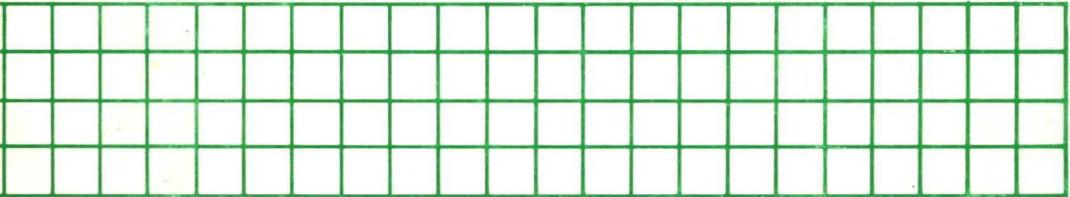
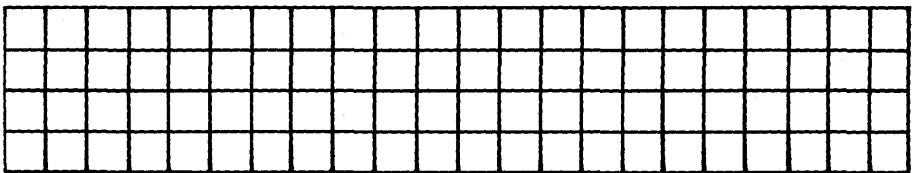


T.M.

Perfect Calc



Perfect Calc^{T.M.}



Perfect Calc

by Robert B. Wesson, Ph.D.

To my wife Nancy, who convinced me the microcomputer revolution is real.

COPYRIGHT

Copyright, 1983 by Perfect Software, Inc. All rights reserved worldwide. No part of this publication may be reproduced, transmitted, transcribed, stored in a retrieval system or translated into any human or computer language in any form or by any means, electronic, mechanical, magnetic, optical, chemical, manual or otherwise, without the express written permission of Perfect Software, Inc., 1400 Shattuck Avenue, Berkeley, California 94709.

DISCLAIMER OF WARRANTY

Perfect Software, Inc. makes no representations or warranties, either express or implied, with respect to this manual and accompanying software and specifically disclaims any implied warranties of merchantability or fitness for any particular purpose. This manual and accompanying software are sold "as is" and Perfect Software will in no event be liable for direct, indirect, incidental or consequential damages resulting from any defect, error or failure to perform.

TRADEMARK

Perfect WriterTM, Perfect SpellerTM, Perfect MailerTM, Perfect SortTM, Perfect TerminalTM, Perfect MessengerTM, Perfect CalcTM, Perfect LedgerTM, Perfect SoftwareTM, and the PerfectTM prefix are trademarks of Perfect Software, Inc.

Documentation: Howard H. Wade

PERFECT CALC USER'S GUIDE

I.	Overview	1
PART I FUNDAMENTALS OF AN ELECTRONIC SPREADSHEET		
II.	Getting Acquainted	13
III.	Beginning and Quitting Perfect Calc	35
IV.	Moving About the Spreadsheet	41
V.	Entering Data: Labels and Numbers	69
VI.	Entering Formulas	97
VII.	Inserting and Deleting	135
VIII.	Moving, Copying and Restoring Deletions	167
IX.	Setting Up the Spreadsheet As You Like It	187
X.	Storing and Printing Your Spreadsheet	205
PART II ADVANCED MULTIPLE SPREADSHEET ANALYSIS		
XI.	Working With Multiple Spreadsheets	219
XII.	Multiple File Display	239
XIII.	Associated Files	263
PART III SPREADSHEET ANALYSIS PROGRAMS		
XIV.	Personal and Home Use Programs	283
	The Family Budget Program	283
	The Financial Net Worth Program	286
	The Check Register Program	288
	The Individual Tax Return Analysis Programs	289

PART III, continued

XV.	Small Business Applications	295
	The Income Statement Program	295
	Cost of Goods Statement	297
	Sales Expense Statement	298
	General and Administrative Expenses Statement	298
	The Cashflow Assessment Program	299
	The Accounts Receivable Worksheet	300
	The Accounts Payable Worksheet	302
	The Payroll Analysis Program	304
XVI.	Spectrum of Applications	307
	The Stock Portfolio Evaluation Program	307
	The Real Estate Analysis Program	308
	The Chi-square Analysis Program	309
	The Professional Fee Analysis Program	310

APPENDICES

A.	Installing Perfect Calc	313
B.	Structure of Perfect Calc Data Files	323
C.	Tips on Using Perfect Calc Efficiently	325
D.	Command Summary	331
E.	Glossary	341
F.	Index	349

Chapter I OVERVIEW

Welcome to Perfect Calc! The powerful new electronic spreadsheet from Perfect Software Inc., which will change your world of financial and scientific computing forever. Perfect Calc brings advances and innovations that make electronic spreadsheet calculation easier, faster, more convenient than ever before. Not since Perfect Writer revolutionized word processing has such power and versatility been brought to the world of microcomputers.

Why is Perfect Calc the most powerful of its type?

Using the sophistication of virtual memory architecture and multiple file buffers, Perfect Calc has revolutionized the electronic spreadsheet. With Perfect Calc you can have several different spreadsheets in memory at the same time and have them all working together and sharing data. This means, for example, that you can post data from expense reports, accounts payable, and accounts receivable to your income statement and ledger files. You can compute your tax by accessing separate tax tables automatically. Plus, you can display two separate spreadsheet files on the screen at the same time, allowing you to perform a "what if?" analysis while continuing to see "what is!"

Other electronic spreadsheet programs differ from each other primarily in the number of functions they provide, such as 'net present value', 'square root', and various statistical and mathematical formulas. Perfect Calc introduces an entirely new dimension to the use of functions by providing a built-in 'functions library' which allows more than twice the number of functions provided by any other spreadsheet program. What is even more important is that Perfect Calc's 'functions library' is 'user extendable', meaning that if you have equations, financial ratios or other formulas that you use often, you can simply add them to your library. This feature makes Perfect Calc capable of supporting an almost unlimited number of functions.

With all this sophistication, is Perfect Calc complicated?

At Perfect Software we believe the 'power' of the computer lies in its ability to make your life simpler and easier. Although Perfect Calc is the most powerful spreadsheet program available, it is also the easiest to learn. If you have already used Perfect Writer, then you know how to use most of Perfect Calc because the commands of the two programs mirror each other. But even if this were not so, the command structure of Perfect Calc is in itself very clear and simple. Once you start using it you'll be surprised at how quickly and efficiently you'll be able to apply it in a wide variety of ways.

What is a Spreadsheet?

Rare is the person today who has not used a spreadsheet in one form or another. If you have ever kept a household budget or balanced a checkbook, you have used a spreadsheet. The monthly bills you receive from the phone and utility companies are nothing more than simple spreadsheets listing in an orderly fashion the calls you have made and the electricity you have used.

A spreadsheet is just that: a time-tested way of organizing numeric data for easy calculation and comprehension. Financial records and scientific calculations that might otherwise be a jumbled mess of scribblings on scratch paper, can be neatly arranged and labeled on a spreadsheet, enabling them to be comprehended at a glance.

In its pure form a spreadsheet is a rectangular grid of lines and columns, like a large sheet of graph paper, onto which data can be entered. Until recently spreadsheets were made only of paper, and although they worked well enough, they were awkward and tedious to use. You entered the data with a pencil and performed the calculations mentally, or with the aid of a hand calculator. Any change in the data required a laborious manual recalculation of major portions of the spreadsheet. As well, you had to remember the relationships that existed between the columns and rows of figures: what must be added to or subtracted from what, and when and where and how. And despite what anyone could do there was always the risk of error in performing the arithmetic, not to mention a slip of the pen in writing down the answer—all of which required checking and rechecking.

Perfect Calc's Electronic Spreadsheet

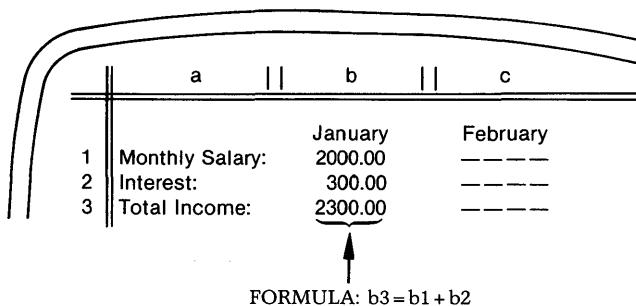
With Perfect Calc all of this has changed. There is no need for pencils, paper, or the hand operated calculator, no need for endless manual recalculations, or for having to remember the sometimes intricate relationships among the spreadsheet entries. Gone is the risk of calculation error. Gone is the drudgery of the whole business!

Perfect Calc creates, inside your computer, an 'electronic spreadsheet', a large matrix of labeled lines and columns, resembling the paper spreadsheets you are familiar with. Your computer screen, in turn, functions like a 'window', capable of moving about and viewing whatever portion of this spreadsheet that you wish.

	a	b	c	d	e	h	i	j	k	l	m	X	Y	Z	
1	ohlu	sen	i	s mih	i snh	ml o	h	selu	isnlt	ol o	shin	rochu	sh in	aj	
2	hlmn	c	oir	chn	l	scor	miles	oc	lo n	anh	mac	nsel	ml	ci	
3	nesn	lho	e	oanh	moel	n snc	ise	ohlu	en nk	min	nsuh	nsel	in	ci	
4	seln	ensir	o	ol	ihm	rochu	les	hlmn	omls	in lo	scor	intz	sh in	aj	
5	lo n	coml	almoc	nlsc	nsch	nsel	ben	nesn	io ol	nhur	lnoc	in snc	in	ci	
6	ohlu	sen	i	s mih	i snh	ml o	hu	selu	isnlt	ol c	shin	rochu	up	abc	
7	hlmn	c	oir	chn	l	scor	miles	oc	lo n	anh	mac	nsel	ml	ci	
8	nesn	h	e	oanh	moel	n snc	ise	ohlu	en nk	anh	nsuh	nsel	in	ci	
23	hlmn	: om	chn	l	scor	oc	inc	sl	selu	isnlt	ol c	shin	rochu	oh	
24	nesn	ho	e	oanh	moel	n snc	miles	15	ohlu	en nk	anh	nsuh	nsel	in	ci
25	seln	ensim	o	ol	hm	rochu	snol	ise	hlmn	omls	in le	scor	intz	sh in	aj
hlmn	lo n	coml	almoc	nlsc	nsch	nsel	hlmn	nesn	io ol	nhur	lnoc	in snc	in	ci	
ochu	ohlu	sen	i	s mih	i snh	ml o	o	selu	isnlt	sh in	nsuh	nsel	in	ci	
nsel	hlmn	c	oir	chn	l	scor	miles	nscc	ohlu	en nk	anh	mac	ml	ci	
253	ohlu	sen	i	s mih	i snh	ml o	ns	in	h	ohlu	en nk	nhu	o	ns	
254	hlmn	: om	nl	om	les	ho	nsel	nscc	hlmn	omls	hn	le	scor	mls	
255	nesn	ho	e	anh	nsel	no	les	nscc	nesn	ho	ol	nhu	lnoc	in	

On the spreadsheet itself the intersections of the lines and columns form many thousands of tiny 'boxes'—entry positions—which can hold not only labels and numbers, but also formulas that define the relationships existing between the various data that you enter.

For example, suppose that you are doing a simple budget analysis of your household. One box might hold the figure representing your monthly salary, while a second box might hold your earnings from other sources: stocks, bonds, savings accounts. A third box might hold a formula that computes the sum of these two, your 'Total Income'.



The real power of an electronic spreadsheet comes from its ability to 'remember' the figures and formulas that you enter, automatically recalculating the formula values whenever data in any box is changed. For example, should you get a raise to \$2,500/month and enter this figure into the 'Monthly Salary' box, the electronic spreadsheet automatically recalculates your Total Income according to the formula you have previously supplied.

This recalculation capability is sometimes called the "What if?" capability, since it allows you to postulate changes in the data contained in your spreadsheet and to see the results of those changes immediately.

With Perfect Calc you can easily analyze and manage the most sophisticated aspects of your business and financial affairs, including cash flow, expenses, depreciation, taxes, accounts receivable, sales, inventory, and much more. Perfect Calc can be used to manage stock and bond portfolios, calculate promissory notes and loan schedules, and prepare income statements and balance sheets. It can be used to project economic and financial 'models'—essential tools in the long-range planning of business or other financial ventures.

But Perfect Calc is not limited to business and financial applications alone. It has numerous personal and scientific applications as well. In fact, its use is limited only by the range of your imagination!

THE FEATURES PERFECT CALC BRINGS TO YOU

You won't be using Perfect Calc long before you will be celebrating the numerous features that make it the unquestioned leader among today's electronic spreadsheet systems:

Simplicity of use

Perfect Calc is tailored especially for users of Perfect Writer, the powerful word processing system. If you already use Perfect Writer, you already know most of what is in this User's Guide, since Perfect Calc's command structure is identical in most respects to that of Perfect Writer. But if not, you will be amazed at how quickly and easily you will be able to understand and to use Perfect Calc. As in all pieces of Perfect Software, the key is **simplicity!**

Versatile Command structure

Perfect Calc provides a logical, flexible, and efficient set of commands capable of performing any task you may wish to do—inserting, deleting, copying, formatting, storing, printing, moving about the spreadsheet. 'Mnemonically' assigned (the first letter of the command matches the command key), the commands are quick to learn and easy to remember.

Multiple Spreadsheet Buffers and Virtual Memory

Like Perfect Writer, Perfect Calc offers the outstanding technological advances of Multiple Spreadsheet Buffers, and Virtual Memory Architecture.

With Perfect Calc, you may have as many as **seven** different spreadsheets active in machine memory at once, switching back and forth between them with ease. Unlike other spreadsheet programs, this feature allows you to actually share data between spreadsheets, with formulas referencing variables in spreadsheets other than their own. This means you can have spreadsheets transfer (access or post) data to other spreadsheets. For example, data from an 'Aging Report' can be posted to an accounts receivable spreadsheet; later the accounts receivable data can be posted to a trial balance statement—all automatically.

Multiple Spreadsheet Display

Through Perfect Calc's Multiple File Display capability, two different spreadsheets may be viewed simultaneously, a feature not offered by any other major electronic spreadsheet program. This means you can ask a "what if?" while still seeing "what is." For instance, if you want to see the impact on your company's income should sales increase by 10 percent, you could view the original income statement on the right side of the screen and the hypothetical "what if?" statement on the left side of the screen. You could also compare your company's 1981 and 1982 budgets on the same screen, with the 1981 budget in the top window and the 1982 budget in the bottom window. You could even compute a projected 1983 budget based on the increase from 1981 to 1982 and display it compared to the current budget.

User Extendable Functions Library

One of the major differences between Perfect Calc and other electronic spreadsheet programs is that Perfect Calc allows more than twice the number of functions as do other programs. It does this using a 'functions library'—a file which holds nothing but functions, everything from intricate financial formulas to simple statistical indicators. You don't need complex programming to perform routine tasks; simply call in the functions library and select the function you need.

The functions library is 'extendable', meaning that you can add your own specialized functions and formulas, or even modify those provided by Perfect Calc to meet your particular needs.

Regional Recalculation

Users familiar with other electronic spreadsheets will be delighted to know that with Perfect Calc, 'recalculation' of formulas can be limited to specific 'regions' of the spreadsheet. No longer must you wait for the entire spreadsheet to be recalculated when data changes affect only a small portion of that spreadsheet.

Spreadsheet Analysis Programs

To illustrate the many potential uses of Perfect Calc, a number of pre-structured Spreadsheet Analysis Programs have been provided, ranging from such simple but necessary activities as balancing a personal checkbook to such sophisticated procedures as managing a business payroll account. The programs, of course, are simply models, intended for illustrative purposes only. Before using them in any serious capacity, you should check the appropriateness of their formulas and calculations to insure that they are correctly suited for the purposes you intend.

Locking Formulas, Replication, Variable Formats, etc.

All of the advanced spreadsheet features that sophisticated users have come to expect are available in Perfect Calc, such as logical formula components, variable format displays, individually-variable column widths, complete report generation facilities, formula replication and locking, and much more!

Spreadsheet editing with Perfect Writer

Perfect Calc is fully integrated with the Perfect Writer word processor. In addition to having a common command structure, Perfect Calc spreadsheets can be transferred directly to Perfect Writer for further editing and inclusion in written documents and reports. This provides greater freedom and flexibility for both programs. You can have all the math and financial functions you need for Perfect Writer and all the editing and formatting functions you need for Perfect Calc!

True ASCII Data Files

Perfect Calc stores data either as a screen image or as a data file for subsequent analysis. In contrast to other electronic spreadsheet programs, Perfect Calc does not use an 'encoding scheme' to store data; rather, it uses the 'American Standard Code for Information Interchange' (ASCII), meaning that you can transfer Perfect Calc files between computer systems for use with other programs, with only slight modifications.

ORGANIZATION OF THE USER'S GUIDE

For your easy understanding and reference, this User's Guide is presented in the following parts:

PART I. Using an electronic spreadsheet

This section presents the basic information about Perfect Calc needed to begin using the system. Included are the basic command procedures for beginning and quitting the system, moving the cursor, entering data (labels, numbers, and formulas), inserting and deleting lines and columns, and storing and printing the spreadsheet. Numerous illustrations and examples are provided to help explain the material.

PART II. Multiple spreadsheet analysis and display

Presented here are those more advanced command options that you will want to begin using when you have become familiar with Perfect Calc's basic mode of operation. These include: handling multiple spreadsheets, multiple file display, and moving and copying data.

PART III. The spreadsheet analysis programs

This section contains descriptions of the model Spreadsheet Analysis Programs. Three groups of applications are provided: Personal and Home Use Applications, Small Business Applications, and a 'Spectrum' of Applications.

Appendices

The Appendices contain a number of useful miscellaneous discussions which you may find helpful in working with Perfect Calc, including technical explanations of Perfect Calc's data files, the operation of Perfect Calc's unique system of 'overlays', hints on managing data storage, as well as a comprehensive index and glossary to help you quickly find and understand whatever you want to know about Perfect Calc.

Tutorials

Step-by-step tutorials are provided at the end of various chapters, to help you practice the command procedures previously discussed. As well, all command procedures are presented singly, their component steps clearly structured, leaving no doubt about when or why or how to do something. Referencing the manual is made simple and convenient.

Installing Perfect Calc

Complete installation instructions, which **must** be completed before you can begin using Perfect Calc, are presented in Appendix A.

Part I

FUNDAMENTALS OF AN ELECTRONIC SPREADSHEET

In This Section

- Beginning and Quitting
- Moving the Cursor
- Labels and Numbers
- Functions, Variables and Formulas
- Inserting and Deleting
- Moving, Copying and Replicating
- Customizing the Spreadsheet
- Saving and Printing

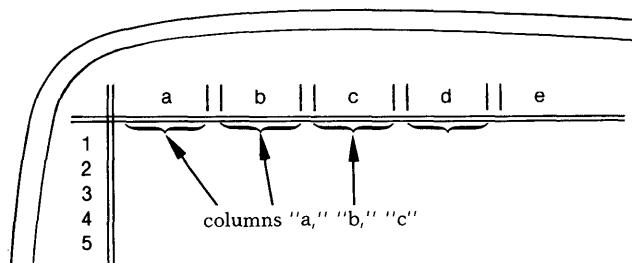
Chapter II GETTING ACQUAINTED

This chapter introduces Perfect Calc's basic operating components: the spreadsheet, the window, the cursor, the Prompt Line. Also included are explanations on the Help Menu, filenames, and the mechanics of operating Perfect Calc's command keys. Even if you have had experience with other spreadsheet calculating systems it will be good to read through this chapter before continuing.

The Spreadsheet

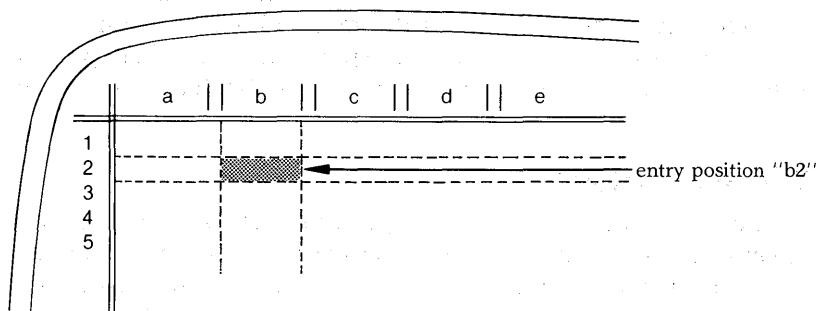
As pointed out in the Overview, the Perfect Calc spreadsheet is an electronic tablet of lines and columns, much larger than your computer screen. The entire spreadsheet contains a total of 255 lines and 52 columns. Lines are numbered from 1 to 255, while the columns are labeled using letters of the alphabet. The first 26 columns are labeled 'a' to 'z', while the last 26 are labeled 'A' to 'Z'.

Besides being designated on the screen by letters of the alphabet, columns are marked by pairs of vertical bars across the top of the screen. Columns may vary in width, containing a maximum of 76 characters each, although Perfect Calc initially sizes them identically at 9 characters each.



Although no grid of lines is actually displayed on the screen, the lines and columns can be thought of as creating an invisible matrix of 'boxes' into which labels, numbers, and formulas can be entered.

Each entry box is identified by a pair of 'coordinates' which indicates the particular line and column that intersect to form the entry box. For example, the shaded box in the illustration below is entry position 'b2'.



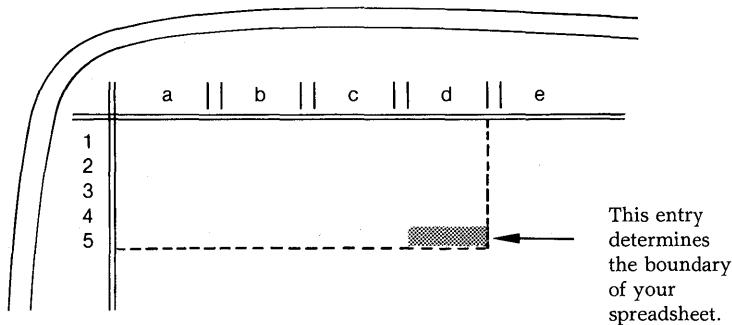
Coordinates always begin with the 'letter' of the column, followed by the 'number' of the line. For example:

'b2' . . . (not '2b')

'M23' . . . (not '23M').

Spreadsheet Boundaries

The boundaries of any spreadsheet that you create are determined by the lowest, most right-hand entry. For example:



Despite irregularities in the placing of entries, Perfect Calc will normally consider your spreadsheet as a rectangle with evenly drawn sides.

The Window

The screen of your computer is like a 'window' through which you view the spreadsheet being held in machine memory. Using Perfect Calc's cursor and screen commands (see Chapter IV), the window can be moved about to view whatever portion of the spreadsheet that you wish.

Depending upon the characteristics of your computer, a single window will display roughly 23 lines of a spreadsheet, and a varying number of columns (the wider the columns, the fewer displayed).

When the screen is divided in half during 'multiple file display' (see Part Two), 'two windows' are said to exist, positioned either one atop the other or side by side. The windows are designated the 'current window', meaning the window which holds the cursor, and the 'other window'.

The Cursor

The cursor marks the position on the spreadsheet where data is being entered. The cursor will be displayed differently depending, again, on the characteristics of your computer screen. Normally it will appear in 'inverse video' to the rest of your screen. That is, if the screen is normally black (or green), with the characters highlighted in white, the cursor will display as a white box, while the numbers or characters within it will appear black (or green). On some screens the cursor will appear as a pair of facing 'angle brackets', < >, enclosing the box it occupies. It may also appear as an 'underline'.

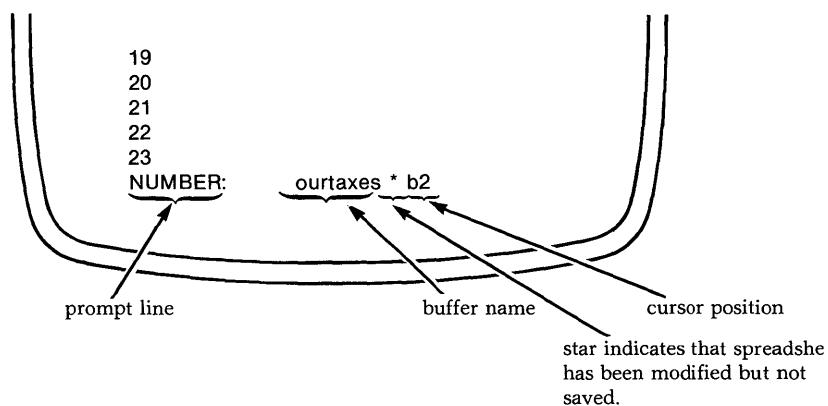
The Prompt Line

The Prompt Line is a partially blank line located at the bottom of the screen which is used for, among other things, displaying data prior to its being entered onto the spreadsheet. When you type a number, formula, or a label (some word used to identify data on the screen), it will appear initially in the Prompt Line, and will not become part of the spreadsheet until you 'enter' it by typing either a command key, or a carriage return, <CR>.

In addition, Perfect Calc uses the Prompt Line to display system messages and questions whenever further information or instructions are needed to complete a command procedure.

On the right side of the Prompt Line, Perfect Calc will constantly display three pieces of information:

- The **name** of the 'spreadsheet buffer' currently being used.
- A **star**, or asterisk, (*) which, when present, indicates that the spreadsheet currently being displayed has not yet been 'saved' on disk since the last change was made to it. (If no changes have been made to a spreadsheet, this star will not appear.)
- The **location coordinates** of the entry box which the cursor is currently occupying.



Spreadsheet File Names

Each of the spreadsheets that you create will have a distinctive filename which you will provide, either at the time you create the spreadsheet, or when you 'save' it to disk (see Chapter X). This is done so you will be able to later identify and retrieve it from disk.

Your operating system requires that filenames follow a particular form. They cannot be longer than eight (8) characters, and cannot contain any of the following:

< > . , ; = ? * _

In addition, Perfect Calc filenames can contain the optional suffix '.pc', as in:

ourtaxes.pc

Optional Perfect Calc suffix

When creating filenames for your spreadsheets, it is helpful to be as descriptive of the contents as possible. For example, use names like checkbk.pc, payroll.pc, and invntory.pc—not names like spread1.pc, spread2.pc, spread3.pc, etc. We recommend that you develop the habit of using the '.pc' suffix with your Perfect Calc data files.

Default Spreadsheet File Name

Should you neglect to supply a filename, Perfect Calc will store your spreadsheet in the file 'default.pc'.

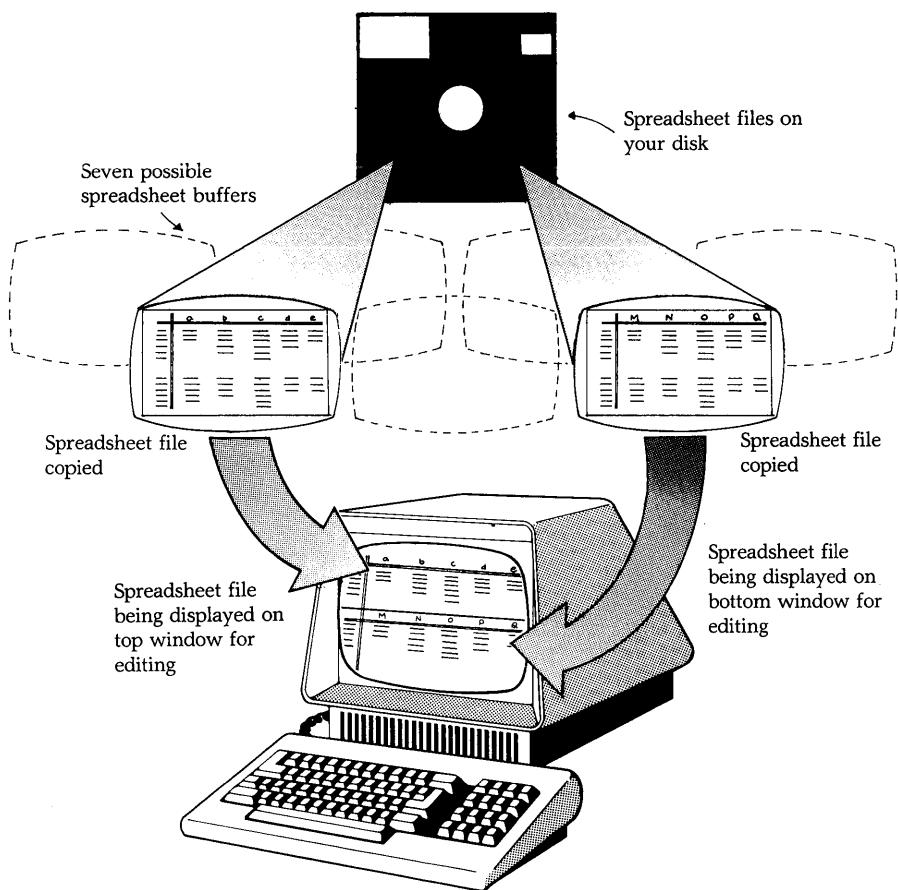
Important: The file default.pc is a temporary 'backup' file in which Perfect Calc stores unnamed spreadsheets. Each time it stores a spreadsheet in default.pc, **it overwrites whatever may have been stored there previously**. Therefore, don't risk losing your spreadsheets by not giving them proper filenames!

Buffers

When you want to work with a spreadsheet you have previously stored as a file, Perfect Calc retrieves a copy from disk storage and places it in a temporary workspace in machine memory, called a 'spreadsheet buffer'.

Such a procedure is convenient because changes made to the copy in the buffer do not affect the original version of your spreadsheet, which remains safely stored on disk. Thus, you can ultimately decide against keeping whatever changes you make and know that your original version is unaltered.

Perfect Calc allows up to **seven** spreadsheets to be held at one time in separate editing buffers. As you shall see, the use of multiple buffers allows easy comparison, exchange, and referencing of data between spreadsheets. Also, through the use of multiple buffers it is possible to split the screen into two windows, and display a different spreadsheet in each window, as in the following illustration.



It is important to understand that buffers are distinct and separate entities, and although it is possible to create 'associated files' which are linked together, command procedures executed in one buffer are normally restricted to that buffer alone, having no effect on spreadsheets held in other active buffers.

Buffer Names

Every buffer temporarily assumes a name when in use. This name is displayed in the lower right corner of the screen, together with the coordinates that mark the present position of the cursor in the spreadsheet.

The buffer name is derived from the name of the last file **from** which a spreadsheet has been copied. Specifically, it consists of the first eight characters of the filename, disregarding both disk drive designation, and suffix. For example:

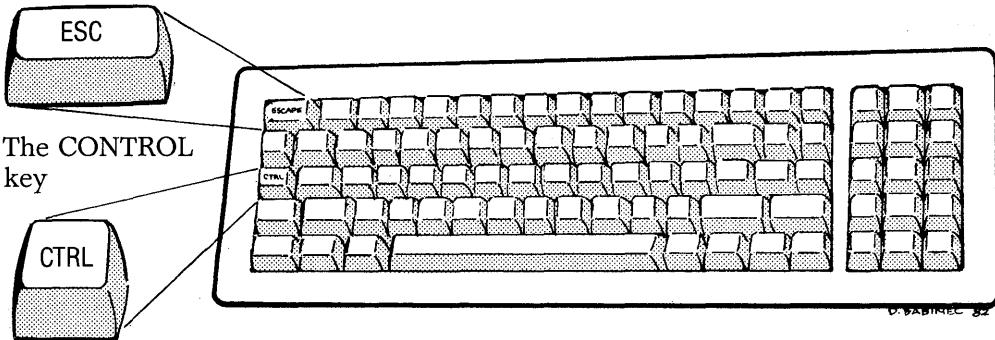
B:OURTAXES.pc ourtaxes * J33
 filename buffer name

If the spreadsheet is new and still bears the default filename 'DEFAULT.PC' (see page 19), the buffer is simply designated the 'default' buffer. Should the spreadsheet be stored using the WRITE FILE command (see Chapter X, page 207), or should another spreadsheet be read into this default buffer using the READ FILE command (see Chapter XI, page 224), the filename will change to reflect the new filename. The buffer will remain named the 'default' buffer, however.

USING COMMAND KEYS

Perfect Calc uses two keys in combination with other characters to produce command codes that will move the cursor, delete, copy, store, print, or otherwise manipulate the data on your spreadsheet.¹ These are:

The ESCAPE key



The CONTROL key

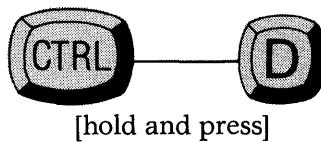
CTRL

¹ The emphasis in this discussion will be upon the MECHANICS of typing commands, and therefore it will not be necessary to remember the specific commands given as examples. All commands will be discussed individually and in detail in the subsequent sections of this manual.

The CONTROL Key

The Control key functions somewhat like a 'shift key' on an ordinary typewriter. Depressing and releasing it by itself produces no effect. However, holding the Control key while typing another character results in a command recognized by Perfect Calc.

For example, holding the Control key and typing the letter 'd' tells Perfect Calc to delete an entry.



Note: Throughout this manual a **solid** line connecting two keys will mean that **both** keys are depressed at the **same time**.)

The ESCAPE Key

Like the Control key, the Escape key is also used in conjunction with other characters to produce commands. However, unlike the Control key, the Escape key should not be held. It has only to be depressed and then released. When the Escape key is typed, Perfect Calc will interpret whatever character that follows as the concluding half of a command. To alert you that this command procedure has been set in motion, Perfect Calc displays the word 'Escape:' in the Prompt Line:



'Escape:' indicates that the Escape key has been typed and that the next character will be interpreted as a command.² (If you type the next character quickly enough, the word Escape will not appear.)

Example: One Escape command is the OPEN COLUMN command, which instructs Perfect Calc to insert a column at the cursor.



(Simply type the Escape key, followed by the lowercase letter 'o')

Note: Throughout this manual a **dotted** line connecting two keys will mean that the two keys should not be depressed at the same time, but rather sequentially in the order listed.

² Perfect Calc uses this echoing technique in other contexts as well to reduce screen clutter for experienced users.

Control vs. Escape

An essential relationship which is orthogonal exists between the Control and Escape keys which aids in understanding and learning them. Specifically, the Control key in conjunction with a letter governs those commands which affect 'entries' and 'lines', while the Escape key followed by the same letter will cause a similar action on 'columns' and 'regions'. See the table below. Both Control and Escape are used for command procedures that are executed frequently.

The Relationship between Control and Escape Command

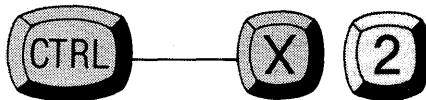
Character	Control Command	Escape Command
d	Control----D DELETE ENTRY	Escape.....D DELETE REGION
w	Control----W COPY ENTRY	Escape.....W COPY REGION
o	Control----O INSERT (open) LINE	Escape.....O INSERT (open) COLUMN
y	Control----y YANKBACK ENTRY	Escape.....y YANKBACK ACROSS REGION

The 'Control----x' Prefix

The Control key is used to create what is known as an 'extended command'. When the Control key is depressed together with the letter 'x', an effect similar to that produced by the Escape key is created. However, instead of 'Escape', the word 'Extended' appears in the Prompt Line, indicating that Perfect Calc will interpret whatever character or characters that follow as an extended command.

Extended commands commonly deal with some larger aspect of the editing process, such as formatting, window, file, and buffer manipulation. These commands are usually performed far less frequently than simple Control and Escape commands. Two examples are:

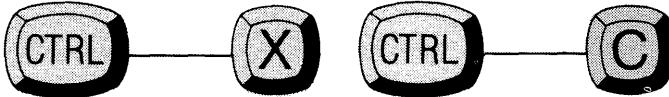
1. The CREATE TWO WINDOWS command: the Control----X prefix is followed by a second character, '2', which is typed by itself.



(While holding the Control key, depress 'X'; after releasing these, depress '2'.)

Extended control commands which are followed by a single character generally deal with screen manipulation and spreadsheet formatting operations, such as: the OTHER WINDOW command, CHANGE COLUMN WIDTH, JUSTIFY ENTRY, LOCK TITLES, etc.

2. The QUIT command in which the Control----X prefix is followed by a 'second' Control command.



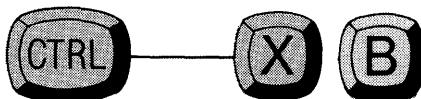
(While holding the Control key, depress lowercase 'x'; then, while **continuing to hold** the Control Key, depress lowercase 'c'.)

Extended Control commands of this form are used in operations that occur with the least frequency, but which sometimes have far reaching implications for the spreadsheet (e.g. the SAVE FILE and WRITE FILE commands, the PRINT SPREADSHEET command, and the QUIT command). In some cases, this form is used to accommodate an odd command (e.g. the EXCHANGE CURSOR & MARK command), or to provide a needed contrast, as with:



Control----X, Control----B

The BUFFERS DIRECTORY command



Control----X, b

The SWITCH BUFFER command

'Control----X, Control-character' vs. 'Control----X, Lowercase Character' Commands

A relationship similar to that between 'Control' and 'Escape' commands holds between 'Control----X, Control----Character' and 'Control----X, Lowercase Character' commands. 'Control----X, Control----Character' commands tend to govern single entries, entities or lines, while 'Control----X, Lowercase Character' commands perform the same function over columns and specified regions (see table below).

The Relationship between Control----X, Control Character and Control----X, lowercase character commands.		
Character	Control----x, Control-Character	Control----x, character (specified regions)
I	C-x C-l LOCK ENTRY	C-x, l LOCK REGION
u	C-x C-u UNLOCK ENTRY	C-x, u UNLOCK REGION
p	C-x C-p PRINT SPREADSHEET	C-x, p PRINT REGION
k	C-x C-k CLEAR MEMORY	C-x, k DELETE BUFFER

Command Key Letters Identify the Command

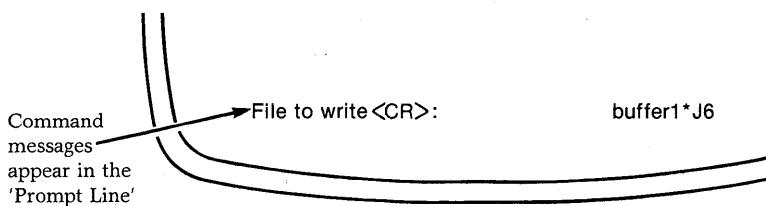
As much as possible, all command keys have been 'mnemonically' assigned to help you more easily remember what action they cause. For example, 'Control----F' moves the cursor 'forward', 'Control----D' 'deletes' an entry, 'Control----X, Control----P' causes the spreadsheet to be 'printed', etc.

All Commands Use Lowercase Letters

Although the illustrations and examples in this manual may not always indicate it, the characters used in commands are always **lowercase**. Capital, or uppercase letters, will not correctly invoke the command.

COMMAND MESSAGES

When Perfect Calc requires additional information regarding a command, it will 'prompt' you for this information with a message. For example, at the WRITE FILE command (Control---x, Control---w), Perfect Calc will ask in the Prompt Line:



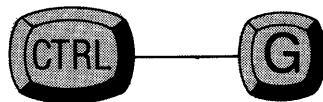
Perfect Calc is asking for the name of the file in which the current spreadsheet should be saved. Note that the message ends with a *<CR>*, which indicates that you should hit a carriage return **after** typing the name of the file.

Yes/No Responses

Sometimes in response to a command Perfect Calc will ask for a simple 'yes/no' decision from you. Typing either 'Y' or 'N' is sufficient to tell Perfect Calc what you want.

