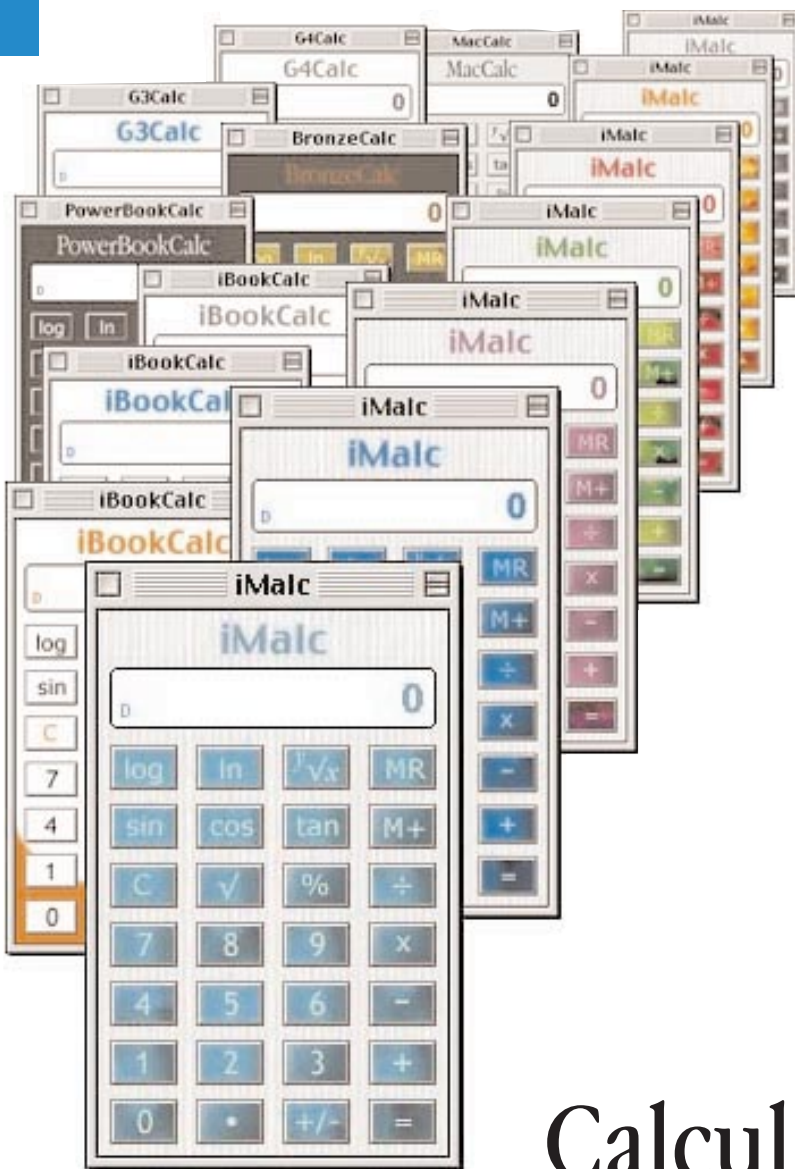# Fruit is good for you.

Enrich your life.

iMalc Manual
version 1.1.2

incorporating G3Calc, G4Calc
PowerBookCalc, BronzeCalc
iBookCalc and MacCalc

Calculate Different. NotTM

Mac

eyecatcher

▲ **WARNING:** Eyecatcher Productions make no claims as to the reliability or accuracy of any of the data, conversion factors, constants or equations supplied with iMalc. The data has been assembled from a variety of sources, and has often differed between sources. Definitions of standards are always open to differing interpretations and international agreements, as well as to conformation by countries to those agreements, and by local practice. In addition the data has been reformatted and there is always the possibility of formatting and typographical errors.

ABOUT THIS MANUAL:

This manual was typeset on a Macintosh PowerBook G3 Series laptop computer. It was laid out using QuarkXPress. Image and graphic manipulation was carried out using Adobe Photoshop, GraphicConverter and ResEdit. Fonts used are Trebuchet, Verdana, Zapf Dingbats.
It was converted into Adobe Acrobat PDF format using Adobe Acrobat Distiller.
In the Contents, Index and List of illustrations, you can click a topic to jump to that location. Clicking on blue email and website URLs will launch the relevant software and in the case of URLs will take you to that website.

Published in London, England.

iMalc

incorporating G3Calc, G4Calc, PowerBookCalc,
BronzeCalc, iBookCalc and MacCalc

Made with **REAL**basic®

iMalc version 1.1.2

System requirements

iMalc runs under Mac OS 7.1 and later. It may work on earlier system versions. iMalc has been tested on a PowerBook G3 and iMac running Mac OS 8.5.1, a G3, iMac and 4400 running 8.6, a 7100/80 running 7.5.3 and a Colour Classic running 7.1. It requires approximately 5Mb of RAM with virtual memory turned off or 4.5Mb of RAM with virtual memory turned on, and 3.5Mb of hard drive space. You can get by with 3Mb of RAM, but if you cycle through all the flavours, you will need 5Mb of RAM. iMalc displays best with thousands or millions of colours and using Charcoal as the system font.

Conventions

The following conventions are used throughout this manual to draw your attention to certain details.

▲ **WARNING:** This is used where there is a slight warning required, more for informational purposes than likely to be a serious problem.

This warning triangle icon indicates a far more serious warning, and relates to information that is likely to be critical or lead to potential loss of data or accuracy.

iv

Assumptions

This manual assumes you are already familiar with the basic Macintosh operations and actions, such as double-clicking and click-dragging, and metaphors such as the desktop.

Refer to your computer's manual or help system if you are unsure about basic Macintosh operation.

Contents

v

List of illustrations

Introduction

iMalc is a fully featured scientific calculator designed to replace the standard Apple calculator. It offers more functions and allows you to customise its appearance to complement your flavour of iMac, iBook, PowerBook, G3, G4 or platinum Mac.

iMalc features include 18 mathematical functions in addition to the basic operators and a saveable memory.

Functions available:

- ◆ Base$_{10}$ (common) logarithm and inverse, 10^x

- ◆ Natural logarithm and inverse, e^x

- ◆ Sine, cosine, tangent and their inverses arcsine, arccosine and arctangent

- ◆ Square root, x^2, y^{th} root of x, x^y, $\frac{1}{x}$

- ◆ Percent, π (Pi) and +/- keys

iMalc also has memory plus, minus, recall and clear buttons. The trignometrical functions may be calculated in degrees, radians or grads.

iMalc supports drag and drop and copying and pasting to and from the display window.

You can customise iMalc to complement your flavour of iMac, iBook, PowerBook, G3, G4 or platinum Mac. The key and display colours change to your favourite flavour.

Choose between Bondi Blue, Blueberry, Grape, Lime, Strawberry, Tangerine or Graphite Special Edition.

iBook owners can choose between Blueberry, Tangerine and Graphite, PowerBook owners have the choice of the original Charcoal or the new Bronze keyboard Lombard, while G3 and G4 owners are catered for as well as those with an original Platinum Mac.

iMalc remembers the value stored in memory, the flavour and window position on quitting.

Versions of iMalc are available for Macintosh PowerPC, 68k and fat binary (runs on both PPC and 68k). A Windows version will be available soon for those who love Macs but have to use Wintel boxes and wish to make a statement :-)

iMalc is shareware and costs just £10/US$15. This means that you may try it for 30 days and must then either register the program and pay for iMalc or delete it from your hard drive.

iMalc runs under Mac OS 7.1 and later. It may work on earlier system versions. iMalc has been tested on a PowerBook G3 and iMac running Mac OS 8.5.1, a G3, iMac and 4400 running 8.6, a 7100/80 running 7.5.3 and a Colour Classic running 7.1.

It requires approximately 5Mb of RAM with virtual memory turned off or 4.5Mb of RAM with virtual memory turned on, and 3.5Mb of hard drive space. You can get by with 3Mb of RAM, but if you cycle through all the flavours, you will need 5Mb of RAM.

iMalc displays best with thousands or millions of colours and using Charcoal as the system font.

iMalc is built using REALbasic from REAL Software Inc, a stunning graphical object orientated Integrated Development Environment for Macintosh, which also ports Macintosh applications to Windows. Other compilers are planned.

The latest version of iMalc can be downloaded from:
http://homepage.mac.com/eyecatcherprods/iMalc.html

The iMalc package should contain the application iMalc, a Read Me file, this iMalc manual, the Register iMalc application and a Registering Read Me.
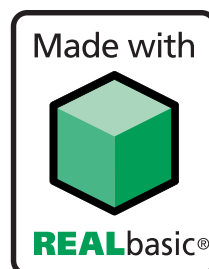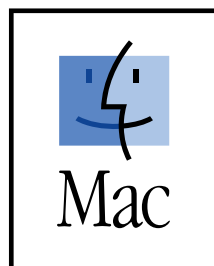
Quick Start

If you're in a hurry to start using iMalc and don't want to read the manual at this stage then here are some quick start instructions.

Calculator operation

iMalc is easy to use and works similar to any other calculator that you might have used.

The logic employed is algebraic rather than Reverse Polish Notation (RPN). This means that to do a calculation such as 15+2=17, you enter it as seen. You would enter 15, followed by +, followed by 2 and then the equals.

If you are employing the +/- sign change key, then use it before the equals eg for 100 x (-2)=-200 you would enter 100 followed by x then 2, then the +/- before finally pressing the equals key. Or you can use the +/- before entering the number: 100 then x, then +/-, then 2, then=.

When employing the % key, you do not need to press equals, just the operator eg 100+20% will give 120. 200 x 20% will give 40.

The C key will totally clear the calculator. The running total is always remembered until the C key is pressed. eg if you enter 15+2=17 then press 12+ the answer will be 29.

You can use the delete key to backspace one character.

The trig and log functions are applied by pressing the relevant key after you've entered the number. To obtain their inverses, hold down the option key.

The M+ key will add the number displayed into the memory. The MR key will recall the memory and show it in the display.

Hidden functions

To obtain further functions press and hold the option key. In general, the function obtained is the inverse of the normal one eg sin becomes arcsine, log natural (ln) becomes e^x, square root becomes x^2.

The M+ key becomes the M- key, subtracting the displayed value from the memory, and MR becomes CM, clearing the memory.

The C key becomes CE, which will clear the last entry, but not totally clear the calculator.

The two keys that have new functions are the % which becomes π (pi), and +/- which becomes $\frac{1}{x}$.

The last two keys are the yth root of x and (with option key down) x^y (x to the power y). To use these keys enter the x value first, then press the key. You will be prompted to enter the y root or power, and when you've done so, press equals to get the result. Unlike many calculators, iMalc lets you correctly determine odd number integer roots of negative numbers.

The Display

The display will show up to 12 digits (including a minus if present).

Also in the display, on the left hand side are two indicators. The one on the bottom left is the degree/radian/grad indicator and shows which mode the calculator is in, displaying a D, R or G respectively. This is always lit.

Initially unlit is the memory light, M, above the trig mode indicator. If there is a number stored in memory other than zero then this indicator will be lit.

To change the trig mode, just click on or near the indicator light. It will cycle through D,R and G.

You can copy, cut and paste to and from the display window using either the Edit menu, or use the standard Macintosh keyboard shortcuts of ⌘-**C**, ⌘-**X** and ⌘-**V** respectively. Clear in the edit menu will act like the C button and completely clear the calculator. You can also drag and drop numerical values on to the display.

4

Using the keyboard

You can also use the keyboard to operate the calculator. The standard number and keypad keys work, as do the operator and backspace. Return and Enter also act as the equals key. Comma also functions as the decimal separator in addition to point (dot).

All of the functions except the memory are accessible via the keyboard too. Log base 10 is L, natural logs is N, percent is P, +/- is M, sin is S, cosine is C, tangent is T, square root is R, y^{th} root of x is Y, and clear is X or Escape.

You obtain the inverse functions in the same way as the buttons, by holding down the option key and selecting the appropriate key, eg Option-S is arcsine.

The calculator keys will depress when the keyboard is used.

Changing the appearance

Changing iMalc's appearance is easy. Simply choose your flavour from the Flavour menu. Or use the shortcut keys.

The flavours of iMalc available are original Bondi Blue, Blueberry, Grape, Lime, Strawberry, Tangerine and Graphite Special Edition.

