.
- **x** is a scalar variable found in **f**.

This function requires a guess value to start the root finding process. For functions with many roots, the root returned depends on the guess value.

If the guess value is very close to a minimum or maximum of **f**, the **root** function may fail to converge, or converge to a root far away from the guess value.

Solving systems of equations

Find(x,y,...) This returns the values of x, y... that satisfy the equations and inequalities in a solve block. If you are solving for n variables, the solve block must have n equations.

Arguments:

- x, y,... are scalar variables found in the system of equations.

When the solve block involves only one unknown, the Find function returns a scalar. Otherwise it returns a vector whose first element is x, second element is y and so on. If you are solving for n variables, the solve block must have n equations.

This function requires a guess value for each unknown to begin the search for solutions. For systems having more than one solution, the solution returned depends on the guess values.

Approximate solutions

Minerr(x,y,..) This returns the values of x, y... that come closest to satisfying the equations and inequalities in a solve block.

Arguments:

- x, y,... are scalar variables found in the system of equations.

When the solve block involves only one unknown, the Find function returns a scalar. Otherwise it returns a vector whose first element is x, second element is y and so on.

This function requires a guess value for each unknown to begin its search. In general, the result returned depends on the guess values.

f

This can be any scalar function or expression involving any number of variables.

It should contain the variable used as the second argument to the root function.

x

This is the variable to be solved for.

It should match one of the variables in the expression used as the first argument to the root function.

Only a variable name can appear here. Expressions and functions cannot be used.

x, y,...

A list of variables to be solved for.

The **Find** and **Minerr** functions return a vector with as many elements as there are entries in this list. The first element is the solution for the first element of the list, and so on.

Guess value

The root function requires a value from which to start its search for a solution. The variable name you use here should match the second argument of the root function.

For expressions with multiple roots, the root Mathcad converges to will depend on this guess value.

If you expect a complex root, you should use a complex guess value.

SORT FUNCTIONS

Click on the function names for an example.

sort(v)
Returns a vector with the values from **v** sorted in ascending order.

csort(A,n)
Returns an array formed by rearranging rows of **A** until column n is in ascending order.

rsort(A,n)
Returns an array formed by rearranging columns of **A** until row n is in ascending order.

reverse(v)
Returns a vector in which the elements of **v** are in reverse order.

SYMBOLIC CALCULATION

The symbolic calculation feature is disabled in this demo version of Mathcad.

The full version of Mathcad 3.1 can perform a variety of symbolic manipulations and derivations, from solving equations to inverting matrices. **Pull down the Symbolic** menu to see what it can do.

STATISTICS AND DATA ANALYSIS

Data analysis functions

Describes elementary data analysis functions: mean, variance, standard deviation and correlation.

Distribution functions

Describes the built-in gamma function, the error function and the cumulative normal distribution.

Random numbers

Describes how random numbers are generated and how to use them to generate normal or exponential distributions.

Histograms

Describes how to get the frequency distribution of a data sample.

Interpolation

Described linear and cubic spline interpolation.

Curve fitting

Describes linear and multilinear regression.

DATA ANALYSIS FUNCTIONS

mean(v) Arithmetic mean of the elements in vector **v**.

stdev(v) Standard deviation of the elements in vector **v**.

var(v) Variance of the elements in **v**.

corr(v,w) Pearson's correlation coefficient for the vectors **v** and **w**.

MEAN FUNCTION

Arguments:

- **v** is an n element vector.

STANDARD DEVIATION FUNCTION

$| n \quad {}_{i}$

Arguments:

- **v** is an n element vector.

Mathcad divides by n rather than n-1 in computing the variance. To find the standard deviation corresponding to a denominator of n-1, multiply Mathcad's result by the square root of n/(n-1).

VARIANCE FUNCTION

Arguments:

- **v** is an n element vector.

Mathcad divides by n rather than n-1 in computing the variance. To find the variance corresponding to a denominator of n-1, multiply Mathcad's result by n/(n-1).

CORRELATION FUNCTION

Arguments:

- **v** is an n element vector with mean mv.
- **w** is an n element vector with mean mv.

DISTRIBUTION FUNCTIONS

Γ(z)	Euler gamma function
erf(x)	Error function
cnorm(x)	Cumulative normal distribution with mean 0 and variance 1.

EULER GAMMA FUNCTION

Arguments:

- z is a real or complex scalar. z cannot be 0, -1, -2...

For complex z, $\Gamma(z)$ is the analytic continuation of real function

For z = 0, -1, -2 $\Gamma(z)$ is undefined.