The CANCEL (or GO BACK) Command

Any command which Perfect Calc has **not yet begun executing** can be canceled by typing the CANCEL command. If you change your mind and want to stop and go back, or if you simply get stuck, type:

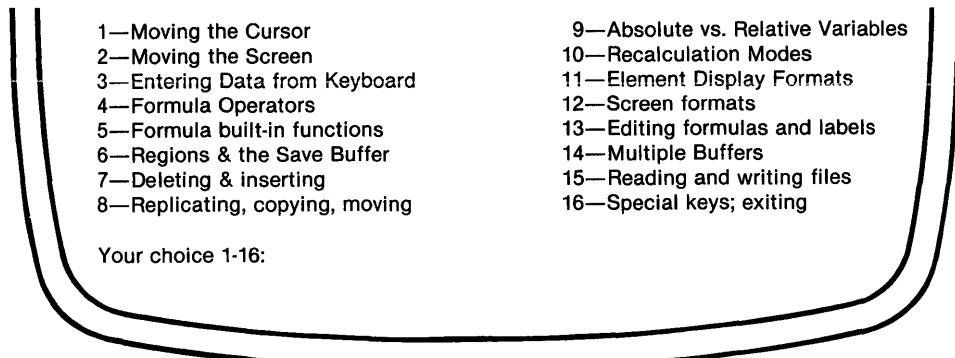


The HELP Command

There will certainly come a time when you will know all of Perfect Calc's commands by heart. However, until that time arrives you may need a little assistance in remembering the various command options. There is no need for you to go thumbing through the manual, because Perfect Calc provides a Help Menu, an on-line summary of its commands. At any time, simply type a question mark:



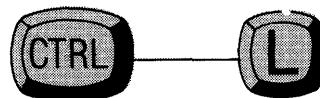
A Help Menu similar to the one shown below will appear at the bottom of the screen.



When this Help menu appears type the number of the command information you need, followed by a carriage return. For example, typing '1' will display all the commands that move the cursor. If you type '5' Perfect Calc will display the various predefined functions available to you.

You should get used to using this Help Menu. In fact, it would be wise to take time at the very start to call up each option on the Help Menu, one after another, just to see what information is available.³

Note: To erase the Help Menu, type the REDRAW SCREEN command:



In fact, any command which causes the screen to be redrawn will also erase the Help Menu. If you have the alternate command set the REDRAW SCREEN command is:



³ Unlike some other spreadsheet programs, you do not 'leave' your spreadsheet to see the help information and then 'return' to it. The help information in the lower portion of your screen is continuously available if you so desire—the message 'Next key continues operation' below it reminds you of this fact. You can remove this help information at any time, however, by using the REDRAW SCREEN command.

the first time I had been in a room full of people and I was nervous. I had been told that I would have to make a speech in front of them. I did not know what to say or how to act. I was afraid that they would not like me or that they would laugh at me. I was also worried about what they might think of me. I tried to calm myself down by taking deep breaths and focusing on my speech. I practiced my speech several times before the big day. I also asked my friends to help me with my speech and gave me some tips. On the day of the event, I was still nervous but I knew that I had prepared well. I took a few deep breaths and started my speech. I spoke clearly and confidently. When I finished, there was a round of applause from the audience. I felt relieved and happy. I realized that public speaking can be a great way to express yourself and connect with others. It takes practice and courage, but it can also be rewarding.

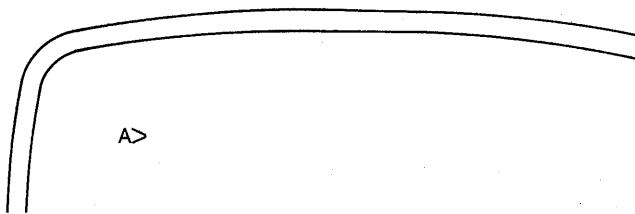
Chapter III

BEGINNING AND QUITTING PERFECT CALC

BEGINNING

After you have installed Perfect Calc on your computer (See Appendix A), and created a system disk holding the Perfect Calc programs, insert this disk into the first drive of your computer.

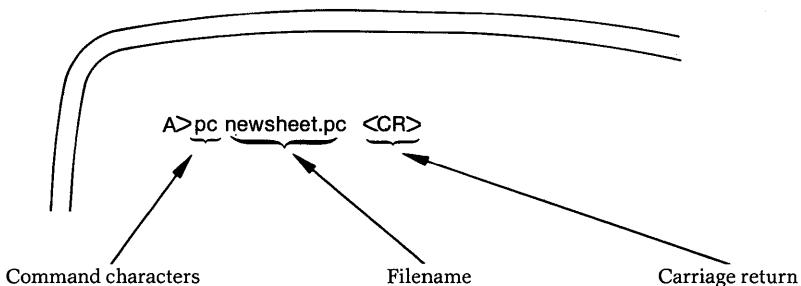
An operating system prompt will appear indicating that your computer is ready for you to give it a command:



At this point you have two options: you can begin with a new, empty spreadsheet, or you can begin by calling up a spreadsheet that you have previously created and saved.

Beginning With a New Spreadsheet

To create a new spreadsheet, simply enter 'pc', the operating system command characters that invoke Perfect Calc, followed by a filename for the new spreadsheet that you wish to create. For example:



Perfect Calc will present a new, blank spreadsheet, showing the first portion of the filename you have given (in this case, 'newsheet') in the lower right corner of the screen.

Note: If you fail to supply a filename, Perfect Calc will present you with a new, empty spreadsheet. However, it will be unnamed, and located in the 'default' buffer. If you later save it to disk without naming it, it will be placed in a special file called 'default.pc' where it is in **danger** of being overwritten by the next unnamed file that you may create.

Beginning with an Existing Spreadsheet

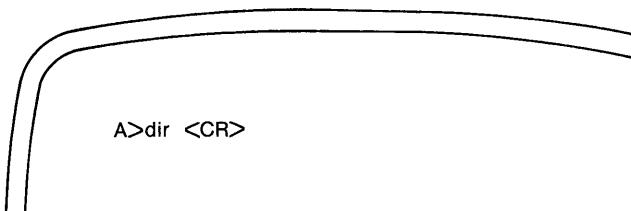
When you wish to work with a spreadsheet that you have previously saved, Perfect Calc places a **copy** of that spreadsheet into a 'buffer' for you, a temporary working space in machine memory. Whatever changes you make to this copy do not affect your original.

An existing spreadsheet can be retrieved from disk storage in one of two ways, either through a command to your computer's disk operating system when beginning Perfect Calc, or through the FIND FILE command. (Here we present only the first method.) The FIND FILE command, which involves an explanation of multiple 'buffers', is presented in Chapter XI.

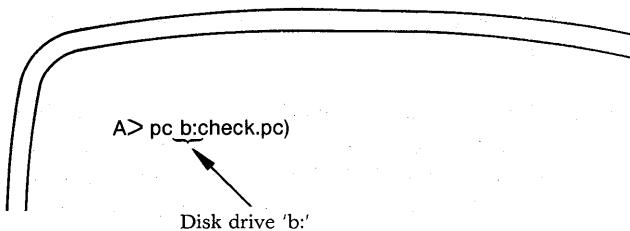
Steps:

You are about to enter Perfect Calc and want to edit a spreadsheet you have previously saved.

1. Consult your system directory for the correct file name of the spreadsheet you wish to retrieve. Use the 'dir' Command:

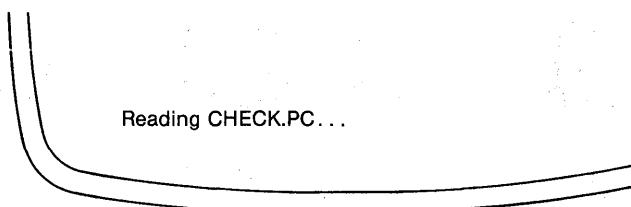


2. Type the filename after the 'pc' command letters that are used to invoke Perfect Calc. For example:



Important: Always remember to specify the disk drive where the file is located. For instance, the spreadsheet file 'check.pc' is on disk drive 'B'.

Perfect Calc will first present a blank spreadsheet accompanied by the message in the Prompt Line:



A few moments later the spreadsheet "CHECK.PC" will be displayed on the screen.

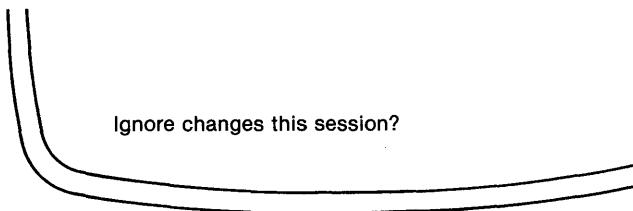
Note: If Perfect Calc cannot find the file you request, it assumes that it is a new file and presents you with a blank spreadsheet.

QUITTING

When you wish to stop working with Perfect Calc, type the QUIT command:



If you have made any changes to the spreadsheet being displayed and have not yet saved those changes, Perfect Calc will respond with the message:



Answering "yes" causes Perfect Calc to discard the spreadsheet you have been working with (i.e. it is **not** saved on disk).

Answering "no" cancels the QUIT command, in effect giving you an opportunity to save the current spreadsheet if you wish. After saving the spreadsheet, enter the QUIT command again, and Perfect Calc will return you to the computer's operating system.¹

¹ Commands for saving your spreadsheet are presented in Chapter X.

Chapter IV **MOVING ABOUT THE SPREADSHEET**

CURSOR AND SCREEN COMMANDS

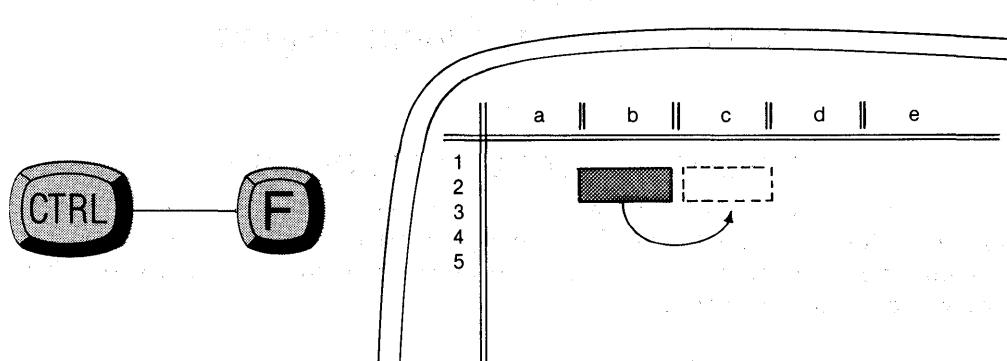
In this chapter we present the various means of moving the cursor, not only column by column and line by line, but also page by page (in whatever direction), including even a command that skips the cursor directly to an entry position anywhere in the spreadsheet.

Elementary Cursor Commands

On most computer terminals the cursor can be moved using 'arrow keys'. In addition to the arrow keys, Perfect Calc provides the following cursor commands. These elementary commands are identical to those used by the Perfect Writer word processing system, and can be used to move the cursor anywhere on the Perfect Calc spreadsheet.

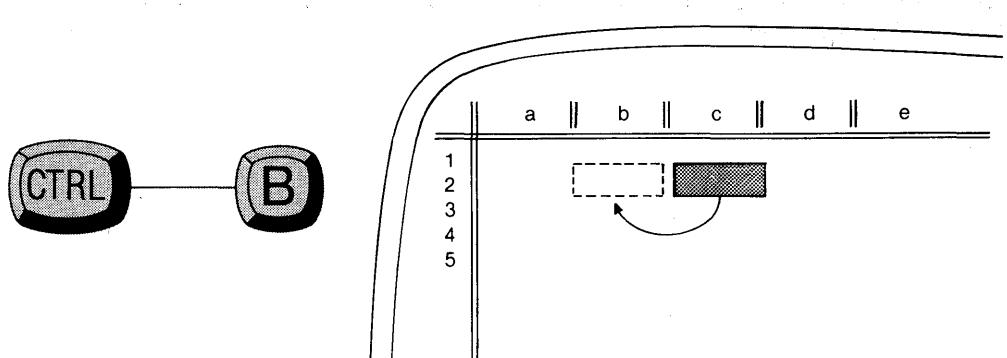
The FORWARD COLUMN Command

Moves the cursor forward (i.e. to the right) one column, same line.



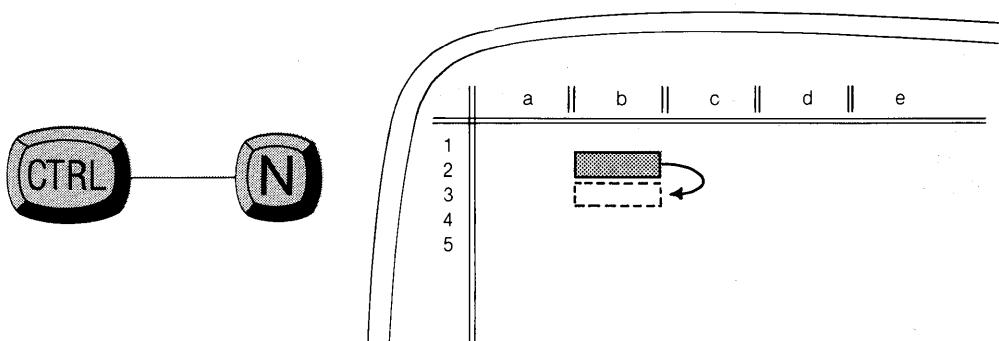
The BACKWARD COLUMN Command

Moves the cursor backward (i.e. to the left) one column, same line.



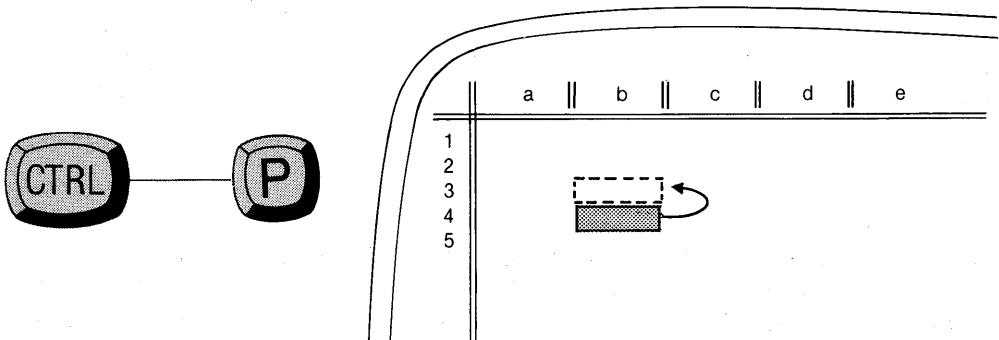
The NEXT LINE Command

Moves the cursor to the next line (same column).



The PREVIOUS LINE Command

Moves the cursor to the previous line (same column).

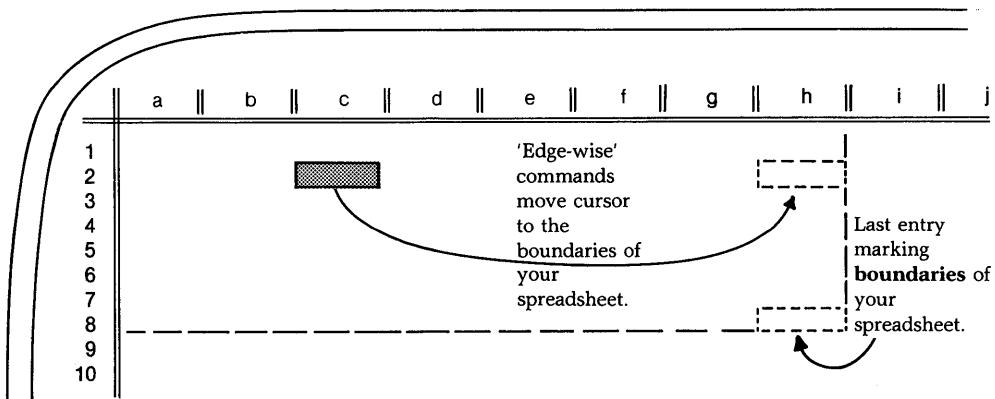


Note: If the cursor is positioned at the edge of the screen and any of the above commands would cause it to move to an off-screen position, Perfect Calc will erase and redraw the window to include the new entry position. Repeatedly entering the command will "drag" the window along with the cursor, thus scanning either horizontally or vertically through the spreadsheet. If the spreadsheet contains many filled entries, this scrolling may take some time, since the entire screen must be redrawn each time. You may want to use one of the 'edge-wise' or 'page-wise' cursor commands presented in the following pages.

Edge-Wise Cursor Commands

It is often useful to move the cursor in large jumps to the edges of the currently defined spreadsheet. The following 'edge-wise' commands will jump the cursor to the ends of a line or column.

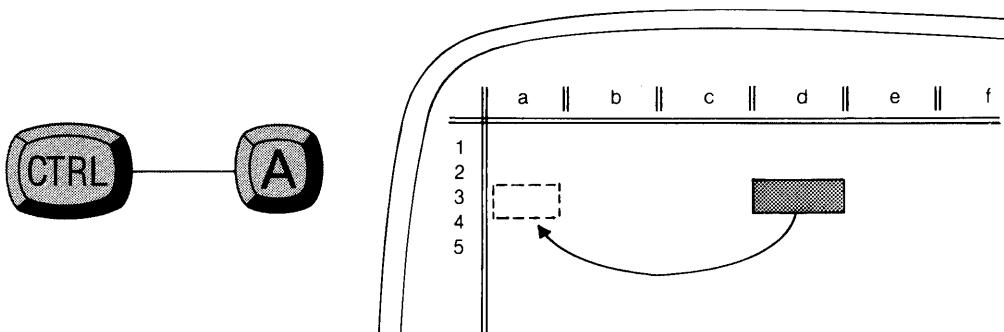
Note that for each of these commands the move is made only to the edges of the currently defined rectangular spreadsheet and not to the extreme limits of Perfect Calc's spreadsheet. If your lowest, rightmost entry is at position "h6", and your cursor is at "c2", typing the END OF LINE command will position the cursor at position "h2" and not at "Z2".



If the end of a line or column is beyond the present 'window', the screen will be redrawn at the new location. Perfect Calc will always attempt to position the window so that the most information is showing. For instance, the END OF LINE Command, after moving the cursor, will adjust the window so that the rightmost column will appear at the righthand edge of the screen.

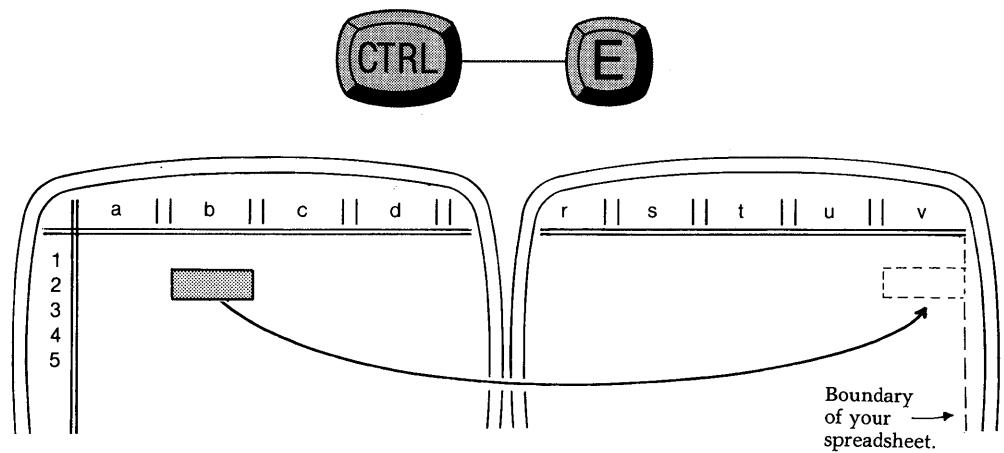
BEGINNING OF LINE Command

Jumps the cursor from anywhere in a line to the **first** entry in the line, redrawing the window as needed.



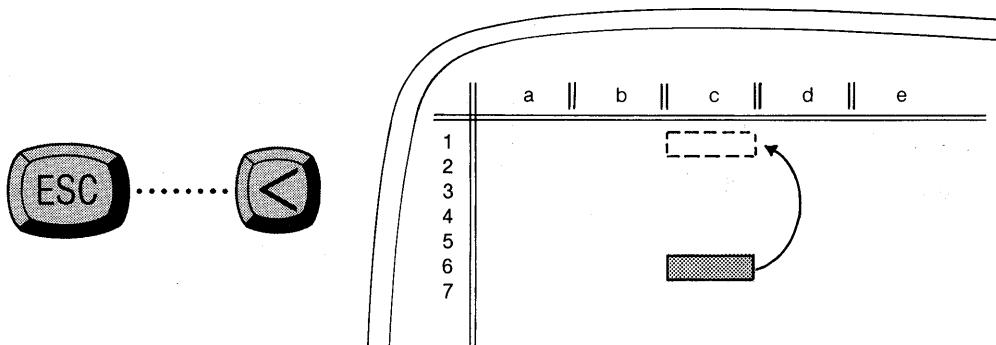
END OF LINE Command

Jumps the cursor from anywhere in a line to the **last** entry in the line, redrawing the window as needed.



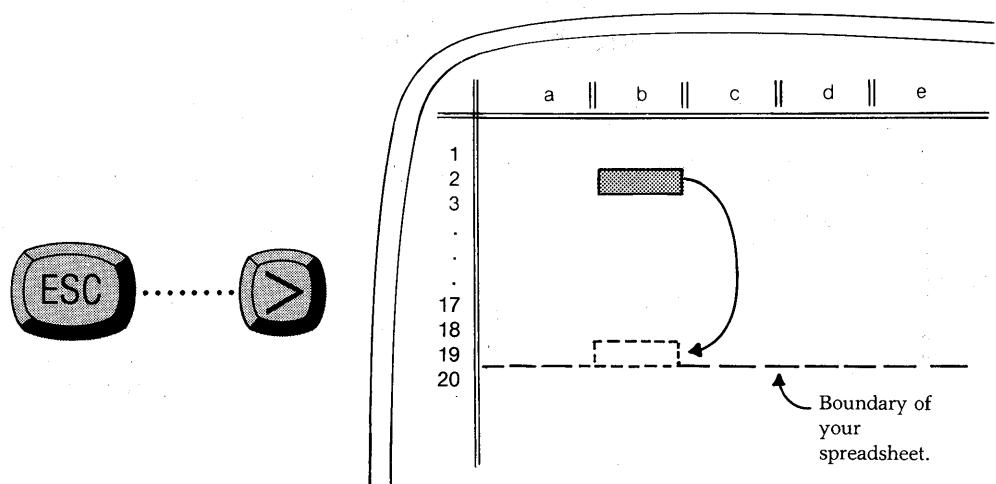
TOP OF COLUMN Command

Jumps the cursor from anywhere in a column to the **first** entry in that column, redrawing the window as needed.



BOTTOM OF COLUMN Command

Jumps the cursor from anywhere in a column to the **last** entry in that column, redrawing the window as needed.

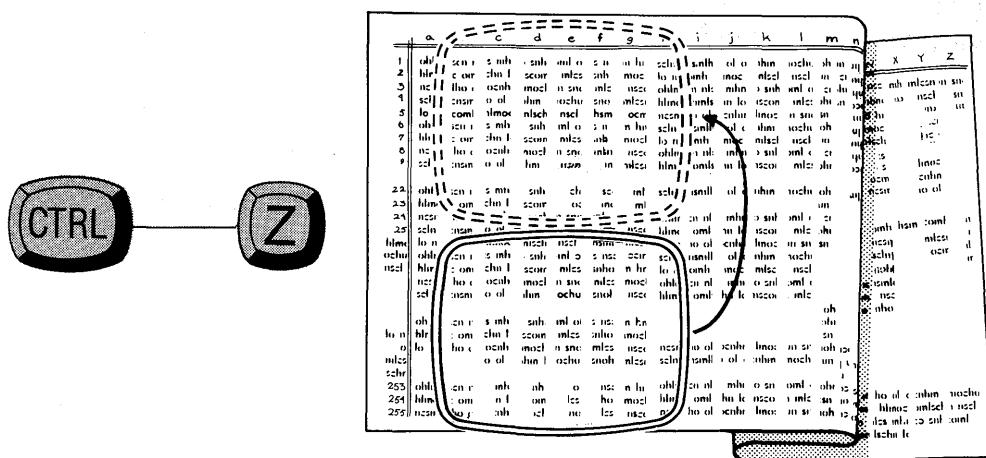


Page-Wise Screen Commands

Another useful group of cursor commands allows you to move about the spreadsheet a 'page' at a time—a 'page' being defined as what you can see in the current window. (A 'page' thus depends on the number and sizes of the 'windows' that you have activated¹, as well as on the size of your screen.)

The PAGE UP Command

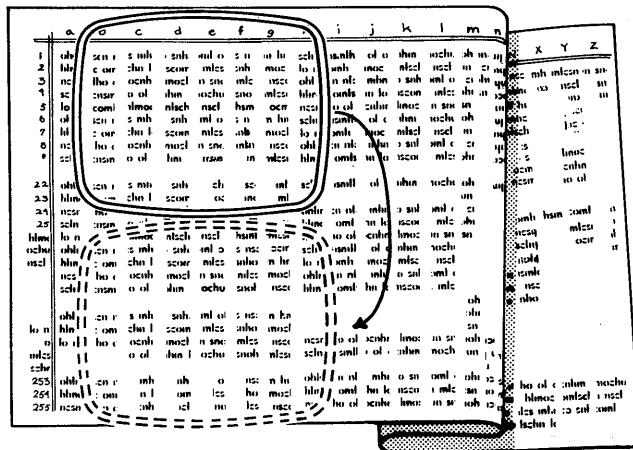
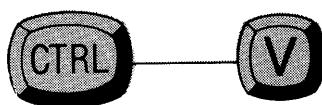
Moves the screen 'up' a page-full of lines.



¹ See Chapter XI for a discussion of multiple spreadsheet analysis.

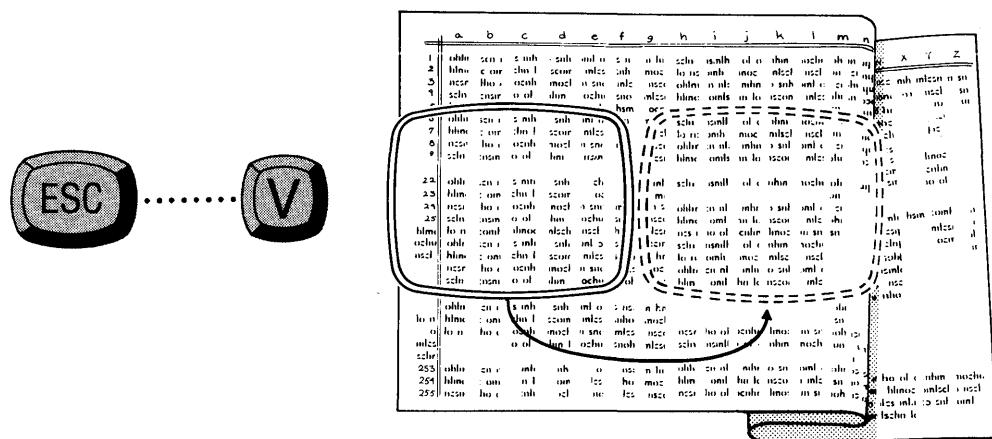
The PAGE DOWN Command

Moves the screen 'down' a page-full of lines.



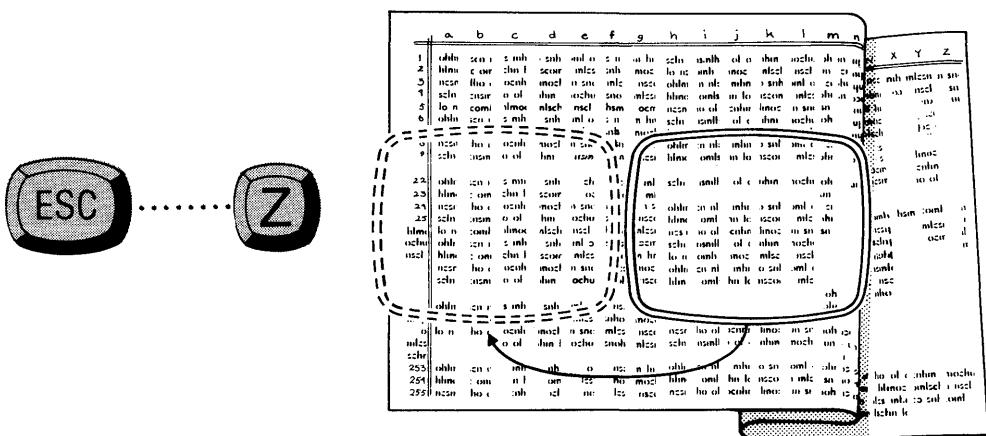
The PAGE RIGHT Command

Moves the screen to the right a page-full of lines.



The PAGE LEFT Command

Moves the screen to the left a page-full of lines.

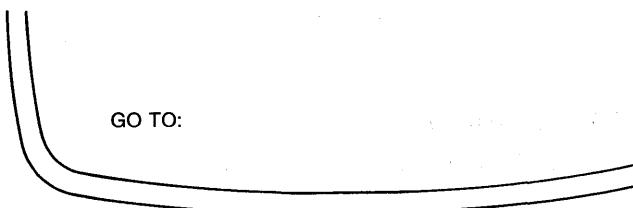


The GOTO Command

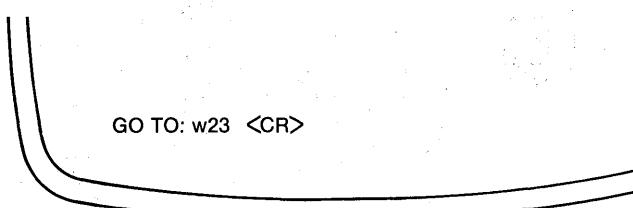
The cursor commands introduced on the previous pages are the primary movement commands you will use. There are times when you will want to move to a specific location on the spreadsheet. For this occasion Perfect Calc provides a GOTO command, which moves the cursor directly to a specified position, redrawing the screen as needed. The command form is invoked by typing a 'greater than' key (using the shift key if needed):



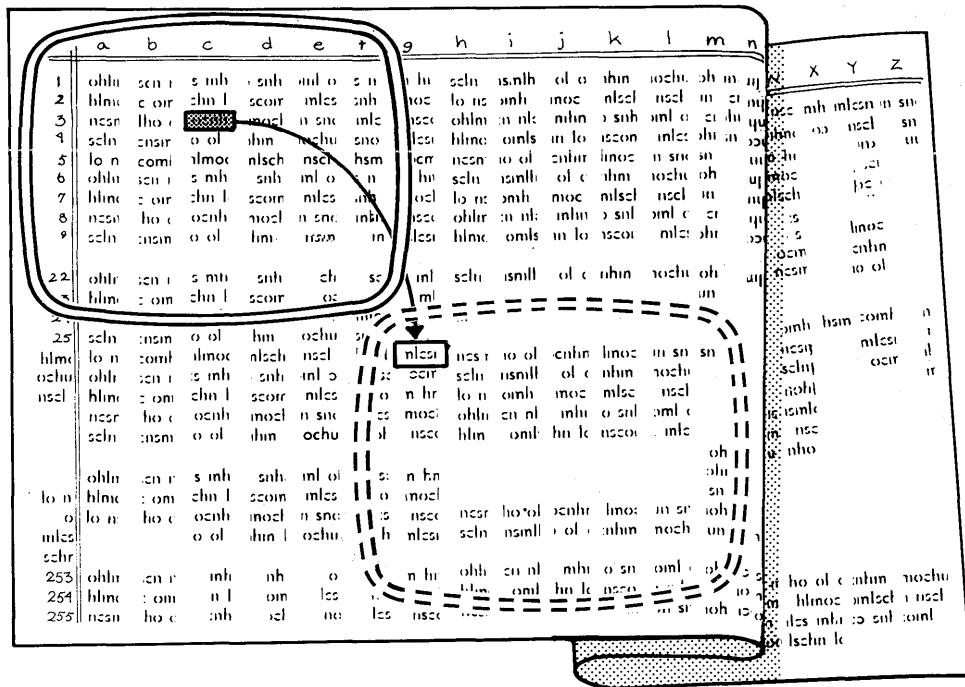
Perfect Calc will respond in the Prompt Line with the message:



Simply type the coordinates of the position you wish to move the cursor to, followed by a carriage return. For example:



If the new position is somewhere in the current window, the cursor will move directly to it. If it is somewhere off-screen, the window will be blanked and redrawn so that the new location will be shown.



Illegal Coordinates

Perfect Calc will check to see that you enter the coordinates for a GOTO correctly. It will refuse to accept any that are "illegal". For example, if the first character of the coordinate that you enter is not an alphabetic character, 'a - z' or 'A - Z', Perfect Calc will beep and refuse to accept it.

Cancelling the GOTO Command

If you change your mind while typing a new position for the GOTO command, you can "back out" of the GOTO by repeatedly hitting the Delete key, or by typing the CANCEL command, (Control----G).²

² If you have the alternate implementation of Perfect Calc then only the Delete key will move the cursor back and erase. Otherwise, you may also be able to use the backspace key.

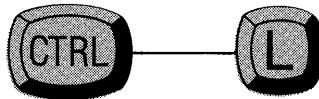
Two Additional Commands

Two commands that are often used when moving the cursor are the REDRAW SCREEN command and the REPEAT command.

The REDRAW SCREEN Command

Causes the entries surrounding the cursor to be redisplayed in the middle of the screen. This command is extremely useful with cursor commands. It is also useful in clearing the screen of the Buffer Directory or Help Menu.

For example, if the cursor is positioned on a line near the bottom (or top) of the screen, this command will cause the spreadsheet to be redisplayed so that the line and cursor are now in the center of the screen.



Note: If you have an alternate version of Perfect Calc, then you may need to type:

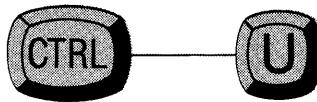


The REPEAT Command

Causes other commands to repeat. The number of repetitions may be specified, but if not, a default value of '4' is supplied by Perfect Calc.

Steps:

1. Type the REPEAT command:



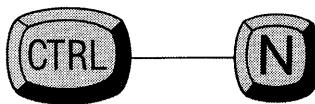
Perfect Writer echoes the command with the message:



2. Enter the number of repetitions desired. The default value of '4' will be replaced by the number that you type.

3. Enter the command you wish to have repeated. Perfect Calc immediately repeats the command the number of times you have specified.

For example, if you enter: The NEXT LINE Command



The cursor now moves down four lines (i.e., it repeats the NEXT LINE command four times).

Note: The argument of the REPEAT command can be enlarged by simply repeating the command itself. Each time the command keys are typed, the argument is multiplied by '4'—Control----U...Control----U...Control----U = Arg: 64.

Further Note: The message 'Arg: 4' will not appear if the steps of this procedure are performed rapidly enough. Perfect Calc intentionally delays displaying the message in order to avoid inconveniencing the experienced user who, familiar with the command, may not need to work through the added step.

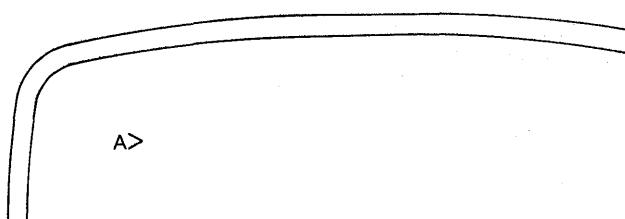
MOVING ABOUT A LARGE SPREADSHEET

TUTORIAL

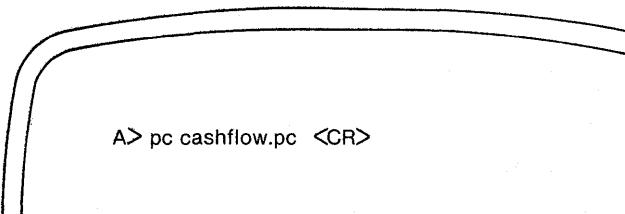
In this tutorial you will practice moving about a spreadsheet using the cursor and screen commands presented in this chapter. For your spreadsheet you will use the 'Cash Flow Assessment' program provided on the Perfect Calc system diskette.

Steps:

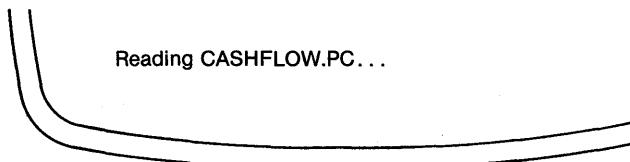
1. After making sure that your computer is turned on, insert the Perfect Calc system diskette into drive 'A'.
2. Your Disk Operating System (DOS) will display a system 'prompt' indicating that it is ready to receive a command:



3. Type the command letters which invoke Perfect Calc, followed by the filename of the Cash Flow Assessment program. Terminate the entry with a carriage return, <CR>:



Perfect Calc presents you with a blank spreadsheet accompanied by the message in the Prompt Line:



A few moments later the 'Cash Flow Assessment' Program is displayed on the screen, the cursor located in the upper left-hand corner. It should look like this:

CASH FLOW ASSESSMENT						
Copyright, 1982, Perfect Software, Inc.						
	MONTHLY					
	Jan	Feb	Mar	...	Dec	TOTAL
Cash Flow In	0	0	0	...	0	0
Sales	0	0	0	...	0	0
Services	0	0	0	...	0	0
Accounts Receivable	0	0	0	...	0	0
Contract Accounts	0	0	0	...	0	0
Other	0	0	0	...	0	0
Total Cash Flow In	0	0	0	...	0	0
Cash Flow Out	0	0	0	...	0	0
Rent	0	0	0	...	0	0
Utilities	0	0	0	...	0	0
Phone	0	0	0	...	0	0
Payroll	0	0	0	...	0	0
Supplies	0	0	0	...	0	0
Insurance	0	0	0	...	0	0
Postage	0	0	0	...	0	0
Advertising	0	0	0	...	0	0
Maintenance	0	0	0	...	0	0
Other	0	0	0	...	0	0
Total Cash Flow Out	0	0	0	...	0	0
Monthly Cash Position	0	0	0	...	0	0
WORKING CAPITAL	0	0	0	...	0	0

4. Although the program is empty of entries, it will assist you in practicing the cursor commands. To obtain a screen listing of the cursor commands type a question mark:



5. Perfect Calc immediately displays its Main Help Menu in the lower half of your screen. 'Cursor' commands are listed as option '1' on the menu. Type '1' followed by a carriage return, <CR>.

Perfect Calc presents the following display in the lower portion of your screen:

MOVING THE CURSOR		
C-F*	Cursor Forward	ESC< Cursor to top line
C-B*	Cursor Backward	ESC> Cursor to bottom line
C-N*	Cursor to Next Line	C-A Cursor to beginning of line
C-P*	Cursor to Previous Line	C-E Cursor to end of line
C-S	Search for Next Unlocked Numeric Entry Go to a specific position, entered as a column letter, line number	
C = Control Key	ESC = Escape key	* = May be repeated via C-U
Next key continues operation		Control-L redraws window†

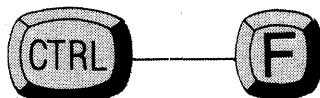
You may erase this help menu by typing Control----L, † or leave it displayed while practicing the commands. Perfect Calc will continue to function even with the help menu displayed. (However, moving the cursor onto the help menu will erase portions of it.)

† If you have the alternate command set you'll need to type Escape...Control----L.