In addition you can also choose G3Calc, G4Calc, two flavours of PowerBook including the Bronze keyboard Lombard G3s, Blueberry, Tangerine and Graphite flavours of iBook, and if you have an older Mac, standard platinum MacCalc.

The keyboard shortcuts are ⌘-**1** for Bondi, ⌘-**2** to ⌘-**7** for Blueberry, Grape, Lime, Strawberry, Tangerine and Graphite respectively, ⌘-**8** for G3, ⌘-**9** for G4, ⌘-**0** for Wallstreet PowerBooks, ⌘-**minus** for Lombard PowerBooks, ⌘-**[** for Blueberry iBook, ⌘-**]** for the tangerine flavour, ⌘-**** for the graphite flavour and ⌘-**=** for platinum.

iMalc will remember and display your chosen flavour the next time it starts up.

Error codes

You may occasionally see some error codes. You will get the warning 'imaginary' if you try to take the square root of a negative number. You will also see NAN, often followed by a number in brackets, or INF.

5

INF means infinity, and can be obtained through erroneous operators such as division by zero.

NANs are Not A Number and are other error codes, generally for numbers which are out of range, such as trying to take the arcsine of 100. The numbers in brackets sometimes following NANs are specific error codes and are listed later in this manual.

Preferences file

iMalc uses a preferences file to store certain values, including the current flavour, the window position and the contents of the memory.

If you trash the iMalc Preferences file, which is located in the Preferences Folder of the System Folder, these values will be reset to their default values.

Interface

In common with most Macintosh applications, iMalc has a familiar interface that consists of icons, windows and menus and conforms to the Apple *Human Interface Guidelines*.

Although the main calculator has one window, there are other windows, such as the About Box and Registration window, that can be called up.

The menu layout is also fairly simple, with just one additional menu, Flavour, added to the standard Apple, File, Edit and Help menus.

Icons

iMalc uses the following icons on the desktop:

16 bit icon	*256 colour icon*	*black and white icon*
small 16 bit icon	*small 256 colour icon*	*small black and white icon*

Windows

When you double-click on iMalc for the first time to start up the application, the first window you see is a Welcome screen:

8

This screen offers you a choice of three buttons to press, Register Now, Register Later and Try iMalc. If you wish to register straight away, press the Register Now button. That option will take you on to the Register screen.

If you wish to register later, press that button, and if you want to try iMalc for a trial period before deciding whether to register and pay the shareware fee, press the Let Me Try iMalc button. Pressing either button will dismiss the Welcome screen.

This screen only appears the very first time that you start up iMalc. You must press one of the buttons to exit the screen. Clicking on the welcome screen or pressing return will not dismiss the screen

After you have dismissed the Welcome screen, you are presented with the main iMalc calculator window in the default flavour of Blueberry:
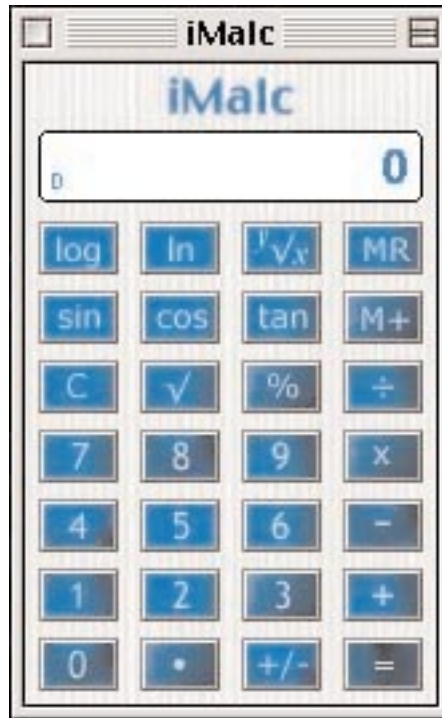
This is a normal style document window, except that it doesn't have a resize box at the top right of the title bar or a grow handle at the bottom right, meaning that you can't resize this window.

The close box at the upper left of the title bar will not only close the window, but quit iMalc.

The windowshade box on the top right of the title bar will reduce the window to just the title bar:

Clicking on the windowshade box again will expand the window back to normal size.

You can also use the standard keyboard shortcuts to close (and quit) the window and iMalc application: ⌘-**W** and ⌘-**Q**.

There are other windows that you can call up at any time. If you choose About iMalc from the Apple menu, you are presented with the About Box screen:
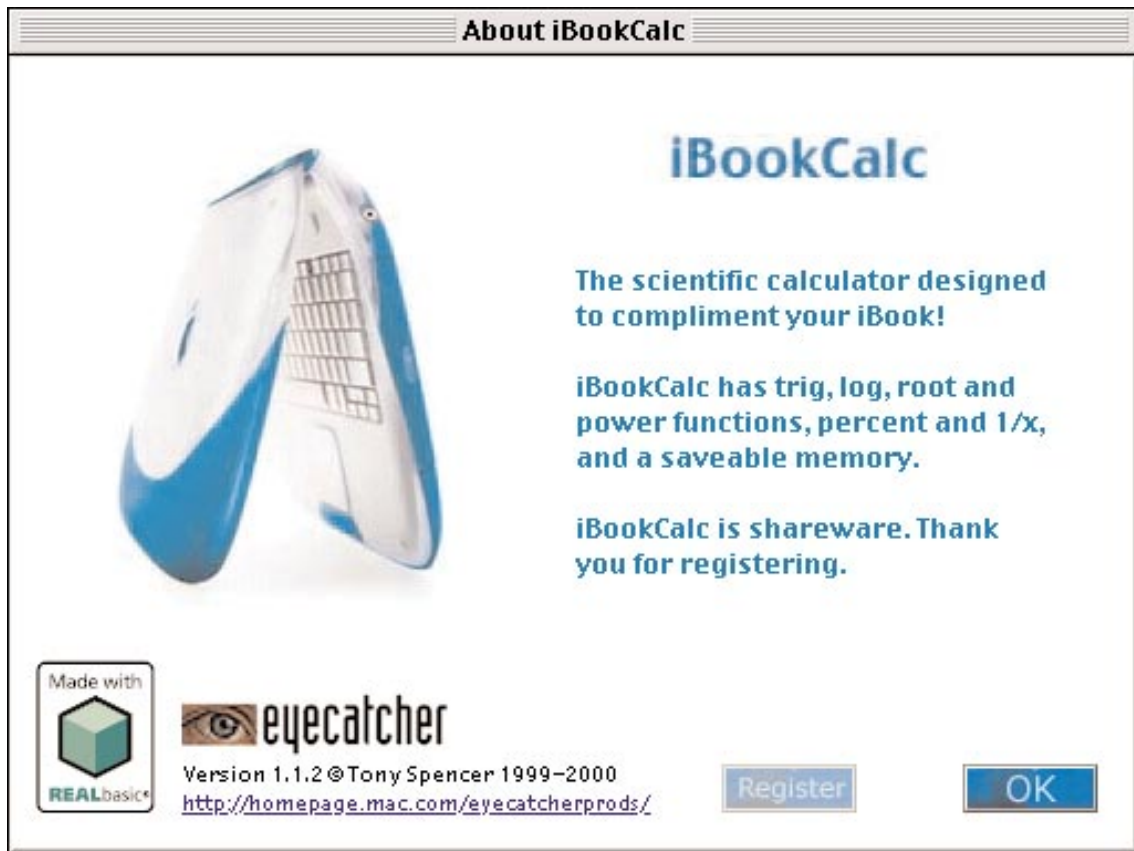
The About Box gives you some very basic information about iMalc's features. The About Box will reflect your choice of flavour.

It contains two buttons, Register and OK. Pressing Register will take you to the Register screen and the OK button will dismiss the About Box. Clicking anywhere in the About Box or pressing return will also dismiss the About Box.

The About Box also contains an Eyecatcher logo, the version number, copyright statement, and the Eyecatcher Productions web site URL. Clicking on the URL will launch your browser and take you to the Eyecatcher web site.

Also visible is a Made with REALbasic logo, the programming application used to develop iMalc.

Under the Help menu, there are two windows you can call up, the iMalc Help window, and the Contacts and Credits window.

The online iMalc help covers all the basics, but is not as

informative as this manual. The iMalc Help window looks like this:
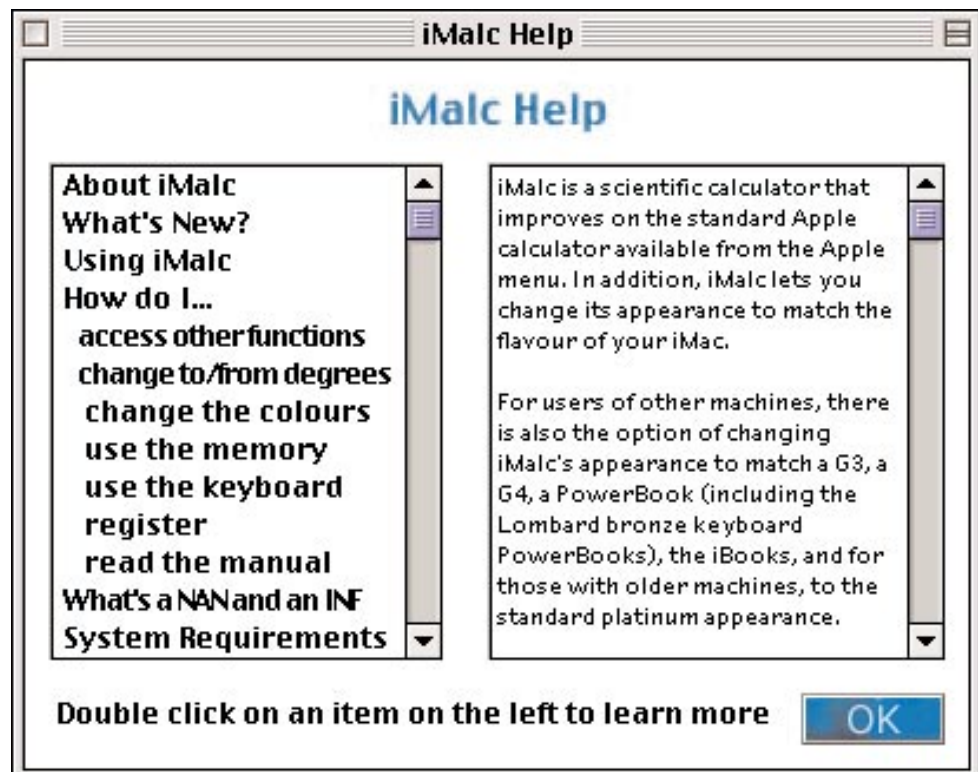
You can scroll down the contents in the box on the right to read the help. Double click on a topic on the left to go to the online Help for that topic. The selection is highlighted:
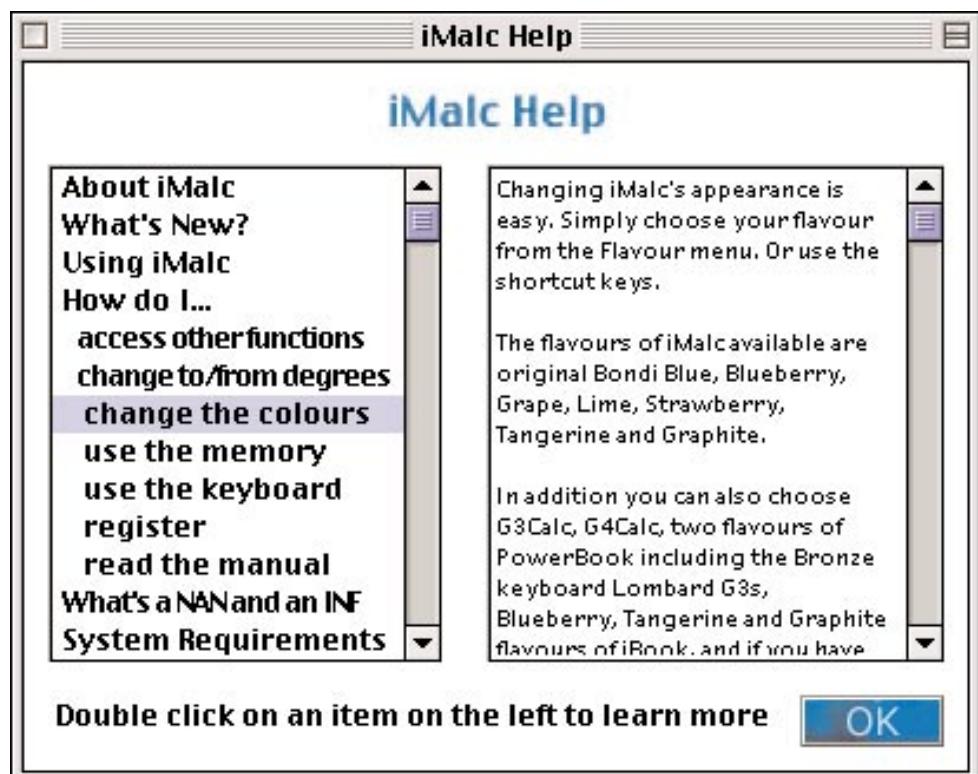
The Contacts and Credits window has buttons down the left:

The top button is the default, and shows the version number, email and website addresses for Eyecatcher. You can double click on the email address to launch your email client and open a blank email mailto: message, or on the website URL to launch your browser and go to the Eyecatcher website.