The following identities involving the gamma function may be useful:

$\Gamma(z+1) = z(z)$

$\Gamma(z)\,(1-z) = \csc(z)$

$\Gamma(n+1) = n!$

ERROR FUNCTION

$\sqrt{\pi} \quad \int_0$

Arguments:

- **x** is a real scalar.

For x complex, erf(x) is undefined.

To define the complementary error function, add the following definition to your document:

cerf(x) := 1 - erf(x)

CUMULATIVE NORMAL DISTRIBUTION

$$\sqrt{2\cdot\pi} \quad \int_{-\infty} \quad (\; 2 \;)$$

Arguments:

- **x** is a real scalar.

For x complex, cnorm(x) is undefined.

RANDOM NUMBERS

rnd(x) Returns uniformly distributed random number between 0 and x.

Arguments:

- **x** is a real scalar.

To generate uniformly distributed random numbers between a and b, define:

RND(a,b) := (b - a)rnd(1) + a

To generate uniformly distributed random integers between m and n, define:

RND_INT(m,n) := m + floor(rnd(n-m+1))

The rnd function uses a seed value of 0 to generate a sequence of random numbers. To generate a different sequence of random numbers, change the seed value by choosing **Randomize** from the **Math** menu and entering a seed value.

To restart the sequence without changing it, choose **Randomize** from the **Math** menu and use the same seed value.

Other distributions:

Gaussian (Normal) distribution
Exponential distribution

x

Must be real.

seed

Must be an integer between 1 and 32000.

Each seed value selects a sequence of uniformly distributed random numbers.

NORMAL DISTRIBUTION

The Box-Muller method allows you to generate normal deviates with mean μ and standard deviation by applying the formula:

$$NORM(\mu,\sigma) := \mu + \sigma\cdot\sqrt{-2\cdot\ln(rnd(1))}\cdot\cos(2\cdot\pi\cdot rnd(1))$$

A document in which this task is performed efficiently is among those in the Mathcad Statistics II Applications Pack.

Other distributions:

Uniform distribution
Exponential distribution

EXPONENTIAL DISTRIBUTION

Given a uniformly distributed random deviate on [0,1], you can generate an exponentially distributed random deviate with mean 1/α by applying the formula:

$$\mathrm{EXPL}(\alpha) := \frac{-1}{\alpha} \cdot \ln(\mathrm{rnd}(1))$$

A document in which this task is performed efficiently is among those in the Mathcad Statistics II Applications Pack.

Other distributions:

Uniform distribution
Gaussian (Normal) distribution

HISTOGRAMS

hist(interv,data) Returns a vector whose ith element is the number of points in **data** falling betwwen the ith and (i+1)th element of **interv**.

Arguments:

- **data** is a vector of real data values.
- **interv** is a vector of real values in ascending order. These represent the intervals into which the elements of **data** will be sorted.

INTERPOLATION

Mathcad provides two interpolation methods:

- **Linear interpolation:** For connecting points with a straight line.
- **Cubic spline interpolation:** For connecting points with a cubic section.

LINEAR INTERPOLATION

linterp(vx,vy,x) Returns a linearly interpolated value at x for data vectors **vx** and **vy**.

Arguments:

- **vx** is a vector of real data values in ascending order. These correspond to the x values.
- **vy** is a vector of real data values in ascending order. These correspond to the y values. The number of elements is the same as **vx**.
- **x** is the value of the independent variable at which you want to interpolate a result. For best results, this should be in the range encompassed by the values of **vx**.

CUBIC SPLINE INTERPOLATION

cspline(vx,vy) Returns vector of second derivatives for data vectors **vx** and **vy**. This vector becomes the first argument of the **interp** function. The resultant spline curve is cubic at the endpoints.

pspline(vx,vy) Same as **cspline**, except the resultant spline curve is parabolic at the endpoints.

lspline(vx,vy) Same as **cspline**, except the resultant spline curve is linear at the endpoints.

interp(vx,vy,vs,x) Returns spline interpolated value of **vy** at a point x.

Arguments:

- **vx** is a vector of real data values in ascending order. These correspond to the x values.
- **vy** is a vector of real data values in ascending order. These correspond to the y values. The number of elements is the same as **vx**.
- **vs** is a vector generated by either cspline, pspline or lspline.
- **x** is the value of the independent variable at which you want to interpolate a result. For best results, this should be in the range encompassed by the values of **vx**.

CURVE FITTING

There are two kinds of curve fitting discussed here:

- **Linear Regression:** Finding the line that best fits a set of data points.
- **Multilinear Regression:** Finding the plane that best fits a set of data points.