Practice moving the cursor one entry at a time using the first four commands listed in the left hand column of the help menu. These commands are equivalent to the arrow keys which many computers possess, namely:

- (**CTRL**) — (**F**) 'Right-Arrow' or FORWARD COLUMN Command
- (**CTRL**) — (**B**) 'Left-Arrow' or BACKWARD COLUMN Command
- (**CTRL**) — (**P**) 'Up-Arrow' or PREVIOUS LINE Command
- (**CTRL**) — (**N**) 'Down-Arrow' or NEXT LINE Command

6. Move the cursor to the entry position at the right edge of the screen by repeatedly entering the 'right arrow' key, or FORWARD COLUMN command, if your computer does not have arrow keys, or for some reason they do not function with Perfect Calc.



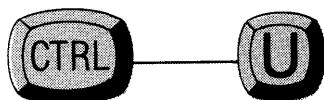
7. When you reach the right edge of the screen, attempt to move the cursor to the next 'off-screen' position on line 1 by again typing the right-arrow key (or FORWARD COLUMN command).

Perfect Calc responds by erasing and redrawing the entire screen in order to include the next entry column 'f'. Simultaneously, at the left-hand side of the screen, column 'a' disappears.

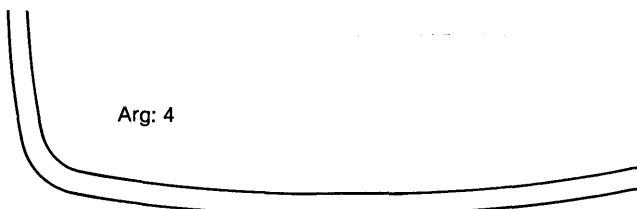
This action represents a crude method of 'scrolling' the window (i.e moving the screen across the spreadsheet). Note that if you repeat the command while Perfect Calc is redrawing the window, the 'redraw' is terminated and the new command immediately executed. Thus, to move right four columns rapidly (and not have the screen redrawn each time), type Control----F four times in rapid succession

All four of the directional cursor commands move the cursor in a similar fashion. Practice using them to move the cursor in various directions. Continue to observe Perfect Calc's response as you move the cursor to off-screen positions.

8. Notice that on the Help Menu the elementary cursor commands are tagged with a star, *. This means that they can be repeated using the REPEAT command, Control---U. Practice this command procedure. With the cursor at any location on the screen, type:

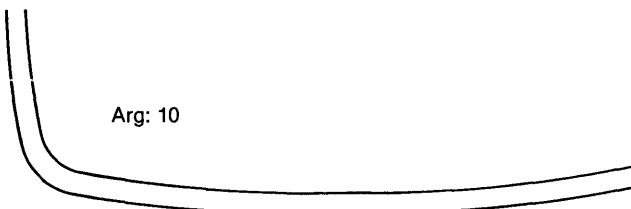


Perfect Calc responds with the message in the Prompt Line:



This means that Perfect Calc will repeat four times whatever command or character (excepting digits) that you type next. To change the 'argument' type any number. For example, type: '10'.

The message in the Prompt Line immediately changes to:



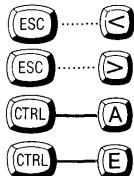
9. Now type any of the elementary cursor commands. For example, type:



Perfect Calc immediately repeats this cursor command 10 times, moving the cursor 10 entry positions to the right. (Return the cursor to its original position by repeating the opposite cursor command 10 times.)

Edge-Wise Cursor Commands

1. Now you will practice several commands which move the cursor longer distances. These are the 'edge-wise' commands, which are displayed on the right-hand side of the cursor Help Menu. (If you have erased the Help Menu, call it up again by typing '?', followed by option '1'.)



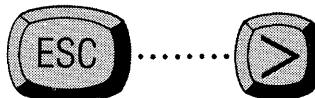
TOP OF COLUMN command
BOTTOM OF COLUMN command
BEGINNING OF LINE command
END OF LINE Command

2. With the cursor anywhere in the middle of the 'Cash Flow' spreadsheet, type the TOP OF COLUMN command:



Perfect Calc immediately moves the cursor to the top entry in the column that it occupies.

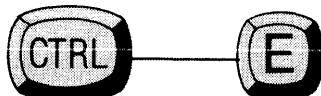
3. Now move the cursor in the opposite direction. Type the BOTTOM OF COLUMN command:



Perfect Calc jumps the cursor to the bottom 'boundary' of the Cash Flow spreadsheet, redrawing the screen to do so. Notice that the command does not move the cursor to the extreme limit of Perfect Calc's spreadsheet (i.e. line 255), but only to the bottom boundary of your spreadsheet, which is determined by the lowest entry in the spreadsheet.

The other edge-wise commands function similarly: they move only to the boundaries of the defined or currently used spreadsheet.

- Now move the cursor to the right-most boundary of the Cash Flow spreadsheet, by typing the END OF LINE command:



Again, the command does not skip the cursor to column 'Z', the last and farthest column in Perfect Calc's spreadsheet, but rather to the last column in the Cash Flow spreadsheet.

- Give the BEGINNING OF LINE command to move the cursor back to column 'a':



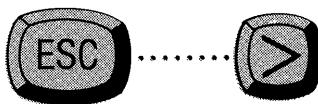
followed by the TOP OF COLUMN command to move the cursor to position 'a1':



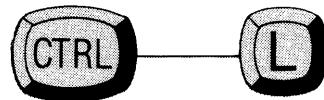
Note that all of the edge-wise cursor commands will always, after redrawing the screen as necessary, position the cursor on the screen so that the maximum amount of data contained in the line or column is showing. That is, in moving the cursor to the end of a line, Perfect Calc will position the cursor at the right margin of the screen. Similarly, when moving it to the bottom of a column, it will place the cursor at the bottom margin of the screen.

Because it may at times be inconvenient to view or to work with a data entry at the extreme lower margin of the screen, Perfect Calc provides a command for centering the line which the cursor is occupying. This is the REDRAW SCREEN command.

6. Move the cursor to the last entry in its column using the BOTTOM OF COLUMN command:



followed by the REDRAW SCREEN command:



(or Escape...Control----L for the alternate command set.)

Perfect Calc immediately adjusts all text upward, positioning the last line in the center of the screen.

Move the cursor back to the top of the column using the TOP OF COLUMN command:



Page-Wise Screen Commands

Perfect Calc provides several commands for scrolling rapidly through the spreadsheet, to areas beyond even the boundaries of the currently defined spreadsheet. These are the 'page-wise screen commands', which are presented in the Help Menu (option '2': Moving the Screen).

C-V* View next page downward
C-Z* View next page upward
C-X,C-V* Page down, other window
C-X,C-Z* Page up, other window

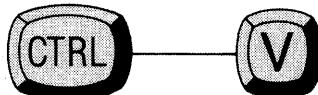
ESC v* view next page to the right
ESC z* view next page to the left
C-X,v* page right, other window
C-X,z* page left, other window

C-L redraw window, centering cursor's line

C- = Control Key ESC = Escape key * = May be repeated via C-U
Next key continues operation Control-L redraws window

We will examine only those screen commands in the top two lines of this menu. The 'other window' commands will be presented in Chapter XII.

1. With the cursor positioned anywhere in the middle of the screen, type the PAGE DOWN command:



Perfect Calc moves the screen down one full screen of lines. The cursor continues to display in the same relative position on the screen. That is, if the cursor is near the top margin of the screen when the command is given, it will appear at the same spot when the screen is redrawn one 'page' down. This action is typical of all of the screen commands. (To center the cursor, use the REDRAW SCREEN Command [Control----L or Escape...Control----L] for the alternate command set.)

2. Continue to practice the four screen commands, moving the screen to the extreme perimeters of Perfect Calc's spreadsheet (and back). When you have finished, position the cursor at the lowest, right-most entry position of Perfect Calc's spreadsheet, entry position 'Z255'.

Note: The universal repeat command may be used with the screen motion commands for large jumps like these. All "repeatable" commands are marked with an asterisk in the help information area.

The GOTO Command

The last command in this tutorial is Perfect Calc's GOTO command, which allows you to skip directly to any entry position in the spreadsheet. Use this command to return from 'Z255' to position 'a1' of the Cash Flow Assessment program spreadsheet.

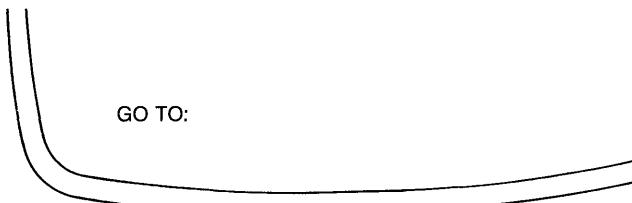
1. Type the GOTO command:



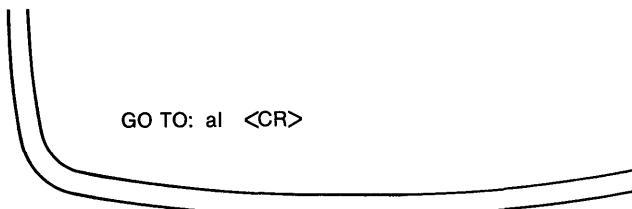
(right 'angle-bracket')

(You may need to use the shift key to type this character.)

Perfect Calc responds with the message in the Prompt Line at the bottom of your screen:



Enter the coordinates of the position you wish to 'go to', followed by a carriage return:



Perfect Calc jumps the cursor to position 'a1' in the top left-hand corner of the spreadsheet.

This concludes the practice tutorial on cursor and screen commands. To exit Perfect Calc, type the QUIT command:



Perfect Calc returns you to your disk operating system (DOS).

END OF TUTORIAL

Chapter V

ENTERING DATA: LABELS AND NUMBERS

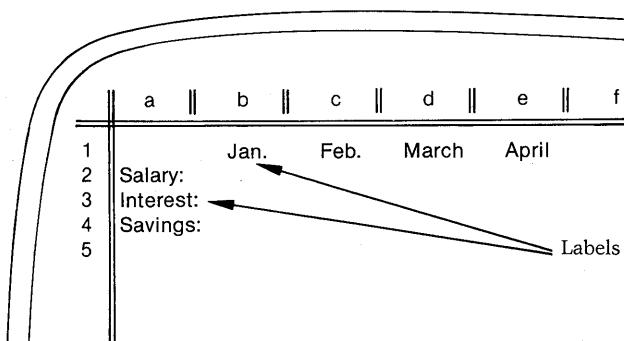
Introduction

All data is entered into the spreadsheet by positioning the cursor at the desired location and typing directly. A data entry, the type of which is determined by the first character, can consist of a label (usually a word or words), a number, or a formula. Data entries are terminated either by a carriage return, <CR>, or by any command key, which will perform the requested action after terminating the entry. Ending an entry with a command key instead of a carriage return allows you to reduce the number of keystrokes. Be sure to use this handy feature.

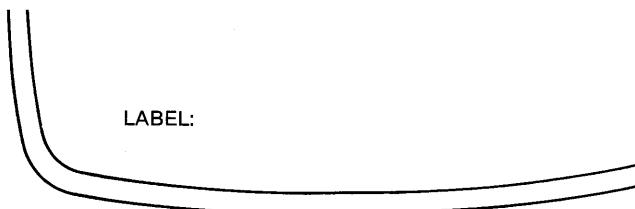
As you type a label, number, or formula, it will be displayed in the Prompt Line. Since the data does not become part of the spreadsheet until you 'enter' it, a partial entry can be corrected or canceled using either the backspace key, or the CANCEL command, {Control----G}.

LABELS

The simplest data entry is a 'label', which is a sequence of printable characters used to visually identify the lines, columns, and regions of the spreadsheet. No computations are performed on label entries.



Any printable character which does not begin either a command or one of the other types of data entry (a number or a formula) may begin a label. When the first character is typed, Perfect Calc will display the following message in the Prompt Line:

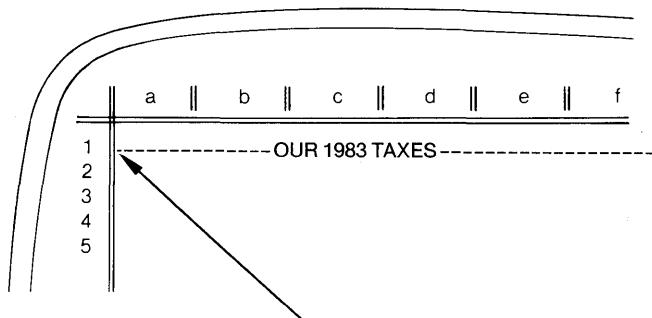


As you type the label, it will appear in the Prompt Line following this message, to be transferred to the spreadsheet at the location of the cursor when the entry is terminated by a carriage return, <CR>, or any command key.

Forcing a Label

If the first character of a desired label would normally begin another type of entry (e.g., a hyphen, '-' , signals a negative number), or would cause some action to occur (e.g., the right angle bracket, '>' invokes the GOTO command), the double quote character (") may be typed to force a label. This character tells Perfect Calc that a label will follow. It is not included as part of the label.

For example, the double quote would have to be the first character typed when entering the following label:

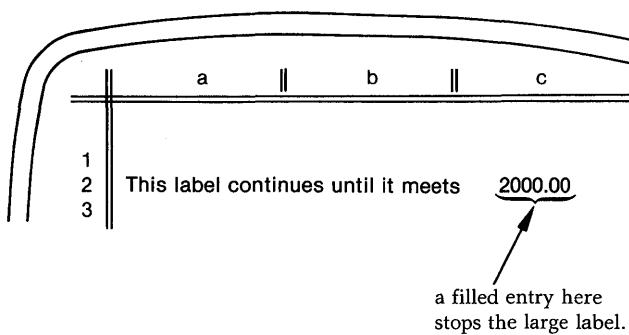


Use a double quote ("') to force a label beginning with a hyphen.

Size of Labels

A label entry may be as wide as the maximum column width, which is 76 characters. If you attempt to enter more than this maximum number of characters when constructing a label, Perfect Calc will beep and disallow any further keystrokes until you terminate the label entry.

If the label is larger than the column it occupies only the initial portion of the label that will fit into the column will be shown, **unless** the entry positions to the right of the label are empty. In this case, the label will extend beyond its column limits until it has been fully printed, or until another filled entry position is encountered.¹



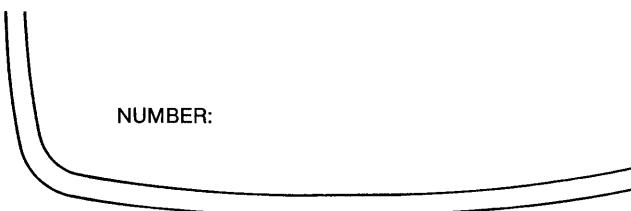
¹ Perfect Calc allows you to modify the widths of columns, see Chapter V, page 79, following.

Positioning Labels

Labels are normally left-justified within a column. To center or right-justify a label within a column, either change the default justification parameter (see Chapter V, page 81), or pad the label with leading blanks (using the space bar).

NUMBERS

Like labels, numbers are entered into the spreadsheet by positioning the cursor at the desired location and typing directly. When you type the first character, Perfect Calc will display the message "NUMBER: " in the Prompt Line. The number you type will initially be displayed following this message and will not be entered into the spreadsheet until you type a carriage return, <CR>, or another command key, which will perform the requested action after entering the number.



Although we may think that numbers are easy things to recognize, Perfect Calc has special rules regarding numbers that will not permit it to accept a wide variety of notations that we might say were 'numbers'. For example, Perfect Calc would not recognize as a number the word "twelve", or the roman numeral "VII", or the notation \$50, or #64, or 25%, 75 cents, 1/2, (1200), or even 3,464.78.

Because Perfect Calc must be able to recognize all kinds of numbers (positive and negative numbers, decimal numbers, 'whole' numbers, very large and very small numbers, numbers that are represented in 'scientific notation'), it uses a standard definition of what constitutes a number.

How does Perfect Calc recognize a Number?

When you begin typing a number, Perfect Calc looks at the first character and attempts to decide whether or not what you are entering is, in fact, a number. To be a number the **first** character must be either:

- a hyphen, (i.e. minus sign): “-”
- a period, (i.e. decimal point): “.”
- or
- Any digit: 0-9.

Once Perfect Calc has determined that you are indeed entering a number, it watches to see that the number follows certain rules.

Specifically, a number can have, **in the following order:**

A minus sign (optional)
followed by
Zero or one or more digits
followed by
A decimal point (optional)
followed by
Zero or one or more digits
followed by
An ‘e’ or ‘E’ (optional exponent sign)
followed by
A ‘+’ or ‘-’ (optional)
followed by
One or two digits of the exponent

Legal Numbers

Examples of numbers which Perfect Calc will accept, according to the above definition, include:

Sample Legal Numbers

17

-12.345e+10 (i.e. -12.345×10^{10})

2E5 (i.e. $2 \times 10^5 = 200,000$)

.25

Illegal Numbers

If an illegal number (one not meeting the above criteria) is entered, Perfect Calc will beep and ask for another corrected entry. The following represent illegal numbers for the reasons given:

+1	A plus sign is not allowed at the beginning of a number
10E125	The exponent must be between -99 and +99
204.0E2.0	Exponents may not contain decimal points
E5	A number may not begin with an exponent
1/2	Numbers may not contain arithmetic operators (i.e. '/')
1,234	Numbers may not contain commas
(678)	Numbers may not be enclosed within parentheses

Note: The last two restrictions come from the special meaning that commas and parentheses have in Perfect Calc.

Display Notation

When a number entry is terminated by a carriage return, <CR>, the number is transferred from the Prompt Line to the spreadsheet, and entered at the position of the cursor. Numbers will normally be displayed in the spreadsheet as a figure possessing two decimal places (or in scientific notation, if needed—see below).² A number entered as:

43

would display on the spreadsheet as:

43.00

while a number entered as:

172359

would display in a column of standard width as:

1.72E + 05

Note here that the last three digits, "...359", of this second number are not lost, **they are simply not being displayed.** Perfect Calc maintains 13 significant figures at all times, which means that even a number as large as:

9,123,456,789,123

and which would appear as:

9.12E + 12

will not be rounded but remembered exactly, even though all of the digits are not displayed on the screen.

² To change how many decimal places are displayed, use the command procedures presented in Chapter IX.

Scientific Notation

Exponential, or 'scientific' notation, is automatically used for any number that would 'overflow' its entry box, i.e. require more character positions to display than are currently available in the column.³

Numbers displayed using scientific notation contain an integer or decimal fraction (termed the mantissa) followed by the letter 'e' or 'E' and a positive or negative integer exponent. This notation indicates that the mantissa is multiplied by 10 raised to the power of the exponent. For example:

4.7E+3 is equivalent to 4700 (or 4.7 times 10^3)

Exponents can be either positive or negative, and must lie within the range -99 and +99. Also, as indicated above, an exponent may **not** be used to begin a number.

Underflow

Numbers which would 'underflow' the display (rather than 'overflow' as above) are treated somewhat differently. Whereas Perfect Calc automatically shifts to scientific notation for numbers which exceed the column width, numbers which are too small to display any significant figures are displayed with a decimal place followed by zeroes. A true zero is displayed as a single zero, enabling you to distinguish between numbers that are exactly zero and those that are approximately zero.

Consider the numbers '1,000,000,000', '1.01', '0.001', and '0' (zero). In a column of standard width 9, these would display as (assuming a display format of two decimal places):

1,000,000,000	as	1.0E+9
1,01	as	1.01
.001	as	00
0	as	0

³ Column widths can be changed to accommodate larger numbers, using the procedures described on the next page.

CHANGING COLUMN WIDTHS

Perfect Calc allows the widths of columns to be changed in order to accommodate entries of various lengths. A change in column width can affect only the column occupied by the cursor, or every column in the spreadsheet.

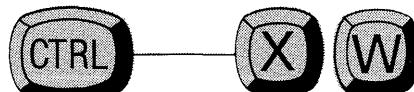
Changing column widths may change the number of columns displayed on the screen, although Perfect Calc will try to fit as many columns onto the screen as possible. On an 80-column terminal, for example, choosing a column width of seven characters will cause 11 columns to display (i.e. $80/7 = 11.4$), while a column width of 19 will allow only four columns to display (i.e. $80/19 = 4.21$). Initially, columns are set at a default width of nine characters, displaying a total of eight columns on the screen at once. Columns can be expanded to a maximum width of 76 characters, allowing only one column to be displayed.

Changing column widths may affect the format in which numbers are displayed on the screen. For example, especially large numbers which would exceed the new width (if you are reducing it), will be reformatted in scientific notation to fit the smaller column width. In such cases the numbers will be adjusted so that as many digits of precision as possible will be shown at all times.

For labels that do not fit a new column width, only those characters that do fit will be shown, unless the succeeding line positions are empty (See Chapter V, page 72).

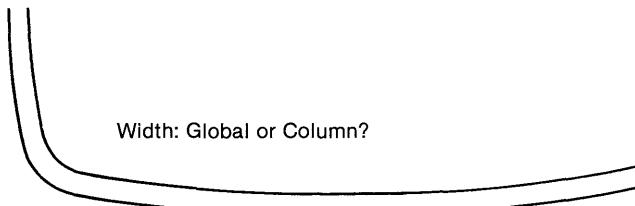
Steps:

1. At any time during editing type the CHANGE COLUMN WIDTH command:



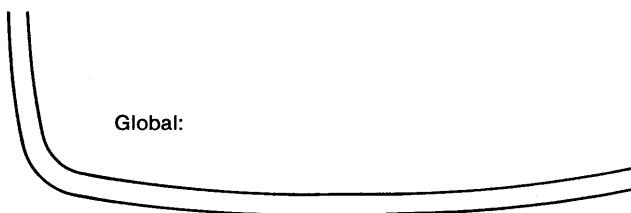
Note: The command letters, such as "w" are always lowercase.

Perfect Calc responds with the message:



2. Answer 'g' if the change in column width is to apply to every column in the spreadsheet, or 'c' if only the current column is to be changed.

Perfect Calc echoes your answer, and asks for the new column width. For example:



Enter any number between 0 and 76 followed by a carriage return, <CR>.

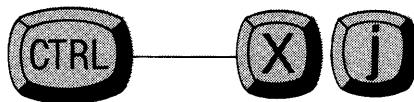
Note: If you simply type a carriage return without entering a number, a default column width will be used. When setting a single column, this default will be whatever 'global' column width is already being used. For global changes, the default width is nine characters.

POSITIONING ENTRIES

Initially all numbers are right-justified within a column while labels are left-justified. However, it is possible to alter this positioning of entries, either for a single line or column, or globally throughout the spreadsheet. Entries may be 'left-', 'center-', or 'right-justified' within a column.

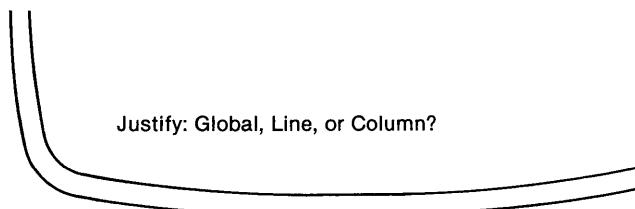
Steps:

1. At any time during editing type the JUSTIFY ENTRY command:



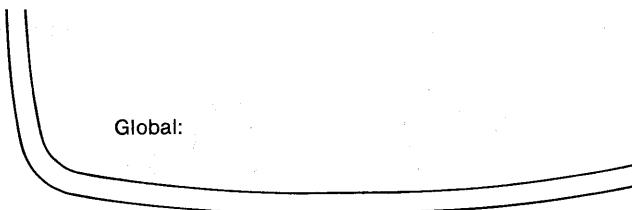
Note: The "j" is lowercase.

Perfect Calc will respond with the message:



2. Answer 'g' if you wish to change the justification of every entry in the spreadsheet; and 'l' or 'c' if you wish to change just the entries in the current line or column.

Perfect Calc echoes your answer and asks for the new justification parameter. For example:



3. Answer 'l', 'r', or 'c'.

Note: Typing just a carriage return, <CR>, will return the justification to the default parameter (i.e. left-justified for labels and right-justified for numbers.)

BUILDING A SPREADSHEET

TUTORIAL

In this tutorial we will structure a simple spreadsheet to illustrate how labels and numbers are entered. In the following chapters we will use this spreadsheet to practice constructing and replicating formulas, inserting and deleting lines and columns, and moving and copying data.

A spreadsheet is an interlocking mechanism of formulas which express relationships among and between the numeric data that is supplied to it. Although you may have some general notion of the kind of spreadsheet you want, it may be difficult to determine at first just what kinds of data are to be included, and what formulas must be created to manipulate that data. Often it will take a good bit of 'fiddling' with the format, adding and deleting elements, altering formulas, moving and copying lines here and there, before you arrive at a spreadsheet that will do what you want it to do.

After you have decided upon a general spreadsheet structure, placing the labels on the spreadsheet will typically be the first step in building the spreadsheet. Labels on a spreadsheet are nothing more than tags which visually identify the various numeric elements and formulas that you are using. When you place labels on a spreadsheet you are actually 'formatting' the screen, breaking it into logical, coherent, and aesthetically pleasing segments, making it not only pleasing to view but easier to work with than it certainly would be otherwise.

The spreadsheet we will construct in this tutorial will be a simple household budget analysis:

	a		b		c		...m	
1	-----	OUR 1983 BUDGET	-----					
2								
3	Month:	Jan.		Feb.		...Dec.		
4	---	---		---		---		
5	Income:	2000.00						
6								
7								
8	-----	Expenses	-----					
9	Taxes:							
10	Rent:							
11	Food:							
12	Misc. Exp.:							
13								
14	-----	Savings	-----					
15	Regular:							
16	Disposable:							
17	% Saved:							

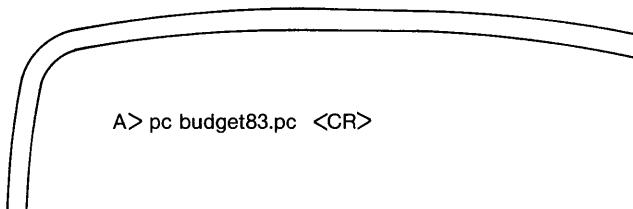
As you can see, this simple budget takes an initial numeric value (Income), and from it computes several other figures: Taxes, Regular and Disposable Savings, and '%' Saved. Included in the computations are several other expenses: Rent, Food, and Miscellaneous.

We will structure this spreadsheet so that a change in 'Income' will automatically result in changes for all variables which are dependent upon Income (i.e. Taxes, Savings, % Saved).

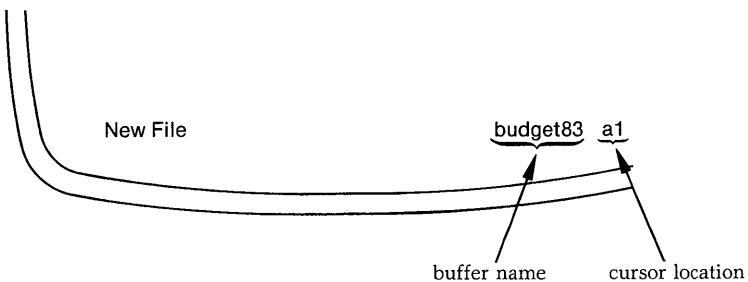
The first step is to place the labels which will identify the various lines and columns.

Steps:

1. Turn on your computer and insert your Perfect Calc program diskette into drive 'A'. Following the operating system command prompt, type the letters which invoke Perfect Calc, followed by the name of the new file ('budget83.pc'):



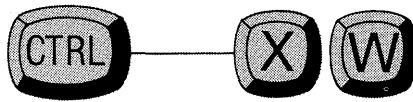
Perfect Calc presents a new blank spreadsheet. In the Prompt Line at the bottom of the screen are the words:



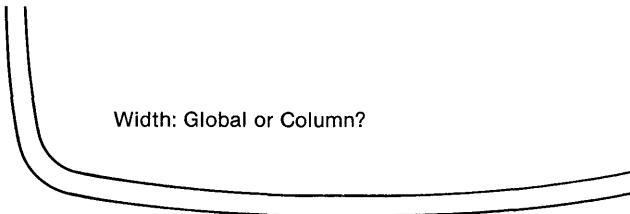
The message 'New File' will disappear as soon as we begin working on the spreadsheet. 'Budget83' is the buffer name¹, while 'a1' are the coordinates of the cursor's position on the spreadsheet, which will change as you begin moving around the spreadsheet.

¹ 'Buffers' are discussed in Chapter XI.

2. Begin by entering the labels that go into column 'a': Month, Income, Taxes, Rent, etc. Before doing this however, change the width of this column so that it will comfortably accommodate these labels. With the cursor positioned anywhere in column 'a' type the CHANGE COLUMN WIDTH command:



Perfect Calc responds with the message in the Prompt Line:



Answer 'c' for 'column'.

Perfect Calc echoes this response, and asks for the new column width:



Enter a column width of 17 characters. Type '17' followed by a carriage return, <CR>.

Perfect Calc redraws the screen, enlarging Column 'a' to 17 character spaces wide.

3. Move the cursor to line '3' of column 'a'. Begin typing the label 'Month:'. As you type the first character of this label Perfect Calc will present the message in the Prompt Line:



The label Month: will initially appear on the screen following this message. If you make a mistake while typing, use the delete key, or rubout to correct the error. Terminate the label with a carriage return, <CR>, or down motion command.

Perfect Calc immediately transfers the label to the position occupied by the cursor, positioning it against the left margin of the entry box.

4. Move the cursor to line 5 (column 'a') and type the second label, Income. Except this time **precede** the label with **two** blank spaces (type the space bar twice before typing the label). Again, terminate the label with a carriage return, <CR>.

Perfect Calc transfers the label to the spreadsheet, **indenting** it two spaces from the left margin of its entry box. 'Padding' with blankspaces is a convenient way of positioning a label, and sometimes makes the label more readable and attractive. (We will correct 'Month:' later.)

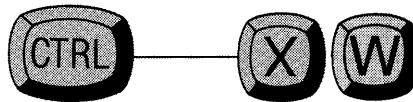
5. Move the cursor to line '9' and enter the label 'Taxes' in the same way. However, instead of terminating the label with a carriage return <CR>, type the 'down arrow' key (or the NEXT LINE command (Control----N).

Perfect Calc transfers the label to the spreadsheet, afterwards moving the cursor to the next line. As mentioned earlier in this chapter, a data entry can be terminated either with a carriage return, <CR>, or with any command key, which will perform the requested action after entering the data. This eliminates the need for a cumbersome two-step command procedure of first entering the data and then moving the cursor.

In this manner continue entering the remaining labels in their respective positions in column 'a'.

6. Move the cursor to line 3 (position 'b3') in preparation for entering the names of the months.

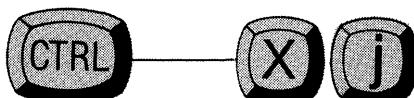
The default setting for every column in the spreadsheet, except column 'a', is '9' characters. Although this is enough room to hold the largest month, September, it will probably result in an overcrowded appearance for the spreadsheet. Widen all of these columns. Give the CHANGE COLUMN WIDTH command again:



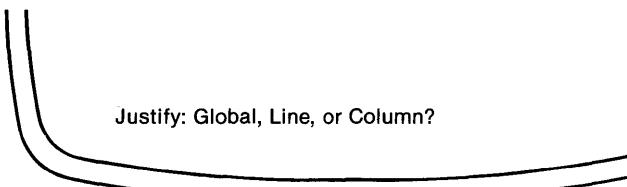
This time answer 'g' (for 'global'), followed by the number '11'.

Perfect Calc immediately redraws the screen, enlarging all spreadsheet columns to 11 characters. (Have any columns been pushed to off-screen positions?) Also note that column 'a' has not changed.

7. Now, the name of each month will head its own column, and we know that Perfect Calc will (by default) initially position each name against the left margin of its entry box. It would be nice, however, if we could center the names in their columns. Padding with blanks is a way to do this, but it would be tedious, involving counting and subtracting characters, etc. Instead, use the JUSTIFY ENTRY command, which positions the names of the months automatically. With the cursor positioned anywhere in line 3, type:



Perfect Calc responds with the message:



Answer 'l' (small letter 'L') to indicate that this will apply to every entry in line '3'.

Perfect Calc then responds:



Hit the question mark (?) and Perfect Calc presents the choices available (Left, Right, Center or <CR>) for default of left justification for labels and right justification for numbers). Select 'c' for center.

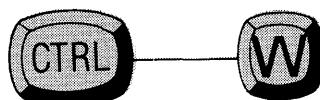
Perfect Calc immediately redraws the screen, repositioning the first label 'Month:' in the center of its column.

8. Begin entering the names of the months into line 3, columns 'b' to 'm'. Notice that as you enter these labels each is automatically centered in its column.
9. Move the cursor to the beginning of line '4' (entry box 'a4'). Line '4' will contain short lines of hyphens separating the column headings from the numeric entries below them. Normally a hyphen signals a negative number. When entering these hyphens as a label you will have to 'force' Perfect Calc to accept them by first typing a double quote: (").

One way to enter these lines is to move across line 4, from one entry box to another, entering the hyphens as you go—again, a tedious thing to do. Perfect Calc has a better way, called 'copying'. Although it is not formally discussed until Chapter VIII, the steps to this very easy and convenient command procedure are presented here in order to allow you to start using it as soon as possible.

Copying a Label

10. With the cursor positioned at 'a4' type a line nine hyphens long. (Remember to precede this label with the double quote ["].)
11. Instead of a carriage return, enter this label using the COPY ENTRY command:

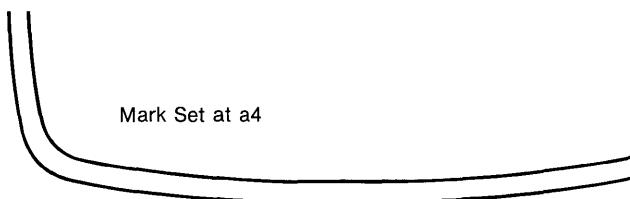


Perfect Calc enters the label into position 'a4', simultaneously 'remembering' the label for subsequent copying.

12. With the cursor still at position 'a4' type the MARK SET command:



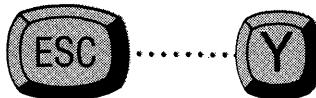
Perfect Calc responds with the message:



This means that an invisible mark has been set at entry position 'a4', marking the beginning of the 'region' over which the label of hyphens will be copied.

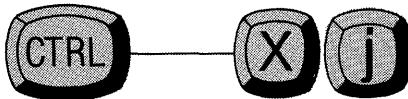
13. Move the cursor to the end of line 4 using the END OF LINE command (Control-----E). (This automatically defines the end of the region over which the label will be copied.) The end of the current region is the farthest column to the right in which an entry has been made. In our example that is column 'm' where you entered the label 'December'.

14. Copy the row of hyphens to every entry box between entry 'a4' and the cursor by typing the YANKBACK command:



Perfect Calc immediately copies the row of hyphens to the entry boxes below the names of the months. How does it look? All right? Or is something wrong?

Actually, if everything is working as it should, the hyphens will **not** be centered in their columns, but justified against the left margin of their entry boxes. To center them beneath the names of the months, give the JUSTIFY ENTRY command:



answering 'l' for 'line' and 'c' for 'center'. Perfect Calc automatically centers the hyphen labels.

Only the long lines that extend across the entire width of the spreadsheet, and which hold the labels OUR 1983 BUDGET, Expenses, and Savings, remain to be entered. Enter these lines using the same procedure as before, by copying column-wide lines of hyphens to every entry box in the line. Again, the steps:

15. Postion the cursor at the beginning of line 8, position 'a8', and enter a label 17 hyphens long (Again, remember to force the label with a double quote ['']).
16. Enter this label of hyphens using the COPY ENTRY command (Control----W).

Perfect Calc enters the hyphens into the first column, simultaneously remembering it for copying to other entry positions.

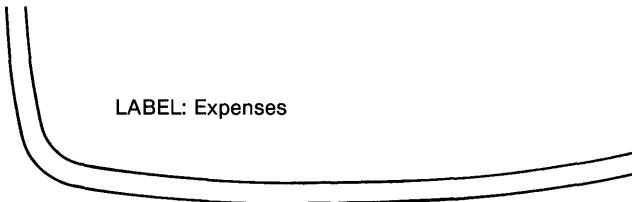
17. Set the invisible boundary mark at position 'a8' by typing the MARK SET Command (Escape.....space bar).

18. Move the cursor to the end of line 8 using the END OF LINE command (Control-----E).
19. Copy the hyphens across the entire region by giving the YANKBACK command (Escape Y).

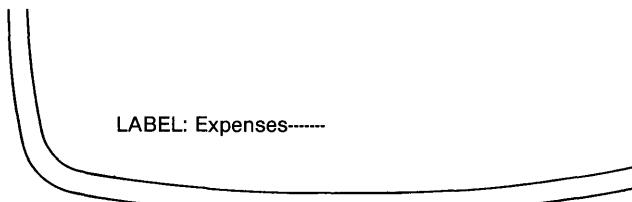
Perfect Calc copies the line of 17 hyphens into every entry box in the region, completely filling each one and giving the appearance of a continuous line across the spreadsheet. (Note that 17 hyphens are actually present in each entry box, except that, in all but column 'a', only 11 of these are actually being displayed.)

20. We must now insert the word 'Expenses' into this line. Move the cursor to position 'd8'. We are going to change the label in this box from a row of hyphens to 'Expenses'. Type the word 'Expenses'.

Perfect Calc accepts the word as a new label displaying it in the Prompt Line:



Transfer it to position 'd8' with a carriage return, <CR>. Does it look all right? No? Are some hyphens missing that break the line? If so, type the label again, adding some hyphens to fill up the line:



This should give the appearance of the word solidly embedded in a continuous row of hyphens.

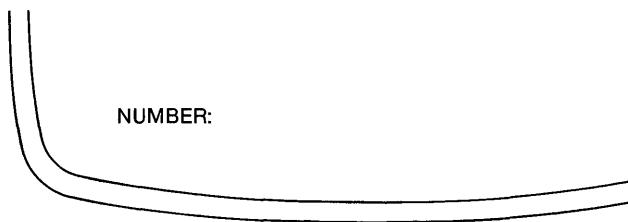
21. Move the cursor to line 14 and enter another row of hyphens, using the same steps as above, afterwards inserting the word Savings at entry box 'd14'.
22. Move the cursor to line 1 and enter the entire title:

"-----OUR 1983 BUDGET-----

23. This completes the spreadsheet. It should look like the illustration at the beginning of this tutorial. Move the cursor around on it to see that the off-screen positions are as they should be.
24. For the final exercise, enter a few values to see how numbers will display. Move the cursor to entry box 'b5' and type any number, followed by a carriage return, <CR>. For example, type:

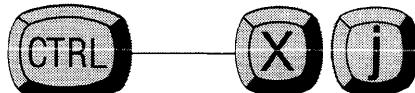
2000 <CR>)

At the first digit, Perfect Calc recognizes that you are entering a number. It responds by displaying the message in the Prompt Line:



As with a label, the number appears on the screen following this message. Only when you type a carriage return, <CR>, or a command key, is the number transferred to the spreadsheet at the location of the cursor.

Notice that the number is right justified in its entry box (i.e. it ends against the right margin of the box). As with labels, this position can be changed by using the JUSTIFY ENTRY command

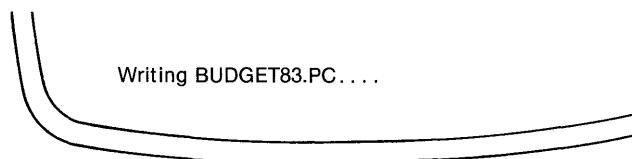


Also, notice that the number is displayed showing two decimal places: 2000.00. This is the default display notation which Perfect Calc will use unless told otherwise. (In Chapter IX you will learn how to change it.)