The second button leads to a credits screen, and the third button

lists the names of the beta testers who helped test and debug the first version of iMalc.

There are two other windows that you will see, the Reminder window and the Registration window.

After a while, you will start to see the Reminder window pop up at start up and when you quit. It is a reminder to pay the shareware fee:

It gives you details of the number of times you have used iMalc, and the total time you have been using it. This window will appear more and more frequently as time goes by. Once you register iMalc, this screen will no longer appear.

There are three buttons. When the window appears, all three buttons are dimmed. After a few seconds, they start to be enabled, with Register Now appearing first, then Register Later, and finally, after 10 seconds the Still Trying... button.

Pressing Register Now will take you to the Register window, and the other two, after the pause, will exit the window.

The Register window, shown overleaf, is accessible from the About Box window, and the Reminder window above.

The Registration window has some very brief instructions on how to pay, using the Register iMalc program that comes with iMalc.
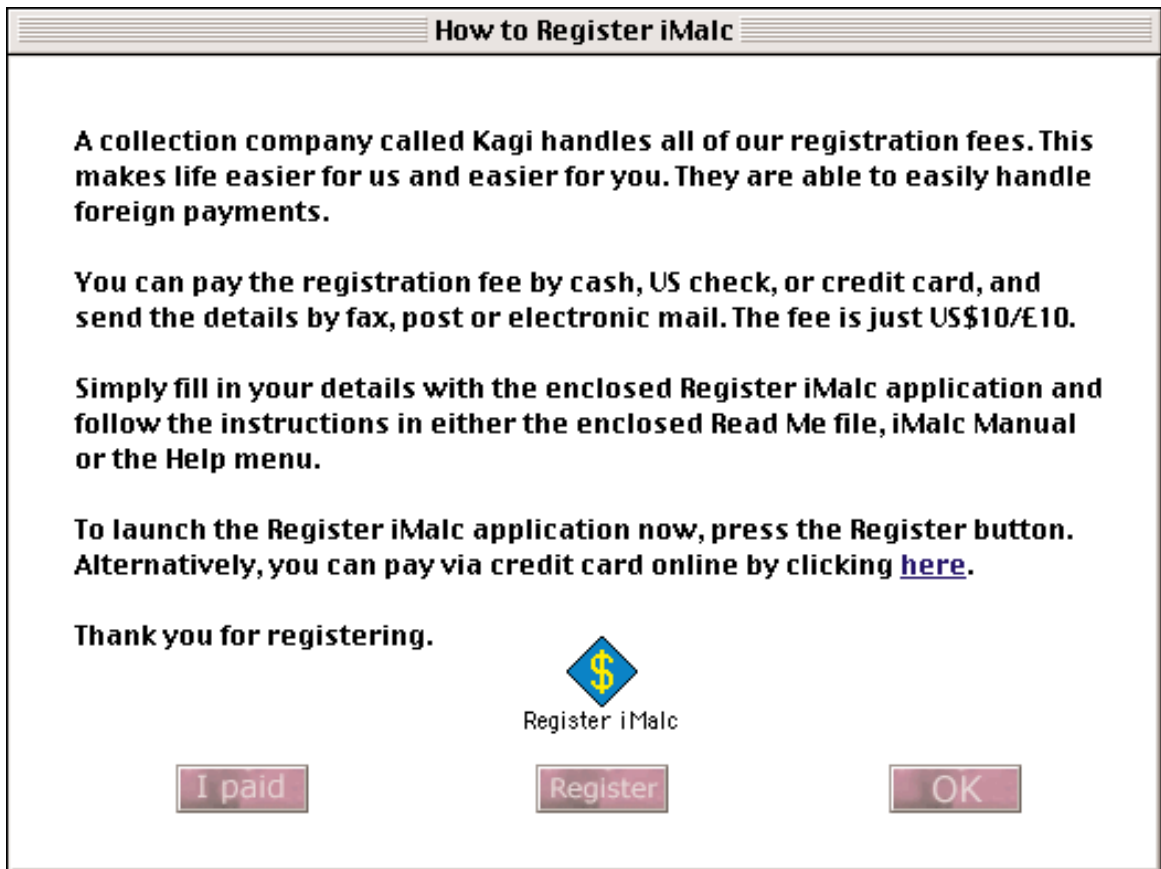
There are three buttons, I Paid, Register and OK. The OK button dismisses the window. The Register button will open the Register iMalc application for you to fill in your registration details. See later in this manual for full details on how to use Register iMalc.

The I Paid button is for when you have registered iMalc and paid the shareware fee. Pressing it will give you a spoken thank you, and will dim the Register button in the About Box window. It will also stop the Reminder screen from appearing.

You can also go direct to our page on the Kagi website to pay by credit card online by clicking on the blue word "here" as instructed. This will launch your browser and point you to the Eyecatcher page on Kagi.

▲ **WARNING:** Note that if you trash the iMalc Preferences file in the Preferences folder of the System Folder, then the Reminder screens will begin to appear again, and the Register button in the About Box window will no longer be dimmed.

Menus

The menus in iMalc follow the Apple standard as outlined in Apple's *Human Interface Guidelines*, with only one addition to the standard Apple set of Apple menu, File menu, Edit menu, and Help menu: the Flavour menu.

They run from left to right: Apple, File, Edit, Flavour and Help.

Under the Apple menu at the top is the About iMalc menu item:

Choosing this item will open up the About iMalc window, previously described.

Under the File menu are two items, Close and Quit, with the standard keyboard shortcut commands:
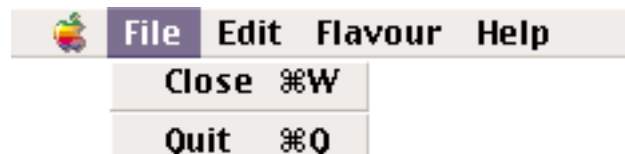
Choosing Close or ⌘-**W** will close the About Box, Help, Contacts and Register windows, but not the Reminder window. If only the main iMalc Calculator window is open, then it will not only close the window, but also quit iMalc. The command is added to the calculator window for added convenience, and to follow Apple guidelines.

Choosing Quit or ⌘-**Q** from any window will quit iMalc. Just before quitting, iMalc will save to its Preferences file, retaining the value in memory, if any, your flavour preference, and the window position.
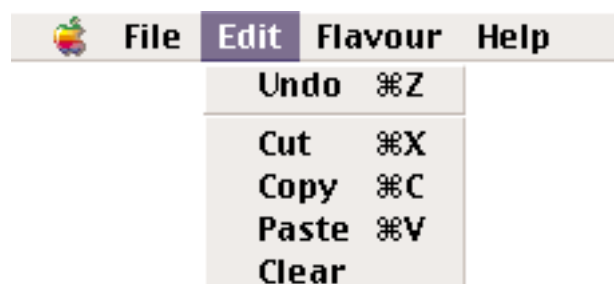
The Edit menu has the standard items of Undo, Cut, Copy, Paste and Clear. There is no Preferences menu item, since they are handled under the Flavour menu.

The Undo menu item is effectively the same as the CE, clear last entry, key. It will clear the last operand input, but not the calculation total or previous entry. It will not undo the last calculation. See the chapter on Calculator Operation for further details.

The Cut, Copy and Paste items are available in the Calculator window and the Copy item in the Help window.

Copying in the calculator window will copy the current display into the system clipboard, and in the Help window, the currently selected text in the right hand info box into the system clipboard.

Using Cut in the calculator window will copy the display to the clipboard and then set the display to zero, and the operand to zero.

If there is already a numerical value in the system clipboard, then using Paste will put the value in the display and set the operand to that value.

The Edit menu uses the standard keyboard commands for Undo, Cut, Copy and Paste.

The Flavour menu has fifteen items:

🍎	**File**	**Edit**	**Flavour**	**Help**	
			Bondi		⌘1
			Blueberry		⌘2
		✓	**Grape**		⌘3
			Lime		⌘4
			Strawberry		⌘5
			Tangerine		⌘6
			Graphite Special Edition		⌘7
			G3		⌘8
			G4		⌘9
			PowerBook G3		⌘0
			PowerBook G3 Bronze		⌘-
			iBook Blueberry		⌘[
			iBook Tangerine		⌘]
			iBook Graphite		⌘\
			Platinum		⌘=

It allows you to choose your flavour, or appearance, of iMalc. You can either choose the menu item or use the relevant keyboard shortcut. A checkmark appears to the left of the current flavour.

Depending on your choice of flavour, both the About iMalc menu item in the Apple menu, and the iMalc Help items will change their name to reflect your flavour choice. For example if you choose G3 the items will change to About G3Calc and G3Calc Help respectively.

The final menu is the Help menu:
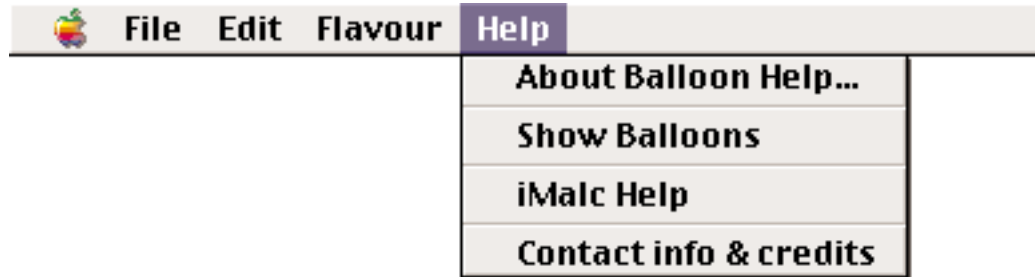
The first two items are Apple standards: About Balloon Help... which calls up the system info on Balloon Help, and Show Balloons, which turns on Balloon Help. This item is a toggle. If Balloon Help is on then the item changes to Hide Balloons.

Balloon Help is a system-wide Apple learning aid to give you help on items on the screen. iMalc fully supports Balloon Help. If balloons are turned on, you can get brief help on buttons, keys, windows and menus, which will give you a quick introduction to the iMalc interface.

The iMalc Help item will call up iMalc's online help, namely the iMalc Help window.

The Contact info & Credits item will display the Contacts and Credits window.

Apple Guide and the newer Apple Help HTML based system are not currently supported.

Interface

17

Calculator Operation

iMalc is easy to use and looks like, and works similar to, any other calculator that you might have used. You can click on the buttons, or use the keyboard to operate iMalc.