LINEAR REGRESSION

slope(vx,vy) Slope of line that best fits data in **vx** and **vy**.

intercept(vx,vy) Intercept of line that best fits data in **vx** and **vy**.

Arguments:

- **vx** is a vector of real data values in ascending order. These correspond to the x values.
- **vy** is a vector of real data values in ascending order. These correspond to the y values. The number of elements is the same as **vx**.

Other regression techniques:

Multilinear regression

MULTILINEAR REGRESSION

Given N ntuple's of data, the goal is to find a plane that best fits these datapoints in a least squares sense.

- Place one column of data in an N-vector **b**.
- Place the other n-1 columns in an N-column matrix **X**.
- The first column of **X** should be a column of 1's.
- The coefficients of the plane that best fits the data are found in the vector:

$$\left(X^T \cdot X\right)^{-1} \cdot X^T \cdot b$$

A document illustrating this method is among those in the Mathcad Statistics II Applications Pack.

Other regression techniques:

Linear regression

TEXT PROCESSING

Inserting Text

Press **Ctrl+T** for text bands stretching across the page. Use " to create text regions of arbitrary shape.

Editing text

Select text, then use **Shift+Del**, **Ctrl+Ins**, and **Shift+Ins** to cut, copy and paste selection.

Fonts

Select, text then choose **Change Font** from the **Text** menu.

Default Fonts

Choose **Change Default Font** from the **Text** menu.

Changing text width

First select the text region with a dashed selection rectangle. Resize text regions by moving the pointer to the edge and dragging the mouse. Change width of text bands by choosing **Set Right Margin** from the **Edit** menu.

Moving text

First select the text region or text band with a dashed selection rectangle. Drag a text region or text band, by moving the pointer inside it and dragging the mouse.

Importing text

This feature is disabled in the demo version.

Inserting text

There are two ways to insert text:

Text bands:

These stretch all the way across the page. Use text bands for large amounts of text.

To create a text band, press **Ctrl+T** and begin typing.

Text region:

Unlike text bands, text regions can have any width. Use them for small blocks of text.

To create a text region, press **"** and begin typing.

To stop entering text, click outside the text band or region. Do not press **Enter**. This will simply add line breaks without leaving the text band or region.

Text band

An area stretching across the page and reserved for typing text.

Text region

A rectangular area reserved for typing text. A text region can have any width and be placed anywhere in the document.

Default text font:

This is the font you see when you first type in a text band or text region.

Selecting text

To select text,

- Place the cursor at one end of the selection.
- Press and hold down the left mouse button while dragging the mouse to the other end of the your selection.
- Once there, let go of the mouse button. The selected area will be highlighted.

Selecting text bands and regions

To select text an entire text band or region:

- Click the mouse just outside the text band or region. This anchors one corner of the selection rectangle.
- Press and hold down the left mouse button. With the button still held, drag the mouse. A selection rectangle framed by dashed lines emerges from the anchor point.
- When the selection rectangle just encloses the text band or region, let go of the left mouse button.

To select more than one text band or region, drag the mouse until the dashed selection rectangle encloses everything you want to select.

Editing text

To copy or cut text:

- Select the text
- Press **Ctrl+Ins** to copy or **Shift+Del** to cut.

To paste text:

- Move the cursor into the text band or region into which you wish to paste text.
- Press **Shift+Ins** to paste the text.

If your clipboard contains equations, an entire text band or an entire text region, Mathcad will not paste it inside a text band or region.

Changing Fonts

To change font, size or style:

- Select the text
- Choose **Change Font** from the **Text** menu.
- Choose a font, size and style from the scrolling list.

Default fonts

To change Mathcad's default text font:

- Choose **Change Default Font** from the **Text** menu.
- Choose a font, size and style from the scrolling list.

Changing the default font affects

- All new text bands or text regions you create.
- All existing text except text whose font has been changed using **Change Font** from the **Text** menu.

Changing text width

To change the width of a text region:

- Select the text region.
- Move the cursor to the right edge of the region. It should change to a double headed arrow.
- Drag the mouse to the right or left. The text region will change size as you drag the mouse.
- Once the text region is the right size, let go of the mouse button.

To change the width of all text bands

- Choose **Set Right Margin** from the **Edit** menu.

You cannot change the width of individual text bands.

Moving text

To drag text bands or regions:

- Select the text band or region with a dashed selection rectangle.
- Move the mouse inside the dashed selection rectangle.
- Drag the mouse. The selected text will move with it.