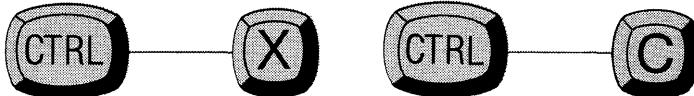
This ends the tutorial on entering labels and numbers. You will be using this spreadsheet in subsequent tutorials, and therefore, before exiting the system, save it to disk by typing the SAVE FILE command:



Perfect Calc responds by copying the spreadsheet to a disk file, displaying the message in the Prompt Line:



You may now exit Perfect Calc. Type the QUIT command:



Perfect Calc returns you to your operating system.

END OF TUTORIAL

Chapter VI

ENTERING FORMULAS

Formulas will probably be your most important type of spreadsheet entry. Formulas instruct Perfect Calc to perform a variety of arithmetic and logical operations using the numeric data which you enter, and to display the results of these operations on the spreadsheet.

Formulas themselves are not normally displayed on the spreadsheet, only the values which they compute. A formula will display only in the Prompt Line and then only when the cursor is occupying the entry box that 'contains' it.¹

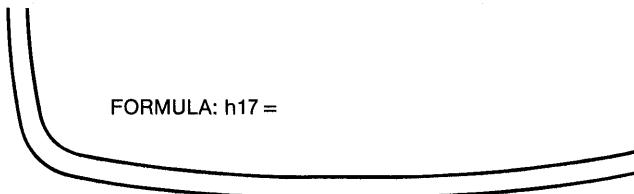
In the following pages we will introduce some very powerful tools (formulas and functions) for analyzing numbers. Some of these may be unfamiliar to you. Do not let this bother you. If you are not familiar with a particular formula or function, you will probably never have occasion to use it anyway. Many of the trigonometric functions, for example, are useful only in highly specialized applications. Know that these tools are here when you need them, but do not try to understand all about them solely by reading this User's Guide.

¹ Perfect Calc allows you to display formulas at their entry positions. For this command procedure, see Chapter IX, page 190.

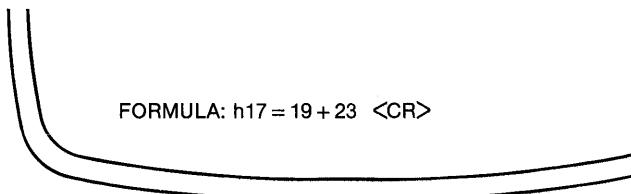
Entering Formulas

To enter a formula, position the cursor at the desired entry position and type an 'equals' character, "=" . Perfect Calc will display the word "FORMULA" in the Prompt Line, followed by a pair of coordinates indicating the present position of the cursor.

For example, if the cursor is at entry position 'h17' when the equals character is typed, the message will read:



This means that Perfect Calc is ready to accept a formula for entry position 'h17'. At this point you may enter whatever formula you wish. For example, you might enter:



Terminate the formula using a carriage return, <CR>, or any command (which will perform the requested action after terminating and 'entering' the formula).

Perfect Calc scans the formula that you enter checking for incorrect or inappropriate characters, syntax, operators, etc., afterwards entering the formula into the entry position indicated by the coordinates. (Remember that the formula itself will not be displayed at the entry position, only the value which it computes.)

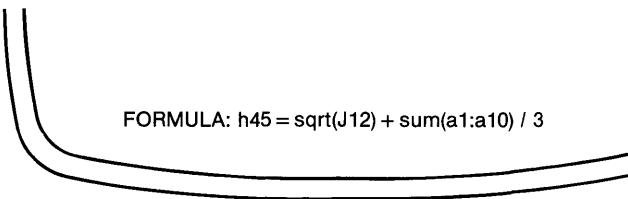
For the above example the sum '42' will be computed and entered into entry position 'h17'. If the formula could not be scanned correctly because it was improperly constructed, (you mistakenly entered "19&23", or "19+23", using a lower-case 'L' instead of the digit '1'), or if the value resulting from computing it were, for some reason, erroneous, Perfect Calc would beep and report 'Bad Formula'. However, it will accept the formula, so that you can edit the erroneous portion rather than retype the entire formula.

What's in a Formula?

Formulas in Perfect Calc are the same common arithmetic expressions that most everyone is familiar with, containing: numbers, arithmetic operators, variables, and predefined functions.

When constructed using common rules of arithmetic, a formula will result in a single numeric value for a spreadsheet entry.

A simple formula might look like this:



This formula instructs Perfect Calc to add one-third of the sum of the values contained in positions 'a1' through 'a10' to the squareroot of the value in position 'J12', and to display the result in position 'h45'.

Blanks embedded in a formula are ignored and may be inserted for readability. However, since formulas are limited in length to the 76 characters available on the Prompt Line, complicated formulas may necessitate compacting the characters as much as possible. (For extremely long formulas it may be necessary to break the formula up into separate parts, each part located in a separate entry box, but linked to one another with appropriate variable references.)

Numbers in a Formula

When numbers are used in a formula, they are identical to the raw numbers used elsewhere in the Perfect Calc spreadsheet. Numbers can be used in formulas in either integer form (e.g. 576897) or scientific notation (e.g. 5.76897e5). All of the rules for constructing numbers must be observed when using numbers in formulas (see Chapter V, page 75).

Formula Operators

'Operators' are those symbols for addition, subtraction, multiplication, etc. that specify the arithmetic relationships existing between the numbers, variables, and functions of a formula. The following arithmetic and relational operators are available in Perfect Calc:

Arithmetic Operators

	Action
\wedge (Up-arrow)	Raises a number to the power indicated (e.g., $3^2=9$). In the expression x^y when $x < 0$ and $y < 1$, y is internally rounded to the nearest integer.
*	Multiplication (e.g., $3*2=6$)
/	Division (e.g., $3/2=1.5$)
+	Addition (e.g., $3+2=5$)
-	Subtraction (e.g., $3-2=1$)

Relational Operators

	Action
<	Less than
\leq	Less than or equal to
=	Equal to
\neq	Not equal to
\geq	Greater than or equal to
>	Greater than

Note: Relational operators return a value of '1' to indicate that the relation is 'true', and a value of '0' (zero) to indicate that it is 'false'.

For example, when computing the formula $j23=h17\geq=100$, Perfect Calc will display in position 'j23' a value of '1' if the value in position 'h17' is greater than or equal to 100. Otherwise it will display '0'.

Operator Precedence

The arithmetic and logical operators in the previous tables are arranged in order of 'precedence'. That is, in any arithmetic expression those operators toward the top of the table are performed before those below. The expression $1 + 2 * 3$ yields a value of 7 rather than 9 since multiplication takes precedence over (is performed before) addition.

Parentheses may be used to alter the order of operations within a formula. For example, $(1 + 2) * 3$ gives the value of 9 rather than 7, since operations enclosed within parentheses are performed before other arithmetic operations.

Parentheses may be 'nested' to any depth. When parentheses are nested, the operation contained in the innermost parentheses is performed first, followed by the next innermost, and the next, etc., with the expression contained within the outermost parentheses being performed **last**. For example, the expression:

$$(((1 + 2) * 4) / 3) - 5$$

becomes

$$((3 * 4) / 3) - 5$$

then

$$(12 / 3) - 5$$

then

$$4 - 5$$

and finally

$$-1$$


(The Answer)

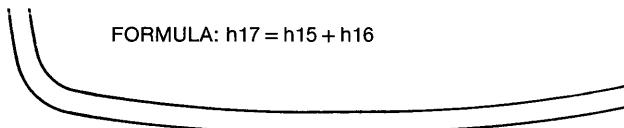
When nesting arithmetic expressions within parentheses, it is easy to become confused and omit a left or right parenthesis. As a general rule of thumb, all expressions should contain a matching number of left and right parenthesis:

$$(((1 + 2) * 4) / 3) - 5$$

3 here 3 here

Variables

In Perfect Calc variables are simply references to other entry positions in the current spreadsheet, and are expressed as the 'coordinates' of these positions.² For example, the formula:



instructs Perfect Calc to compute the sum of whatever values are located in entry positions 'h15' and 'h16' and to display that sum in entry position 'h17'. If the numbers '19' and '23' have been entered into 'h15' and 'h16', the sum displayed in 'h17' will be '42'.

A screenshot of a Perfect Calc spreadsheet. The columns are labeled f, g, h, i, j, k. The rows are numbered 14 through 18. Row 15 contains 'First' and '19.00'. Row 16 contains 'Second' and '23.00'. Row 17 contains 'Result' and '42.00'. An arrow points from the formula $h17 = h15 + h16$ to the cell containing '42.00'.

	f	g	h	i	j	k
14						
15		First	19.00			
16		Second	23.00			
17		Result	<u>42.00</u>			
18						

formula:
 $h17 = h15 + h16$

Variables thus allow us to express formulas in general rather than specific terms. The usefulness of this is apparent when you wish to alter, or 'vary', the numeric data. It would defeat the very purpose of Perfect Calc if you always had to insert raw numeric data directly into the formula, for then you would have to rewrite the formulas each time you changed a single piece of data.

With variables there is no need for this. Change whatever entry values you wish, and Perfect Calc, knowing the formula in general terms, substitutes the new values for the old and recalculates the answer.

For example, if you change the value contained in position 'h15', from '19' to '21', Perfect Calc **automatically** recalculates the sum displayed in position 'h17' to '44'.

² It is possible for a formula in one spreadsheet to reference a variable in another spreadsheet. For this command procedure, see Chapter XI.

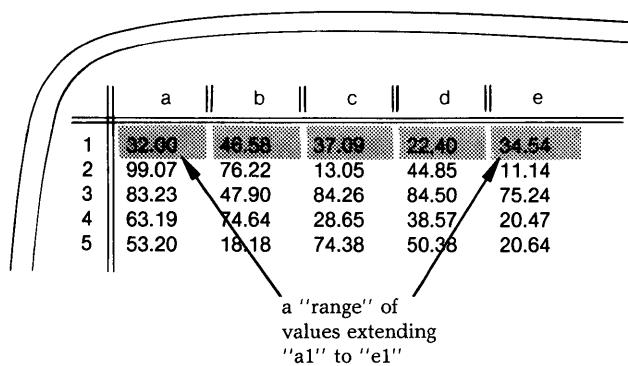
FUNCTIONS

'Functions' in Perfect Calc can be thought of as prepackaged formulas that perform common and frequently used mathematical computations, and which would be difficult and/or time consuming to compute on your own.

Functions are similar to operators in that they take variables and numbers as input and produce a single value as a result. Functions have a higher precedence than any arithmetic operator, so that, for example, 'sqrt(x)/2' will first compute the square root of x, before dividing by 2.

Range of Values

While some of Perfect Calc's functions take a single input value, others take two values, and still others take a 'range' of values. A range is simply any linear sequence of spreadsheet values, either horizontal across a line, or vertical down a column. In addition, a range may be a rectangular array.



The diagram shows a 5x5 grid of numbers. The columns are labeled 'a', 'b', 'c', 'd', and 'e'. The rows are labeled 1 through 5. The grid contains the following values:

	a		b		c		d		e
1	32.00		46.56		37.89		22.40		34.54
2	99.07		76.22		13.05		44.85		11.14
3	83.23		47.90		84.26		84.50		75.24
4	63.19		74.64		28.65		38.57		20.47
5	53.20		18.18		74.38		50.36		20.64

A curved line starts from the top-left corner of the grid and ends at the bottom-right corner. Two arrows point from the text 'a "range" of values extending "a1" to "e1"' to the first cell of the second row ('99.07') and the last cell of the fifth row ('20.64').

a "range" of values extending "a1" to "e1"

The method for expressing a range is:

(starting position:ending position)

A range consisting of the first 10 entry positions down the first column of the spreadsheet would be expressed as:

$(a1:a10)$

To compute the sum of the first 10 values down the first column we would use the 'sum' function, writing it:

$\text{sum}(a1:a10)$

Ranged functions will operate over a region of values. For example, it is possible to calculate the sum of all values in a rectangular region, as in:

$\text{sum}(a1:c5)$

	a	b	c
1	32.00	46.58	37.09
2	99.07	76.22	13.05
3	63.23	47.90	84.26
4	63.19	74.64	28.05
5	53.20	18.16	74.35

Labels and non-filled entry positions are ignored when values over a range are being computed.

Perfect Calc's Functions

The functions available in Perfect Calc are presented below. The 'arguments' to functions are enclosed in parentheses and follow directly after the function names. No blank spaces are permitted between the function name and the first parenthesis.³

Arithmetic Functions	Action
abs(x)	Computes the absolute value of 'x'.
avg(range)	Computes the average value of a range of values. The average or 'mean' is equal to the sum of a range of values divided by the number of values in the range.
count(range)	Computes the number of entry positions within a range filled with numeric values (e.g. formulas or numbers).
exp(x)	Computes e^x , where x is a number or variable, and 'e' is the base of the natural logarithm (2.7182818...). Here, 'x' must be less than approximately 230. This is the inverse of the natural logarithm function.
int(x)	Computes the largest integer not exceeding x . (The absolute value of x must be less than 32,768.)
ln(x)	Computes the natural logarithm (in base e) of x . (X must be non-negative.)
log(x)	Computes the log (in base 10) of x .
lookup(x, range)	Searches for 'x' in the range specified, and returns the value held in the entry position adjacent to 'x'.

³ The designations 'x', 'y', 'z', and 'range' are used here to represent the 'arguments' of the functions. In fact, the functions take numbers, variables (expressed as entry position coordinates), or arithmetic expressions for their arguments.

max(range)	Identifies the maximum value among a range of values
min(range)	Identifies the minimum value among a range of values.
npv(x, range)	Computes the 'net present value' across the cash flow range specified, discounted at a rate specified by 'x'.
sqrt(x)	Computes the square root of x. (X must be non-negative.)
sum(range)	Computes the sum of a range of values.

Logical Functions and(x,y)

or(x,y)

Action

Returns 'true' (the value 1) if **both** x and y are 'true' (i.e., x and y are both non-zero). Otherwise returns 'false' (the value 0).

if(x,y,z)

Returns 'true' (the value 1) if **either** x or y is 'true' (i.e., either x or y is non-zero). Otherwise returns 'false' (the value 0).

not(x)

Returns y, if x is 'true' (i.e. non-zero); otherwise returns z.

Returns 'true' (the value 1) if x is 'false' (i.e., 0). Otherwise returns 'false' (the value 0).

Trigonometric Functions**atan(x)****Action**Computes the arc tangent (inverse tangent) of x.
(X must be in radians.)**cos(x)**

Computes the cosine of x. (X must be in radians.)

sin(x)

Computes the sine of x. (X must be in radians.)

FORMULAS FOR REMAINING TRIGONOMETRIC FUNCTIONS

$$\text{arc cosine}(x) = 3.1415926535979 - \text{atan}(x/\sqrt{1 - x^2})$$

$$\text{arc sine}(x) = \text{atan}(x/\sqrt{1 - x^2})$$

$$\tan(x) = 1/\text{atan}(x)$$

Note: The last three remaining trigonometric functions are not built-in functions in Perfect Calc. For each you must insert the formula as shown. For example, to solve for ' $\tan(x)$ ', enter the formula ' $1/\text{atan}(x)$ '.

Use of the Lookup Function

The **lookup** function is used to search for a value x in a table (or range) and returns the value of the entry box adjacent to x . If the range is a column, then the value returned will be the one located in the adjacent entry box in the columns to the right. For example, the formula:

FORMULA: $c1 = \text{lookup}(15, a1:a5)$

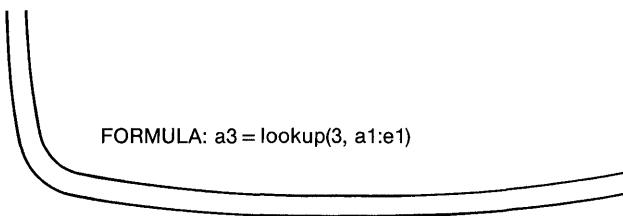
for the following spreadsheet:

	a		b		c
1	5.00		10.00		30.00
2	10.00		20.00		
3	15.00		30.00		
4	20.00		40.00		
5	25.00		50.00		

result

would return the value 30 in entry box c1.

If the range is a line, then the value returned will be located in the adjacent cell below the specified range. For instance, the formula:



for the following spreadsheet table:

	a		b		c		d		e
1	1.00		2.00		3.00		4.00		5.00
2	5.00		10.00		15.00		20.00		25.00
3	15.00								

would return the value 15 to entry box 'a3'.

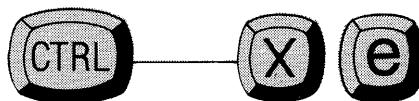
Editing a Formula

Sometimes, after a formula has been entered, it needs changing, either because it was incorrect, or because the spreadsheet has changed in some way and the formula now needs altering (perhaps slightly) to accommodate this change. If the formula is complicated, it would be a cumbersome procedure to first write down the formula on paper and then enter it again in its new form.

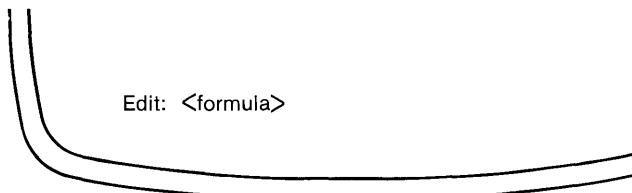
Perfect Calc provides a simple command procedure for 'editing' a formula (or label, for that matter), allowing us to make changes in the formula without having to retype the formula from scratch.

Steps:

1. Position the cursor on the entry box containing the formula to be edited. (The formula will appear in the Prompt Line.)
2. Type the EDIT ENTRY command:



Perfect Calc enters a special editing mode, displaying the message "Edit:" in the Prompt Line, followed by the formula to be edited:



In this editing mode, the cursor is displayed as a blinking underscore on the Prompt Line. Also, certain of Perfect Calc's cursor and deletion commands have been altered to allow you to edit the formula. These command changes are as follows:⁴



Moves the cursor to the BEGINNING of the Prompt Line.



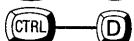
Moves the cursor to the END of the Prompt Line.



Moves cursor FORWARD one character.



Moves cursor BACKWARD one character.



DELETES the character occupied by the cursor.



Deletes characters backward.



Cancels the editing process, returning the formula to the spreadsheet in its original form.

Carriage return.

Ends the editing process and returns the altered formula to the spreadsheet.

Illegal Values

Illegal values passed to a function will cause the function to return an 'Error!' value, which will be displayed wherever the illegal value is used or referenced. Such errors may be corrected by changing the input values to the function, or by changing the function definition itself.

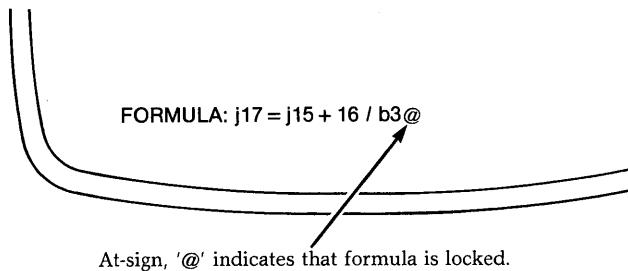
⁴ Note that these editing commands are identical to those used by Perfect Writer.

Locking Formulas

As a safeguard against accidental deletion of a formula, or formulas, Perfect Calc provides a means of 'locking' formulas in place, either singly or throughout a region. Once locked into an entry position, formulas may not be moved, changed, edited, or deleted, until they have been 'unlocked'.

Locking a formula does not affect Perfect Calc's use of the formula in calculating values—i.e., the formula's 'value' is free to change, only its definition is locked in place.

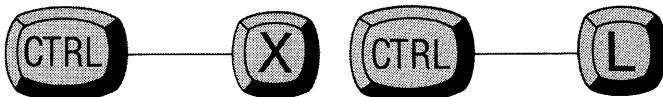
Formulas which have been locked into place can be identified by an 'at-sign', '@', which Perfect Calc appends to them. For example, the following formula has been locked into place:



The commands for locking and unlocking formulas are as follows:

The LOCK CURRENT FORMULA Command

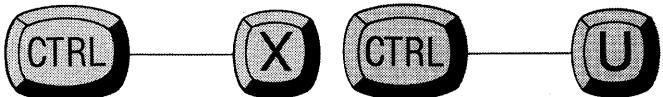
Locks the formula which the cursor is currently occupying:



Perfect Calc locks the current formula, displaying the message "Locked" in the Prompt Line.

The UNLOCK CURRENT FORMULA Command

Unlocks the formula which the cursor is currently occupying.



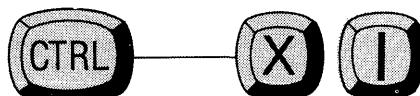
Perfect Calc unlocks the current formula, displaying the message "Unlocked" in the Prompt Line. (The 'at-sign', '@', has been removed from the end of the formula.)

The LOCK FORMULAS IN REGION Command

Locks the formulas in a predefined region.

Define the region in which the formulas will be locked, using the Mark Boundaries Procedure (see Chapter VII, page 13).

Type the LOCK FORMULAS command:



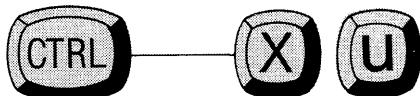
Perfect Calc locks all formulas in the defined region, displaying the message "Locked" in the Prompt Line. The '@' sign is appended to the end of each formula.

The UNLOCK FORMULAS IN REGION Command

Unlocks the formulas in a predefined region.

Define the region in which the formulas should be unlocked, using the Mark Boundaries Procedure (see Chapter VII, page 147).

Type the UNLOCK FORMULAS IN REGION command:



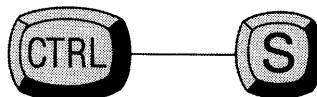
Perfect Calc unlocks the formulas in the defined region, displaying the message "Unlocked" in the Prompt Line. The '@' sign is removed from each of the formulas.

The SEARCH NEXT UNLOCKED ENTRY Command

It is sometimes useful to be able to identify those entries displaying numeric values which are not locked.

For example, suppose that you are entering numeric data into a large spreadsheet containing many formulas, and in which it is not immediately evident which entry positions contain formulas and which are actually reserved for the numeric data you are entering. It would be tedious and time-consuming to hunt through the spreadsheet for the positions where the data needs to be entered, always having to take caution that you do not inadvertently overwrite a formula, which on the screen looks just like a numeric entry.

The SEARCH NEXT UNLOCKED ENTRY command searches in a line-wise direction from the current position of the cursor to the next unlocked entry containing a numeric value, either formula or number. Since numbers cannot be locked, the solution to the problem lies in locking all the formulas in the spreadsheet, (using the command procedure presented previously), and then, by invoking the SEARCH NEXT UNLOCKED ENTRY command, searching for the various numeric entries which remain unlocked and which can only be intended for the data that you are entering. This command is entered by:



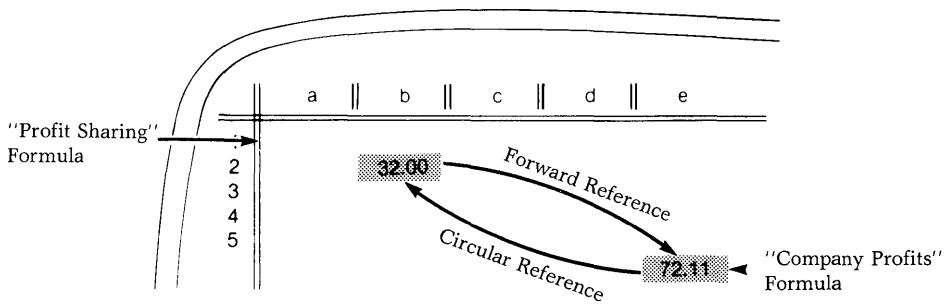
Perfect Calc, searching from line to line, positions the cursor at the next unlocked entry position which displays a numeric value. To continue entering data into unlocked positions, simply continue to type 'Control----S' and Perfect Calc will lead you through the spreadsheet.

Recalculation of Formulas

Whenever a number or formula is changed, Perfect Calc automatically recalculates the values of all relevant spreadsheet formulas. This recalculation occurs from left to right across the spreadsheet, starting at position 'a1', and continuing line-by-line to the last formula on the spreadsheet.⁵

It is important to consider this sequential nature of updating when constructing a spreadsheet with many interrelated formula values. 'Circular' and 'forward' formula references should be avoided. That is, a formula should not reference an entry that will be updated **after** itself.

For example, consider a formula in a spreadsheet that computes employee profit sharing, and in so doing, references a later formula which computes the company's final profits. However, the later formula in turn references the earlier profit sharing formula in order to compute the final profits. During a recalculation, 'final profits' would first be referenced in computing 'profit sharing', as intended, but then later would itself be changed. Both formula values would be faulty, even though some value would have been computed for each, with everything appearing normal and correct to a casual inspection.



⁵ The direction of this recalculation may be changed using the command procedure described on Chapter IX, page 198.

The caution here is to ensure that your formulas always reference spreadsheet entries which will be updated **before** they themselves will—i.e., use ‘backward’ references only. Typically, **this means that your formulas should reference only formula entries which are above and to the left of the current entry.**

Should any formula produce an error as its value, any other formula which references that entry will also produce an error. However, correcting the original formula will serve to correct all referencing formulas as well.

Replicating a Formula

In creating a spreadsheet it will often happen that many formulas will be structured identically, except that their variables will reflect varying line or column positions.

For example, suppose that you are structuring the following simple spreadsheet, which computes your taxes as a percentage of your monthly income. Here, your taxes are figured at 20 percent of your gross monthly income.

	a		b		c		d		...m
1			Jan.		Feb.		March		...Dec.
2	Income:		2000		2000		2000		...2000
3	Taxes:		400		400		400		... 400
4									
5									

The **only** non-formula entry in this spreadsheet, besides labels, is entry position 'b2', which holds a single numeric value. Every other position will contain a relative formula that will reference this position.

If you displayed this spreadsheet so that its formulas were visible at their entry position⁶, it would look like this:

	a		b		c		...m
1			Jan.		Feb.		...Dec.
2	Income:		2000	= b ₂			... = l ₂
3	Taxes:		= b ₂ * 0.20	= c ₂ * 0.20			... = m ₂ * 0.20
4							
5							

⁶ See Chapter IX, page 190 for the procedure for displaying formulas at their entry positions.

Notice that entry position 'c2' takes its value from whatever is entered into 'b2', and position 'd2' from whatever is in 'c2', etc. Similarly, the formulas in entry positions 'c3', 'd3', 'e3', etc., which compute the taxes for February, March, April . . . , are identical to the formula in 'b3', except that their variables have been changed to reflect their positions in columns 'c', 'd', 'e' . . . etc.

This is a typical way of structuring consecutive formulas in a spreadsheet. The advantage is that should your income increase at some time in the middle of the year, say, March, entering a new monthly income at that point will cause all entries to the right (from April through December) to change, while leaving the months January and February, unchanged.

It would be a tedious chore to enter the same formula, varied slightly, into these adjacent entry positions (23 in all, for this example). Fortunately, Perfect Calc provides an easy method of doing this through a command procedure called 'replication', which utilizes a form of the YANKBACK command.⁷ The steps for replicating a formula across a region of the spreadsheet are presented here. Some of the command procedures for doing this have not yet been discussed in the manual. Where this is the case, chapter and page numbers are noted for your easy reference.

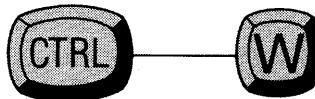
⁷ For a complete discussion of the YANKBACK command, see Chapter VIII.

Procedure for Replicating a Formula

To replicate a formula across a linear or rectangular region of a spreadsheet, proceed as follows:

Steps:

1. Position the cursor at the **first** entry position in which the formula will occur.
2. Type the formula as it will appear at this position.
3. Terminate the entry with the COPY ENTRY command:⁸



This command will both enter the formula into the spreadsheet **and** copy it to a temporary location in memory called the 'Save Buffer'⁹. Once in the Save Buffer, the formula can be copied to other locations in the spreadsheet.

4. Set a boundary mark at the **first** entry position by typing the SET MARK command:¹⁰

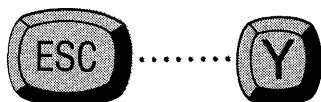


⁸See Chapter VIII.

⁹ See Chapter VIII.

¹⁰ See Chapter VIII.

5. Move the cursor to the **last** position in the region, thus defining the region (see Mark Boundaries Procedure, Chapter VII, page 147).
6. Replicate the formula across the region by typing the YANKBACK command form:



Perfect Calc displays the formula in the Prompt Line, the first variable highlighted, followed by the question: "...Relative?"¹¹

7. If the variable that is highlighted must be changed to reflect its position in the various columns, answer 'yes'. If not, answer 'no'.

Perfect Calc immediately replicates the formula across the defined region, changing or maintaining the variables as you have specified.

¹¹ See Chapter VIII for a discussion of 'Relative vs. Absolute Variables'.

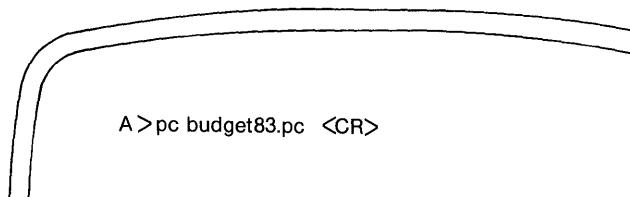
ENTERING FORMULAS

TUTORIAL

In this tutorial you will practice entering formulas, using the spreadsheet created in Tutorial 2: 'budget83.pc'.

Formulas will be your most common and valuable spreadsheet entry. Indeed, the quality of your spreadsheet will be largely determined by the quality and structure of your formulas. The better the formulas perform, the more valuable the spreadsheet. In this tutorial you will enter several simple formulas, illustrating the strategy to be followed in structuring them, and how they are replicated across regions of the spreadsheet.

Begin Perfect Calc by calling up the spreadsheet created in Tutorial 2:



Perfect Calc retrieves the file from disk and displays it to your screen. As you recall it looks like this:

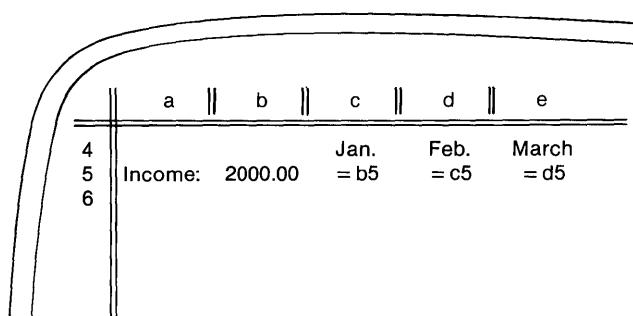
	a		b		c		...m	
1								
2								
3	Month:		January	February	...December			
4								
5	Income:		2000.00					
6								
7								
8								
9								
10								
11								
12								
13								
14								
15								
16								
17								

Although you may have entered several numbers into the spreadsheet during the previous tutorial, most of the entries on this spreadsheet will not contain numbers, but formulas. In fact, the **only** numeric entries in the entire spreadsheet will be in positions 'b5' (the initial entry for Income), 'b10' (the initial entry for Rent), 'b11' to 'm11' (the monthly entries for Food), and 'b12' to 'm12' (the monthly entries for Miscellaneous Expenses.) Every other entry will be represented by a formula. Begin by entering the formulas.

Steps:

1. Move the cursor to line '5', the Income line, positioning it at entry box 'b5'. If you have not already done so, enter a number value in this box to represent an initial monthly income (e.g. 2000), afterwards moving the cursor to the entry box in the next column: 'c5'.

2. You want to structure the entries in the spreadsheet in such a way that they reference one another. That is, the Income value displayed in February should be determined by whatever value is displayed in January, and the value in March by what is held in February , and the April value by the March value, etc. That way, should you get a raise at some time during the middle of the year, say June, the monthly income values for July through December will be updated automatically, while those from January to May will remain unchanged.



As the illustration shows, every entry on the Income line will possess a formula that is identical in structure to the one preceding it, except that for each the variable will be changed to reflect its particular entry position.

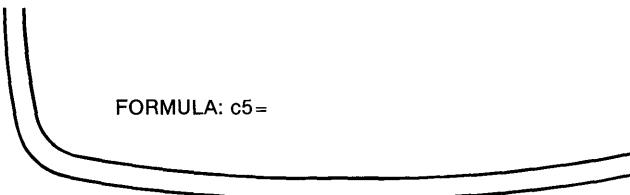
Such a formula variable is called a 'relative' variable, because it changes according to its location on the spreadsheet.¹

3. With the cursor at position 'c5' type an equals sign to signal that you wish to enter a formula:

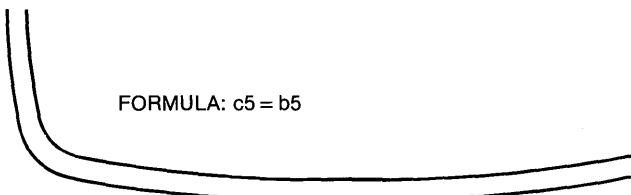
"=" (Equals sign)

¹ This concept of 'relative' variables is very important and is discussed at length in Chapter VIII, page 183.

Perfect Calc responds with the message:



Perfect Calc is ready to accept a formula for position 'c5'. Type the formula, which will consist of a single variable, namely the coordinates for the preceding entry box 'b5':

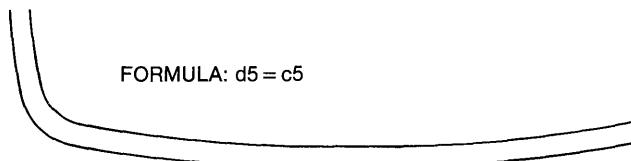


This formula, instructs Perfect Calc to display in position 'c5' whatever numeric value is located in entry position 'b5'. When this formula is entered with a carriage return, <CR>, Perfect Calc will display in 'c5' the value held in 'b5', i.e. '2000'. (As a test, move the cursor to January and change the value from '2000' to '2500'. Immediately, the February value changes also.)

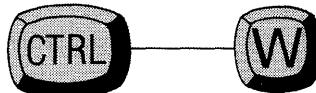
4. The problem now presents itself: How to quickly and efficiently enter all of these nearly identical formulas into their proper locations in the spreadsheet? Fortunately, Perfect Calc provides a very handy means of doing this, called 'Replication'. Replicating a formula is almost identical to 'copying' a label, with only one slight modification.

Replicating a Formula

5. Position the cursor in the March column, position 'd5'.
6. Type the formula for this position using an equal sign to begin:

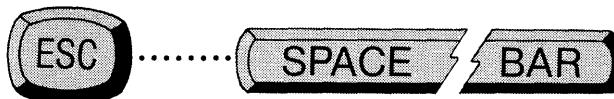


7. Instead of entering this formula with a carriage return, <CR>, enter it using the COPY ENTRY command (see Chapter VIII, page 175).

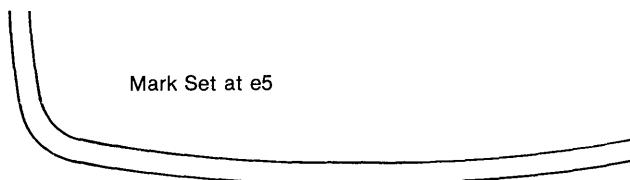


Perfect Calc enters the formula, simultaneously saving it in a temporary storage space called the 'Save Buffer' for copying to other locations.

8. Move to position e5 (use the Control----F command) and set an invisible 'mark' at position 'e5' by giving the SET MARK command:



Perfect Calc responds with the message:

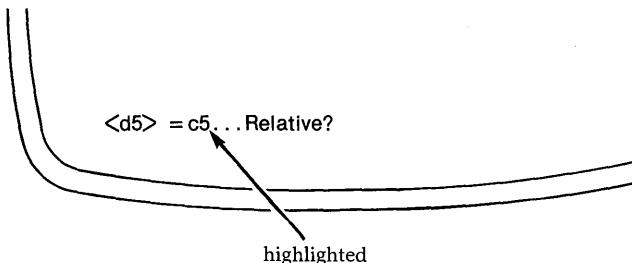


This mark defines the beginning of the region over which the formula will be replicated.

9. Move the cursor to the end of line 5, to the December column, using the END OF LINE command (Control----E).
10. Replicate the formula through this region of entry boxes by giving the YANK-BACK command:



Up to this point replicating a formula is identical to copying a label. However, Perfect Calc needs some extra information concerning formulas. It needs to know if the variable in the formula is 'relative' or not. That is, should it be changed to reflect its position in each of the successive entry boxes in which it will be replicated. It asks for this information by displaying the formula in the Prompt Line and highlighting the variable in question:



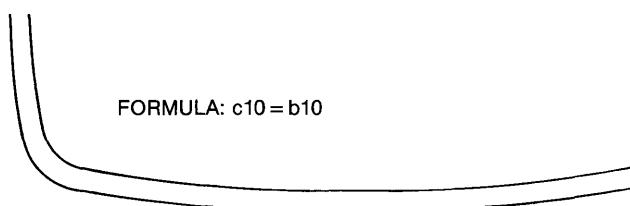
Here, the variable 'c5' is highlighted.² Perfect Calc wants to know if it should change this variable to reflect the various entry positions that the formula will be copied to. The answer, of course, is Yes. As we will see, when a formula contains more than one variable, Perfect Calc will highlight each variable in turn, asking this question.

² The variable will be highlighted in inverse video, if your terminal supports this. Otherwise it will be underlined, or bracketed with facing angle brackets, <>.

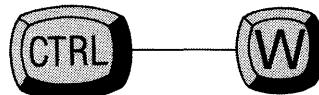
11. Type 'Y' to indicate that the variable is relative. Perfect Calc immediately replicates the formula to each of the remaining entries in line 5, April to December. At each entry, the value '2500' is displayed, indicating a related string of variable references extending back to the original numeric value entered for January.

It is now possible to see the value of such a structure. Move the cursor to June, and, on the hypothetical assumption that your income will rise, substitute a new numeric value (say '3000') for the formula. Immediately the entries from July to December display this value, while those from January to May remain unchanged.

12. The remaining formulas on the spreadsheet are entered in a similar fashion. Move the cursor to Rent on line 10. This line functions identically to Income, which is to say that rent is a value that can be expected to continue unchanged from month to month. Line 10 will thus contain a series of successive formulas structurally identical to those of Income.
13. Position the cursor at entry box 'b10' and enter a value for rent, say '350'.
14. Move the cursor to the next position, entry box 'c10', and type the formula:

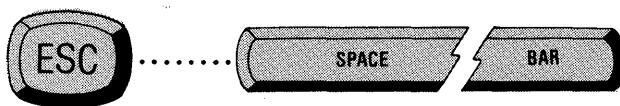


15. Enter this formula using the COPY ENTRY command:

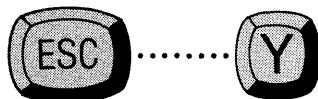


Perfect Calc enters the formula, simultaneously saving it in the save buffer for copying to other spreadsheet locations.

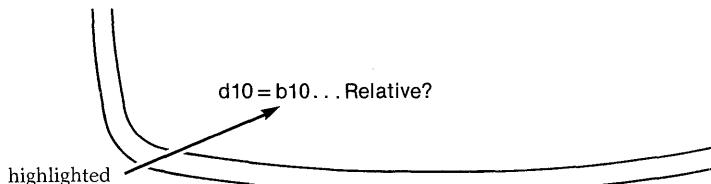
16. Move the cursor to 'd10' and set an invisible mark at position 'd10' by giving the SET MARK command:



17. Move the cursor to the last entry in the line, the December position 'm10', using the END OF LINE command (Control----E).
18. Replicate the formula across the region of entries between the mark and the cursor by giving the YANKBACK command:



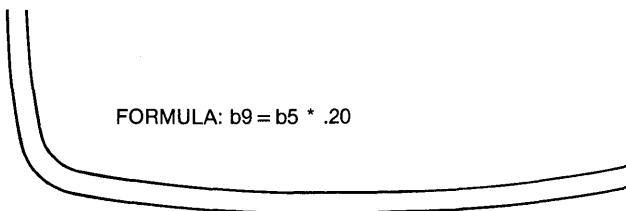
Perfect Calc immediately displays the formula in the Prompt Line, highlighting the variable 'b10':



Perfect Calc wants to know if variable 'b10' is a relative variable? Answering 'Yes' causes Perfect Calc to replicate the formula throughout the region. The value '350' appears as the 'rent' for each month. As a simple test change the July value to '400'. Do the months August to December change as well?