The layout of the iMalc calculator is as follows:

Close box — Title bar — Windowshade or collapse box — Memory indicator — Display — Trig mode indicator — y^{th} root of x key — Logarithm base$_{10}$ key — Natural logarithm key — Trignometry functions — Memory keys — Clear key — Percent key — Square root key — Number keys — Operator keys — Decimal point key — Plus/Minus key
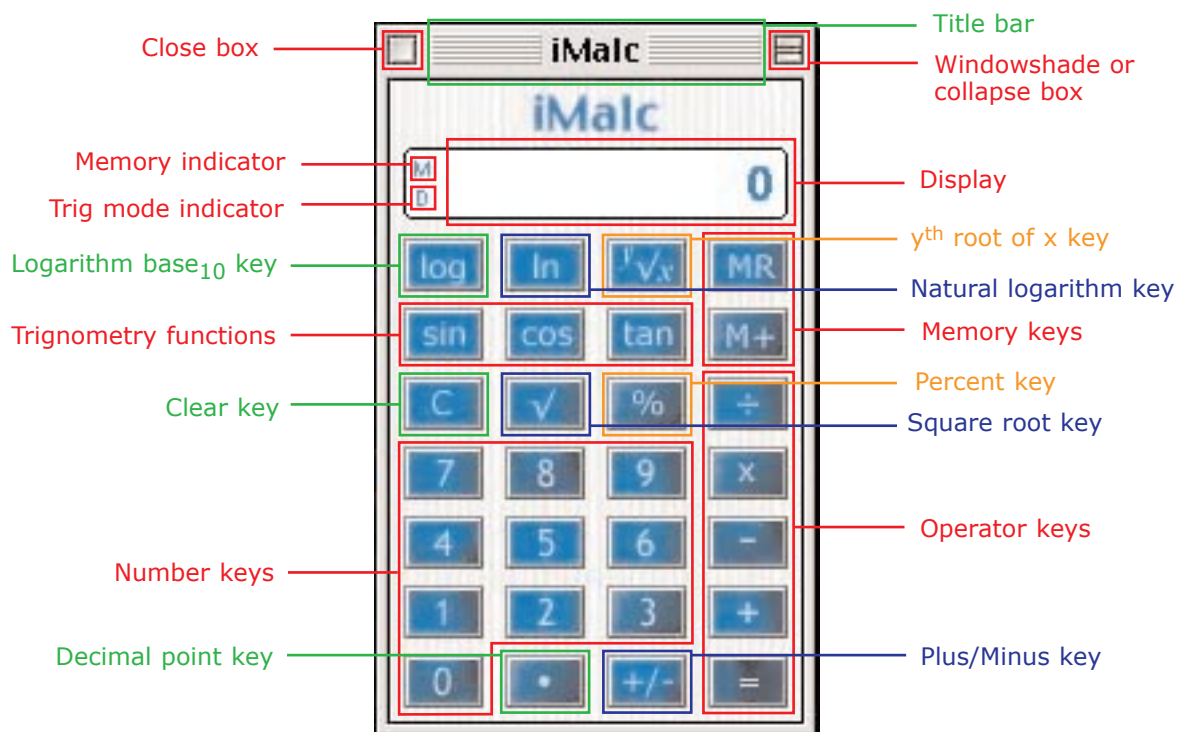
The key layout follows fairly standard layout. Various types of keys are grouped together.

The Number keys are based at the bottom in a standard layout, with the Decimal Point and Plus/Minus keys just below. The Operator keys (+,-,x,÷,=) are in a column on the right. The Memory keys are located just above the operator keys on the top right. The Clear key is on the left just above the number keys.

Above the number keys are the various function keys: trignometry, logarithm, power, root and percent keys. Above the keys is the display, including the memory and trig mode indicators.

In addition, more functions can be obtained by holding down the Option (Alt), ⎇ key:

10^x key ——— [10^x]

Trignometry inverse functions ——— [asin] [acos] [atan]

Clear last entry key ——— [CE]

x^y key ——— [x^y]

e^x key ——— [e^x]

Memory keys ——— [M−]

π (Pi) key

x^2 (Square) key

$^1\!/x$ (inverse) key
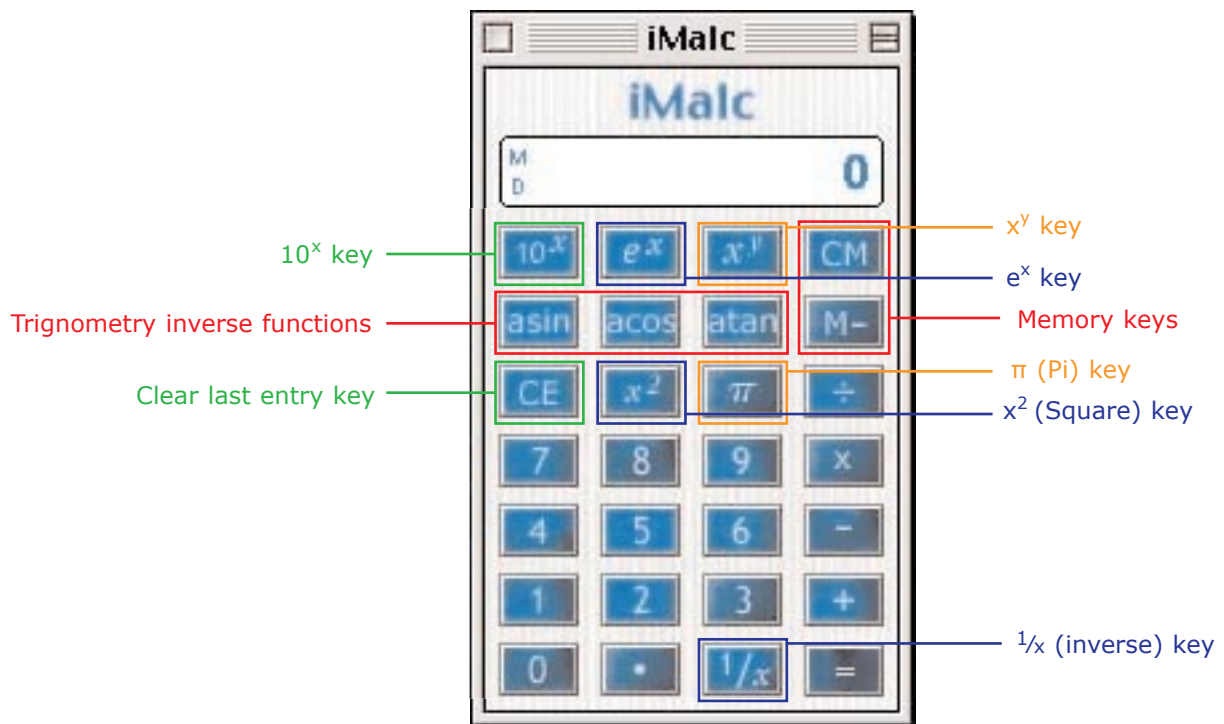
Generally, the inverse of the function is obtained when holding down the Option button and pressing the key.

Thus the sine (sin) key becomes arcsine, the cosine (cos) key becomes arccosine, the tangent (tan) key becomes arctangent, the logarithm base$_{10}$ (log) key becomes 10^x, the natural logarithm (ln) key becomes e^x, the square root ($\sqrt{}$) becomes a square key, x^2, and the yth root of x becomes the x^y (x to the power y) key.

The memory plus (M+) key becomes the memory minus (M-) key and the memory recall (MR) key becomes the clear memory (CM) key.

The three exceptions are the percent (%) key, which becomes the π (pi) key, the plus/minus key which becomes the inverse key, $^1\!/x$, and the clear (C) key which becomes the clear last entry (CE) key.

The calculator window is a standard window with a title bar, close box and windowshade (collapse) box. The window is not resizable and hence there is no grow or zoom box. Clicking on the close box will quit iMalc, and clicking on the windowshade box will collapse the window to just the title bar. You can move the window around by click-dragging the title bar (ie clicking and holding the mouse down while dragging).

iMalc Manual

The keys

7 8 9 4 5 6 1 2 3 0	Number keys
÷ x − + =	Operator keys

·	Decimal point key
log	Logarithm to base 10 key
10^x	10^x or anti-logarithm to base 10 key (+ Option key)
ln	Natural logarithm (base e) key
e^x	e^x or anti-natural logarithm key (+ Option key)
$\sqrt[y]{x}$	y^{th} root of x key
x^y	x^y (x to the power y) key (+ Option key)
sin	Sine x key
asin	Arcsine x key (+ Option key)
cos	Cosine x key
acos	Arccosine x key (+ Option key)
tan	Tangent x key
atan	Arctangent x key (+ Option key)
C	Clear key
CE	Clear last entry key (+ Option key)

	Square root x ($\sqrt{\ }$) key
	x^2 (x squared) key (+ Option key)
	Percent (%) key
	π (pi) key (+ Option key)
	Plus/Minus key
	$\frac{1}{x}$ (inverse of x) key (+ Option key)
	Memory recall key
	Clear memory key (+ Option key)
	Memory plus key
	Memory minus key (+ Option key)

Operation

The logic employed by iMalc is algebraic rather than Reverse Polish Notation (RPN). This means that to do a calculation such as 15+2=17, you enter it as seen, or as you would when writing an algebraic equation. You would enter 15, followed by +, followed by 2 and then the equals.

To get 18x3=54 you would therefore:
    Press 1
    Press 8
    Press x
    Press 3
    Press =
and the result, 54, will be displayed.

If you are employing the +/- sign change key, then use it before the equals key eg for 100 x (-2)=-200 you would:
    Enter 100
    Press x
    Press 2
    Press +/-
    Press =

Or you can use the +/- before entering the number:
> Press 1
> Press 0
> Press 0
> Press x
> Press +/-
> Press 2
> Press =

When employing the % key, you do not need to press equals key, just the operator key eg 100+20% will give 120. 200 x 20% will give 40.

The C key will totally clear the calculator. The running total is always remembered until the C key is pressed. eg if you enter 15+2=17 then press 12+ the answer will be 29.

You can use the delete key to backspace one character.

The trig and log functions are applied by pressing the relevant key after you've entered the number. To obtain their inverses, hold down the option key and press the relevant arc key. The trig function will return the correct result for the selected trig mode, degrees, radians or gradients.

The M+ key will add the number displayed into the memory. The MR key will recall the memory and show it in the display.

Hidden functions

To obtain further functions press and hold down the option key. In general, the function obtained is the inverse of the normal one

so   sin (sine) becomes asin (arcsine)
> cos (cosine) becomes acos (arccosine)
> tan (tangent) becomes atan (arctangent)
> log (base 10) becomes 10^x (anti-log base 10)
> log natural (ln) becomes e^x
> √ (square root) becomes x^2

The M+ key becomes the M- key, subtracting the displayed value from the memory, and MR becomes CM, clearing the memory if the value in memory is other than zero.

The C key becomes CE, which will clear the last entry, but not totally clear the calculator ie iMalc remembers any calculation total.

The two keys that have new functions are the % which becomes π (pi), and +/- which becomes $\frac{1}{x}$.

The last two keys are the yth root of x and (with option key down) x^y (x to the power y). To use these keys enter the x value first, then press the key. You will be prompted to enter the y root or power, and when you've done so, press equals to get the result. Unlike many calculators, iMalc lets you correctly determine odd number integer roots of negative numbers.

The display

Memory indicator ———

Trig mode indicator ———

The display will show up to 12 digits (including a minus if present). Numbers in exponential form will be shown with an e followed by a plus or a minus and then exponent. eg
$1.633124e+16 = 1.633124 \times 10^{16}$

Also in the display, on the left hand side are two indicators. The one on the bottom left is the degree/radian/grad indicator and shows which mode the calculator is in, displaying a D, R or G respectively. This is always lit.

Initially unlit is the memory light, M, above the trig mode indicator. If there is a number stored in memory other than zero then this indicator will be lit.

To change the trig mode, just click on or near the indicator light. It will cycle through D,R and G.

You can copy, cut and paste to and from the display window using either by selecting the relevant command in the Edit menu, or use the standard Macintosh keyboard shortcuts of ⌘-**C**, ⌘-**X** and ⌘-**V** respectively. Cutting will copy the displayed value to the clipboard and then set the display and operand to zero.

The operand is the number to be operated on when the next operator key is pressed, and is effectively the number currently in the display. An operator key is any one of +, -, ÷, x, or =, or any of the mathematical function keys such as log or sin.

Clear in the edit menu will act like the C button and completely clear the calculator.

You can also drag and drop numerical values on to the display from other applications. When the cursor is over the display it changes from an arrow to a hand and the display takes on a border colour to signify you can drag and drop:

If you then click and hold the mouse button you start dragging. The cursor changes to a closed hand and a dotted rectangle appears indicating where the drag has moved to. You can simply continue dragging and drop the rectangle onto any application, or on to the desktop to create a clipping containing the value in the display, as per the usual Macintosh Drag and Drop function:

Using the keyboard

You can also use the keyboard to operate the calculator. The standard number and keypad keys work, as do the operator and backspace. Return and Enter also act as the equals key. Comma also functions as the decimal separator in addition to point (dot).

All of the functions except the memory are accessible via the keyboard too. Log base 10 is L, natural logs is N, percent is P, +/- is M, sin is S, cosine is C, tangent is T, square root is R, y^{th} root of x is Y, and clear is X or Escape.

You obtain the inverse functions in the same way as the buttons, by holding down the option key and selecting the appropriate key, eg Option-S is arcsine.