To cut and paste text bands or regions:

- Select the text band or region with a dashed selection rectangle.
- Press **Shift+Del** to cut it to the clipboard.
- Click the mouse where you want to put the text and press **Shift+Ins**.

Pasting text

When you press **Shift+Ins** to paste text, Mathcad places whatever text was last copied or cut, and inserts it at the location of the cursor.

Only text can be pasted into a text band or region.

Copying text

When you press **Ctrl+Ins** to copy text, Mathcad replaces whatever is on the clipboard with whatever text you have selected.

Cutting text

When you press **Shift+Del** to cut text, Mathcad

- Replaces whatever is on the clipboard with whatever text you have selected.
- Removes the selected text from your document

Dragging the mouse

To drag the mouse:

- Press and hold down the left mouse button.
- Without letting go of the mouse button, move the mouse.

UNITS AND DIMENSIONS

Renaming dimensions:
How to change the default names for the fundamental dimensions L, T, Q, and M.

Defining a unit:
How to define a unit in terms of either existing units, or the fundamental dimensions themselves.

Unit conversions:
How to display the results of your computations in whatever units you wish.

Choosing a system of units:
For convenience, Mathcad comes with mks units installed. You can also install cgs or US customary (Imperial) units by choosing **Units** from the **Math** menu.

Renaming dimensions

The four fundamental dimensions are:

M	mass
L	length
T	time
Q	charge

If your application requires a dimension having different name, you can rename a dimension by choosing **Units** from the **Math** menu.

A common application of this is to rename Q to be degrees for absolute temperature.

Keep in mind that conversions involving Fahrenheit and Celsius require addition as well as multiplication. For this reason, they cannot be considered fundamental units.

Defining a unit

You can define units by multiplying and dividing appropriate combinations of the fundamental dimensions M, L, T, and Q.

For example:

Smoot := 71 in

You can also define units in terms of existing units for example:

Kilosmoot := 1000 Smoot

In defining units, it is often convenient to use the global equals sign, ≡, instead of the := Mathcad evaluates all expressions involving the before it evaluates those involving a :=. This allows you to place unit definitions inconspicuously at the bottom of your document.

To insert the global equals sign, type ~ .

Unit conversions

Mathcad displays results in terms of the four fundamental dimensions. For example:

this placeholder
and press Enter

You can use this same technique to display a number in terms of any other number. For example, to display a result in terms of π,

- Type the Greek letter into the placeholder.

Systems of Units

By default, Mathcad loads the mks units. To select a different system of units, choose **Units** from the **Math** menu.

You can choose MKS, CGS, US Customary, or no units at all. Once you choose a system of units, Mathcad defines all the common units in that system without your having to do so.

To see the complete list of available units for each system of units, click on the appropriate name below:

MKS (or SI)
CGS
US Customary

MKS units

The following units are available by default or when you choose MKS from the Units dialog box:

acre	kW
amp	lb
atm	lbf
BTU	liter
cal	m
cm	mi
coul	min
day	mL
deg	mm
dyne	mph
erg	newton
farad	oersted
fl_oz	ohm
ft	oz
g	Pa
gal	poise
gm	psi
hectare	rad
henry	sec
hp	siemens
hr	slug
in	stokes
in_Hg	tesla
joule	ton
kcal	tonne
kg	torr
kgf	volt
km	watt
kph	weber
	yd
	yr

CGS units

The following units are available when you choose CGS from the Units dialog box:

acre	liter
amp	m
atm	mi
BTU	min
c	mL
c_	mm
cal	mph
cm	newton
coul	ohm
day	oz
deg	Pa
dyne	poise
erg	psi
farad	rad
fl_oz	sec
ft	slug
g	statamp
gal	statcoul
gm	statfarad
hectare	stathenry
henry	statohm
hp	statseimens
hr	stattesla
in	statvolt
in_Hg	statweber
joule	stokes
kcal	ton
kg	tonne
kgf	torr
km	volt
kph	watt
kW	yd
lb	yr
lbf	

US Customary units

The following units are available when you choose US from the Units dialog box:

acre	kW
amp	lb
atm	lbf
BTU	liter
cal	m
cm	mi
coul	min
day	mL
deg	mm
dyne	mph
erg	newton
farad	oersted
fl_oz	ohm
ft	oz
g	Pa
gal	poise
gauss	psi
gm	rad
hectare	sec
henry	siemens
hp	slug
hr	stokes
in	tesla
in_Hg	ton
joule	tonne
kcal	torr
kg	volt
kgf	watt
km	weber
kph	yd