You have completed the second of six formulas for this spreadsheet. By now you have noticed that the actual formulas themselves are not displayed to the screen, only the values which they compute. However, every formula on the spreadsheet will display in the Prompt Line when the cursor is moved into the entry box containing it. Do this now. Move the cursor from entry box to entry box along both lines '5' and '10', noticing how the replicated formulas vary from entry to entry.

19. Position the cursor at position 'b9', the line which will hold the values for 'Taxes'. Taxes, of course, are computed as a percentage of Income. For this spreadsheet we will say that taxes will amount to 20 percent of Income. The formula for position 'b9' will then be:

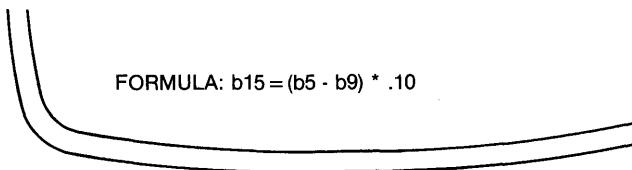


This tells Perfect Calc to multiply (*) whatever value is in position 'b5' (the Income position for January) by .20 and to display the result in position 'b9'.

In fact, as with Income and Rent, this will be the structure for all of the formulas throughout line 9. For each month, Perfect Calc will find the value in the Income entry and multiply it by .20. Again, it is a problem of replicating a formula. The variable 'b5' will become 'c5', 'd5', 'e5', 'f5', etc. Replicate this formula now. The steps are:

- Type the formula.
- Enter it with the COPY ENTRY command (Control-----W)
- Move over one entry and set the invisible mark (Escape... <space bar>)
- Move the cursor to the end of the region, (i.e. line).
- Type the YANKBACK command (Escape... Y)
- Answer 'Y' to the 'relative?' question.

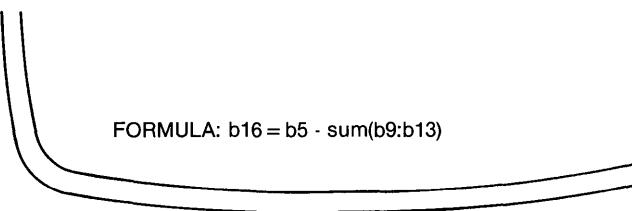
20. The next three formulas concern Savings. Move the cursor to entry box 'b15', the line which will calculate 'Regular' savings. Regular savings will be 10 percent of Income, computed after taxes. The formula will be:



FORMULA: $b15 = (b5 - b9) * .10$

This formula instructs Perfect Calc to subtract monthly Taxes from monthly Income and multiply the result by '.10'. As with the Taxes formula, this operation will be performed for each month, and so again, the problem is one of replicating a formula across a region of entries. The only difference is that this formula contains two variables 'b5' and 'b9'. **Both** are relative variables, dependent upon their column positions. (Perform the replication now, following the steps outlined in step 19.)

21. 'Disposable' savings, line 16, is defined as the amount of money left over after everything else has been paid—taxes, rent, food, even regular savings. The formula is:

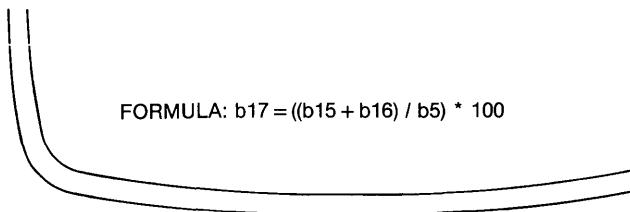


FORMULA: $b16 = b5 - \text{sum}(b9:b13)$

This formula, which makes use of the function 'sum' instructs Perfect Calc to subtract the total of all Expenses (including regular savings) from Income. The result will constitute disposable savings. Since this operation will be performed every month, the formula will need to be replicated to every entry position in line 16.

This formula, like the formula for Regular Savings contains several variables. All are relative, dependent upon their column position.

22. The last formula, '% Saved', simply instructs Perfect Calc to calculate each month what percentage of Income has been saved. The formula is:



Replicate this formula to every entry in line 17. As before, all variables are relative.

The spreadsheet should now be fairly well filled with values, the only empty entries being the two which must be entered singly as the months pass: Food and Misc. Expenses. Such blank entries will be ignored by Perfect Calc when performing calculations.

Order of Recalculation

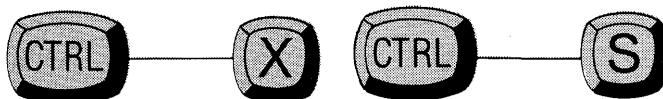
At this point the order of recalculation should be discussed. You have probably already noticed that Perfect Calc calculates entry values in a line-wise direction, moving left to right, from one line to the next.³ It is important to take note of this order of recalculation when structuring the formulas on the spreadsheet.

For example, in this spreadsheet, the formulas were placed in roughly the order in which they needed to be calculated. That is, Income had to be calculated first, because everything else in the spreadsheet depended upon it. Taxes needed to be calculated as soon after Income as possible, since the value it generated was used by nearly all of the subsequent formulas. Faulty values, or 'Errors!' would have resulted had the formulas for either Income or Taxes been placed in the last lines of the spreadsheet.

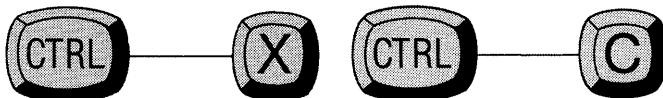
³ This can be changed using the command procedure described in Chapter IX, page 198.

In general, formula variables should reference entries **above and to the left** of their own positions. If this is not possible, and a variable must reference a position that is updated after itself, a careful check should be made to insure that recalculation does not result in a faulty value to the formula.

This concludes the tutorial on formulas. You will need the spreadsheet as we have structured it for use in the tutorials which follow. Therefore, before exiting, save the program to disk by giving the SAVE FILE command:



To exit Perfect Calc, type the QUIT command:



END OF TUTORIAL

Chapter VII

INSERTING AND DELETING

Perfect Calc provides a very powerful and efficient command structure for inserting and deleting entries, lines, columns, and even regions to and from the spreadsheet. These commands enable you to produce a highly complex and inter-related spreadsheet with a minimum of effort.

However, because these commands are so powerful, they must be used correctly and with care. This section discusses the steps that must be used when inserting and deleting data from the spreadsheet. It is recommended that you study the explanations and examples in this section carefully.

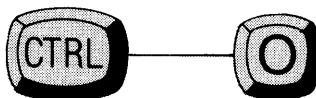
INSERTING LINES AND COLUMNS

From time to time you will need to insert a line or column into an existing spreadsheet. You may have forgotten to leave room for a particular type of data, or new information has become available which must now be included. The following commands are provided for inserting new, blank lines and columns into the spreadsheet to hold data that must be entered.

The OPEN LINE Command inserts a blank line at the position of the cursor.

Steps:

1. Position the cursor where you want the new line to be inserted. (It will necessarily occupy an existing line.)
2. Type the OPEN LINE Command,



(lowercase letter 'o')

Perfect Calc moves the line the cursor is occupying and all lines **below** it down by one, simultaneously inserting a new, blank line at the position of the cursor.



The OPEN COLUMN Command

Inserts a blank column at the position of the cursor.

Steps:

1. Position the cursor where you want the new column to be inserted. (It will necessarily occupy an existing column.)
2. Type the OPEN COLUMN command:



(lowercase letter 'o')

Perfect Calc moves the column the cursor is occupying and all columns to its **right** over by one, simultaneously inserting a new, blank column at the position of the cursor.

before

	a		b		c		d		e
1	jan		Feb.		Mar.		April		
2	350.00		350.00		350.00		350.00		
3	250.00		250.00		250.00		250.00		
4	125.00		125.00		125.00		125.00		
5									



after

	a		b		c		d		e
1		Jan.	Feb.		March				
2		350.00	350.00	350.00					
3		250.00	250.00	250.00					
4		125.00	125.00	125.00					
5									

new column inserted

Note: Both the OPEN LINE command and the OPEN COLUMN command can be repeated using the REPEAT command (Control----U), Chapter IV, page 53.

Adjusting Variable References

For both the OPEN LINE and OPEN COLUMN commands Perfect Calc will automatically adjust all variable references in the lines and columns to reflect their new positions.

For example, suppose that on a spreadsheet containing 10 filled lines you have just inserted a line at position '7'. The old lines 7, 8, 9, and 10 have been moved down to become lines 8, 9, 10, and 11. Any references to variables on these lines will be adjusted to reflect their new line numbers. A formula such as:

$$b10 = \text{sum}(b1:b9) + a2$$

will automatically become

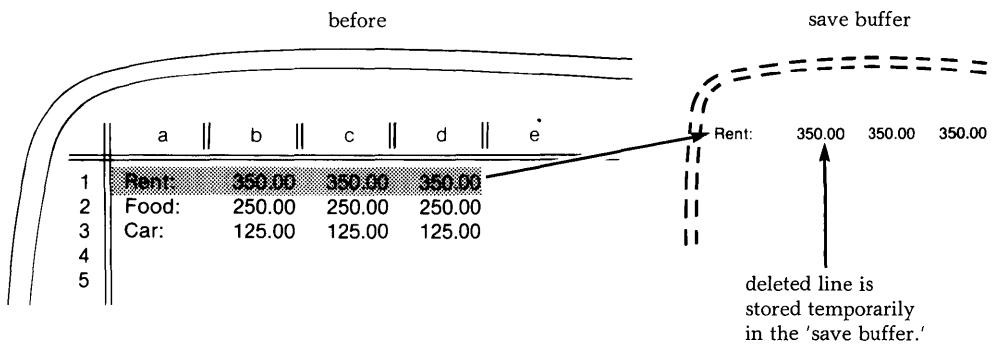
$$b11 = \text{sum}(b1:b10) + a2$$

However, variable references to entry positions which are **above**, or **to the left of**, the inserted line or column will remain unchanged in each formula.

DELETING

The 'Save Buffer'

Perfect Calc provides several powerful commands for deleting entries, lines, columns, and regions of data. Whenever a deletion is made, Perfect Calc temporarily saves the deleted material in a special storage buffer called the 'Save Buffer'. This is a safeguard against the chance that you might make a mistake or change your mind about a particular deletion. The Save Buffer is simply an area in machine memory that has been set aside to hold deletions temporarily. Imagine it something like this:



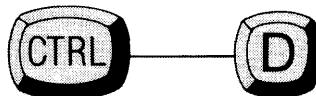
Only the most **recent** deletion is stored in the Save Buffer. That is, if you make a deletion, move the cursor, and then make another deletion at another location, Perfect Calc, assuming you no longer want the first deletion, discards it, keeping the second deletion in its place.

The most recent deletion you make can be restored to the spreadsheet from the Save Buffer using the YANKBACK command (see Chapter VIII). When used in connection with the YANKBACK command, the Save Buffer has important applications in copying and moving data, (which will be discussed in the next chapter).

The DELETE ENTRY Command

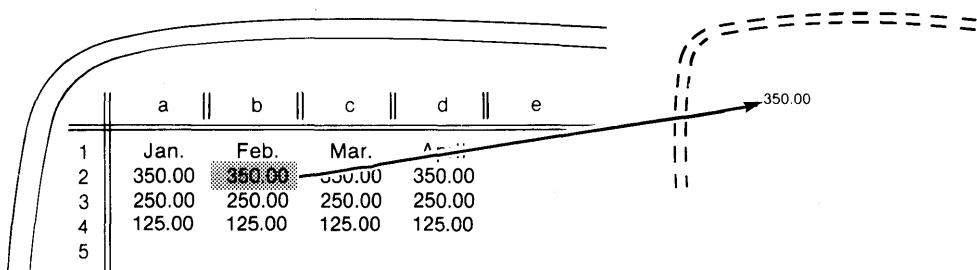
Deletes the entry occupied by the cursor.

Position the cursor on the entry to be deleted and type the DELETE ENTRY command:



before

save buffer

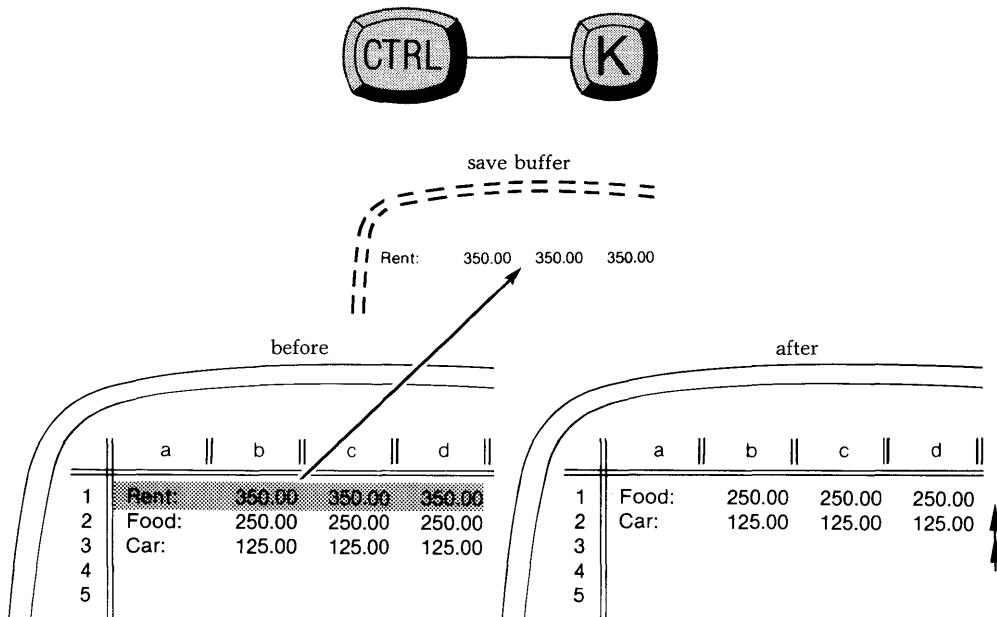


Perfect Calc deletes the entry, saving it temporarily in the Save Buffer.

The DELETE LINE Command

Deletes or kills the entire line which the cursor is currently occupying.

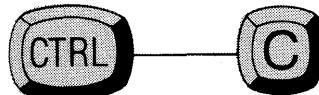
Position the cursor anywhere on the line to be deleted and type the DELETE LINE command:



Perfect Calc deletes the line, saving it temporarily in the Save Buffer.

All lines below the deleted line are automatically renumbered to adjust for the missing line. That is, if line 2 has been deleted, line 3 is renumbered as line 2, line 4 as line 3, etc. As well, variables within these lines are renumbered to reflect their new positions.

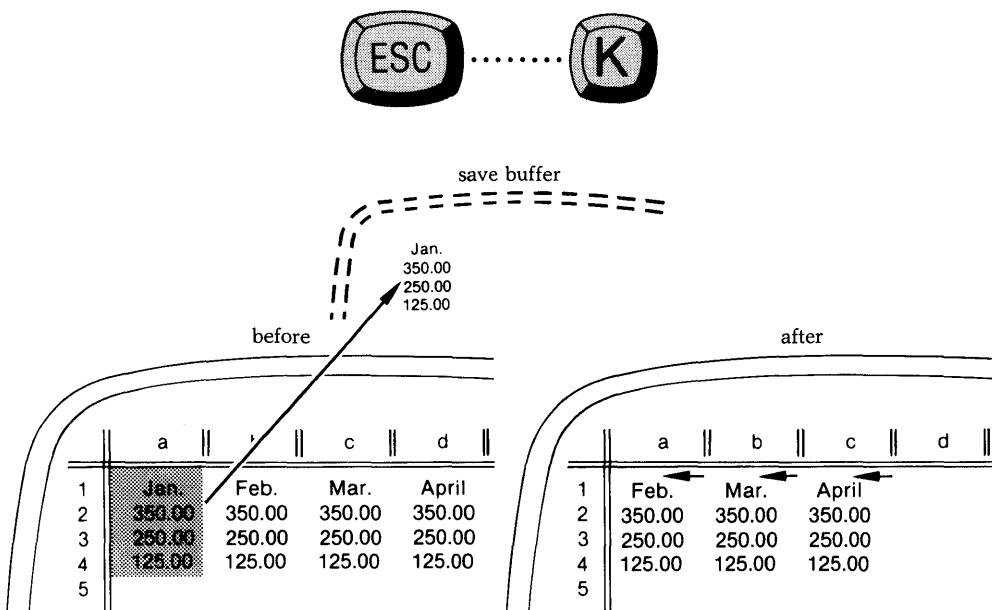
Note: If you have the alternate command set, to delete the line you will need to type:



The DELETE COLUMN Command

Deletes the entire column which the cursor is currently occupying.

Position the cursor anywhere in the column to be deleted, and type the DELETE COLUMN command:



Perfect Calc deletes the current column, saving it temporarily in the Save Buffer.

All columns to the right are shifted one space to the left and relabeled to adjust for the missing column. That is, if column 'b' has been deleted, column 'c' is now relabeled as column 'b', column 'd' as column 'c', etc. All variables within these columns are relabeled to reflect their new positions.

Note: If you have the alternate command set, to delete the column you will need to type:



The DELETE REGION Command

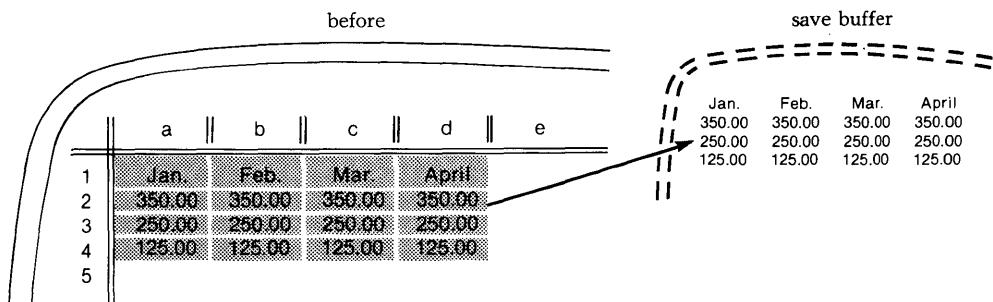
Deletes an entire 'region' of the spreadsheet at one time. This may be a single entry, a column, a line, or a region—a rectangular area including several partial or full columns or lines (see below).

Define the region to be deleted using the Mark Boundaries Procedure, Chapter VII, page 147.

Type the DELETE REGION command:



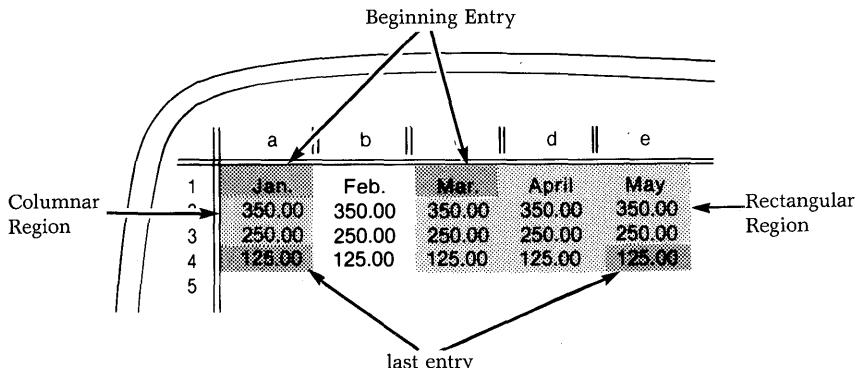
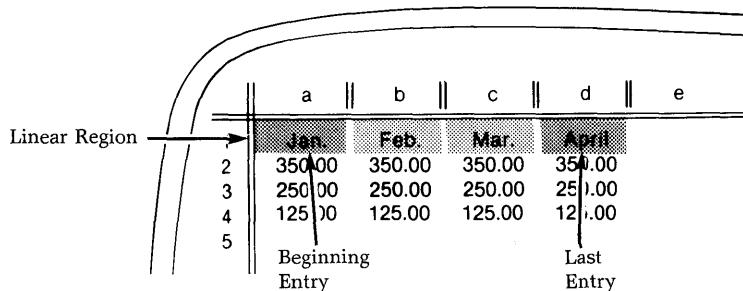
Perfect Calc deletes the contents of the region, saving it temporarily in the Save Buffer. Unlike the DELETE LINE and DELETE COLUMN commands, the DELETE REGION command does not erase the lines and columns of the spreadsheet. Only the data occupying these lines and columns is removed.



REGIONS

A region is simply a portion of the spreadsheet. It may be a line, column, or rectangular area encompassing several lines and columns.

Possible Regions:



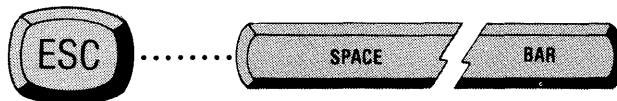
DEFINING REGIONS

The Mark Boundaries Procedure

Before any command involving a region can be executed, the region must be identified using a procedure called the Mark Boundaries Procedure:

Steps:

1. Position the cursor on the first entry position of the region, which will usually (but not necessarily) be the upper left-most entry box.
2. Set a boundary mark at this position by typing the MARK SET command:



Perfect Calc will display the message 'Mark Set at <coordinates>' in the Prompt Line, indicating that an invisible boundary mark has been set at the first entry position in the region.

3. Move the cursor to the 'last' position in the region. For linear and columnar regions this will be the last entry in the line or column; for rectangular regions, it will be the lowest, right-most position.

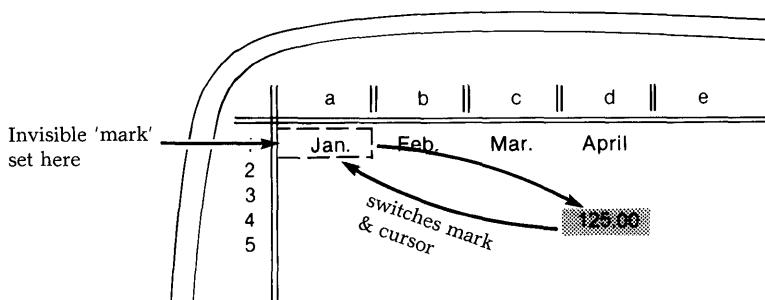
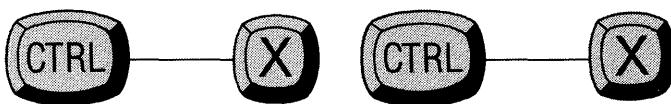
Note: Once the forward boundary mark has been set, it is the position of the cursor which defines the region. As you move the cursor, the definition of the region changes.

4. Once you have positioned the cursor at the other end of the region, type the command you wish to execute.

Perfect Calc performs the action for the command affecting the region you have defined.

The EXCHANGE CURSOR & MARK Command

When defining regions, you may forget where the invisible forward mark has been set, since the message 'Mark set at <coordinates>' disappears from the Prompt Line when you move the cursor. An easy means of verifying the position of the mark is to use the EXCHANGE CURSOR & MARK command, which switches the positions of the mark and cursor. Type:



Perfect Calc switches the cursor and mark. Repeating the command will switch the cursor and mark again to their original positions. An alternative method of examining the current region is to use the STATUS command (Control----X=) which will list, among other things, the currently defined region on the Prompt Line (see Chapter IX, page 203).

Note: When deleting regions the relative positions of the cursor and mark are immaterial, therefore it is perhaps a wise policy to always exchange the cursor and mark at least once as a final check of the boundaries before deleting the region.

A Word of Caution

When deleting data the chance exists that you could inadvertently affect areas of the spreadsheet beyond the particular entry, or region of entries, being deleted.

This circumstance stems from the nature of a spreadsheet itself, which is closely integrated, its various elements referencing and cross-referencing each other. When any entry, line, or column is deleted, the possibility exists that another entry—a formula somewhere in the spreadsheet, which was referencing a variable in the deleted line or column—will be left with a 'blind', or invalid, variable reference.

When this happens Perfect Calc substitutes in those formulas that are affected a line number of '0' (zero) for every variable still referencing a deleted **line**, and a column letter of '?' for every reference to a deleted **column**. Thus, for a deleted line '13' in the formula:

FORMULA: c17=b13+b14

the formula would be changed to:

FORMULA: c17=b0+b14

Had column 'b' been deleted, the formula would become:

FORMULA: c17=?13+?14

Formulas containing such blind references automatically compute 'Error!', and can be corrected only by changing, or otherwise altering, the blind reference within the formula. They **cannot** be corrected by simply restoring the deleted line or column, using the YANKBACK command. It can be a dismaying experience to see Error! flags sprout everywhere throughout your spreadsheet due to one inadvertent deletion¹. Nevertheless, it can and does happen.

¹ See the command procedure on 'locking' formulas as a way to prevent this, Chapter VI, page 113.

INSERTING AND DELETING

TUTORIAL

In this tutorial we will practice inserting, deleting, restoring and copying lines and columns. We will be using the exercise spreadsheet which was created in previous tutorials, 'budget83.pc'. At this time, retrieve this program from disk, displaying it to the screen in Perfect Calc.

INSERTING

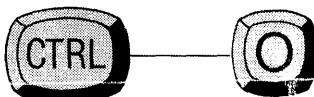
Inserting a line or column is generally an easy thing to do. It is usually employed when making room for some additional data that is either new, or was forgotten when the spreadsheet was created.

For example, suppose that in our spreadsheet, 'budget83.pc', we have forgotten to provide a line recording the monthly payments for our car.

Steps:

1. Position the cursor at the point in the spreadsheet where you wish the line to be inserted. This should be somewhere below Taxes and above Regular Savings. Let us insert the line between Rent and Food. Therefore, position the cursor on line 11, which records expenses for food.

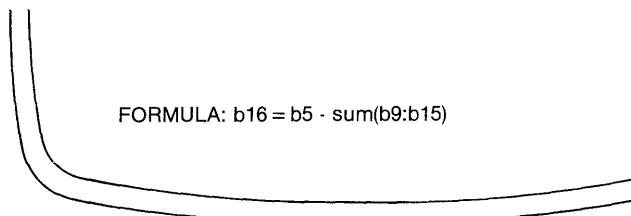
2. Type the OPEN LINE command:



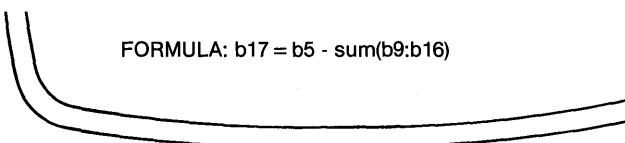
Perfect Calc inserts a blank line between Rent and Food, shifting all lines below the insert down one line. Food has become line 12, Misc. exp. line 13, etc. All of the formulas contained in these lines have been modified to reflect their new line positions.

3. Position the cursor at the beginning of the new line 11 and enter the label 'Car'.
4. In the next position, entry box 'b11', enter the numeric value representing the monthly car payment, e.g. \$35.
5. Move the cursor to entry position 'c11' and enter the formula which will represent the monthly car payments. This will be identical to the formula for Income and Rent, and should be replicated to every entry in the line.

Notice that the new line has been automatically integrated into the spreadsheet. That is, the single formula in the spreadsheet which referenced this position (Disposable Savings) was automatically expanded to include the values of the new line. The original formula was:



After the new line was inserted, this became:



Perfect Calc will automatically alter formulas which define a range of entries, and among which a new entry has been inserted. Of course, if the line is inserted beyond the defined range of a formula, then the formula will have to be altered manually to take account of the insertion.

For example, if we had inserted Car above Taxes, at line 8, the formula which computes Disposable Savings, and which originally included only values in the range 'b9' to 'b15', would have had to be manually altered to include position 'b8'.

In general, if one is inserting a line among several others of the same type, all of which are being referenced by formulas which include the lines in a range, the formulas will be automatically altered to include the insertion.

If, on the other hand, the insertion is not among a range of values, but still affects the calculations, then all of the relevant formulas will have to be edited (or edited and then replicated) to take account of it.

For example, if we wanted to insert a line recording 'Interest Income' at line 6 just beneath Income, we would clearly have to change a number of important formulas. For one, the tax formula would have to be changed from:

FORMULA: $b9 = b5 * .20$

to

FORMULA: $b9 = (b5 + b6) * .20$

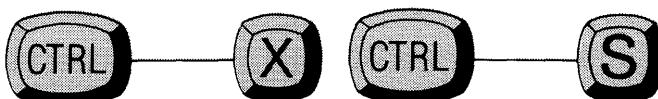
Inserting a column is performed in a similar fashion, except that Perfect Calc opens a new column, instead of a line, at the position of the cursor.

DELETING

As was mentioned earlier in this chapter, deleting entries, lines, and columns from a spreadsheet entails some risk, because the chance exists that the deleted data is being referenced by some variable in another formula which we did not notice. When this happens, the referencing formula may compute to Error! at the next recalculation. Other formulas which reference **that** formula will also begin computing to Error!

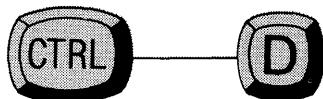
It is interesting to see how this happens:

1. Save the spreadsheet by giving the SAVE FILE command:



(Saving the spreadsheet should be your first step before beginning deletions, for if the deletion process goes awry, you can simply exit Perfect Calc and call up another copy of the spreadsheet to start all over.)

2. Move the cursor to position 'b5' the first entry that records Income. Give the DELETE ENTRY command:



Perfect Calc deletes the data from entry position 'b2' and recalculates the spreadsheet. Notice how many formulas are affected by the deletion. The spreadsheet should look something like this:

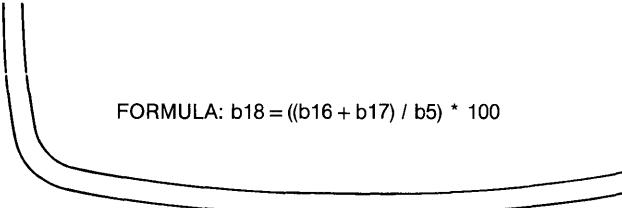
	a		b		c		d		e		f		g	
1														
2														
3	Month		January		February		March		April		May		June	
4														
5	Income		(deleted)		0		0		0		0		0	
6														
7														
8														
9														
	Expenses													
10	Taxes		0		0		0		0		0		0	
11	Rent		425.00		425.00		425.00		425.00		425.00		425.00	
12	Car		35.00		35.00		35.00		35.00		35.00		35.00	
13	Food		235.78		189.00		95.43		154.34		175.00		134.00	
14	Misc. exp.		457.00		325.00		552.45		605.11		389.99		531.00	
15														
	Savings													
16	Regular		0		0		0		0		0		0	
17	Disposable		(1,327.78)		(1,149.00)		(1,282.88)		(1,394.45)		(1,199.99)		(1,300.00)	
18	% saved		Error!											

The various formulas which referenced this position are affected differently depending upon the mathematics involved. 'Taxes', which are computed by the formula:

$$\text{FORMULA: } b9 = b5 * .20$$

computes automatically to 'zero', as do the monthly 'Income' formulas, 'c5' to 'm5'.

However, the formula which computes percentage of income saved:

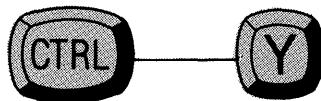


FORMULA: $b18 = ((b16 + b17) / b5) * 100$

computes to Error because the formula is mathematically incorrect now that position 'b5' contains no value. One cannot divide a number by zero.

YANKBACK

As we know, Perfect Calc saves any deletion made in its Save Buffer, a reserved space in computer memory. It is possible to restore the most recent deletion from this buffer using the YANKBACK command:



To yankback the deleted Income entry at position 'b5':

- Do not make any further deletions (since these will replace the Income data being held in the Save Buffer).
- Place the cursor at position 'b5' if you have since moved it.
- Give the YANKBACK command (Control----Y).

Perfect Calc immediately replaces the entry and recalculates the spreadsheet, correcting the 'Error!' values in line 18.

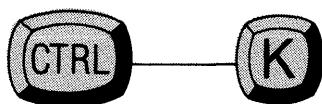
Important Note: It should be emphasized that although a 'yankback' resulted here in the correction of Error! values, this will not always be the case. When a line or column is deleted, not only is the data removed but also the actual line or column itself. This means that variables referencing entries in that line will not just be left with a value of 'nothing', which may or may not compute to Error!. They will be left with a 'blind' reference to a non-existent line or column. In this case they will compute to Error! (see Chapter VII, page 149).

Deleting a Line

Let us delete the Rent line of our spreadsheet, and then restore it, to see just how Perfect Calc removes the line and reinserts it.

Steps:

1. Position the cursor anywhere on line 10.
2. Give the DELETE LINE command



(or Control----C if you have the alternate command set.)

Perfect Calc removes line 10, moving all lower lines up by one and renumbering them. Examine the formulas in the lower lines. Their variables have been readjusted to account for the deleted line 10.

3. Since the line has been temporarily saved in the Save Buffer, we can recall it simply by typing the YANKBACK command (Control----Y). You will need to indicate that the variables yanked back are relative.

Perfect Calc reinserts the line at the position of the cursor, pushing all lines below the insertion down by one and renumbering.

Deleting a Region

One important operation is deleting and restoring a 'region', a rectangular area of the spreadsheet involving both lines and columns. Let us suppose that we wish to delete data from the second two columns of our spreadsheet.

Steps:

1. Position the cursor at position 'b9', the second entry position in the Taxes line.
2. Set an invisible mark here using the SET MARK command (Escape.....<space bar>). This defines the beginning of the region to be deleted.
3. Move the cursor to position 'e13', the entry recording April's Miscellaneous expenses. (The new position of the cursor defines the other boundary of the rectangular region to be deleted.)
4. Type the DELETE REGION command:

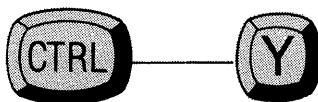


Perfect Calc immediately deletes the data from the lines and columns in the defined region, saving it temporarily in the Save Buffer. Notice that the lines and columns themselves have not been deleted, just the data which they held.

Restoring a Region

A region is restored to the spreadsheet in a slightly different manner than are lines and columns. Whereas the lines and columns are 'reinserted' into the spreadsheet, pushing other lines and columns aside, a region 'overwrites' entry positions **beginning** at the position of the cursor. Therefore our first step in restoring this deleted region of data is:

1. Move the cursor from position 'e13' to position 'b9', the original beginning position of the region.
2. Type the YANKBACK command:



Perfect Calc proceeds to restore the data to the region, pausing to display each formula in the Prompt Line and to ask 'Relative?'

Since every formula variable in this particular region is relative, typing an exclamation point (!) causes Perfect Calc to consider every variable in the region as relative. The data is immediately written to the region.

Note: As mentioned in Chapter VIII, page 172, the procedures for moving lines and columns to new locations on the spreadsheet are identical to those we have just practiced in restoring deletions, except, of course, that the material is restored to a different location in the spreadsheet. Therefore, in this tutorial, we will not examine a move operation but continue with several examples of 'copying' lines, columns, and regions.

COPYING LINES, COLUMNS, AND REGIONS

Copying is similar to restoring and moving in that data is first transferred to the Save Buffer, and from there to the spreadsheet. However, instead of material being 'deleted' into the Save Buffer, it is 'copied' into the Save Buffer. No actual entries, lines, or columns are ever removed from their original locations in the spreadsheet, as with deleting. Also, when the item, or items, are restored to the spreadsheet, they are not inserted between existing lines and columns; instead, they overwrite whatever data may be present at the new location. When the operation is complete, the data exists at both its old and new positions. Two copy commands are used to transfer data to the Save Buffer, depending upon whether the item is a single entry or a region (i.e. a line, column, or region).



COPY ENTRY Command



COPY REGION Command

Two YANKBACK command forms exist to recall the data from the Save Buffer to the spreadsheet:



'Yankback' to Single Location



'Yankback' to Multiple Locations

Notice that, as with other Perfect Calc Commands, the Control Key deals with a single entry, while the Escape Key functions with a region of entries.

Let us now attempt several kinds of copy operations to see just how these two commands function together. Although our sample spreadsheet 'budget83.pc' does not provide convincing reasons for moving lines and columns around, we will use it anyway, to practice the various command procedures.

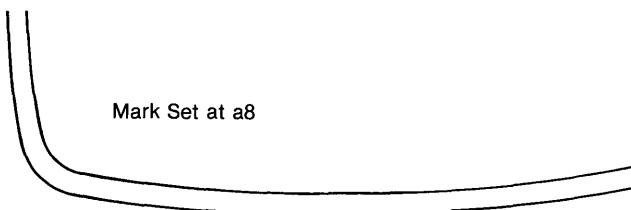
Copying a Line

Let us suppose that we wish to copy line 8, which holds the label "-----Expenses-----". We will copy it first to another single line position, and then to a region of line positions.

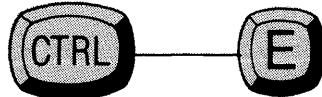
Steps:

1. Position the cursor at the first entry in the line, entry box 'a8'.
2. When copying, Perfect Calc regards a line as a linear region of entries. Therefore, we must first define line 8 as a region, using the Mark Boundaries Procedure. Type the MARK SET command, establishing an invisible mark at position 'a8'.

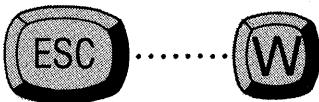
Perfect Calc responds with the message:



3. Move the cursor to the end of line 8 using the END OF LINE command:

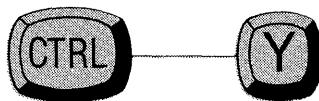


4. Copy the line into the Save Buffer by typing the COPY REGION command:



Perfect Calc immediately copies the line to the Save Buffer. (There is almost no indication that this action has taken place, except that the cursor may blink and redisplay when the operation is complete.)

5. Move the cursor to position 'c19', three columns from the left-margin.
6. Type the YANKBACK command form which restores the line to a single location on the spreadsheet:

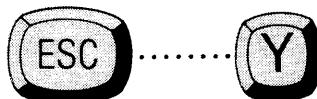


Perfect Calc immediately copies the line to the spreadsheet.

Notice that the newly copied line **begins** at position 'c19'—inset three columns from the left margin of the spreadsheet. This is an important characteristic of copying: any line, column, or region of data that is being copied to another single location is restored to the spreadsheet **beginning** at the position of the cursor.

Furthermore, the copied material will **overwrite** whatever data happens to be present at the new position.

7. Let us suppose that we wish to continue copying line 8 five more times, to lines 20 through 24. Since the line is still present in the Save Buffer, we need only define a 'column-wise' region that will include these 5 line positions:
 - Set a mark at position 'c19'.
 - Move the cursor down five lines to position 'c24'.
8. Type the YANKBACK command form used for copying a data item to multiple locations (i.e. to a defined region) in the spreadsheet.



Perfect Calc immediately copies line 8 to lines 20-24, beginning at positions 'c20' through 'c24'.

Copying a Column

Copying a column works identically to copying a line. Again, prior to copying to the Save Buffer, the column must be defined as a region. For an example of this, let us copy column 'a' to an off-screen location, say column 'p'.