The calculator keys will appear to be depressed when the keyboard is used to give a visual clue as to which keys have been pressed.

The Preferences File

iMalc uses a preferences file to store certain values, including the current flavour, the window position and the contents of the memory.

If you trash the *iMalc Preferences* file, which is located in the Preferences Folder of the System Folder, these values will be reset to their default values.

25

Flavours

Changing iMalc's appearance is easy. Simply choose your flavour from the Flavour menu. Or use the shortcut keys.

The flavours of iMalc available are original Bondi Blue, Blueberry, Grape, Lime, Strawberry, Tangerine and Graphite Special Edition.

In addition you can also choose G3Calc, G4Calc, two flavours of PowerBook including the Bronze keyboard Lombard G3s, Blueberry, Tangerine and Graphite flavours of iBook, and if you have an older Mac, standard platinum MacCalc.

The keyboard shortcuts are ⌘-**1** for Bondi, ⌘-**2** to ⌘-**7** for Blueberry, Grape, Lime, Strawberry, Tangerine and Graphite respectively, ⌘-**8** for G3, ⌘-**9** for G4, ⌘-**0** for Wallstreet PowerBooks, ⌘-**minus** for Lombard PowerBooks, ⌘-**[** for Blueberry iBook, ⌘-**]** for the tangerine flavour, ⌘-**** for the graphite flavour and ⌘-**=** for platinum.

iMalc will remember and display your chosen flavour the next time it starts up.

The About Box and Help windows change to reflect your choice of flavour, as do the menu items.

26

Bondi iMalc

Blueberry iMalc

Grape iMalc

Lime iMalc

Strawberry iMalc

Tangerine iMalc

Graphite iMalc
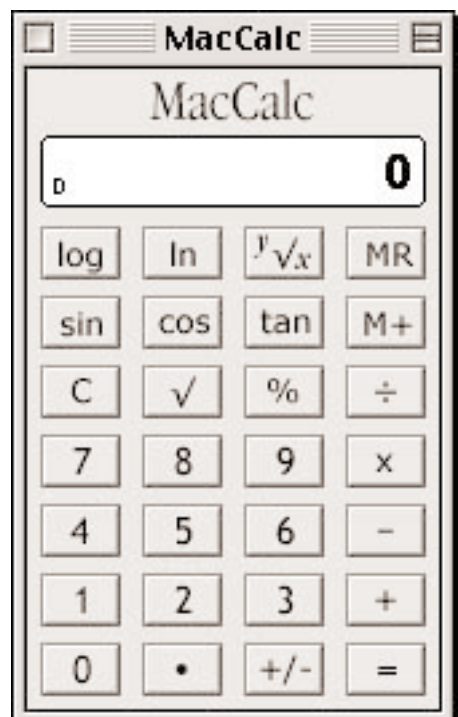
Classic platinum MacCalc
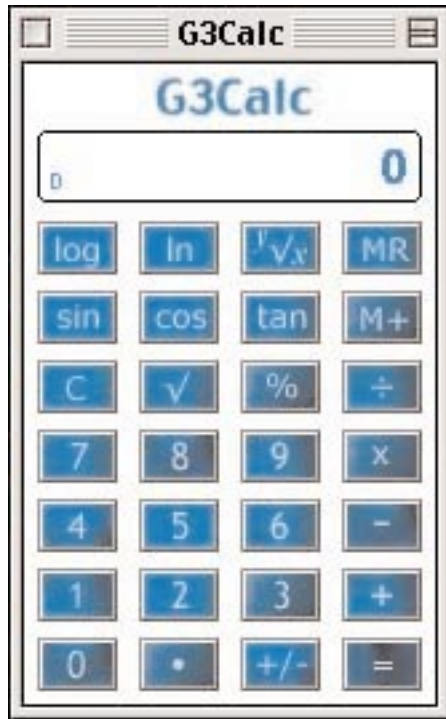
G3Calc

G4Calc
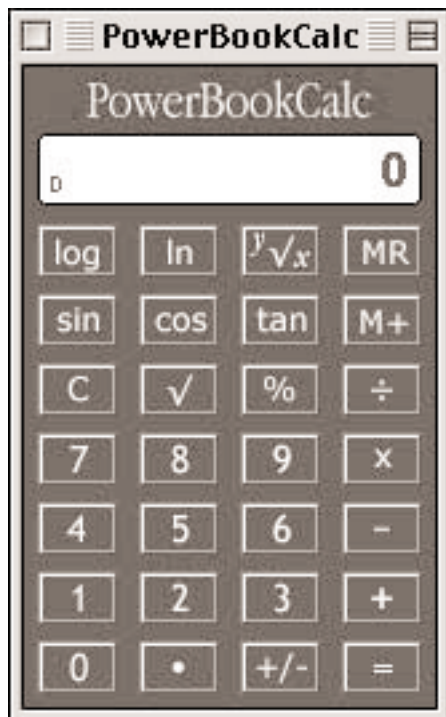
PowerBookCalc

BronzeCalc

Blueberry iBookCalc

Tangerine iBookCalc

Graphite iBookCalc

Technical Considerations

There are various technical considerations that should be borne in mind when using a computer to perform mathematical calculations. Most of the following has been gleaned from Apple's *Inside Macintosh* series of books, available via Apple's web site or in print form from Addison-Wesley.

Apple Macintosh computers that use the PowerPC chip (601, 603, 603e, 604, 604e and 750 [G3] microprocessors) use PowerPC Numerics as a floating point environment for mathematical computation. Earlier Apple computers that use a 68k chip (68000, 68020, 68030, 68040 and LC68040 microprocessors) use the earlier SANE (Standard Apple Numerics Environment) for mathematical computation. There are differences between the two numerics systems.

Any computer will also have certain numerical limitations when it comes to mathematical computation, which is a result of the way computers use binary to represent and handle numbers and computations.

How computers handle maths

PowerPC Numerics is a floating-point environment that complies with **IEEE Standard 754**. There are two IEEE standards for floating-point arithmetic: IEEE Standard 754 for binary floating-point arithmetic and IEEE Standard 854 for radix-independent floating-point arithmetic.

The IEEE standards ensure that computers represent real numbers as accurately as possible and that computers perform arithmetic on real numbers as accurately as possible. Although there are infinitely many real numbers, a computer can represent only a finite number of them. Computers represent real numbers

as binary floating-point numbers. Binary floating-point numbers can represent real numbers exactly in relatively few cases; in all other cases the representation is approximate.

For example, 1/2 (0.5 in decimal) can be represented exactly in binary as 0.1. Other real numbers that can be represented exactly in decimal have repeating digits in binary and hence cannot be represented exactly, as shown below. For example, 1/10, or decimal 0.1 exactly, is 0.000110011 . . . in binary.

Errors of this kind are unavoidable in any computer approximation of real numbers. Because of these errors, sums of fractions are often slightly incorrect. For example, 4/3 – 5/6 is not exactly equal to 1/2 on any computer, even on computers that use IEEE standard arithmetic.

Fraction	Decimal approximation*	Binary approximation†
1/10	0.1000000000‡	0.0001100110011001100110011001101
1/2	0.5000000000‡	0.10000000000000000000000000‡
4/3	1.333333333	1.0101010101010101010101011
5/6	0.8333333333	0.110101010101010101010101
4/3 – 5/6	0.4999999997	0.10000000000000000000000001

* 10 significant digits † 23 significant digits ‡ Exact value

The IEEE standard defines data formats for floating-point numbers, shows how to interpret these formats, and specifies how to perform operations (known as floating-point operations) on numbers in these formats. It requires the following types of floating-point operations:
■ basic arithmetic operations (add, subtract, multiply, divide, square root, remainder, and round-to-integer)
■ conversion operations, which convert numbers to and from the floating-point data formats
■ comparison operations, such as less than, greater than, and equal to
■ environmental control operations, which manipulate the floating-point environment

The IEEE standard requires that the basic arithmetic operations have the following attributes:
■ The result must be accurate in the precision in which the operation is performed. When a numerics environment is performing a floating-point operation, it calculates the result to a predetermined number of binary digits. This number of digits is called the precision. The result must be correct to the last binary digit.
■ If the result cannot be represented exactly in the destination data format, it must be changed to the closest value that can be represented, using rounding.

■ If an invalid input is provided or if the result cannot be represented exactly, a floating-point exception must be raised.

Careful Rounding

If the result of an IEEE arithmetic operation cannot be represented exactly in binary format, the number is rounded. IEEE arithmetic normally rounds results to the nearest value that can be represented in the chosen data format. The difference between the exact result and the represented result is the roundoff error.

The IEEE standard requires that users be able to choose to round in directions other than to the nearest value. For example, sometimes you might want to know that rounding has not invalidated a computation. One way to do that would be to force the rounding direction so that you can be sure your results are higher (or lower) than the exact answer.

Because it conforms to the IEEE standard, PowerPC Numerics gives you a means of doing that. Fully developed, this strategy is called *interval arithmetic* (Kahan 1980).The following example is a simple demonstration of the advantages of careful rounding.

Suppose your application performs operations that are mutually inverse; that is, operations $y=f(x)$, $x=g(y)$, such that $g(f(x))=x$. There are many such operations, such as
$y=x^2$, $x=\sqrt{y}$
$y=375x$, $x=y\div375$

Suppose $F(x)$ is the computed value of $f(x)$, and $G(y)$ is the computed value of $g(y)$. Because many numbers cannot be represented exactly in binary, the computed values $F(x)$ and $G(y)$ will often differ from $f(x)$ and $g(y)$. Even so, if both functions are continuous and well behaved, and if you always round $F(x)$ and $G(y)$ to the nearest value, you might expect your computer arithmetic to return x when it performs the cycle of inverse operations, $G(F(x))=x$.

It is difficult to predict when this relation will hold for computer numbers. Experience with other computers says it is too much to expect, but IEEE arithmetic very often returns the correct inverse value.

The reason for IEEE arithmetic's good behaviour with respect to inverse operations is that it rounds so carefully. Even with all operations in, say, single precision, it evaluates the expression 3x1/3 to 1.0 exactly; some computers that do not follow the standard do not evaluate this expression exactly.

Exception Handling

The IEEE standard defines five exceptions that indicate when an exceptional event has occurred. They are
■ invalid operation
■ underflow
■ overflow
■ division by zero
■ inexact result

There are three ways an application can deal with exceptions:
■ Continue operation.
■ Stop on exceptions if the programmer thinks they will invalidate results.
■ Include code to do something special when exceptions happen.

The IEEE standard lets programs deal with the exceptions in reasonable ways. It defines the special values NaN (Not-a-Number) and Infinity, which allow a program to continue operation. The IEEE standard also defines exception flags, which a program can test to detect exceptional events.

About the FPCE Technical Report

Even though many computers now conform to the IEEE standard, the standard has suffered from a lack of high-level portability. The reason is that the standard does not define bindings to high-level languages; it only defines a programming environment.

For instance, the standard defines data formats that should be supported but does not tell how these data formats should map to variable types in high-level languages. It also specifies that the user must be able to control rounding direction but falls short of defining how the user is able to do so.

However, the definition of a binding is in progress for the C programming language. The Floating-Point C Extensions (FPCE) branch of the Numerical C Extensions Group (NCEG), or **ANSI X3J11.1**, has proposed a general floating-point specification for the C programming language, called the **FPCE technical report**, that contains additional specifications for implementations that comply with IEEE floating-point standards 754 and 854.

The FPCE technical report not only specifies how to implement the requirements of the IEEE standards, but also requires some additional functions, called **transcendental functions** (sometimes called *elementary functions*). These functions are consistent with the IEEE standard and can be used as building

34

blocks in numerical functions. The transcendental functions include the usual logarithmic and exponential functions, as well as $\ln(1+x)$ and e^x-1; financial functions for compound interest and annuity calculations; trigonometric functions; error and gamma functions; and a random number generator.

PowerPC Numerics Versus SANE

Although PowerPC Numerics is an implementation of the IEEE Standard, it is not the Standard Apple Numerics Environment (SANE). SANE is the numerics environment used on 680x0-based Macintosh computers, and it is the numerics environment used when you run a 680x0 application on a PowerPC processor-based Macintosh computer.

PowerPC Numerics is the environment used when you run an application built for a PowerPC processor-based Macintosh computer.

There are fundamental differences between PowerPC Numerics and SANE because of the differences in the microprocessors on which the two environments are used. The major difference is that SANE supports an 80-bit extended type and performs all floating-point computations in extended precision. This protects the user from roundoff error, overflows, and underflows that might occur in an intermediate value when determining the result of an expression.

Because the PowerPC processor is double-based, support of an 80-bit data type would be inefficient. It instead supports a 128-bit type (in software) called *double-double* (which corresponds to the *long double* type in C).

PowerPC Numerics provides this wide type only for cases where precision greater than that provided by the double format is necessary; PowerPC Numerics does not perform all computations in double-double precision. Instead, PowerPC Numerics recommends a method by which an expression is evaluated in the widest precision necessary.

Another fundamental difference is that PowerPC Numerics conforms to the FPCE recommendations as well as to the IEEE standard. C implementations using SANE do not necessarily comply with the FPCE recommendations.

About Floating-Point Data Formats

The IEEE standard defines several floating-point data formats, one required and the others recommended. IEEE requires that

35

each data format have a **sign bit** (s), an **exponent** field (e), and a **fraction** field (f). For each format, it lists requirements for the minimum lengths of these fields. For example, the standard describes a 32-bit single format whose exponent field must be 8 bits long and whose fraction field must be 23 bits long.

The only required data format is the 32-bit single format. A 64-bit double format is strongly recommended. The IEEE standard also describes two data formats called *single-extended* and *double-extended* and recommends that floating-point environments provide the extended format corresponding to the widest basic format (single or double) they support.

To conform to the IEEE requirements on floating-point data formats, the PowerPC Numerics environment provides three data formats: single (32 bits), double (64 bits), and double-double (128 bits).

The single and double formats are implemented exactly as described in the standard. The double-double format is provided in place of the recommended double-extended format. IEEE requires that the double-extended format be at least 79 bits long with at least a 15-bit exponent. The double-double format is 128 bits long and has an 11-bit exponent.

The double-double format is just what its name sounds like: two double-format numbers combined. The PowerPC assembly-language multiply-add instructions, which multiply two double-format numbers and add a third with at most one roundoff error, make implementing the double-double format much more efficient than implementing a true IEEE double-extended format.

The IEEE standard also makes requirements about how the values in these data formats are interpreted. PowerPC Numerics follows these requirements exactly.

iMalc uses doubles for all calculations.

Interpreting Floating-Point Values

Regardless of which data format (single, double, or double-double) you use, the numerics environment uses the same basic method to interpret which floating-point value the data format represents.

Every floating-point data format has a sign bit, an exponent field, and a fraction field. These three fields provide binary encodings of a sign (+ or −), an exponent, and a **significand**, respectively, of a floating-point value. The value is interpreted as
$\pm$ significand x $2^{\text{exponent-bias}}$

where

±	is the sign stored in the sign bit (1 is negative, 0 is positive).
significand	has the form $b_0.b_1b_2b_3\ldots b_{precision-1}$ where $b_1b_2b_3\ldots b_{precision-1}$ are the bits in the fraction field and b_0 is an implicit bit whose value is interpreted as described in the sections "Normalized Numbers" and "Denormalized Numbers" below. The significand is sometimes called the *mantissa*.
exponent	is the value of the exponent field.
bias	is the bias of the exponent. The **bias** is a predefined value (127 for single format, 1023 for double and double-double formats) that is added to the exponent when it is stored in the exponent field. When the floating-point number is evaluated, the bias is subtracted to return the correct exponent. The minimum biased exponent field (all 0's) and maximum biased exponent field (all 1's) are assigned special floating-point values.

In a numeric data format, each valid representation belongs to exactly one of these classes, which are described in the sections that follow:

- normalized numbers
- denormalized numbers
- Infinities
- NaNs
- zeroes

Normalized Numbers

The numeric data formats represent most floating-point numbers as **normalized numbers**, meaning that the implicit leading bit, b_0, of the significand is 1. Normalization maximizes the resolution of the data type and ensures that representations are unique.

Using only normalized representations creates a gap around the value 0. If a computer supports only the normalized numbers, it must round all tiny values to 0. For example, suppose such a computer must perform the operation $x\text{-}y$, where x and y are very close to, but not equal to, each other. If the difference between x and y is smaller than the smallest normalized number, the computer must deliver 0 as the result. Thus, for such **flush-to-zero** systems, the following statement is *not* true for all real numbers:

$x\text{-}y{=}0$ if and only if $x{=}y$

Denormalized Numbers

Instead of using only normalized numbers and allowing this small gap around 0, PowerPC processor-based Macintosh computers use denormalized numbers, in which the leading implicit bit, b_0, of the significand is 0 and the minimum exponent is used.

The use of denormalized numbers makes the following statement true for all real numbers:
$x\text{-}y=0$ if and only if $x=y$

Another advantage of denormalized numbers is that error analysis involving small values is much easier without the gap around zero (Demmel 1984). The computer determines that a floating-point number is denormalized (and therefore that its implicit leading bit is interpreted as 0) when the biased exponent field is filled with 0's and the fraction field is nonzero.

Infinities

An Infinity is a special bit pattern that can arise in one of two ways:
■ When an operation (such as 1/0) should produce a mathematical infinity, the result is an Infinity.
■ When an operation attempts to produce a number with a magnitude too great for the number's intended floating-point data type, the result might be a value with the largest possible magnitude or it might be an Infinity (depending on the current rounding direction).

These bit patterns (as well as NaNs, introduced next) are recognized in subsequent operations and produce predictable results. The Infinities, one positive and one negative, generally behave as suggested by the theory of limits. For example:
■ Adding 1 to $+\infty$ yields $+\infty$
■ Dividing -1 by +0 yields $-\infty$
■ Dividing 1 by $-\infty$ yields -0

The computer determines that a floating-point number is an Infinity if its exponent field is filled with 1's and its fraction field is filled with 0's. So, for example, in single format, if the sign bit is 1, the exponent field is 255 (which is the maximum biased exponent for the single format), and the fraction field is 0, the floating-point number represented is $-\infty$.

NaNs

When a numeric operation cannot produce a meaningful result, the operation delivers a special bit pattern called a **NaN (Not-a-Number)**. For example, zero divided by zero, $+\infty$ added to $-\infty$, and $\sqrt{-1}$ yield NaNs. A NaN can occur in any of the numeric data formats (single, double, and double-double), but generally, system-specific integer types (non-numeric types exclusively for integer values) have no representation for NaNs.

A NaN may have an associated code that indicates its origin.

NaN	Meaning
1	Invalid square root, such as $\sqrt{-1}$
2	Invalid addition, such as $(+\infty) + (-\infty)$
4	Invalid division, such as $0 \div 0$
8	Invalid multiplication, such as $0 \times \infty$
9	Invalid remainder or modulo, such as x rem 0
17	Attempt to convert invalid ASCII string
21	Attempt to create a NaN with a zero code
33	Invalid argument to trigonometric function (such as cos, sin, tan)
34	Invalid argument to inverse trigonometric function (such as acos, asin, atan)
36	Invalid argument to logarithmic function (such as log, $\log_{10}$)
37	Invalid argument to exponential function (such as exp, expm1)
38	Invalid argument to financial function (compound or annuity)
40	Invalid argument to inverse hyperbolic function (such as acosh, asinh)
42	Invalid argument to gamma function (gamma or lgamma)

Zeroes

Each floating-point format has two representations for zero: $+0$ and -0. Although the two zeros compare as equal $(+0)=(-0)$, their behaviours in IEEE arithmetic are slightly different.

Ordinarily, the sign of zero does not matter except (possibly) for a function discontinuous at zero. Though the two forms are numerically equal, a program can distinguish $+0$ from -0 by operations such as division by zero or by performing the numeric copysign function.

The sign of zero obeys the usual sign laws for multiplication and division. For example, $(+0) \times (-1) = -0$ and $1/(-0) = -\infty$. Because

extreme negative underflows yield -0, expressions like $1/x^3$ produce the correct sign for ∞ when x is tiny and negative.

Addition and subtraction produce -0 only in these cases:
(-0) - (+0) yields -0
(-0) + (-0) yields -0
When rounding downward, with x finite,
x-x yields -0
x+(-x) yields -0
The square root of -0 is -0.

The sign of zero is important in complex arithmetic (Kahan 1987).

iMalc will convert any display of -0 or +0 to read 0.

Ranges

The following are the ranges for numbers and functions allowed by iMalc.

Range of numbers

Maximum positive	$1.8 \times 10^{+308}$
Minimum positive norm	2.2×10^{-308}
Minimum positive denorm	4.9×10^{-324}
Maximum negative denorm	-4.9×10^{-324}
Maximum negative norm	-2.2×10^{-308}
Minimum negative	$-1.8 \times 10^{+308}$

Ranges of functions

Operation	Result
$e^x(+0)$	+1
$e^x(-0)$	+1
$e^x(\text{NaN})$	NaN
$e^x(+\infty)$	+∞
$e^x(-\infty)$	+0
ln(x) for $x<0$	NaN
ln(+1)	+0

ln(+0)	-∞
ln(-0)	-∞
ln(NaN)	NaN
ln(+∞)	+∞
ln(-∞)	NaN
$\log_{10}(x)$ for $x<0$	NaN
$\log_{10}(+1)$	+0
$\log_{10}(+0)$	-∞
$\log_{10}(-0)$	-∞
$\log_{10}(NaN)$	NaN
$\log_{10}(+\infty)$	+∞
$\log_{10}(-\infty)$	NaN
$\sin(\pi)$	0
sin(+0)	+0
sin(-0)	-0
sin(NaN)	NaN
sin(+∞)	NaN
sin(-∞)	NaN
$\cos(\pi)$	-1
cos(+0)	1
cos(−0)	1
cos(NaN)	NaN
cos(+∞)	NaN
cos(-∞)	NaN
$\tan(\pi)$	0
$\tan(\pi/2)$ ie tan90°	+∞ Note: iMalc will return 1.633124e+16
tan(+0)	+0
tan(-0)	-0
tan(NaN)	NaN
tan(+∞)	NaN
tan(-∞)	NaN
asin(x) for $|x|>1$	NaN
asin(-1)	$-\pi/2$
asin(+1)	$\pi/2$
asin(+0)	+0
asin(-0)	-0
asin(NaN)	NaN
asin(+∞)	NaN
asin(-∞)	NaN
acos(x) for $|x|>1$	NaN
acos(-1)	π
acos(+1)	+0
acos(+0)	$\pi/2$
acos(-0)	$\pi/2$
acos(NaN)	NaN
acos(+∞)	NaN
acos(-∞)	NaN
atan(+0)	+0
atan(-0)	-0

```
atan(NaN)          NaN
atan(+∞)           π/2
atan(-∞)           -π/2
```

Error codes

You may occasionally see some error codes whilst using iMalc. You will get the warning 'imaginary' if you try to take the square root of a negative number. You will also see NAN, often followed by a number in brackets, or INF.

INF means infinity, and can be obtained through erroneous operators such as division by zero.

NANs are Not A Number and are other error codes, generally for numbers which are out of range, such as trying to take the arcsine of 100. The numbers in brackets sometimes following NANs are specific error codes and are listed on page 37.

Trigonometry units

Most computers calculate trig functions such as sine, cosine, tangent and their inverses in radians. iMalc will calculate the correct function in whatever units are shown in the trig mode indicator.

Degrees	Radians	Gradients
360°	2π	400
180°	π	200
90°	$\pi/2$	100
45°	$\pi/4$	50

Version History

0.1.0b1   First beta release, 12th September 1999.

1.0.0   First release, 6th October 1999.
- Added flavour for G4. Slightly changed flavour menu shortcuts.
- Colour change to BronzeCalc to better match Lombard keyboards.
- Fixed bug with display showing "Not numerical" with "0."
- Moved buttons slightly and made small changes on reminder screen.
- Fixed bug where Help and Contact windows could draw partly offscreen. These now open centred on the screen.
- Changed $\log_{10}$ calculation to eliminate need of PPC only plug-in.
- Minor code changes to menu items for future Windows version.
- Numbers pressed no longer add to end of memory recall.
- Added names of beta testers in Credits window.
- 68k and fat binary Macintosh versions now available.
- Minor documentation changes.