Steps:

1. Place the cursor at the top of column 'a'.
2. Set an invisible mark at this top position, using the Mark Set command (Escape.....<space bar>).
3. Move the cursor to the bottom of the column, using the BOTTOM OF COLUMN command (Escape.....>).
4. Copy the column to the Save Buffer using the COPY REGION command (Escape.....W).
5. Move the cursor to the top of Column 'p' , using the GOTO command:

(Type a right-angle-bracket followed by the coordinates 'p1')

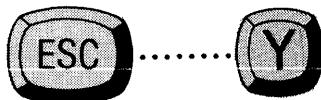
6. Restore the column to its new single location by typing the YANKBACK command form (Control----Y).

Perfect Calc copies the data to column 'p'.

7. Now copy the data to column positions 'q' through 'u' by defining a 'line-wise' region:

- Set an invisible mark at position 'q1' using the Mark Set Command.
- Move the cursor to position 'u1'.

8. Recall the data from the Save Buffer using the YANKBACK form:



Perfect Calc immediately copies the data to the five columns, 'q' through 'u'.

Copying a Region

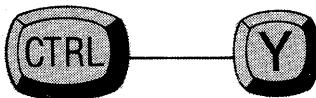
Regions cannot be simultaneously copied to multiple locations as can entries, lines, and columns. In any one command operation they can be copied only to another single location.

Let us suppose that we wish to copy the first three columns of our spreadsheet to another location.

Steps:

1. Define a region encompassing entry positions 'a1' through 'c17'.
 2. With the cursor at position 'c17' type the COPY REGION command (Escape.....W).
- Perfect Calc copies the data in the first three columns into the Save Buffer.
3. Move the cursor to an off-screen position which is still in the current spreadsheet, say position 'j12'. (This time we are going to **overwrite** existing spreadsheet data!)

4. Recall the data to the spreadsheet using the YANKBACK command form which restores a copied item to a single, new spreadsheet location:



Perfect Calc immediately copies the data to the spreadsheet, **overwriting** existing data, pausing only to ask for 'relative' confirmation for the variables contained in the region. Hit (!) to indicate that all the variables in the region are 'relative'.

Notice that Control----Y was used to restore the region and not Escape.....Y. This is because Perfect Calc regards the entire region as a single entity that is simply being restored to another single location. It is not possible to define a region of multiple locations to which a region may be copied. Terminate this tutorial with the QUIT command.

END OF TUTORIAL

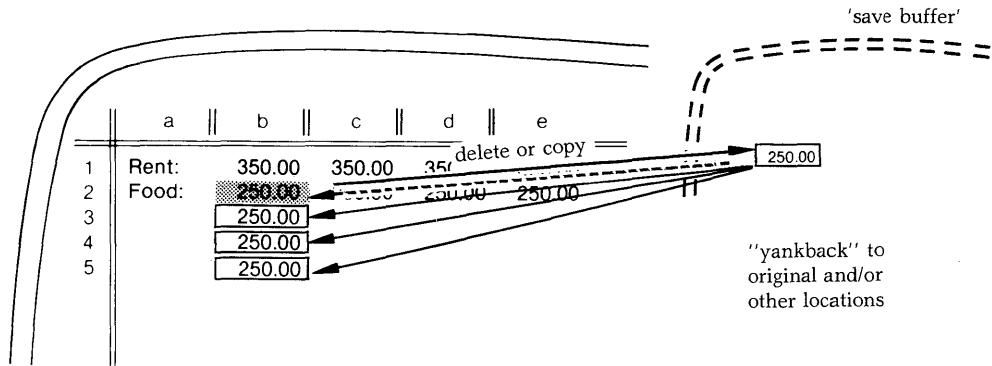
Chapter VIII

MOVING, COPYING, AND RESTORING DELETIONS

The YANKBACK Command

The YANKBACK command causes whatever data is being held temporarily in the Save Buffer¹ to be written to the spreadsheet beginning at the position of the cursor. Functioning as such, it has several very useful applications, including restoring deletions, moving and copying data, and 'replicating' formulas across regions of the spreadsheet.²

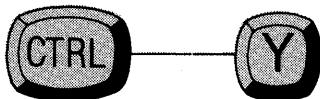
The method for accomplishing all of these is essentially the same. The material is first deleted or copied into the Save Buffer using an appropriate delete or copy command, and then is 'yanked back' to the spreadsheet at its new location(s).



¹ See Introduction to Deleting, Chapter VII, page 141.

² The procedure for replicating formulas is presented in Chapter VI, page 127.

The YANKBACK command possesses two general command forms, depending on whether material is yanked back into a single entry, line, column or region (Control----y), or whether it is yanked back into multiple entries, lines or columns (Escape----y).



Yankback to single location



Yankback to multiple locations

One or the other of these forms is used depending upon the kind of data involved, (i.e. entry, line, column, or region), how that data has been transferred into the Save Buffer, and whether the data is being 'restored', 'moved', or 'copied'.

The YANKBACK is an extremely powerful and efficient command that will enable you to shape and manipulate your spreadsheet with ease. This chapter discusses the various and correct applications of the YANKBACK, and should be studied carefully.

Restoring Deletions

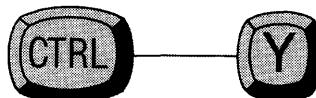
Restoring deletions represents the most basic application of the YANKBACK command. An entry, line, column, or region has been deleted, using one of the delete commands (see Chapter VII), and you now want to replace it, using the YANKBACK.

Restoring An Entry

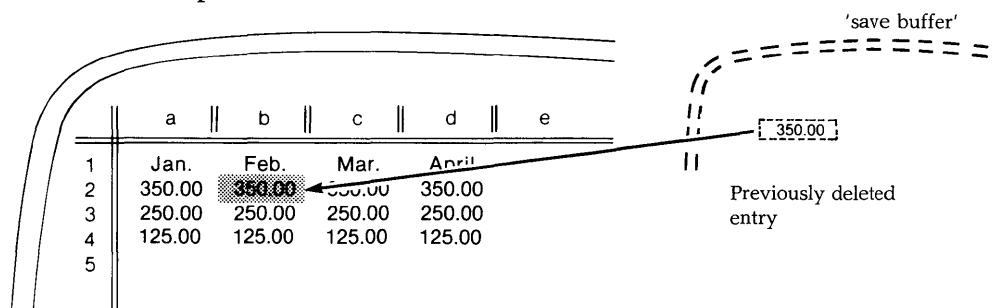
An entry has just been deleted using the DELETE ENTRY command (Control-D). It is still in the Save Buffer.

Steps:

1. Do **not** make any other deletions, since these will overwrite the previously deleted entry being held in the Save Buffer.
2. Position the cursor where it was when you made the deletion.
3. Type:



4. Perfect Calc restores the entry at the position of the cursor **overwriting** whatever data is present.

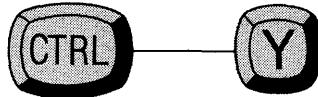


Restoring a Line

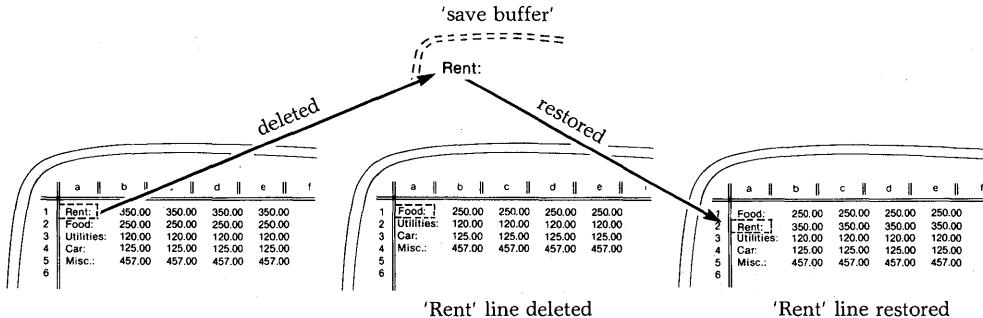
A line has just been deleted using the DELETE LINE command (Control----K or Control----C for the alternate command set), and is being held temporarily in the Save Buffer. On the spreadsheet, all lines below the deletion have been moved up and renumbered.

Steps:

1. Do **not** make any other deletions, since these will overwrite the line being held in the Save Buffer.
2. Position the cursor where you want to restore the deleted line. (This may be on an existing line.)
3. Type:



4. Perfect Calc reinserts the deleted line, moving all lines below the insert, including the line which the cursor is occupying, down one line. All lines are renumbered and formula variable references are readjusted accordingly.



Restoring a Column

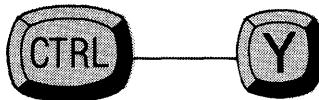
A column has just been deleted using the DELETE COLUMN command (Escape K or Escape C for the alternate command set), and is being held temporarily in the Save Buffer. On the spreadsheet, the columns to the right of the deleted column have been moved to the left and relabeled. The procedure here is the same as used for restoring a line.

Restoring a Region

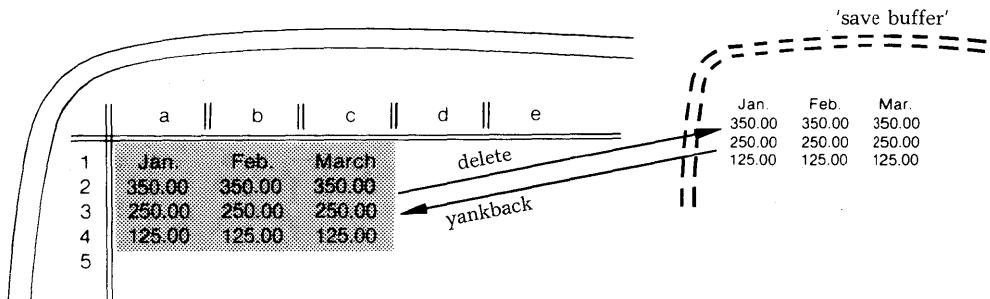
A region of data has just been deleted using the DELETE REGION command (Escape D), and is being held temporarily in the Save Buffer. However, the lines and columns which held this data are still present on the spreadsheet.

Steps:

1. If you have moved the cursor, do **not** make any other deletions, since these will overwrite the data being held in the Save Buffer.
2. There is no need to redefine the region. Simply move the cursor to the upper, left-most position which the region of data will occupy.
3. Type:



Perfect Calc restores the deleted data, **overwriting** whatever new data may have been entered into the region you have defined.



MOVING DATA

Moving entries, lines, columns, and regions of data involves nothing more than 'deleting' these items to the Save Buffer and then 'restoring' them to new locations. The procedures are identical to those used in restoring simple deletions, except that the data is not being restored to its old location, but to a new location.

As mentioned in the previous chapter, you must make sure that formulas elsewhere in the spreadsheet do not reference the line, column, or region being moved, since the moment that the line or column is deleted, these formulas will compute to Error!.

Whenever a line or column is to be moved, find the formulas in the spreadsheet which reference variables in that line or column. Edit these formulas to reflect their **new** locations **before** actually deleting and moving the line or column. Overall, you will probably encounter this problem infrequently, if at all, since moving lines and columns happens rarely, and when it does, usually none or very few formulas reference the line or column in question.

Steps:

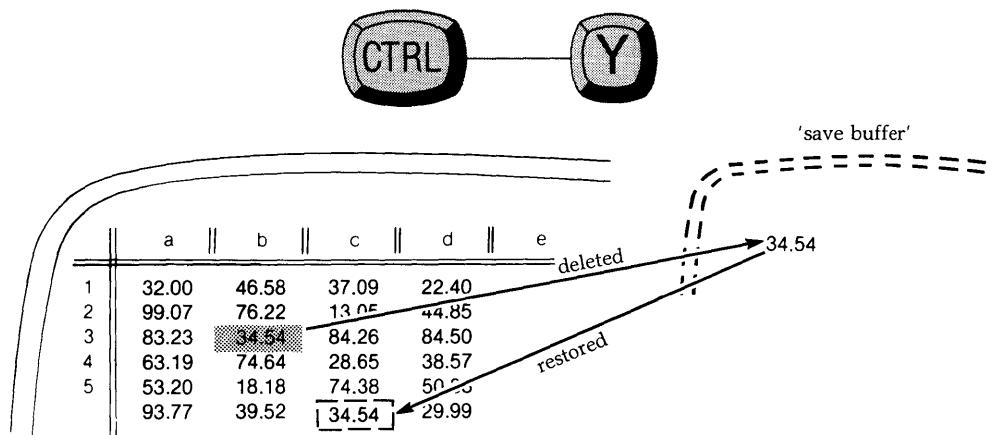
1. Determine if any formulas in the spreadsheet reference the item to be moved.
2. If any formulas do reference the item, change their variables to reflect the item's **new** location.
3. Delete the entry, line, column, or region, into the Save Buffer using one of the following delete commands:
 - The DELETE ENTRY command (Control----D)
 - The DELETE LINE command (Control----K)³
 - The DELETE COLUMN command (Escape.....K)⁴
 - The DELETE REGION command (Escape.....D) (Use the Mark Boundaries Procedure to define the region.)

[**Note:** All of these deletion commands can be repeated using the REPEAT command [Control----U]).

³ For the alternate command set, this is Control----C.

⁴ For the alternate command set, this is Escape.....C.

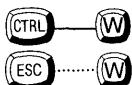
4. Move the cursor to the new location where the deleted material is to be restored.
5. Restore the material to the spreadsheet using the YANKBACK command:



COPYING DATA

'Copying' entries, lines, columns, and regions differs from 'moving', in that a copied item is not removed from its original position, although it is duplicated at another.

The steps for copying an entry, line, or column to a new location are essentially the same as those for moving it, except that instead of using the delete commands to transfer the material into the Save Buffer, the COPY ENTRY command and the COPY REGION command are used.



The COPY ENTRY Command

The COPY REGION Command

These two copy commands represent an important distinction since the YANK-BACK command will operate differently upon material copied into the Save Buffer than it will upon material deleted into the Save Buffer. Specifically, lines and columns copied into the Save Buffer will, when restored to the spreadsheet, **overwrite** existing data on the spreadsheet, whereas lines and columns deleted into the buffer will be inserted between existing lines and columns.

This is an important distinction, and when copying data from one location to another, one should take care not to inadvertently overwrite other existing data on the spreadsheet.

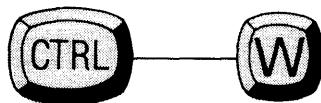
In the following pages the commands for copying entries, lines, columns, and regions of data to new locations in the spreadsheet will be presented.

The COPY ENTRY Command Procedure

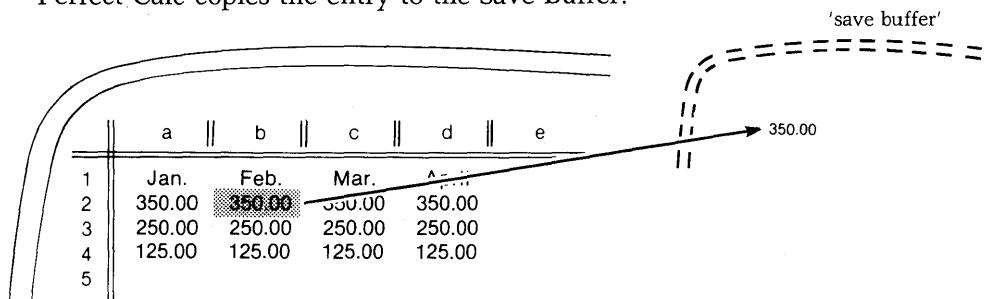
Copies a single entry marked by the cursor to a new location (or locations) on the spreadsheet.

Steps:

1. Place the cursor on the entry to be copied.
2. Type the COPY ENTRY command:

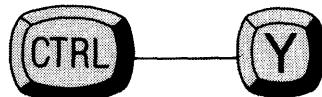


Perfect Calc copies the entry to the Save Buffer.



At this point two options are available. You can copy the entry to another single entry position, or you can copy the entry to numerous positions throughout a defined region.

- **Single Entry Position:** To copy to a single entry position, move the cursor to the new position and type the YANKBACK command:



Perfect Calc copies the entry in the Save Buffer to the new location.

- **Multiple positions in a Region:** To copy the entry to numerous positions throughout a defined region. Define the region using the Mark Boundaries Procedure (Chapter VII, page 147). Type the YANKBACK command form:



Perfect Calc copies the single entry into all the entry positions throughout the defined region.

The COPY LINE Command Procedure

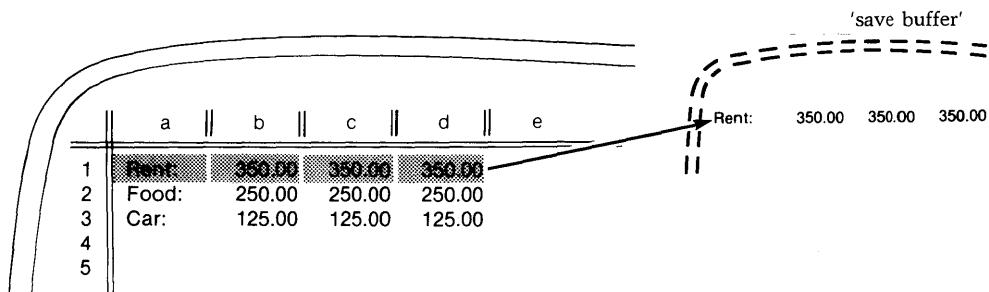
Copies a line to a new location (or locations) in the spreadsheet. (Note that in this procedure a 'line' is considered a 'linear region'.)

Steps:

1. Define the line as a region to be copied, using the Mark Boundaries Procedure, (Chapter VII, page 147).
2. Type the COPY REGION command:

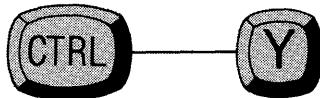


Perfect Calc copies the line into the Save Buffer.



At this point two options are available: You can copy the line to another single line position, or you can copy the line to numerous line positions throughout a defined region.

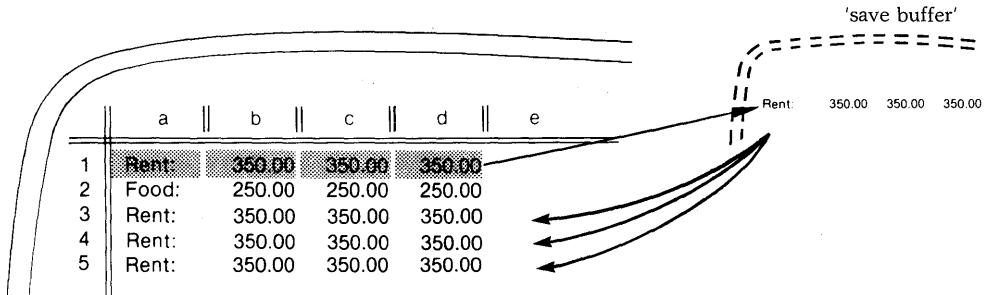
- **Single Line Position:** To copy to a single line position, move the cursor to the first entry position where you wish the line to begin. Type the YANKBACK command:



Perfect Calc copies the line to the spreadsheet, beginning at the position of the cursor, **overwriting** whatever data is present at that position. (If you only want to insert a copy of a line, simply perform the OPEN LINE command, and then yank it back into the new empty line.)

- **Multiple Line Positions:** To copy a line to several other line positions, define a 'column-wise' region as the copy destination, using the Mark Boundaries Procedure, Chapter VII, page 147.

For example, if you have copied line 3 into the Save Buffer, and now want to duplicate it on lines 3-5 of the spreadsheet beginning at column 'a':



- Define a linear region from 'f6' to 'f10' using the Mark Boundaries Procedure.
- When the region is defined, type the YANKBACK command:



Perfect Calc immediately copies line 5 to line positions 6-10, beginning at column 'f', **overwriting** whatever data is present in these locations.

The COPY COLUMN Command Procedure

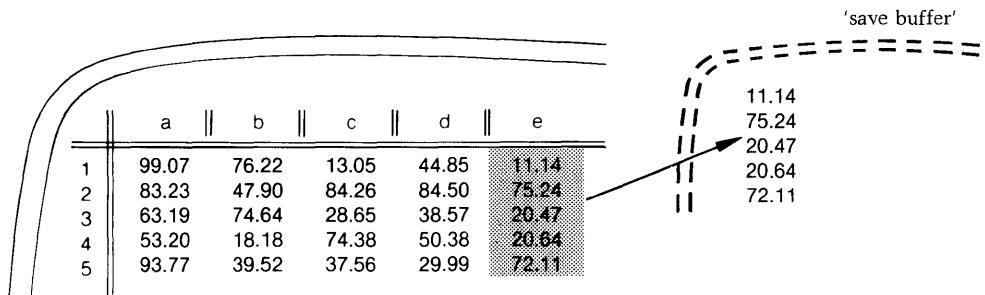
Copies a column to a new location (or locations) in the spreadsheet. (Note that in this procedure a 'column' is considered a columnar 'region'.)

Steps:

1. Define the column to be copied as a region, using the Mark Boundaries Procedure, Chapter VII, page 147.
2. Type the COPY REGION command:



Perfect Calc copies the column into the Save Buffer.



At this point two options are available: You can copy the column to another single column position, or you can copy the column to numerous column positions throughout a defined region.

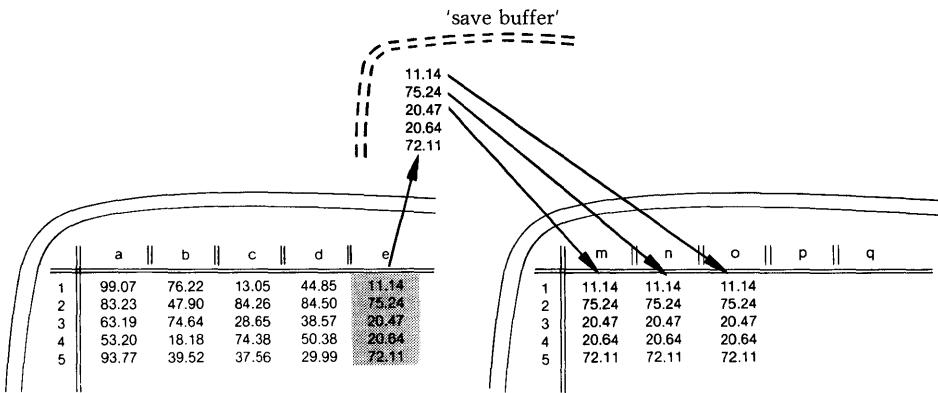
- **Single Column Position:** To copy to a single column position, move the cursor to the first entry position where you wish the column to begin. Type the YANKBACK command:



Perfect Calc copies the column to the spreadsheet, beginning at the position of the cursor, **overwriting** whatever data is present at that column location.

- **Multiple Column Positions:** To copy a column to several other column positions throughout a region, first define a 'line-wise' region as the copy destination, using the Mark Boundaries Procedure, Chapter VII, page 147.

For example, if you have copied column 'e' into the Save Buffer and now wish to duplicate it in columns 'm' through 'o'.



- Define a linear region from 'm1' to 'o1', using the Mark Boundaries Procedure.
- Type the YANKBACK Command:



Perfect Calc copies column 'e' to columns 'm' through 'o', **overwriting** whatever data is present at those column locations.

The COPY REGION Command Procedure

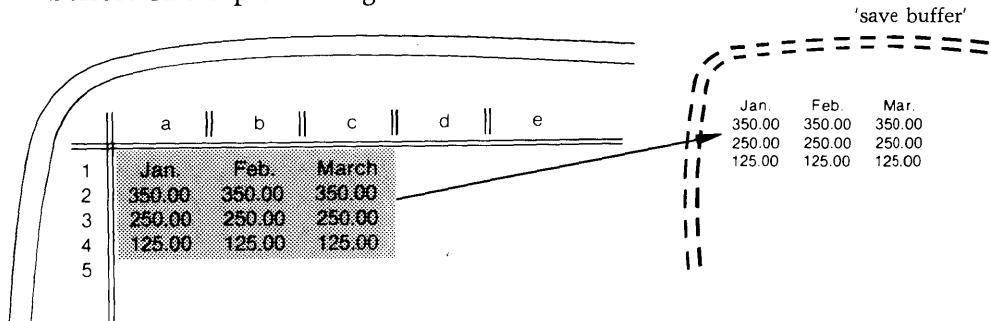
Copies a defined region to a new location in the spreadsheet.

Steps:

1. Define the region to be copied, using the Mark Boundaries Procedure, (Chapter VII, page 147).
2. Type the COPY REGION command:



Perfect Calc copies the region into the Save Buffer.

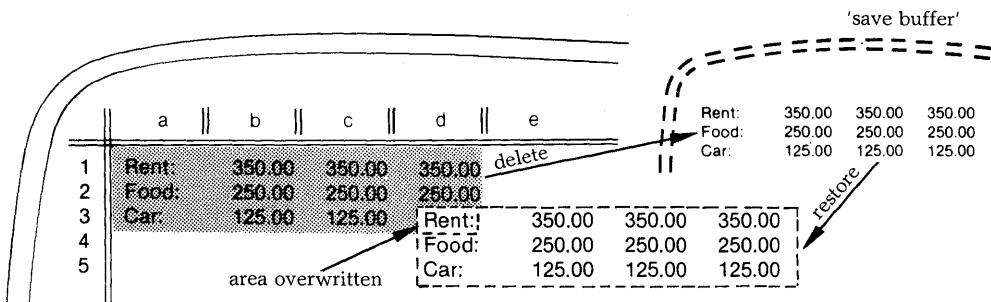


3. Move the cursor to the entry position where the copy of the region is to begin (i.e. this will be the upper, left-most entry position).

4. Type the YANKBACK command:



Perfect Calc copies the region to the new location on the spreadsheet beginning at the position of the cursor, **overwriting** whatever data, if any, may be present in that region.



Notice that in all of the above copy procedures, {Control----Y} is used to restore the deleted entry to a single location on the spreadsheet, whereas {Escape.....Y} is used to restore the deleted material to multiple positions throughout a region.

The COPY REGION command procedure, though it may at first appear inconsistent with this, actually treats a region of material as a single entity that is being copied to another single location on the spreadsheet. It is not possible to define a 'region of regions' to which a single region could be copied.

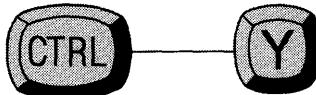
Relative or Absolute?

Whenever you attempt to restore the contents of the Save Buffer to the spreadsheet, either while moving, copying, or replicating data, Perfect Calc will scan the contents of the buffer for formulas. If the item being restored to the spreadsheet contains a formula, Perfect Calc must know whether the variable references it contains are 'relative' or 'absolute'. Perfect Calc will display the formula in the Echo Line and, while highlighting each of the variables in turn, will ask '...Relative?'

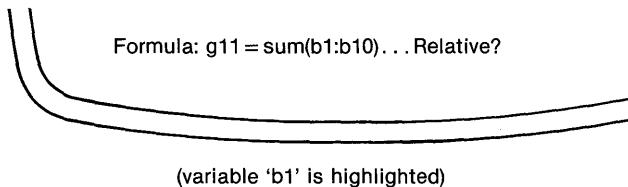
This concept of relative and absolute can best be made clear through an example:

Suppose that column 'b' contains 10 numbers (positions 'b1' through 'b10'), with position 'b11' containing a formula that computes the sum of these numbers.

Suppose that you wish to move column 'b' to column 'g' on the spreadsheet. After checking to see that no other formulas in the spreadsheet reference column 'b', you delete it into the Save Buffer using the DELETE COLUMN command (Escape.....K or Escape.....C). The spreadsheet adjusts itself by moving all other columns to the left by one and relabeling. You move the cursor to column 'g' and type the YANKBACK command form that is used for restoring deleted columns:



Perfect Calc immediately restores the deleted material to column 'g', replacing every entry position in the column except the formula in position 'b11'. It displays this formula in the Prompt Line, highlighting the first variable and asking the question:



Perfect Calc wants to know if this variable should be changed to 'g1' in order to reflect its new position in column 'g'? The answer, of course, is yes. Perfect Calc then highlights the second variable in the formula, 'b10'. Is this variable relative, and should it be changed to reflect its new column position, 'g10'? Again, the answer is yes.

Suppose now that the original formula had been:

Formula: b11 = sum(b1:b10)/a5

where column a contains data common to all subsequent columns.

Which of these variables would be relative and which absolute, if the same move were made from column 'b' to column 'g'? The answer is that the first two variables would be 'relative' while the third, 'a5', would be 'absolute', since it must continue to reference entry position 'a5', regardless of where it is located. Thus, our final formula would look like this:

Formula: g11 = sum(g1:g10)/a5

What About Mistakes?

If you should mistakenly identify the first two variables in our formula as absolute instead of relative, the formula would continue to reference entry positions 'b1' through 'b10', which either would compute to Error!, if these positions were empty, or would compute a false value, depending upon what entries were moved over and relabeled as column 'b'.

... Relative

When copying or moving formula entries, Perfect Calc must know whether the variables are relative and should be changed to reflect their new locations, or whether they are absolute references which must remain intact. You need tell Perfect Calc this only once for each formula; it remembers the information in case you later move or copy the formula again.

You will generally find the question/answer procedure quick and easy. The answers that you should give Perfect Calc in response to the '...relative?' question are as follows:

Relative or Absolute**Response****Action**

Yes, the variable is relative and should be changed to reflect its new position.



Repeat the previous answer for **every** formula variable in the Save Buffer. (Giving this as the initial answer will cause Perfect Calc to consider every variable in the Save Buffer as a relative variable. Note that this response overrides **all** subsequent questions, and should be used **only** when you are quite sure of its effect.)



Made a mistake! Start the questions over beginning with the first variable in this formula.



Cancel the YANKBACK command altogether, and return to the top command level.

(Any Other Key)

No, the variable is absolute, and should be copied to the spreadsheet with no change.

Chapter IX

SETTING UP THE SPREADSHEET AS YOU LIKE IT

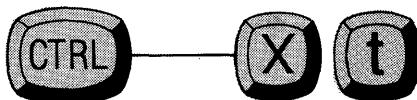
In this chapter, we look at the ways in which you can structure the spreadsheet visually and functionally to suit your particular requirements. Included are command procedures for such things as changing column widths, locking spreadsheet titles, and setting the frequency and order of formula recalculation.

The commands introduced in this chapter usually respond with messages requesting further information. If you are not sure what action is appropriate simply type a question mark "?" and Perfect Calc will display the possible responses.

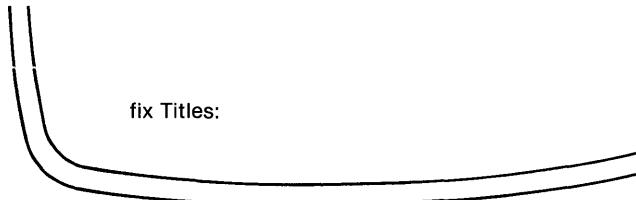
Fixing Titles in place

Your spreadsheet will normally contain a line of label entries across the top and down the left-hand side. For example, you might have a spreadsheet with the names of the months written across the top and budget categories down the left side (as in the spreadsheet 'budget83.pc', created in Tutorial 2). These entries constitute the 'titles' of the spreadsheet. Since Perfect Calc's spreadsheet is much larger than the small section displayed in the current window, moving the window away from the upper left-hand corner of the spreadsheet will necessarily cause these titles to disappear from view. You would be left not knowing what information was contained in the lines and columns, unless you had previously noted on a piece of paper what titles matched which lines and columns.

Perfect Calc allows you to fix into place the labels appearing in the top line of the spreadsheet, as well as those listed down the first column, using the FIX TITLES command:



When you give this command Perfect Calc responds with the message:



You must answer with one of the following:

Fix Titles Responses

Response	Action
a	Fix only the titles across line 1
d	Fix only the titles down column 'a'
b	Fix both sets of titles
<CR>	unlock any previously-fixed titles

You should note that when titles are fixed, you may not move the cursor onto their entry positions to delete or modify them. This is not usually a problem, and may even be considered a safety convenience, since it prohibits the accidental deletion of lines or columns which should remain unchanged.

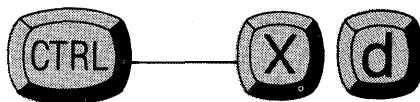
The main disadvantage of fixing titles comes when you try to print the screen image using the PRINT REGION command, (Control-----X, p), (see Chapter X, page 211), only to find that you cannot define a region which encompasses what you see on the screen.

In order to print a region which includes fixed titles, you must first unlock the titles before defining the region to be printed.

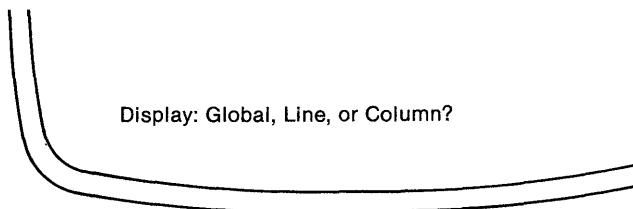
Changing the Numeric Display Format

Normally, Perfect Calc displays all numbers and formula values as figures showing two decimals. For example, the integer (number) '253' would be displayed as '253.00'.

This 'numeric display format' can be changed using the CHANGE DISPLAY FORMAT command:



Perfect Calc will respond to this command with the following message:



If the change is to apply throughout the spreadsheet, answer 'g', for 'global', but if it applies only to the current line or column, answer 'l' or 'c'.

Perfect Calc will acknowledge your answer in the Prompt Line. For example:



Display Format

At this point you can respond in one of several ways:

Response	Result
0-13	Typing a number between 0 and 13 will cause numeric values to be displayed showing this number of decimal places. If '0' (zero) is typed, decimal notation is not used and numeric values are displayed as integers.
s	Typing 's' causes all numeric values to be displayed in scientific notation, showing as many digits of precision as possible.
*	Typing a star or asterisk, '*', will instruct Perfect Calc to display the numeric values in 'graphic notation', that is, using as many stars as the number's value, starting at the lefthand edge of the column. If the numeric value is zero or less, nothing will be displayed. If the value is greater than the column width, the entire column will be filled with stars. (Generally you will select this option only when creating bar graphs in individual columns. (To create a bar graph, see the following page.)
f	Typing 'f' causes all formulas in the particular line, column, or spreadsheet to be displayed at their entry positions (in place of the values which they would normally compute and display). This option is especially useful for determining what formulas reference a line, column, or region about to be deleted or moved. Note that column widths may have to be changed prior to selecting this option, since some formulas may be too long to fit in the normal columns.

\$

Answering with a dollar sign '\$' instructs Perfect Calc to display numeric values in a dollars-and-cents format. However, before doing so Perfect Calc will first ask "Display cents?" Answering 'y' causes the cents portion to be retained. Answering 'n' causes the cents to be rounded to the next whole dollar. Thus, the number '-1234567.89' would display in the first instance as: '(1,234,567.89)', while in the second it would appear as '(1,234,568)'.

(Note: Parentheses are used to indicate a negative amount.)

Creating a Bar Graph

A bar graph is useful when a column of numeric data needs to be displayed in a manner which visually depicts the relationship of the various entries. The procedure for creating a bar graph is connected to the command procedure for changing the numeric display format just described.

In a bar graph, numeric values are displayed using a star or asterisk, '*'. A number will be represented by as many stars as its value warrants. Stars will be inserted in the entry position starting at the lefthand edge of the column. If the numeric value is zero or less, nothing will be displayed. If the value is greater than the column width, the entire column will be filled with stars.

For example, the following column of simple numeric values might be represented as follows:

```
1.00 *
2.00 **
11.00 *****
4.00 ****
25.00 *****
6.00 *****
0.00
4.00 ****
3.00 ***
2.00 **
1.00 *
```

It is evident that this graphic notation depends on the column width which will contain the stars. If there is a one to one correspondence between the stars and the numeric values, such as shown above, the column will have to be wide enough to display the largest value. Such a notation quickly loses its usefulness as the numeric values begin to exceed column widths of any practical size.

To generate a 'normalized' bar graph, which would accurately reflect the relative sizes of numeric values regardless of their size, use a procedure similar to the following:

Steps:

Suppose that listed in column 'b' of a spreadsheet, are the monthly sales figures for a company. You want to display these figures in graphical notation to better visualize the trends of business.

	a		b		c
1	January		8,250.73		
2	February		7,362.58		
3	March		6,782.19		
4	April		9,325.23		
5	May		10,527.22		
6	June		15,629.90		
7	July		23,009.25		
8	August		17,008.91		
9	September		15,872.19		
10	October		13,098.25		
11	November		10,980.72		
12	December		13,980.77		

1. Create an empty column to the right (or left) of the column containing the numeric data. (Use the OPEN COLUMN command, (Escape.....O), if an empty column does not already exist.)
2. Make the column 15 characters wide, using the CHANGE COLUMN WIDTH command (Chapter V, page 79).
3. Type the CHANGE DISPLAY FORMAT command (Control----X, d), followed by 'c' for column, and '*' for graphic notation format. The empty column is now ready to display numeric values in graphic notation.

4. 'Scale' the numeric data in the adjacent column (column 'b') using the MAX function (Chapter VI, page 107). For example, suppose the data occupies positions b1 to b10 in adjacent column 'b'. With the cursor in position 'c1' of the empty column 'c', construct a formula of the form.

FORMULA: $c1 = <\text{column-width of } 'c'\text{}> * b1 / \max(b1:b10)$

where 'x' is the line number of the formula (i.e. '1'), and 'n' is the line number of the last entry pair in column 'b', which holds the numeric values.

For column 'c' which has a column width of fifteen characters, this formula is:

$$c1 = 15 * b1 / \max(b1:b10)$$

5. Enter this formula into entry position 'c1' using the COPY ENTRY command:

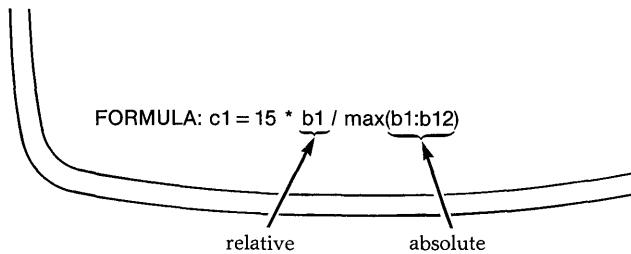


6. Using the Mark Boundaries Procedure (Chapter VII, page 13), define a columnar region from positions 'c2' to 'c10'. (Simply place the cursor on 'c2' and press Escape.....<space bar>. The Prompt Line will display the message 'Mark set at c2'. Next, move the cursor to 'c10'.)

7. Replicate the formula in every position in column 'c' by entering the YANK-BACK command form:



8. Perfect Calc will ask you to identify the relative variables in the formula:



Perfect Calc enters the formula into positions 'c1' to 'c10', simultaneously constructing the bar graph, which should look like this:

	a		b		c
1	January		8,250.73		***
2	February		7,362.58		***
3	March		6,782.19		***
4	April		9,325.23		****
5	May		10,527.22		****
6	June		15,629.90		*****
7	July		23,009.25		*****
8	August		17,008.91		*****
9	September		15,872.19		*****
10	October		13,098.25		*****
11	November		10,980.72		***
12	December		13,980.77		*****

It should be noted here that enlarging the width of column 'c'—which holds the stars may increase the readability of the bar graph. For example, changing the column width of 'c' from 15 characters to 25 characters will make the bar graph look like this (be sure to adjust the formula):

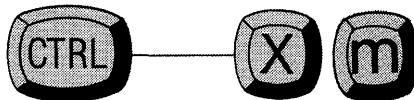
	a		b		c
1	January		8,250.73		*****
2	February		7,362.58		*****
3	March		6,782.19		*****
4	April		9,325.23		*****
5	May		10,527.22		*****
6	June		15,629.90		*****
7	July		23,009.25		*****
8	August		17,008.91		*****
9	September		15,872.19		*****
10	October		13,098.25		*****
11	November		10,980.72		*****
12	December		13,980.77		*****

Note: To insert space between the graph and the numeric display, use the INSERT COLUMN command (Escape.....o), afterwards reducing the column width of the new column to '2', using the CHANGE COLUMN WIDTH command (Control----X, w).

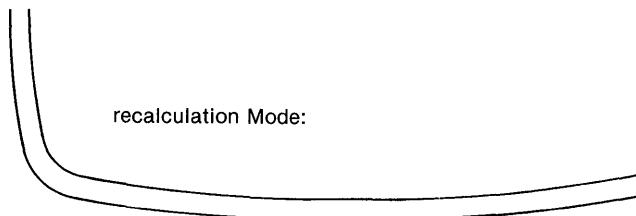
Recalculation Modes

Each time you enter a new number or formula, or change an existing one, Perfect Calc performs a complete recalculation of the spreadsheet. Normally this occurs on a line-wise basis. That is, all the formulas in line 1 are recomputed, followed by all the formulas in line 2, line 3, and so forth. Occasionally this may lead to problems if such an updating scheme would cause some of your formulas to reference values which appear on subsequent lines (recall the problem of forward and circular referencing discussed in Chapter VI, page 117), or in buffers that are not currently activated.

As a spreadsheet gets larger, updating time increases, meaning that while entering a large amount of data, you may find yourself waiting through unneeded or unwanted calculations each time you change or add one entry. You may change this using the CHANGE RECALCULATION MODE command:



Perfect Calc responds to this command with the message:



At this point you may enter one of the following responses:

Recalculation Mode Responses

Response	Action
l	Recalculate formulas by lines (this is the default mode).
c	Recalculate formulas by columns .
a	Recalculate the entire spreadsheet automatically whenever the value of a number or a formula entry is changed (this is the default mode).
m	Recalculate the spreadsheet only upon a manual request to do so. (See Chapter IX, page 16 for MANUAL RECALCULATION command [!].)
r	Recalculate only the currently defined region . (See below for further comments on the use of this option.)

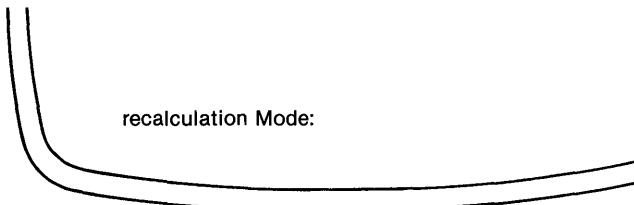
It should be noted here that the first two recalculation modes—lines and columns—are mutually exclusive, as are the second two—automatic and manual. This means that if you recalculate, for example, by lines you may not calculate by columns also. However, recalculation can be performed automatically by lines (or columns), or manually by lines (or columns).

Recalculating by Region

Recalculating by region is very useful when you have a very large spreadsheet which is really a series of loosely-connected smaller spreadsheets. Recalculation time can be reduced significantly by instructing Perfect Calc to recalculate a particular local region only.

The region option can co-exist with any of the other selections, so that it is possible to recalculate within a region either automatically by lines, or manually by columns (and vice versa).

Be sure that you have defined the region using the Mark Boundaries Procedure (Chapter VII, page 147) **before** selecting the 'r' selection from Perfect Calc's prompt:



Perfect Calc will remember the region you define. Thereafter every calculation it performs, whether automatically or manually, by line or by column, will be done on the region that you have defined. To change this region, you must perform the CHANGE RECALCULATION MODE command again.

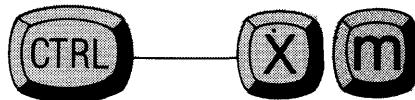
Note: To resume recalculation of the **entire** spreadsheet, simply select 'r' with a single element region defined. That is:

1. Execute the first half of the Mark Boundaries Procedure, by typing:



Perfect Calc sets a forward boundary marker at the position of the cursor.

2. Without moving the cursor to create a region, give the CHANGE RECALCULATION MODE command, selecting 'r' for region:



Perfect Calc assumes the new region to be the entire spreadsheet.

The MANUAL RECALCULATION Command

Perfect Calc provides a command that will force recalculation of formulas, regardless of the recalculation mode currently being used. Simply type:

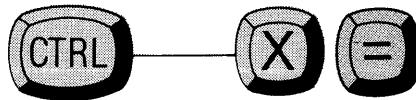


Perfect Calc recalculates all formulas in the spreadsheet (or defined region of the spreadsheet). Use this command to invoke a recalculation when the Manual Recalculation Mode is in effect.

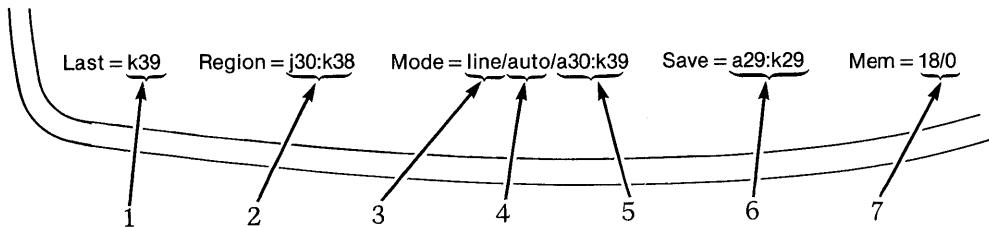
The STATUS Command

The many options in force within a spreadsheet at any one time are difficult to remember, and Perfect Calc does not continuously display them in order to avoid cluttering the screen.

A command is available, however, to display the current status information pertaining to the current spreadsheet. At any time type the STATUS command:



Perfect Calc will display in the Prompt Line a row of information like the following:



These elements represent the following:

1. Location of the lowest, right-most entry of the spreadsheet ('k39' in this example).
2. The currently defined region between the mark and the cursor (j30:k38).
3. The direction of recalculation mode (line).
4. The current recalculation mode (automatic).
5. The current region in which calculations are made (a30:k39).
6. The current contents of the Save Buffer (a29:k29).
7. Kilobytes of main memory available (18) / Kilobytes of virtual memory temporarily stored on disk (0).

Chapter X

STORING AND PRINTING YOUR SPREADSHEET

STORING A SPREADSHEET

After you have finished entering and recalculating data on your spreadsheet you will want to store your spreadsheet on disk for later reference and use. Perfect Calc will save your spreadsheet as a 'file', under a unique filename. In saving your spreadsheet as a file, Perfect Calc creates in effect a 'master copy', which cannot thereafter be erased or destroyed unless you specifically request it.

When you have finished creating or modifying a spreadsheet, you may either:

- Overwrite the original (if one exists) with the newly modified copy, or
- Save the copy in a new file by itself, thereby retaining both versions of your spreadsheet.

File Commands

The following commands instruct Perfect Calc to save files.

The SAVE FILE Command

Immediately copies the spreadsheet being edited to disk, **overwriting** whatever original version of the spreadsheet was there before.



The system message "Writing <filename> . . ." is displayed briefly, after which Perfect Calc returns you to the spreadsheet to continue your calculations.

Use this command frequently, and at any time during the editing process, to guard against accidental losses of data on the spreadsheet.

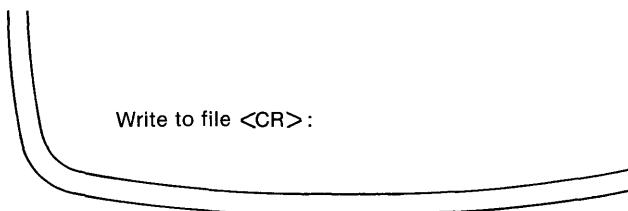
The WRITE FILE Command

Saves the spreadsheet currently being edited. However, this command provides the option of either overwriting the master copy, or of storing your updated spreadsheet in a different file under a new name.

At any time during the editing process type:



Perfect Calc responds with the message in the Prompt Line:



At this point you have two options:

- | | |
|-------------------|---|
| Write File | You can create a new file, and Perfect Calc will store the updated spreadsheet in that file. Simply type the new filename followed by a carriage return, <CR>. (Remember that we suggest the filename end with the extension '.pc'.) |
| Save File | You can overwrite the old master file copy (This option is identical to the SAVE SPREADSHEET command.) Simply type a carriage return, <CR>. Perfect Calc immediately copies the spreadsheet to disk, overwriting whatever version of the spreadsheet was previously there. |

Printing Your Spreadsheet

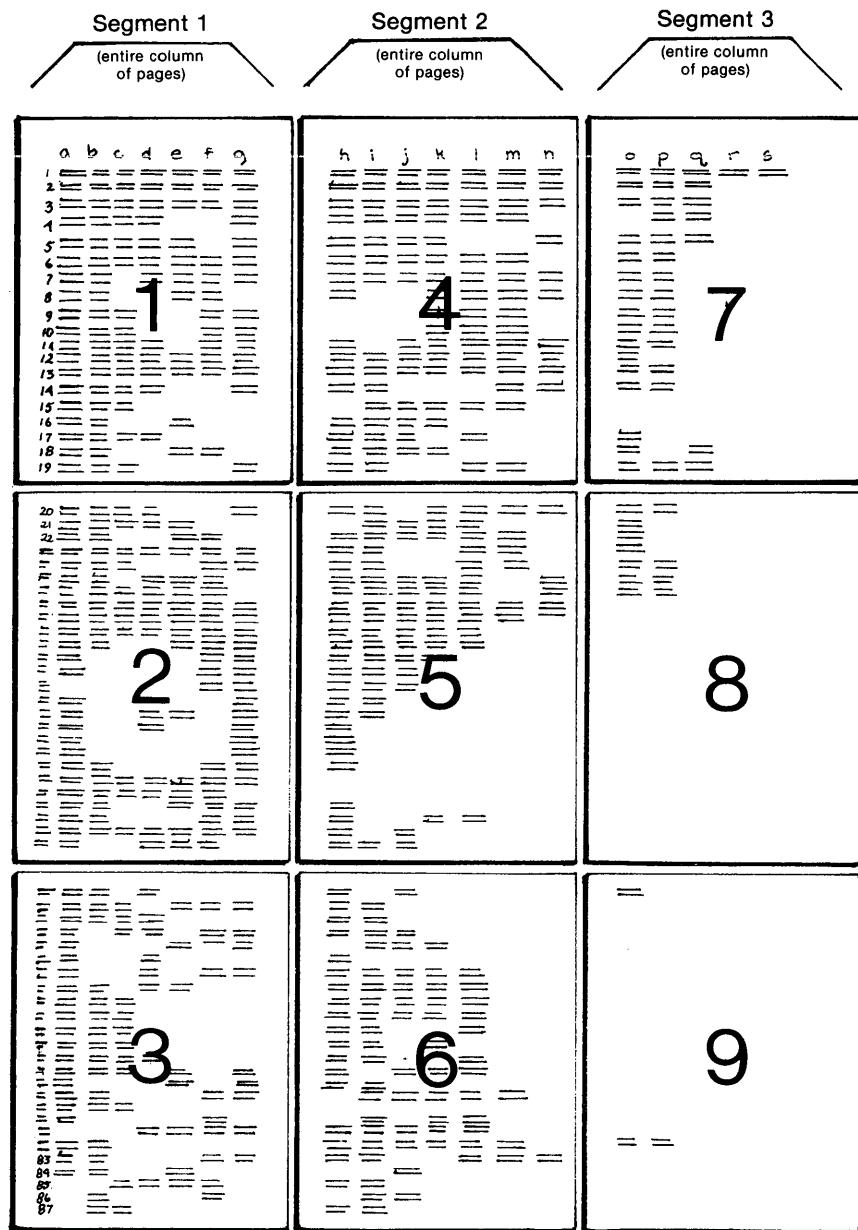
Perfect Calc allows you to print any portion, or region, of your spreadsheet that you wish. If you need only a certain segment of the spreadsheet for a report, it is possible to define this region using the Mark Boundaries Procedure, and print it alone.

Although you may normally wish to print hardcopy of the spreadsheet data, it is possible to 'print' to a disk file, a procedure that allows you to later edit the file using Perfect Writer. This is convenient if you wish to incorporate Perfect Calc data with a written report, but first need to add explanatory text to the Perfect Calc lines and columns.

In all of these, the rule followed is: "what you see is what you get", meaning that the number of decimal places, column widths, and so forth will be printed just as displayed on the Perfect Calc screen. If you wish to change any of these parameters, you must do so before initiating the print operation.

Output Format

The normal action of Perfect Calc's print commands is to output all of the spreadsheet in printer-wide pages, dividing the spreadsheet into segments depending upon the page-width which you specify. If your spreadsheet occupies 19 columns and 87 lines, and each spreadsheet column is 11 characters wide, one printed segment will contain seven columns and 87 lines (which presumably will require two actual paper pages to print). After printing the first segment of the spreadsheet (the first seven columns), Perfect Calc will then proceed to print the second segment (the second seven columns), and then the last segment (the remaining five columns). If you were to line up the printed pages side-by-side, the entire 19-column-by-87 line spreadsheet would appear intact.

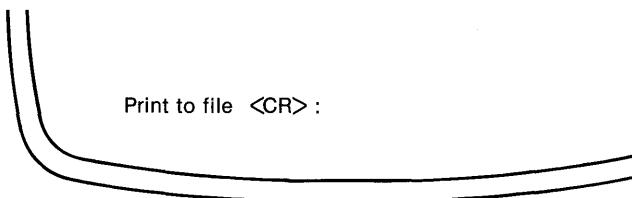


The PRINT SPREADSHEET Command

Causes the entire spreadsheet to be printed, either in hardcopy or to a disk file for subsequent editing with Perfect Writer. At any time during editing, type:



Perfect Calc responds with the message in the Prompt Line:



At this point you have two options:

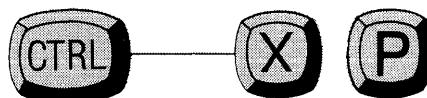
Perfect Writer File You can enter a filename, in which case Perfect Calc will 'print' the spreadsheet to a disk file for use with Perfect Writer. Remember to use a different filename suffix. For example, change '.pc' to '.mss'. Also, print the file to a disk that is accessible to Perfect Writer.

Hardcopy You can print the spreadsheet directly to hardcopy. Simply type a carriage return, <CR>. Perfect Calc outputs the spreadsheet on your printer.

Note: A default printer page width of 80 columns is specified during the printer configuration. If your printer is capable of printing a wider page than this, you must change this printer specification (see Appendix A).

The PRINT REGION Command

Causes a defined region of the spreadsheet to be printed, either in hardcopy or to a disk file for subsequent editing with Perfect Writer. **After** defining the region to be printed using the Mark Boundaries Procedure (see Chapter VII, page 147), type:



Perfect Calc responds with the message:



At this point you have two options as before. See opposite page.

SAVING A FILE FOR USE BY PERFECT WRITER

TUTORIAL

Frequently, you may want your spreadsheet data to be included with a report or other document that you are preparing with Perfect Writer. The simple way to do this is to print the spreadsheet to a disk file that can be conveniently accessed by Perfect Writer. (This may not be the disk you are using for Perfect Calc.) With Perfect Writer you will be able to alter the spreadsheet to conform to your written report, adding explanatory information, notes, and generally enhancing the figures with text and formatting of your own.

Suppose that you have been working with the sample spreadsheet 'budget83.pc', which we have been developing throughout the previous tutorials. You now wish to transfer that spreadsheet to a file that you can later modify using Perfect Writer. You perhaps wish to include the spreadsheet (or a portion of it) in a letter, or other written document.

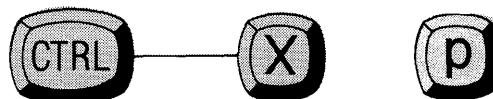
Perfect Writer uses a standard line length of 70 characters. Therefore, you should set the printer's width for Perfect Calc to 70 when you print files for Perfect Writer. This can be accomplished by using the 'PCCONFIG' program (see Appendix A, page 318).

Steps:

1. Type one of the print file commands, depending upon how much of the spreadsheet you wish to transfer to Perfect Writer's disk file:



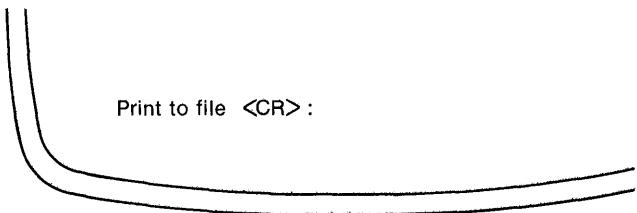
(PRINT SPREADSHEET command)



(PRINT REGION Command)

[This second command must be preceded by the Mark Boundaries Procedure, see Chapter VII, page 147].

Perfect Calc will respond in the Prompt Line:



Enter the name of the disk file to which the spreadsheet will be printed, followed by a carriage return <CR>. Remember to use the filename suffix that is required by Perfect Writer, '.mss'. For example, the Perfect Calc filename would be changed from:

budget83.pc

to

budget83.mss

Also, remember to add the disk drive designation holding the disk that Perfect Writer will use. For example:

b:budget83.pc
disk drive designation

Perfect Calc prints the spreadsheet, or the region of the spreadsheet which you have defined, to the new file, afterwards returning you to the spreadsheet to continue editing.

When editing this screen image with Perfect Writer, it will be necessary to enclose it within a VERBATIM environment. Otherwise, when printing the file, Perfect Writer will group the various lines and columns and figures into 'paragraphs', thus destroying the original format created by Perfect Calc. The VERBATIM environment will tell Perfect Writer to print the spreadsheet exactly as it appears on the screen. For example:

@Begin(verbatim)

Month	January	February	March	April	May	June
Income	2,500.00	2,500.00	2,500.00	2,500.00	2,500.00	2,500.00
<hr/>						
Rent	425.00	425.00	425.00	425.00	425.00	425.00
Car	35.00	35.00	35.00	35.00	35.00	35.00
Food	235.78	189.00	95.43	154.34	175.00	134.00
Misc. exp.	457.00	325.00	552.45	605.11	389.99	531.00

@ End(verbatim)

'verbatim'
environment
required for Perfect
Writer files.

END OF TUTORIAL

Part II

ADVANCED MULTIPLE SPREADSHEET ANALYSIS

In This Section

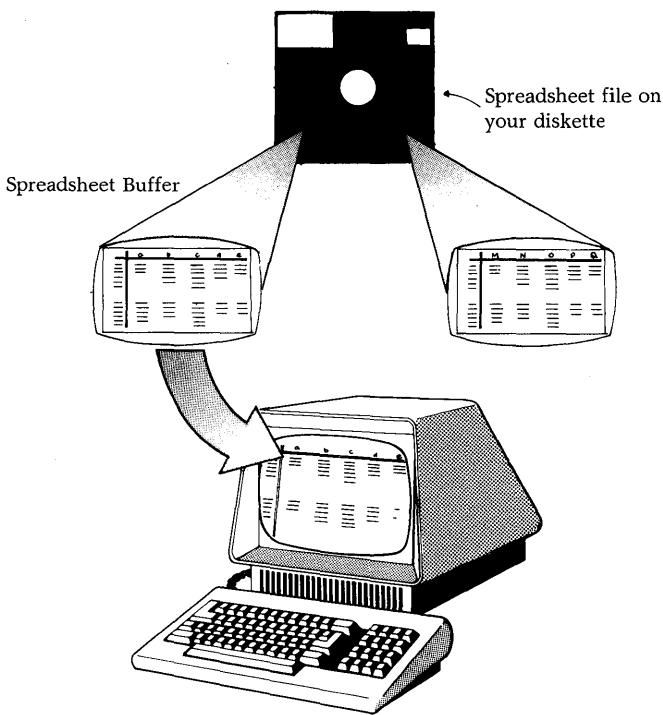
- Multiple Spreadsheet Display
- Multiple Spreadsheet Analysis
- Associative Files

Chapter XI

WORKING WITH MULTIPLE SPREADSHEETS

Introduction

As you may recall from the brief discussion in Chapter II, the editing of a spreadsheet takes place in a temporary workspace in machine memory, called a buffer. A spreadsheet held in a buffer represents merely a copy of the original spreadsheet which is stored as a file on disk (unless, of course, it is a new spreadsheet that has not yet been saved, in which case it represents the **only** copy). It is possible to have more than one spreadsheet buffer in your computer's active memory at a time, as the diagram on the next page illustrates.



In this figure two spreadsheets have been copied into separate buffers. One is being displayed to the computer screen, and one is waiting in another active buffer to be displayed.

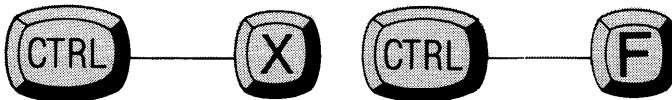
As has been mentioned, up to **seven** spreadsheets may be held at one time in separate spreadsheet buffers. Perfect Calc provides a number of file and buffer commands which allow you to create and delete buffers, copy spreadsheets into and out of buffers, switch the viewing screen from one buffer to another, and to list the names of those buffers currently in use.

FILE AND BUFFER COMMANDS

The FIND FILE Command

Retrieves a spreadsheet from disk storage and places it in its own spreadsheet buffer for you to edit. If the spreadsheet you request is already in a buffer, this command switches you directly from the buffer currently being used to the buffer containing the spreadsheet you requested, without performing any disk read operation.

At any time during editing type the FIND FILE command:



Perfect Calc responds with the message:



Enter the name of the file followed by a carriage return. Perfect Calc retrieves the file, placing it in a buffer for you to use.

Note: When retrieving a file from disk storage, Perfect Calc always attempts to place the file in a buffer having the same name. If you request a spreadsheet whose name is associated with a buffer that is already in use, Perfect Calc will display the message:

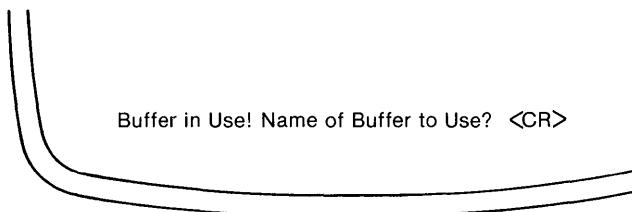
Buffer in Use! Name of Buffer to Use? <CR>

This circumstance often arises when requesting a second 'original' copy of a spreadsheet you have been editing, perhaps to compare the changes you have been making.

For example, you have been working with the spreadsheet 'ourtaxes.pc', and want to examine the original spreadsheet again to check figures which you have since deleted in the working copy. The original version on disk is also called 'ourtaxes.pc'.

Using the WRITE FILE command you save the present spreadsheet to disk under another filename. After this save operation, Perfect Calc returns you to the spreadsheet to continue editing. (Notice that although the filename has changed, the buffer name has not!)

Using the FIND FILE command, you next instruct Perfect Calc to retrieve the original version of the spreadsheet from disk and to place it into another buffer. However, you get the message:



In retrieving a second copy of the spreadsheet 'ourtaxes.pc', Perfect Calc has attempted to create a buffer with the name 'ourtaxes', only to find that a buffer with that name already exists!

When this situation arises, you must supply a different buffer name in which to place the second copy of the document. If you answer by hitting a carriage return, <CR>, Perfect Calc will use the existing buffer, **overwriting** the copy of the spreadsheet that you have been modifying! To avoid this, supply a name for the second buffer—in this case, something like 'ourtax2'.

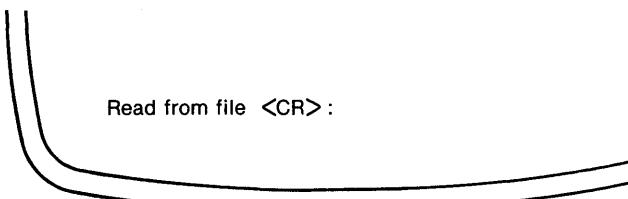
Further Note: The FIND FILE command can be used to create a new spreadsheet. If the file that you request does not exist, Perfect Calc will create a new spreadsheet in an empty buffer, the name of which will be the first eight characters of the filename you have given. Any subsequent 'save' command will create the file you name on disk and copy your new spreadsheet to it.

The READ FILE Command

Retrieves a spreadsheet file from disk and reads it into the current buffer. If a spreadsheet file is already present in the buffer, this file will be partially overwritten. The buffer filename is updated to reflect the presence of the new file. (This command is especially useful for manipulating file overlays—see Chapter XI, page 234.) Type:



Perfect Calc responds with the message:



Type the name of the file which is to overwrite the current file, followed by a carriage return, <CR>.

Perfect Calc displays the message:



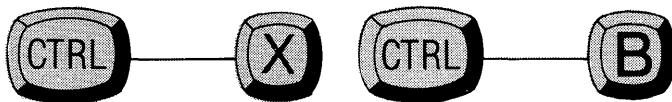
The new file is read into the current buffer, overwriting, or partially overwriting, whatever spreadsheet file is already present.

The BUFFERS DIRECTORY Command

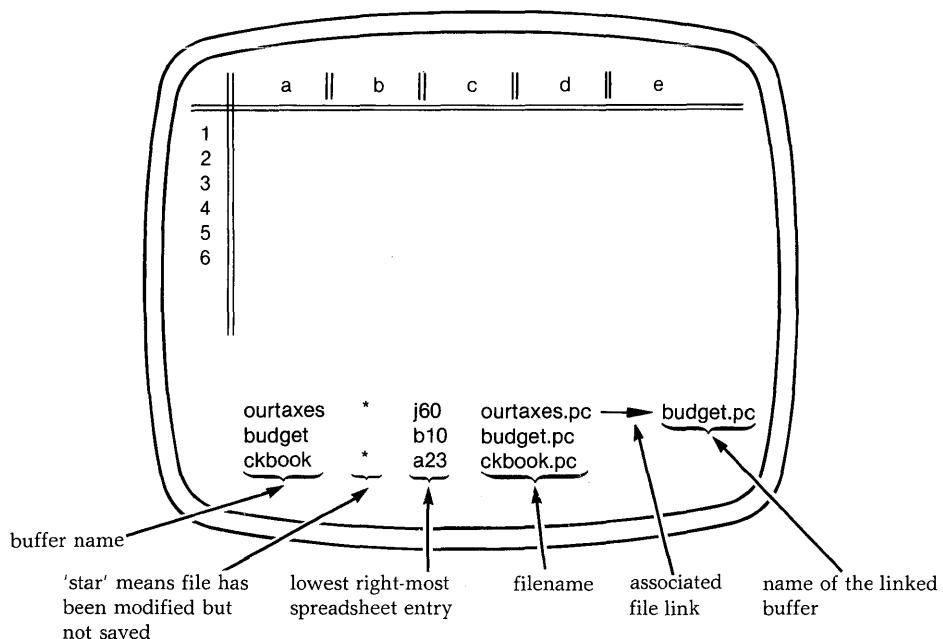
Instructs Perfect Calc to display a directory of all the buffers in use during an editing session. Directory information includes:

- the name of the buffer;
- a star (*), the presence of which indicates that the contents of the buffer have been modified since the last read, save, or write command was given;
- the coordinates of the lowest, right-most entry in the spreadsheet (indicating relative size);
- the filename of the spreadsheet held in the buffer; and
- whether any files are associated.

At any time during spreadsheet analysis type the BUFFERS DIRECTORY command:



Perfect Calc immediately displays a directory of the active buffers in the lower left-hand corner of the screen. For example:

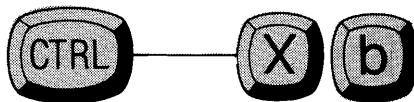


To erase the buffer directory, issue any command that will cause the screen to redraw, such as Control----L (or Escape.....Control----L for the alternate command set).

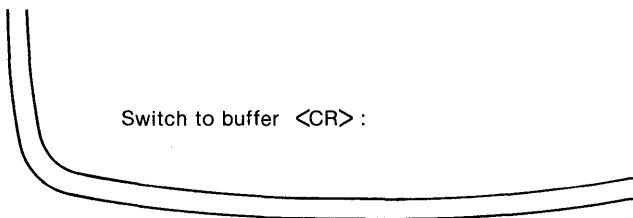
The SWITCH BUFFERS Command

Causes the window to switch from one active buffer to another. Also, will switch to an empty buffer for creating a new spreadsheet.

At any time during editing type the SWITCH BUFFERS command:

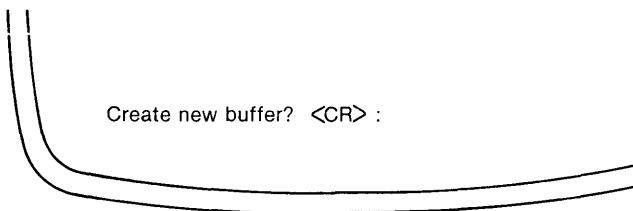


Perfect Calc responds with the message:



Enter the name of the buffer you wish to use, followed by a carriage return. (Remember that the buffer name is typically only the first portion of the spreadsheet filename). Perfect Calc switches you to the buffer you have named.

If the buffer you name does not exist, Perfect Calc will respond with the message:



Answering 'y' causes Perfect Calc to create a new empty buffer identified by the name you have given, while answering 'n' cancels the original command.

Note: The spreadsheet filename for any new buffer will be 'default.pc', and will remain such until you assign a new filename to it, usually when writing the spreadsheet (see Chapter X, page 207).

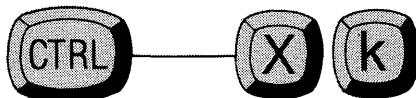
Further Note: The SWITCH BUFFERS command can be used to switch back and forth quickly between two buffers, without having to specify the buffer names each time.

To do this enter only a carriage return, <CR>, when the message 'Switch to Buffer <CR>:' is displayed. Perfect Calc will switch to the buffer that was last switched **from**.

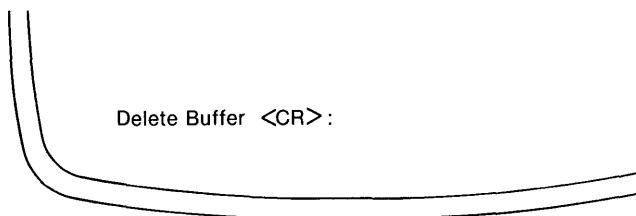
The DELETE BUFFER Command

Deletes a buffer from the set of buffers currently being used. When a buffer is deleted, all spreadsheet information contained within the buffer is lost.

At any time during editing type the DELETE BUFFER command:



Perfect Calc will respond with the message:

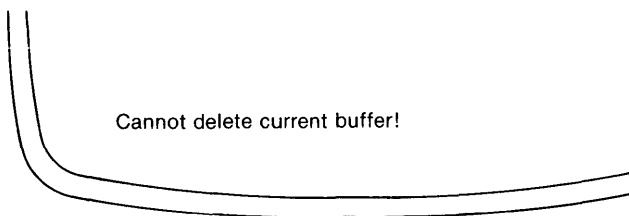


Enter the name of the buffer followed by a carriage return.

Perfect Calc deletes the buffer you have named.

Note: If the file in the buffer has been altered since the last save/write operation, Perfect Calc protects you by requesting confirmation with the message 'Delete modified buffers?'

Further Note: If you should type the name of the current buffer, Perfect Calc will respond with the message:

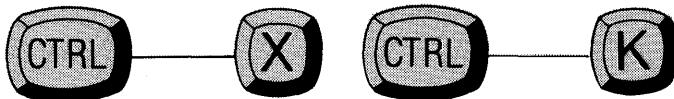


Perfect Calc will delete only buffers that are not currently being viewed. To delete the buffer being viewed you must switch to another buffer and reissue the delete command from there.

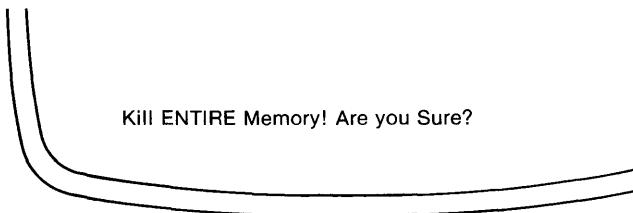
The CLEAR MEMORY Command

Deletes every active buffer, completely erasing all spreadsheet data in machine memory, returning the screen to a new spreadsheet in the buffer default. All deleted data is lost and unrecoverable. To guard against inadvertent loss of spreadsheet data, save desired files first using the SAVE FILE command or the WRITE FILE command (see Chapter X, pages 206 and 207).

At any time during editing type the CLEAR MEMORY command:



Perfect Calc responds with the message:



Typing 'y' causes Perfect Calc to delete all currently active buffers, returning you to a new, blank spreadsheet in the 'default' buffer. Typing 'n' cancels the command.

Referencing Entry Positions in Other Buffers

In general, spreadsheet data held in a buffer is used only in that buffer. That is, column widths, format specifications, labels, numbers, and formulas are valid only within the buffer that holds the particular spreadsheet, while commands issued within a buffer affect only that buffer. However, there are times when it is desirable to share information between buffers.

Perfect Calc provides a convenient mechanism for allowing spreadsheet formulas in one buffer to reference variables in a spreadsheet contained in another buffer. As you recall, a formula consists of numbers, variables, operators, and/or functions, with the variables referring to entries in the current spreadsheet. The reference is made using the coordinates that identify the entry position. For example, the formula

$$a17 = \text{sum}(a1:a16)$$

calculates the sum of all entries from 'a1' to 'a16' in the spreadsheet of the current buffer, displaying the result in entry position 'a17'. -

To reference a variable in another buffer, place the variable in square braces, [], preceded by the name of the buffer which contains it. For example:

$$c23 = a17 + \text{budget}[f43]$$

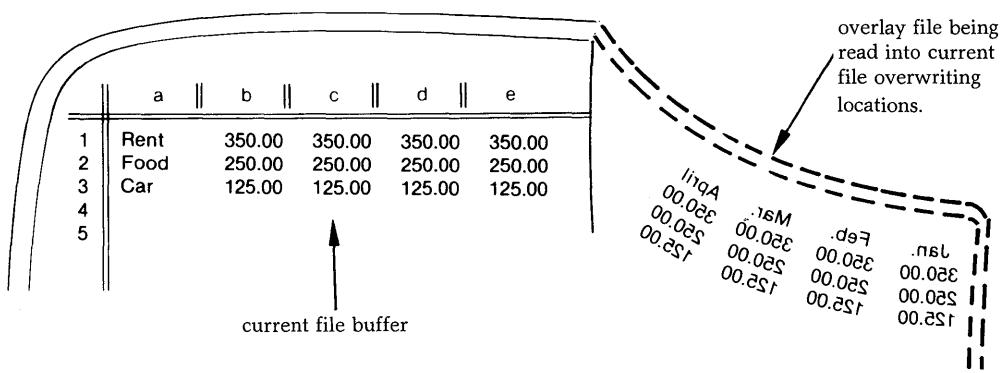
This formula computes the sum of two variables, one of which is held in a buffer other than the current buffer. The variable 'a17' is held in the current buffer, while 'f43' is in the buffer called 'budget'. Whenever the current spreadsheet is recalculating, Perfect Calc, to solve the formula, will reference these variables, afterwards displaying their sum in 'c23' of the current buffer.

An important point to remember about this capability is that references outside the current buffer may become outdated and unreliable if the other buffers (and associated files) are not linked to the current buffer.¹

¹ See Chapter XIII, Associated Files, for the procedures on linking files.

Spreadsheet Overlays

Besides allowing spreadsheets to reference each other, Perfect Calc offers the capability of allowing spreadsheets to 'overlay' one another. While viewing one spreadsheet it is possible to call another spreadsheet into the **same** buffer. Only those entry positions in the overlay spreadsheet which contain data—formulas, numbers, labels—will overwrite on the current spreadsheet. When the overlay is complete, the current spreadsheet automatically recalculates, using the values supplied to it by the overlay.

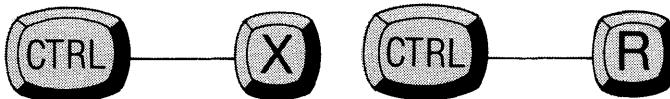


Normally, overlays involve a single 'master' spreadsheet, accompanied by numerous individual 'member' spreadsheets. The master spreadsheet frequently contains lengthy and complicated computations, while the member spreadsheets are simple, but supply values pertaining to individual objects, people, activities which must be calculated by the master spreadsheet. Calculations involving individual employee records, stock inventories, and more, find ideal application in spreadsheet overlays.

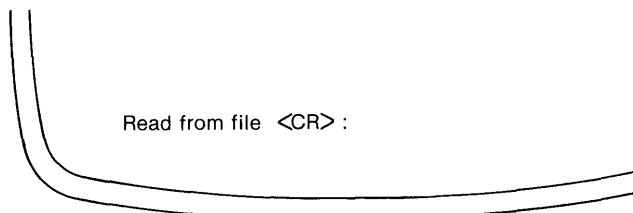
To use this overlay capability proceed as follows:

Steps:

1. Structure the master and member spreadsheets such that their numeric display format, column widths, etc. are identical, and that the data entries that will overwrite are positioned in the same entry boxes. If this is not done exactly, the spreadsheets will overlay incorrectly, causing, perhaps, other valuable data to be lost on the master spreadsheet. Each spreadsheet will reside in a separate file under a unique filename.
2. Call the master spreadsheet into the current buffer.
3. Type the READ FILE command:



Perfect Calc responds with the message:



4. Type the filename of the first member spreadsheet to be overlaid, followed by a carriage return, <CR>.

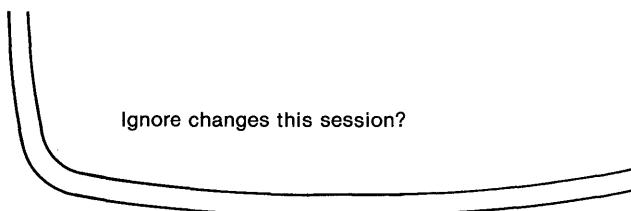
Perfect Calc retrieves the member spreadsheet from disk and overlays it upon the master spreadsheet in the current buffer.

When the overlay is complete, Perfect Calc recalculates the master spreadsheet, using the values supplied by the member spreadsheet. Its results can then be output to disk files or printed to hardcopy.

(Repeat steps 3-4 above to overlay a second, third, fourth, etc. member spreadsheet.)

Quitting with Multiple Buffers

When you have finished editing multiple documents and wish to quit using Perfect Calc, typing the QUIT command (Control-----X, Control-----C), will, if you have not yet stored any of the spreadsheets contained in active buffers, result in the following message from Perfect Calc:



A 'y' will erase all of the modifications you have just made to the current spreadsheet as well as modifications to spreadsheets in other buffers. 'n' will cancel the QUIT command, giving you opportunity to save your newly edited spreadsheets.

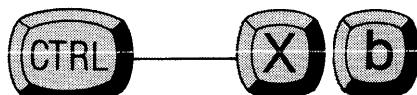
To save the spreadsheets existing in various buffers:

1. Enter the BUFFERS DIRECTORY command:



Perfect Calc will list the active buffers in the lower left corner of the screen. An asterisk, '*', beside a buffer name indicates that the spreadsheet contained in the buffer has not been saved since it was last modified.

2. Switch to the first buffer whose spreadsheet you wish to save, using the SWITCH BUFFERS command:

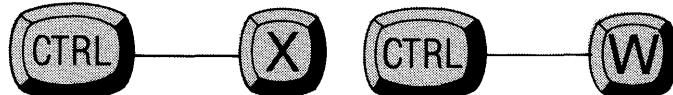


3. Save the document by using the SAVE FILE Command or the WRITE FILE command, depending on whether you wish to overwrite the old file or create a new file.

SAVE FILE:



WRITE FILE:



4. In this way continue switching to various buffers, saving the spreadsheets that you wish.
5. Finally, give the QUIT command (Control---X, Control---C) again, this time ignoring the modified spreadsheets that you did not wish to save.

Chapter XII

MULTIPLE FILE DISPLAY

Introduction

Perfect Calc offers the unique feature of allowing you to divide the single window of your computer console into two separate windows displaying either different parts of the same spreadsheet, or two spreadsheet files held in different buffers.