1.1.0   Second release, 1st September 2000.
- Graphite iMac DV SE flavour added.
- Graphite iBook flavour added.
- iMalc now also accepts and displays commas as decimal separators, according to the settings in the Numbers control panel
- email and web site addresses in windows now draggable.
- Added web site and email clippings to documentation.
- Added option to register online in registration box.
- New improved iBook, G3 and G4 flavours.
- Better match for graphite colour. Previously the graphite colour was taken from the Apple website, web colour being inaccurate.
- email and web links now turn red when the mouse is down.
- Keyboard shortcuts changed to accommodate new flavours.
- Documentation changes. Registering information now in separate Read Me.
- 0.00 display bug fixed. Multiple zeroes following a decimal separator would not show properly.
- Auto text resize display for font not being Charcoal. Fixes a problem where the display would not show properly if a font with larger font metrics than Charcoal was used as the System font.
- Drag now works! Glove cursor appears when over the display which is highlighted with a border colour matching the chosen flavour.
- Help Box text now correctly starts at the top when choosing another topic.
- Added What's New and System Requirements to Help.

1.1.1   Third release, 12th October 2000.
- Leading zero input/display bug fixed.

1.1.2   Fourth release, 15th October 2000.
- Trailing zero input/display bug fixed.

Proposed future enhancements

- Key for input of exponents
- Additional memories
- Support for new flavours Sage, Indigo, Ruby and Snow
- Hyperbolic functions
- Key for factorial
- Base conversions

Where to get the latest version

We hope to keep the latest version of iMalc posted to the Info-Mac HyperArchive, which is searchable at:
http://hyperarchive.lcs.mit.edu/HyperArchive.html

We also hope to be able to provide a list of other local servers where you can get the latest versions for download on our web site, to try and reduce the load on our ISP.

You can also obtain the latest version at
http://homepage.mac.com/eyecatcherprods/iMalc.html
http://eyecatcherproduction.tripod.com/iMalc.html
http://www.geocities.com/eyecatcherproductions/iMalc.html
http://eyecatcherproductions.nav.to/iMalc.html
http://www.tspencer.dircon.co.uk/eyecatcher/iMalc.html

If you are using Apple's iTools, you can access the iMalc iDisk at:
http://homepage.mac.com/imalc or use the iDisk user name:
imalc.

Registration

This section covers registration issues, such as the shareware concept, pricing, how to register and methods of payment. **You must register iMalc and pay the shareware fee if you intend to continue using iMalc.**

Shareware

iMalc is shareware. So what is shareware? Firstly, shareware is *NOT* free software. You have to pay for it like everything else. Shareware is a method of software distribution that allows software authors to share their endeavours with you in a cheap and easy way.

Shareware is effectively try before you buy. This means that you can try the software free for a trial period before deciding whether you want to keep on using the software. A period of 30 days is reasonable. After 20 uses, iMalc will start reminding you to register. After the trial period, if you decide to keep on using iMalc you must register and pay the shareware fee, or you must delete iMalc and its associated files from your hard drive.

Shareware also means you can share a program with a friend, so long as you give them all of the files that accompany iMalc. They, of course, like you, have to pay a shareware fee if they continue to use the software.

Most shareware authors don't earn a fortune. But the fees they do receive provide an incentive to keep producing and updating software that might otherwise be unavailable. Shareware relies on your honesty to succeed.

Pricing

The following software licenses are available:

Single user licenses:

No of copies	Price
1-9	US$15 (£10) per copy
10-19	US$12 per copy
20+	US$10 per copy

Site license:
US$300 (roughly equal to 30 users)

A Site License covers all locations for your organization within a 160 kilometre radius of your site (100 miles). The big advantage of a Site License is that you do not need to keep track of how many people at your site are using the software.

World-Wide License:
US$1,000 (roughly equal to 100 users)

A World-Wide license covers all locations for your organization on the planet earth. Other planets and satellites, such as the moon, require a separate World-Wide License. ;-)   For licensing purposes, anything in orbit around a planet is considered part of the planet. However if you intend using iMalc in orbit, you should contact Eyecatcher as the author will need to be on the mission to check that the software is functioning correctly in weightless conditions. No kidding.

If you are an impoverished student, school, college, university or other academic institution then the following rates apply:

Academic Single user licenses:

No of copies	Price
1-9	US$10 per copy
10-19	US$8 per copy
20+	US$5 per copy

Academic Site license:
US$200 (roughly equal to 30 users)

Payment covers all future versions of iMalc.

For details on how to register and pay see the next two sections.

Registering

Registering is very easy.

You can register and pay online, by email, fax or post.

To handle registrations we use the Kagi Software registration service. The big advantage for you and us is if you have a credit card you can register immediately and being in a different country is no problem no matter how you decide to pay.

You can pay using a credit card (Visa, MasterCard, American Express, Diners Club, Discover or Optima), by cash in a variety of currencies (including €), by US$ cheque or money order, using First Virtual, or if you are a company, by invoice.

Registering Online

To register online with a credit card:
• open your favourite browser
    — You will need to supply your credit card details so you may want to use the secure connection:
    http://order.kagi.com/?UZE&S
    — If that doesn't work or you have an older browser that doesn't support secure connections try the non-secure connection:
    http://order.kagi.com/?UZE
• type in all of the characters including the ? and &
• follow the instructions on the web page.

Registering by email, fax or post

To register by email, fax or post you will need to use the **Register iMalc** program that came with iMalc. See the next section for details on how to use Register iMalc for your preferred payment method.

At the present time you must either register online or via Kagi with the Register iMalc program. Eyecatcher cannot deal with registrations directly (the author is currently travelling around the world if you must know the reason - therefore all fees help!). At a future date direct registration with Eyecatcher may be possible.

Payment Instructions

Kagi handles payments from customers around the world and forwards payments on to software authors around the world. Kagi represents many thousands of software authors and has designed an automated system to keep registrations as speedy and fuss-free as possible.

If you wish to register online with a credit card see left for details.

Otherwise, whatever method you choose to pay by, you will need to use the Register iMalc program that came with iMalc. If you no longer have this, you will need to download the program again. Kagi cannot accept any payments that do not use the Register iMalc program. This is because their fee collection and payment system is highly automated to reduce costs and speed up the registration process.

Using Register iMalc is also very easy. You may want to keep these instructions open onscreen while you fill in your registration details or print them out for later referral.

Using Register iMalc

• Open Register iMalc

• Enter your name and your email address in the corresponding boxes (these may already be filled in from System details, but check them). If you have no email address enter "none".

• Optionally enter your postal address (this is required if you do not have an email address). Please remember that Kagi has no way of knowing which country you come from so please include this.
> ** Note: You must provide an email address or a postal address so Kagi can contact you with confirmation of your registration. Please note that Kagi do not reply by fax. If you do not have an e-mail address, you must check the *Postcard Receipt* box (which automatically adds $1 to your charge) at the bottom right of the window. Otherwise, you will not receive any reply to your registration although you will be registered. After the initial postcard receipt Kagi only replies via email.

• Then, fill in the payment details at the bottom left of the window. Choose your method of payment from the pop up menu

and complete the relevant information (credit card number, type of currency, etc.) below. Make checks payable to "Kagi". Checks must be in US Dollars only. Checks in other currencies are not accepted, solely because the bank processing costs involved make it impractical. Please write the program name on the check as well as your email address, if you have one.

> ** Note: Your credit card data is scrambled. Although this is not completely secure, Kagi have never had a case of credit card fraud.
> ** Note: Purchase Orders must include payment with the invoice.

• Next, indicate the programs you want to register in the middle of the window. Type the number of copies you want to register into the *Single User* box. In most cases you should just type "1". Do not type the purchase price into this box since the Register iMalc application computes the fee for you. The *Site* and *World-Wide* check boxes are intended for companies or organisations that want a license to use the software on all their machines. Double check this before you send in the form as you may be charged significantly more that you expect if you make a mistake.

• If you feel that the program is worth more than the registration fee, you can add an optional bonus in the *Bonus US$* box.

Once you have filled in all this required information, the Register program displays the total amount at the bottom right of the window. Double check the total!

• Choose one of the four buttons.

> **Print** – prints a hard copy of the registration form, which you can mail or fax to Kagi with your payment to the address shown on the form. If you have a fax modem you can also print the form to this via the menu and your normal method of faxing.

> **Copy** – puts a copy of the registration form on your clipboard for you to paste into an e-mail message to send to Kagi. This method works only with credit cards and First Virtual since it is not yet possible to email cash or checks.

> **Save** – saves a copy of the registration form to your hard disk as a SimpleText document for you to print out or e-mail later.

> **Quit** – quits the iMalc register program without doing anything.

Double check your registration request to see that all the information is correct, if not, correct it using the Register program. Do not edit the output produced by the Register program. The Kagi email and postal mail addresses are given on the registration form and above. Save a copy for yourself. We suggest printing it out. Note Eyecatcher's email address.

Payment by credit card or First Virtual

Kagi accepts Visa, MasterCard, American Express, Diners Club, Discover or Optima. If paying with Credit Card or First Virtual, you can email or fax the data to Kagi. Their email address and their fax number are on the form. You can either Copy the data from Register iMalc and paste into the body of an email message or you can Save the data to a file and you can attach that file to an email message. There is no need to compress the data file, it's already pretty small. If you have a fax modem, just Print the data to the Kagi fax number. Alternatively you can print the form and send it by post.

Kagi's email address is: sales@kagi.com

Kagi's fax number is: +1 (510) 652-6589

Payments sent via email are processed within 3 to 4 days. You will receive an email acknowledgement when it is processed. Payments sent via fax take up to 10 days and if you provide a correct internet email address you will receive an email acknowledgement.

Payment by cash or US$ check/money order

If you are paying with cash or US$ check/money order you should print the data using the Register iMalc application and send it to the address shown on the form.

You can pay with a wide variety of cash from different countries but at present if you pay via check, it must be a check drawn in US Dollars. Kagi cannot accept checks in other currencies, the conversion rate for non-US$ checks is around US$15 per check and that is just not practical.

If you have a purchasing department, you can enter all the data into the Register program and then select Invoice as your payment method. Print three copies of the form and send it to your accounts payable people. You might want to highlight the line that mentions that they must include a copy of the form with

their payment. Kagi cannot invoice your company, you need to act on our behalf and generate the invoice and handle all the paperwork on your end.

Please do not fax or email payment forms that indicate cash, check or invoice as the payment method. As far as we know, there is still no technology to transfer physical objects via fax or email and without the payment, the form cannot be processed.

Kagi's address is: Kagi
1442-A Walnut Street #392-UZE
Berkeley, CA 94709-1405
USA

Payments send via postal mail take time to reach Kagi and then up to 10 days for processing. Again, if you include a correct email address, you will hear from Kagi when the form is processed.

After you have registered

After you register, Kagi automatically sends you an e-mail reply titled *Thanks for your registration*. You should print and save this message as your receipt. If you checked the *Postcard Receipt* box in the Register program, Kagi will send you a postcard with this information. Again, if you do not have an e-mail address, you must request a *Postcard Receipt* since otherwise you will not receive any reply.

It can take Kagi up to a week to process a registration from the date they receive it (please make sure to factor in the delivery time if you use postal mail). If you do not receive the *Thanks for your registration* e-mail message, contact Kagi at <support@kagi.com> or the postal address given on the registration form to ask about the status of your registration. See * Kagi need this data: below.

Keep in mind that some registrations can get lost in the mail on the way to Kagi. Please make sure to seal the envelope and write your email address on the check. Also, sometimes the e-mail reply gets lost as well. If you do not receive an email confirmation of your payment, contact Kagi with the "* Kagi need this data:" from below and ask what happened.

* Kagi need this data:

Anything that you can provide Kagi with would be helpful, such as when and how you sent your payment and any previous email

messages sent. Please tell Kagi exactly what you want them to do/send to you.

Payment-method:	Card , Check and check #, cash invoice or First Virtual
Register-to:	Registrant's name
Time/Invoice:	if original had this number
Email:	email address
Address:	If used
Card-name:	card holder's name
Credit-card#:	first 10 digits
Program:	program name & cost

Direct your registration queries to the following. Note that Kagi only deals with registrations of software and not technical support.

Kagi
1442-A Walnut Street #392-UZE
Berkeley, CA 94709-1405
USA

registration:	sales@kagi.com
	send your registration and registration related questions here.
support:	support@kagi.com
	use this address only after you have registered.
internet:	http://www.kagi.com
	read here for more information.
FAQ via email:	FAQ@kagi.com
	send a message and receive the current FAQ as a reply.
Fax:	+1 (510) 652-6589

Technical Support

If you have any technical support questions, problems or comments about iMalc, please direct them to the author at: eyecatcherprods@mac.com

Or visit our FAQ at:
http://homepage.mac.com/eyecatcherprods/techsupport.html

About Eyecatcher

iMalc is the first software available from Eyecatcher Productions.

Eyecatcher Productions is the trading name used by Tony Spencer.

Tony has been using Apple Macintoshes for over ten years in the publishing field. He has been a freelance production editor for eight years, and a freelance photographer for three years.

He is the author of iMalc and is from London in the United Kingdom, and has recently returned from 19 months travelling around the world.

Having done some programming several years ago, Tony is now using REALbasic from Real Software to produce Macintosh applications.

Contacting Eyecatcher

If you have any technical support questions, problems or comments about iMalc, or Eyecatcher Productions, please email us at: eyecatcherprods@mac.com

Our web site, which contains news and information on iMalc and other Eyecatcher products, is at:
http://homepage.mac.com/eyecatcherprods/

Mirror sites are also available at:
http://eyecatcherproduction.tripod.com
http://www.geocities.com/eyecatcherproductions/
http://eyecatcherproductions.nav.to
http://www.tspencer.dircon.co.uk/eyecatcher/

Legal Information

The following is legal information which concerns copyright and your software license which you should read carefully.

Copyright

Trademarks

Machines Corporation. REALbasic is a trademark of Real Software, Inc. Adobe, the Adobe logo, Acrobat, Distiller and the Acrobat logo are trademarks of Adobe Systems Incorporated. QuarkXPress is a trademark of Quark, Inc. Microsoft, Word, Windows 95 and Trebuchet MS are registered trademarks of Microsoft, Inc. Zapf Dingbats is a trademark of International Type Corporation. All other product names are trademarks of their respective holders. Mention of non-Eyecatcher products is for information purposes only and constitutes neither an endorsement nor a recommendation.

Software License

Every effort has been made to ensure that the information in the iMalc software, this Read Me file and the manual is accurate. Eyecatcher Productions is not responsible for typographical errors.

This is a legal agreement between you and Eyecatcher Productions covering your use of iMalc (the "Software"). Be sure to read the following agreement before using the Software.

BY USING THE SOFTWARE (REGARDLESS OF WHETHER YOU HAVE REGISTERED THE SOFTWARE OR NOT), YOU ARE AGREEING TO BE BOUND BY THE TERMS OF THIS AGREEMENT. IF YOU DO NOT AGREE TO THE TERMS OF THIS AGREEMENT, DO NOT USE THE SOFTWARE AND DESTROY ALL COPIES IN YOUR POSSESSION.

iMalc is distributed via the shareware system. You are permitted to use the Software on a trial basis for up to 30 days. If you wish to continue using the Software beyond that period, you must register by paying a fee of $15.00 in U.S. or equivalent currency or remove it from your computer storage medium. Academic, Site and World-Wide licenses are also available.

The Software is owned by Eyecatcher Productions and is protected by United Kingdom copyright laws and international treaty provisions. Therefore, you must treat the Software like any other copyrighted material (e.g., a book or musical recording). Paying the license fee allows you the right to use one copy of the Software on a single desktop computer, and one copy on a portable computer. You may not network the Software or

55

otherwise use it or make it available for use on more than one computer at the same time except on a portable as a second computer.

You may not rent or lease the Software, nor may you modify, adapt, translate, reverse engineer, decompile, or disassemble the Software. If you violate any part of this agreement, your right to use this Software terminates automatically and you must then destroy all copies of the Software in your possession.

Permission is granted for electronic non-commercial distribution of unmodified copies of the Software, so long as this Read Me file, the iMalc Manual and the Register iMalc program is included. This includes CD-ROMS on magazine covers and those supplied by User Groups, posting on the internet and electronic bulletin boards.

For permission about including this Software on commercial compilation disks, CD-ROMs or bundle options, please contact Eyecatcher Productions.

You acknowledge that the Software may not satisfy your requirements or be free from defects.

EYECATCHER PRODUCTIONS MAKES NO WARRANTY OR REPRESENTATION, EITHER EXPRESS OR IMPLIED, WITH RESPECT TO THIS READ ME FILE, THE MANUAL, THE DATA, OR THE SOFTWARE, ITS QUALITY, ACCURACY, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. AS A RESULT, THIS READ ME, MANUAL AND SOFTWARE IS PROVIDED AS SHAREWARE "AS IS," AND YOU, THE PURCHASER, ARE ASSUMING THE ENTIRE RISK AS TO ITS QUALITY AND ACCURACY.

IN NO EVENT WILL EYECATCHER PRODUCTIONS BE LIABLE FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES RESULTING FROM ANY DEFECT OR INACCURACY IN THIS READ ME, MANUAL OR SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. THE WARRANTY AND REMEDIES SET FORTH ABOVE ARE EXCLUSIVE AND IN LIEU OF ALL OTHERS, ORAL OR WRITTEN, EXPRESS OR IMPLIED. IN NO EVENT SHALL EYECATCHER PRODUCTIONS' LIABILITY EXCEED THE LICENSE FEE PAID, IF ANY.

No Eyecatcher Productions dealer, agent, or employee is authorized to make any modification, extension, or addition to this warranty.

This Agreement shall be governed by the laws of England and Wales. If for any reason a court of competent jurisdiction finds any provision of the Agreement, or portion thereof, to be

unenforceable, or contrary to law, that provision of the Agreement shall be enforced to the maximum extent permissible so as to effect the intent of the parties, and the remainder of this Agreement shall continue in full force and effect.

57

Other Information

Here is a selection of other useful information and web sites.

Bibliography

Inside Macintosh reference books from Apple, published by Addison-Wesley. In particular, *Overview*, *Human Interface Guidelines*, *Memory*, *PowerPC Numerics* and *Macintosh Toolbox Essentials*.

American National Standards Institute. *Floating-Point C Extensions*, prepared by the Floating-Point C Extensions (FPCE) branch of the Numerical C Extensions Group. ANSI X3J11.1/93-028, 1993.

Institute of Electrical and Electronics Engineers. *IEEE Standard for Binary Floating-Point Arithmetic*. IEEE Standard 754-1985. New York: IEEE, 1985.

Kahan, W. "Interval Arithmetic Options in the Proposed IEEE Floating-Point Arithmetic Standard." In *Interval Mathematics* 1980, edited by K. E. L. Nickel. New York: Academic Press, 1980.

Kahan, W. "Rational Arithmetic in Floating-Point." Berkeley, CA: Report No. PAM-343, Center for Pure and Applied Mathematics, University of California, 1986a.

Motorola Corporation. *PowerPC 601 RISC Microprocessor User's Manual*, Motorola Corporation, 1993.

Credits

Thanks are due to many people.

Firstly to those who have paid their registration fees - hopefully that's you! Thank you for supporting the shareware concept.

Secondly to Andrew Barry for writing such a great program as REALbasic, the development language in which iMalc is written.

Thanks to Geoff Perlman, Joe Strout, Dave Grogono, Lorin Rivers and the rest of the team at Real Software Inc, http://www.realsoftware.com, for continuing to develop such an awesome RAD IDE, and all those on the REALbasic NUG for their help and tips.

Thanks also to Matt Neuberg, http://www.tidbits.com/matt, who wrote *REALbasic The Definitive Guide* (published by O'Reilly, ISBN 1-56592-657-9 http://www.ora.com/catalog/realbasic/index.html) for his many helpful tips and examples and explanations of the workings of REALbasic.

Love to Jane for making the PowerBook happen.

Also to Steve Jobs, Steve Wozniak, Jeff Raskin, Bill Atkinson and the rest of the guys on the Macintosh development team and at Apple who made such an insanely great machine in the first place and continue to make them.

Gratitude to the beta testers, Andy Pugh, Ric Davis, Dave Wakeman and Steve Loween - who bought iMalc whilst still in beta - superstar! Also to those who pointed out bugs, particularly John Blommers.

Last, to all the developers out there who have made such wonderful software.

And, if you've read this far all the way, good on yer!

Remember: Think Different. Calculate Different.

Tony Spencer
Updated 15th October 2000
1.1.2

Useful contacts

Adobe   http://www.adobe.com

Apple Computer   http://www.apple.com

    Apple Masters - famous people who use Macs

    http://www.apple.com/applemasters/home.html

    AppleStore

    http://www.apple.com/store/

    AppleStore UK

    http://www.apple.com/ukstore/

    Developer Documentation

    http://developer.apple.com/techpubs/macos8/mac8.html

    iTools

    http://itools.mac.com/itoolsmain.html

    Macintosh Product Guide - over 17,000 Mac products

    http://guide.apple.com/

    Manuals

    http://support.info.apple.com/manuals/manuals.qry

AppleDesigns   http://www.appledesigns.com

Apple-Donuts   http://www.apple-donuts.com

Applefritter   http://www.applefritter.com

Apple History   http://www.apple-history.com

AppleLinks   http://www.applelinks.com

Apple Watches   http://www.applewatch.co.uk

Complete Conflict Compendium

    http://www.mac-conflicts.com

Daily iMac   http://www.dailyimac.com

everythingmac.com http://www.everythingmac.com

Eyecatcher   http://www.tspencer.dircon.co.uk/eyecatcher/

                       eyecatcher@kagi.com

Favourite Mathematical Constants

    http://www.mathsoft.com/asolve/constant/constant.html

Fundamental Physical Constants from NIST

  http://physics.nist.gov/cuu/Constants/index.html?/codata86.html

Go2Mac   http://www.go2mac.com

History of Mathematics

    http://www-groups.dcs.st-and.ac.uk/~history/

iMac2day   http://www.imac2day.com

iMacWorld   http://www.imacworld.com

Insanely Great   http://www.insanely-great.com

insanely iMac!   http://members.xoom.com/insanelyiMac/

Kagi   http://www.kagi.com

London Macintosh User Group   http://www.lmug.org.uk

Low End Mac   http://lowendmac.com

Mac4Sale.com   http://www.mac4sale.com

MacAddict   http://www.macaddict.com

MacDirectory   http://www.macdirectory.com

Mac Driver Museum   http://members.xoom.com/macdrivers/
MacEvangeList   http://www.MacEvangeList.com
MacFever   http://www.macfever.com
MacFormat   http://www.macformat.co.uk
MacGeeks.com   http://www.macgeeks.com
MacHyperArchive
      http://hyperarchive.lcs.mit.edu/HyperArchive.html
Macinsites   http://macinsites.com
Macinstein - Mac search engine   http://www.macinstein.com
Macintosh OS.com   http://www.macintoshos.com
Macintouch   http://www.macintouch.com
MacMad   http://www.macmad.org
MacMarines   http://www.macmarines.com
Mac Observer   http://www.macobserver.com
MacOS Web Directory   http://www.macwebdir.com
MacOSZone.com   http://www.macoszone.com
MacPicks   http://www.macpicks.com
Mac Rocks - music   http://www.macrocks.com
Mac Rumors   http://www.macrumors.com
Macsites   http://www.macsites.com
MacSoldiers   http://www.macsoldiers.com
MacSurfShop   http://www.macsurfshop.com
MacTimes   http://www.mactimes.com
Mac Today   http://www.mactoday.com
MacToolbox   http://www.mactoolbox.com
MacTreasures   http://www.mactreasures.com
MacUpdate   http://www.macupdate.com
MacUser UK   http://www.macuser.co.uk
MacWindows - integrating   http://www.macwindows.com
MacWorld UK   http://www.macworld.co.uk
MacWorld US   http://www.macworld.com
MacWorld iMac   http://macworld.zdnet.com/subject/imac/
Making the Macintosh   http://library.stanford.edu/mac/
Mathematical Atlas   http://www.math-atlas.org/welcome.html
MathPages   http://www.seanet.com/~ksbrown/
Microsoft MacTopia   http://www.microsoft.com/mac/
Motorola   http://www.motorola.com
NoBeige.com   http://www.go2mac.com/nobeige/
Real Software   http://www.realsoftware.com
Quark   http://www.quark.com
Secure Mac   http://www.SecureMac.com
Steve Wozniak   http://www.woz.org
The Apple Collection http://www.theapplecollection.com
TheiMac.com   http://www.theimac.com
TidBits - Mac news   http://www.tidbits.com
Version Tracker   http://www.versiontracker.com
Woz's MacLinks   http://www.woz.org/maclinks/index.html
Yahoo! Mathematics   http://dir.yahoo.com/Science/Mathematics/

Index

63