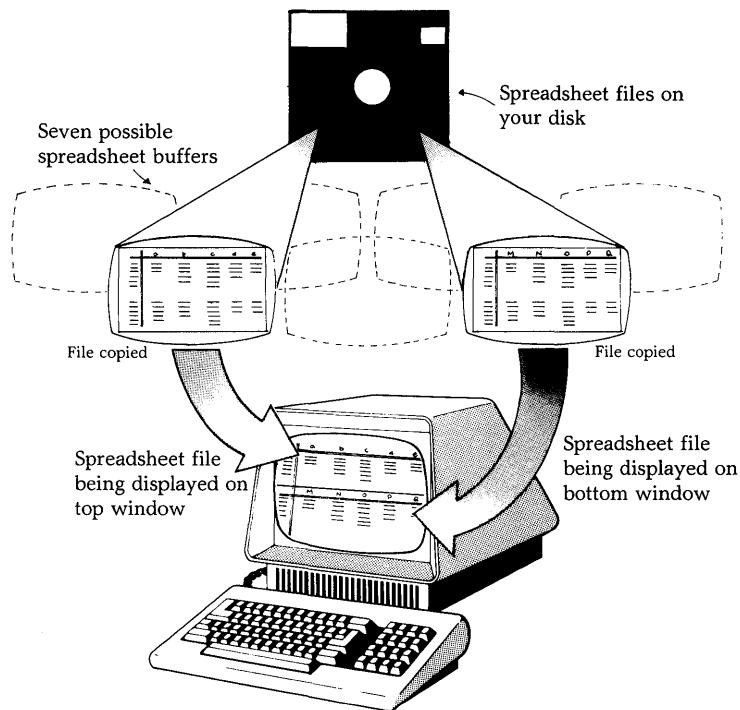
This multiple file display feature is useful not only for simultaneously examining the data and operation of widely separated segments of the same spreadsheet, but also for calculations between spreadsheets whose formulas reference each other. You can change values and formulas in one location and watch the results in the other (faraway) location. You can alter a variable in one spreadsheet and observe the results in another spreadsheet.

Moreover, for those rare occasions when you may need to copy or transfer lines and columns of material between spreadsheets, the multiple file display capability makes the task amazingly simple!

Perfect Calc's window can be split either vertically or horizontally, and can be variously proportioned, with one screen larger or smaller than the other.

When a second window is created it initially displays an identical copy of the spreadsheet which was being edited in the original, full-screen window. Although the cursor is positioned in only one of the two windows, the spreadsheet being displayed derives from the **same** buffer.

However, it is possible to display a different spreadsheet in each window, each held in a separate buffer as the following illustration shows. In this case, changes made in one window only affect the spreadsheet displayed in that window. (Of course, if the buffers are 'associated' or otherwise related through cross-references, changes to data in either window will affect related data in the other window. See Chapter XI, page 233.)



WINDOW COMMANDS

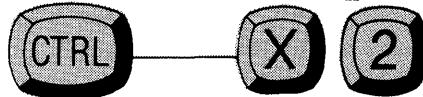
The following pages present the various command procedures used to create and manipulate Perfect Calc's multiple file display. (Note that all of Perfect Calc's other commands continue to function just as they do when only a single window is being used.)

The CREATE TWO WINDOWS Command

Splits the screen display, changing the single screen window into two windows, either horizontal or vertical.

Steps:

1. Position the cursor on the column or line where you want the split to occur and type the CREATE TWO WINDOWS command:

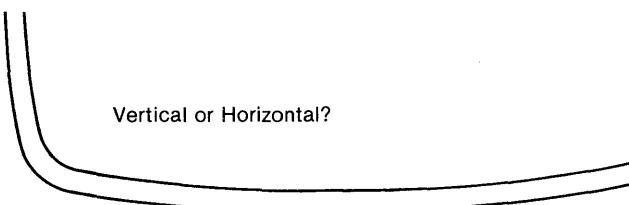


If the cursor is positioned along the top or bottom edge of the screen, Perfect Calc will assume that you want the split to be vertical, since a horizontal split at either of these positions would leave no room for another window.

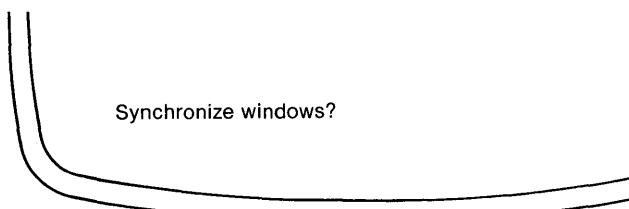
Similarly, if the cursor is positioned along either the left or right edge of the screen, Perfect Calc will assume that you want the split to be horizontal, since, again, a vertical split at either of these positions would leave no space for another window.¹

¹ In certain positions a window cannot be placed at all. For example, if you position the cursor at the top, left-most position of the spreadsheet, Perfect Calc will respond: 'Cannot put a window there!'

2. If the cursor is not positioned along an edge of the screen, Perfect Calc will respond with the question:



3. Answer either 'v' or 'h', Perfect Calc will display the message:



If the split is horizontal it is possible to have Perfect Calc synchronize the **columns** in the two windows such that a left or right movement of one window will cause an equal movement of columns in the other window. When one window is moved horizontally across the spreadsheet, the other window will follow automatically.

Similarly, if the split is vertical, Perfect Calc will synchronize the **lines** in the two windows, so that scrolling up or down the spreadsheet in one window, will cause the other window to follow automatically.

Answering 'n' to the question, tells Perfect Calc that you do not wish the windows to be synchronized. In this case the 'other window' (whichever window does not contain the cursor) will remain fixed in place until you move the cursor into it and manually change its position.

It is often desireable to synchronize windows. For example, if you are comparing 1981 and 1982 budget figures, you might display the 1981 figures in the top window and the 1982 figures in the bottom window.

The image shows two overlapping spreadsheets. The left spreadsheet displays the 1981 BUDGET with columns labeled a through e. The right spreadsheet displays the 1982 BUDGET with columns labeled f through j. Both spreadsheets have rows numbered 1 through 6. The data for the 1981 BUDGET is as follows:

	a	b	c	d	e
1	----- 1981 BUDGET -----				
2	Jan.	Feb.	March	April	May
3	347.33	859.63	332.45	843.12	703.68
4	489.59	723.45	621.90	873.29	932.11
5	872.39	769.08	873.98	765.32	123.00
6	763.23	342.30	984.38	358.25	830.17

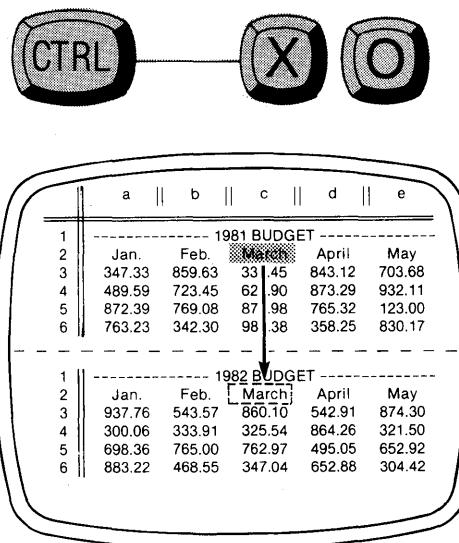
The data for the 1982 BUDGET is as follows:

	f	g	h	i	j
1	----- 1982 BUDGET -----				
2	June	July	Aug.	Sept.	Oct.
3	239.82	304.73	667.54	156.87	763.22
4	774.35	673.32	238.88	643.34	875.43
5	566.45	673.33	765.56	654.55	346.29
6	654.35	876.45	732.49	732.01	964.40

Note that the figures on these spreadsheets are displayed in 12 monthly columns. To examine all 12 months you will have to shift the screen one 'page' to the right. If the windows are synchronized, moving the top screen will cause the bottom screen to follow also, so that the months July to December of each year will display in both the top and bottom windows simultaneously. If the windows are not synchronized, the bottom window will continue to display the columns January to June, even though the top window shifts to view July to December.

The OTHER WINDOW Command

When working with two windows it is often useful to switch from window to window. The OTHER WINDOW command switches the cursor from one window to the other.



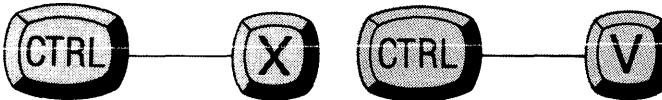
Note: In the Prompt Line the buffer name will change to reflect the new window.

Scrolling the Other Window

When working with two windows it is often necessary to scroll the window which the cursor is **not** occupying. Perfect Calc provides the following commands for moving the other window in page-wise fashion, either up, down, right, or left. (Notice how these several commands mirror the usual Page-Wise Screen Commands, Chapter IV, page 47.)

The PAGE DOWN Other Window Command

Causes the other window to move **down** a screen full of **lines**.



	a		b		c		d		e
<hr/> ----- 1981 BUDGET -----									
1	Jan.	Feb.	March	April	May				
2	347.33	859.63	332.45	843.12	703.68				
3	489.59	723.45	621.90	873.29	932.11				
4	872.39	769.08	873.98	765.32	123.00				
5	763.23	342.30	984.38	358.25	830.17				
<hr/> ----- 1982 BUDGET -----									
1	June	July	Aug.	Sept.	Oct.				
2	854.37	821.20	463.21	972.31	874.55				
3	229.84	202.06	732.88	762.10	764.00				
4	654.28	237.98	454.32	348.99	854.38				
5	654.66	843.20	453.31	544.76	459.92				

	a		b		c		d		e
<hr/> ----- 1981 BUDGET -----									
1	Jan.		Feb.		March		April		May
2	347.33		859.63		332.45		843.12		703.68
3	489.59		723.45		621.90		873.29		932.11
4	872.39		769.08		873.98		765.32		123.00
5	763.23		342.30		984.38		358.25		830.17
<hr/>									
6	654.35		876.45		732.49		732.01		964.40
7	632.30		644.02		832.17		463.28		349.20
8	397.49		501.77		832.19		342.30		721.99
9	400.78		567.29		832.66		832.96		348.93
10	843.21		832.91		832.05		932.15		832.19
11	583.34		930.58		380.24		732.80		483.48

The PAGE UP Other Window Command

Causes the other window to move up a screen full of lines.

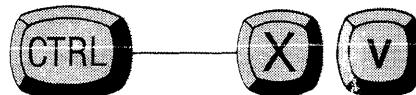


	a		b		c		d		e
<hr/> ----- 1981 BUDGET -----									
1									
2	Jan.		Feb.		March		April		May
3	347.33		859.63		332.45		843.12		703.68
4	489.59		723.45		621.90		873.29		932.11
5	872.39		769.08		873.98		765.32		123.00
6	763.23		342.30		984.38		358.25		830.17
<hr/>									
15	333.54		653.22		718.36		389.61		764.39
16	632.30		644.02		832.17		463.28		349.20
17	397.49		501.77		832.19		342.30		721.99
18	400.78		567.29		832.66		832.96		348.93
19	843.21		832.91		832.05		932.15		832.19
20	583.34		930.58		880.24		732.80		483.48

	a		b		c		d		e
<hr/> ----- 1981 BUDGET -----									
1									
2	Jan.		Feb.		March		April		May
3	347.33		859.63		332.45		843.12		703.68
4	489.59		723.45		621.90		873.29		932.11
5	872.39		769.08		873.98		765.32		123.00
6	763.23		342.30		984.38		358.25		830.17
<hr/>									
9	333.54		653.22		710.36		389.61		764.39
10	632.30		644.02		832.17		463.28		349.20
11	397.49		501.77		832.19		342.30		721.99
12	400.78		567.29		832.66		832.96		348.93
13	843.21		832.91		832.05		932.15		832.19
14	583.34		930.58		880.24		732.80		483.48

The PAGE RIGHT Other Window Command

Causes the other window to move **right** a screen full of **columns**.

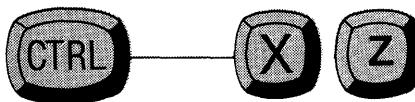


	a		b		c		d		e
<hr/>									
1	----- 1981 BUDGET -----								
2	Jan.	Feb.	March	April	May				
3	347.33	859.63	332.45	843.12	703.68				
4	489.59	723.45	621.90	873.29	932.11				
5	872.39	769.08	873.98	765.32	123.00				
6	763.23	342.30	984.38	358.25	830.17				
<hr/>									
1	----- 1982 BUDGET -----								
2	Jan.	Feb.	March	April	May				
3	937.76	543.57	860.10	542.91	874.30				
4	300.06	333.91	325.54	864.26	321.50				
5	698.36	765.00	762.97	495.05	652.92				
6	883.22	468.55	347.04	652.88	304.42				

	a		b		c		d		e
<hr/> ----- 1981 BUDGET -----									
1									
2	Jan.		Feb.		March		April		May
3	347.33		859.63		332.45		843.12		703.68
4	489.59		723.45		621.90		873.29		932.11
5	872.39		769.08		873.98		765.32		123.00
6	763.23		342.30		984.38		358.25		830.17
<hr/> ----- 1982 BUDGET -----									
1									
2	June		July		Aug.		Sept.		Oct.
3	854.37		821.20		483.21		972.31		874.55
4	229.84		202.06		732.98		762.10		764.00
5	654.28		237.98		654.32		348.99		854.38
6	654.66		843.20		453.91		544.76		459.92

The PAGE LEFT Other Window Command

Causes the other window to move **left** a screen full of **columns**.

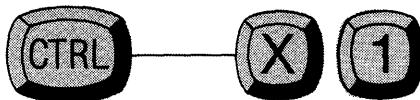


	f		g		h		i		j
1 ----- 1981 BUDGET -----									
2	June	July	Aug.	Sept.	Oct.				
3	239.82	304.73	667.54	156.87	763.22				
4	774.35	673.32	238.88	643.34	875.43				
5	566.45	673.33	765.56	654.55	346.29				
6	654.35	876.45	732.49	732.01	964.40				
1 ----- 1982 BUDGET -----									
2	June	July	Aug.	Sept.	Oct.				
3	854.37	821.20	163.11	972.31	874.55				
4	229.84	202.06	132.98	762.10	764.00				
5	654.28	237.98	654.32	348.99	854.38				
6	654.66	843.20	113.81	544.76	459.92				

	a		b		c		d		e
1 ----- 1981 BUDGET -----									
2	June	July	Aug.	Sept.	Oct.				
3	854.37	821.20	163.11	972.31	874.55				
4	229.84	202.06	132.98	762.10	764.00				
5	654.28	237.98	654.32	348.99	854.38				
6	654.66	843.20	113.81	544.76	459.92				
1 ----- 1982 BUDGET -----									
2	Jan.	Feb.	March	April	May				
3	347.33	859.63	332.45	843.12	703.68				
4	489.59	723.45	621.90	873.29	932.11				
5	872.39	769.08	873.98	765.32	123.00				
6	763.23	342.30	984.38	358.25	830.17				

The CREATE ONE WINDOW Command

Ends split screen editing by making whatever window the cursor is occupying the only window on the screen. At any time during editing with the split-screen, type the CREATE ONE WINDOW command:



Perfect Calc redisplays the current window (the window the cursor is in) as a single window now occupying the full screen. The window that did not contain the cursor is still available in its own buffer, even though it is not now being displayed. You can easily transfer to it using the SWITCH BUFFERS command (Control---X, b).

A diagram showing two overlapping windows side-by-side, each displaying a budget table for 1981 and 1982. The windows are represented by rounded rectangles with a drop shadow. Each window contains two tables separated by a dashed horizontal line.

	a		b		c		d		e
1	----- 1981 BUDGET -----								
2	Jan.	Feb.	March	April	May				
3	347.33	859.63	332.45	843.12	703.68				
4	489.59	723.45	621.90	873.29	932.11				
5	872.39	769.08	873.98	765.32	123.00				
6	763.23	342.30	984.38	358.25	830.17				

	a		b		c		d		e
1	----- 1981 BUDGET -----								
2	Jan.	Feb.	March	April	May				
3	347.33	859.63	332.45	843.12	703.68				
4	489.59	723.45	621.90	873.29	932.11				
5	872.39	769.08	873.98	765.32	123.00				
6	333.54	653.22	710.36	389.61	764.39				
7	632.30	644.02	832.17	463.28	349.20				
8	397.49	501.77	832.19	342.30	721.99				
9	400.78	567.29	832.66	832.96	348.93				
10	843.21	832.91	832.05	932.15	832.19				
11	583.34	930.58	380.24	732.80	483.48				

	a		b		c		d		e
1	----- 1982 BUDGET -----								
2	June	July	Aug.	Sept.	Oct.				
3	854.37	821.20	483.21	972.31	874.55				
4	229.84	202.06	732.98	762.10	764.00				
5	654.28	237.98	654.32	348.99	854.38				
6	654.66	843.20	453.91	544.76	459.92				

WORKING WITH MULTIPLE BUFFERS AND MULTIPLE FILE DISPLAY

TUTORIAL

In this tutorial you will practice using Perfect Calc's multiple buffers and multiple file display.

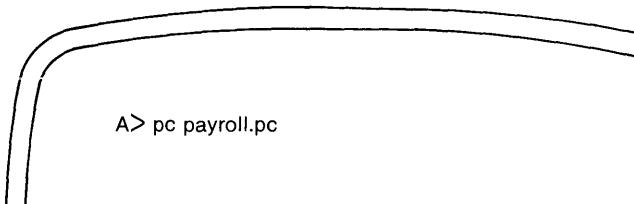
Perfect Calc allows you to work with more than one spreadsheet at a time. In fact, it even allows you to view two spreadsheets simultaneously. All this is made possible by Perfect Calc's unique multiple buffer capability, which allows up to seven spreadsheet files to be active in memory at one time.

To illustrate how the multiple buffers and the multiple file display function, you will use one of the prestructured spreadsheet analysis programs contained in Part 3 of this manual, called the 'Payroll Analysis Program'. This is a multiple-file program, one which involves two separate spreadsheets that cross-reference each other. Because both files must be contained in active buffers, it is an ideal program to illustrate the functioning of multiple buffers and of multiple file display.

The Payroll Analysis Program, like all of the preconfigured spreadsheet analysis programs, is merely for illustrative purposes only, to give you an idea of how Perfect Calc can be made to function. The formulas contained in it, which are intended for weekly wage reporting, as well as the tax tables and social security computations that the spreadsheet uses, may not accurately reflect current government requirements. You should therefore **check all tables and formulas with a professional accountant, or other appropriate source, before attempting to use this program for your business or profession.**

To begin, imagine that you are an employer, who must compute hourly wages and taxes and social security deductions for each of your employees.

1. Call up the first file in the Payroll Analysis Program, named 'payfacts.pc':



Perfect Calc retrieves the file 'payroll.pc' from disk and places it in a buffer called 'payroll' for you to view. In addition, Perfect Calc also reads into another buffer the associated file, 'payfacts.pc'. After reading the two files into memory, Perfect Calc displays the 'payfacts' file (we will discuss associated files in the next chapter).

THE PAYROLL FACT SHEET
Copyright, 1982, Perfect Software, Inc.

This file is used with the file 'payroll.pc'.

	Soc Security Number	Hourly Rate	Single	(yes = 1) Married	FICA rate: Exemptions
Howard Wade	012-34-5678	30.00	0	1	3
	0	0	0	0	0
	0	0	0	0	0

This first spreadsheet of the Payroll Analysis Program records the standard and unvarying facts regarding your various employees. As you can see, the categories include employee name, Social Security number, hourly rate, etc. The data which you enter into these columns will automatically be referenced by the second spreadsheet in the analysis program, called 'payroll.pc', which performs the actual calculations to determine gross pay, income tax and FICA deduction, net pay, etc.

Again, remember that the formulas used here may not be current or accurate. For example, the FICA rate is shown here as '0.06100', which may or may not reflect current government regulations.

2. Into the appropriate entry boxes of line 9, begin entering data on the following employee:

Howard H. Wade

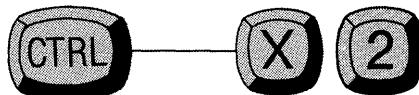
Social Security No: 012-34-5678 (Enter this as a **label**: i.e., you will need to force a label with a "", see Chapter V, page 3)

Hourly rate: \$30 (Omit the '\$' sign)

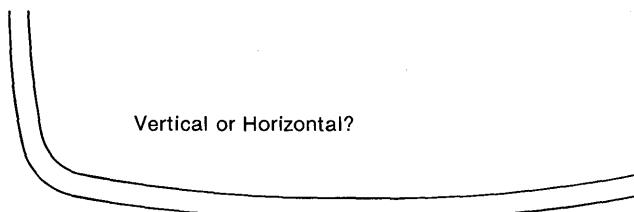
'Married' (Enter a '1' in this column)

Exemptions: 3

3. When this data has been entered, position the cursor in approximately the middle of the screen (about line 11) and give the CREATE TWO WINDOWS command:

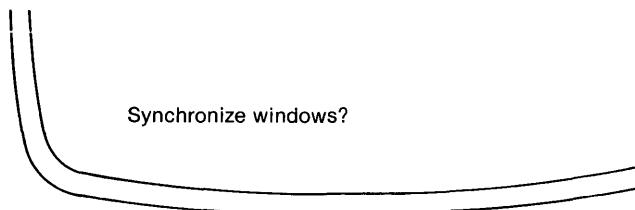


Perfect Calc may respond with the message:



4. Type 'h' for horizontal, if the question is asked.

Perfect Calc will then ask:

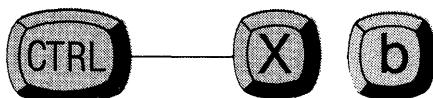


You recall that synchronization will cause the windows to scroll in unison, when a screen or cursor command is given in either window. For this example, answer 'n' (for no) to this question, since we will not need the windows to scroll together.

Perfect Calc immediately splits the screen into two windows at the position of the cursor. The top window holds the upper half of the file you have just finished entering data into, while the bottom window holds the bottom half of this same file.

It should be emphasized here that each window still looks at the same single buffer. Any change that you make in one window will be reflected equally in the other window.

5. You now need to bring the second file of the Payroll Analysis Program, 'payroll.pc' into the window. Give the SWITCH BUFFERS command:



Perfect Calc responds with the message:

Switch to Buffer <CR> :

6. Press the carriage return.

Perfect Calc displays the file 'payroll.pc' in the bottom window of the screen. (Actually, the file is now being viewed through the bottom window.)

Notice that the buffer name displayed in the lower righthand corner of the screen has changed from 'payfacts' to 'payroll', to reflect this new buffer. This name will change to reflect whichever buffer the cursor is currently occupying.

The file 'payroll.pc' should now look like this:

THE PAYROLL FACT SHEET
Copyright, 1982, Perfect Software, Inc.

This file is used with the file 'payroll.pc'.

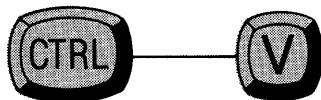
FICA rate:

	Soc Security Number	Hourly Rate	Single	(yes = 1) Married	Exemptions
Howard Wade	012-34-5678	30.00	0	1	3
	0	0	0	0	0
	0	0	0	0	0

PAYROLL ANALYSIS WORKSHEET
Copyright, 1982, Perfect Software, Inc.

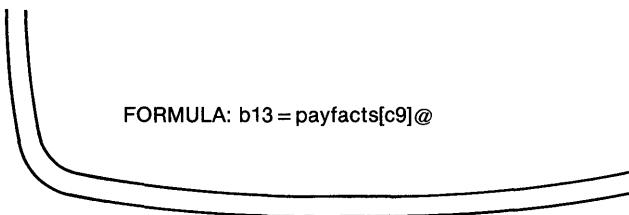
This program automatically reads in the 'payfacts.pc' file needed by this file. This file uses forward referencing, therefore you will need to press the recalculation key (!) twice when updating the file.

7. Actually, only the top portion of this file may be showing in the bottom window. Scroll down a screen-full of lines by giving the PAGE DOWN Command:



Perfect Calc shifts the screen to the center portion of the file.

Notice that data has already been transferred between the two spreadsheets. The '\$30' hourly rate has been transferred from position 'c9' in payfacts.pc to position 'b13' in payroll.pc. Move the cursor to position 'b13' and examine the formula that accomplishes this:



Certain data, however, have not been transferred. Since labels do not transfer between spreadsheets, the employee name has not been entered into payroll.pc.

8. Move the cursor to position 'a13' and type the employee name as it is displayed in the top window.
9. Continue to move the cursor across line 13 adding the following new data in the appropriate entry positions:

Hours: 40
O.T. (overtime hours): 0

As each of these data items is entered into its appropriate entry box, Perfect Calc performs a recalculation of the spreadsheet. Since this program uses forward references you will need to press the recalculation key after entering the appropriate data. The screen will be recalculated to look something like this:

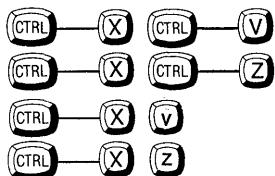
PAYROLL SUMMARY STATEMENT (weekly)

Name of Employee	Hourly Rate	Hours	O.T. Hours	Federal Tax	FICA	State Tax	Gross Pay
Howard Wade	30.00	40.00	0	357.62	73.20	0	1200.00
	0	0	0	0	0	0	0
	0	0	0	0	0	0	0
Totals		40.00	0	357.62	73.20	0	1200.00

11. Change the value under hourly rate from \$30 to \$45.

As you enter the change the recalculation is made in the other window, to reflect new values for all entries. You will need to press the recalculation key (!) because of the forward references used in the file.

12. While in this window, add several other employees, supplying the various data required. Afterwards move the cursor to the payroll window, and enter names and figures for Hours worked and perhaps Overtime hours. Be sure that the employees are entered in the same relative order, in each spreadsheet.
13. While in the top window, practice scrolling the other window, using the following Scroll Other Window Commands:



PAGE DOWN Other Window

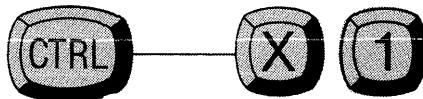
PAGE UP Other Window

PAGE RIGHT Other Window

PAGE LEFT Other Window

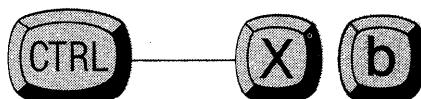
While scrolling you will encounter a large off-screen table of numbers. This is the table used to calculate tax. Switch to the buffer holding payroll.pc and position the cursor on those entry boxes which compute tax values and examine the formula which is displayed in the Prompt Line. Notice how the **lookup** function is used to find the correct tax value in the table.

14. When you have finished scrolling the other window, enter the CREATE ONE WINDOW command:

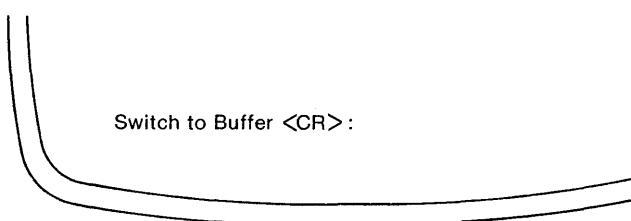


Perfect Calc ends multiple file display, by making the window which the cursor is occupying the only window on the screen.

Although the other file has disappeared from view, it is still present in its active memory buffer, and we can change to it by typing the SWITCH BUFFERS command:



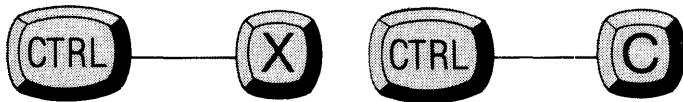
Calc responds with the message:



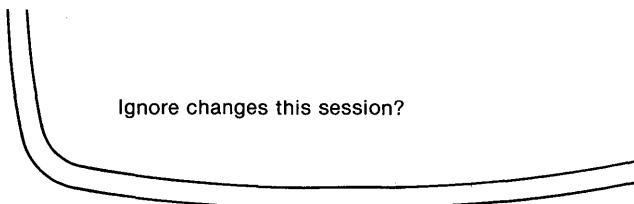
15. Although you could type the name of the other buffer followed by a carriage return, Perfect Calc will automatically switch you to the last buffer switched **from**, if you simply hit a carriage return <CR>.

Practice switching back and forth between buffers, using this command procedure.

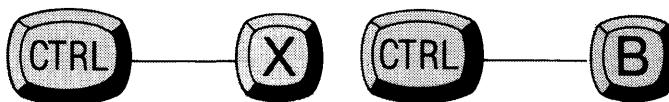
16. When you are ready to quit using Perfect Calc, type the QUIT command:



Perfect Calc responds with the message:



17. If you want to lose the modifications you have made to both spreadsheets answer 'Yes'. Answering 'No' will cancel the QUIT command and return to the position in the current buffer (in effect giving you a chance to save the spreadsheets).
18. Answer 'no', followed by the BUFFERS DIRECTORY command, which will list the active buffers at the bottom of the screen:



Notice in the display which follows that both buffers 'payfacts' and 'payroll' are accompanied by asterisks, '*'. This means that neither of these files has been saved since it was last modified. The choice is yours to save them or not. If you want to save them, you must transfer to each buffer and save it in turn, using either the SAVE FILE command (Control----X, Control----S) or the WRITE FILE command (Control----X, Control----W).

19. Since this is an exercise, you need not save the files. Type the QUIT command again, followed by 'Yes' to ignore the changes.

Perfect Calc discards the modifications to the files, and returns you to your disk operating system.

END OF TUTORIAL

Chapter XIII **ASSOCIATED FILES**

You have already discovered how Perfect Calc, unlike other spreadsheet programs, allows you to manipulate more than one spreadsheet in its virtual memory system at one time. You will recall that up to seven different spreadsheets may be active and share information at once.

However, you may have noticed that during recalculation each spreadsheet is treated separately. When a recalculation occurs, either automatically or by your pressing the '!' key, only the spreadsheet in the current buffer is updated. The others remain unchanged, except to the extent that they may use the revised data from the current spreadsheet when they are eventually updated themselves.

In this chapter, we will discuss how to link spreadsheets together so that recalculation affects not only the current spreadsheet but other spreadsheets as well. Because this material can become complicated, we suggest that you try out the command procedures in this section on sample spreadsheets before attempting to use them on important ones.

File Dependency

In a single spreadsheet, we can see how some entry positions depend on other entry positions. For example, if we define a formula "`a10 = sum(a1:a9)`", we know that the value in entry position "`a10`" depends on each of the values in positions "`a1`" through "`a9`".

This concept of "depends on" can be generalized to include whole spreadsheets as well. Just as certain entry positions within a spreadsheet depend on other entry positions for their values, entire spreadsheets can depend on other spreadsheets. For example, suppose you have two spreadsheets called "`sheet1.pc`" and "`sheet2.pc`", and that a formula in the first references an entry position in the second:

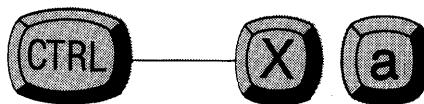
`"c34 = sheet2[q12]`

In such a case we say that "`sheet1.pc`" depends on "`sheet2.pc`". Despite this relationship a recalculation begun in one spreadsheet will not affect the other spreadsheet, since recalculations are normally restricted to the current buffer.

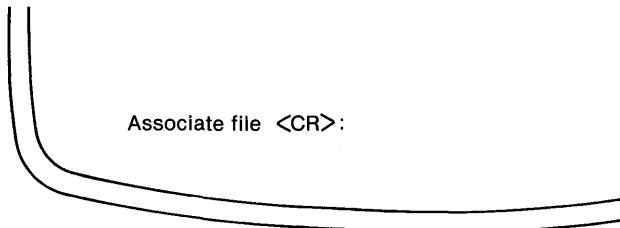
Perfect Calc is designed to operate this way because spreadsheets do not usually depend on each other, and therefore a recalculation of all active spreadsheets, each time any particular one is altered, would in most instances be unnecessary.

The ASSOCIATE FILES Command

If you have spreadsheets which are interdependent, Perfect Calc has a command that will "associate" them, link them together so that changes to one will cause a recalculation in all. This is the ASSOCIATE FILES Command:



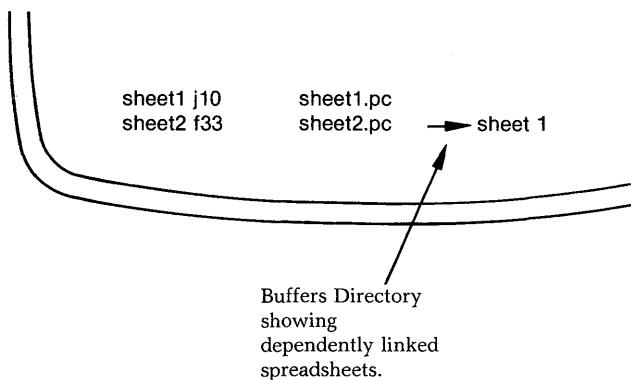
Typing this command causes Perfect Calc to prompt for the filename of the spreadsheet you wish to associate with the current spreadsheet.



Answering with a valid spreadsheet filename, causes Perfect Calc to create a 'one-way' dependency link between the current spreadsheet and the spreadsheet you have named. Perfect Calc appends a coded notation to the end of the current spreadsheet, which specifies it as 'depending on' the named file.

It is perhaps worth noting that the ASSOCIATE FILE command works almost identically to the FIND FILE command (Control----X, Control----F), with one slight difference. When the ASSOCIATE FILE command is given, Perfect Calc checks to see if the file you have named is already active in a buffer. If it is, Perfect Calc switches you to that buffer after creating the one-way dependency 'link'. If the spreadsheet file is not currently in an active buffer, Perfect Calc first activates a buffer, and then retrieves the specified spreadsheet from disk, placing it in the newly activated buffer. As before, it establishes a one-way 'link' between the two files, and switches you to the new file. Technically, the only difference between the ASSOCIATE FILE command and the FIND FILE command is that the former establishes a dependency link, while the second does not.

A 'one-way' dependency means that the current spreadsheet depends on information contained in, or produced by, the second spreadsheet, but not vice versa. This relationship is reflected in the Buffers Directory which will show an arrow pointing from the current buffer to the buffer holding the spreadsheet being depended on. From this time on any recalculations involving either of the two spreadsheets will operate on both of them.



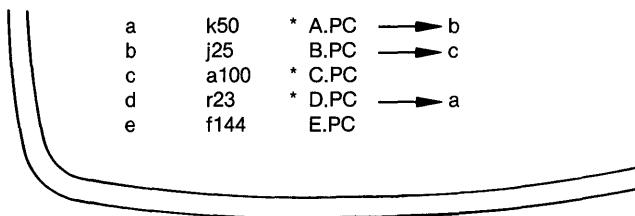
Order of Dependency

The 'order' of the dependency link is important, because it determines the order in which the spreadsheets will be recalculated. Dependent spreadsheets are recalculated **after** all spreadsheets which they depend on have been recalculated. This is done so that the information they use will have been updated before it is needed for their computations. If the current spreadsheet depends on other spreadsheets, while still other spreadsheets depend on the current spreadsheet, the sequence of recalculation is:

- first, those spreadsheets which the current spreadsheet depends on are recalculated;
- then, the current spreadsheet is recalculated;
- finally, the spreadsheets which depend on the current spreadsheet are recalculated.

Example:

Suppose you have five spreadsheets active, named 'A.PC' through 'E.PC'. Suppose further that you have created some dependency links, such that a Buffers Directory shows the following display at the bottom of your screen:



What this display tells us is that spreadsheet 'A' depends on spreadsheet 'B'; spreadsheet 'B' depends on spreadsheet 'C'; spreadsheet 'D' depends on spreadsheet 'A'; and spreadsheets 'C' and 'E' depend on no other spreadsheet. If spreadsheet 'A' is current when a recalculation occurs, the recalculation order will be: 'C', 'B', 'A', and finally 'D'. Because spreadsheet 'A' depends on spreadsheet 'B' and spreadsheet 'B' depends on spreadsheet 'C', these three spreadsheets are updated first in inverse order. Then, because spreadsheet 'D' depends on 'A', it is updated last. Spreadsheet 'E' is not associated with any other spreadsheet and therefore is not updated at all.

You may object that this method of specifying spreadsheet dependencies allows for circular references to be defined. That is, spreadsheet 'C' could be marked as depending on spreadsheet 'A'. However, Perfect Calc blocks such circular dependencies, allowing each spreadsheet to be updated only once in any particular update cycle. Thus, any dependency of spreadsheet 'C' on spreadsheet 'A' will be ignored and the update sequence will remain as before.

Saving and Reading Associated Spreadsheets

Suppose you have established a complex set of interdependent spreadsheets and wish to save them to disk before exiting Perfect Calc. The usual disk write operations (SAVE FILE and WRITE FILE) will automatically preserve the dependency links of each spreadsheet.

However, when reading this set of interdependent spreadsheets back into machine memory, the spreadsheet file which begins the chain of dependencies should be read **first**. Perfect Calc will automatically read in the subsequent files which the main file depends on.

For example, in the above illustration, spreadsheet file 'D.PC' would be the first file read into machine memory. Perfect Calc would then automatically read in the remaining files in the order of their dependencies: i.e., 'A.PC', 'B.PC', and 'C.PC'.

Perfect Calc knows the order of dependency because the linking notation has been appended to the end of each file. At the end of 'D.PC' it encounters the first 'link' pointing to the next file in the chain, 'A.PC'. Reading 'A.PC' it encounters the link to 'B.PC', etc. Should you inadvertently begin in the middle of this sequence, say with spreadsheet file 'A.PC', Perfect Calc would read only 'B.PC' and 'C.PC'. It has no way of knowing that 'D.PC' is earlier in the linked chain of files.

Applications

You may wonder just what associated spreadsheets are good for, particularly if you have successfully used other spreadsheet programs limited to one-at-a-time spreadsheet manipulation. Although we will attempt to suggest some applications, we must admit that the concept is so new and powerful that we ourselves have only scratched the surface of its potential capabilities.

To begin with, associated spreadsheets closely resemble the action of sub-routines in a typical programming language. By associating spreadsheets it is possible to separate "main" information-reporting spreadsheets from "secondary" computing spreadsheets, in effect, isolating computation routines from their input data, thereby saving them for later use in other contexts.

A good application of this capability might involve the functions library provided with your distribution disk. Perfect Calc contains the typical built-in functions present on many other spreadsheet programs—trig functions, net present value, square root, etc. However, by filling a spreadsheet with, say, statistics functions and linking it to your main spreadsheet of information, you are able to extend Perfect Calc's calculating capabilities without having to search through statistics textbooks and type in the formulas each time they are required by a program. The only difficulty you may encounter is in determining exactly how to share the various data items (individual entry positions as well as ranges) between the two spreadsheets.

Let us now consider a short example, presented in the form of a tutorial, to see exactly how this capability of associated spreadsheets works.

WORKING WITH ASSOCIATED SPREADSHEETS

TUTORIAL

In this tutorial we will examine the action of two associated spreadsheet files, one of which (the main information spreadsheet) uses a secondary spreadsheet (the computation spreadsheet) to compute the day of the year, given a date. For example, June 1 (i.e. 6/1) computes to the '152nd' day of the year.

The two spreadsheets look like this:

The MAIN INFORMATION spreadsheet, 'main.pc':

	a		b		c		d		e		f		g
1													
2													
3	Month:								6				
4	Day:								1				
5													
6	Number of Day of the Year								152				
7													

The COMPUTATION spreadsheet, 'compute.pc':

	a		b		c		d		e		f		g		h		i		j		k		l		m
1			JAN		FEB		MAR		APR		MAY		JUN		JUL		AUG		SEP		OCT		NOV		DEC
2	0		1		2		3		4		5		6		7		8		9		10		11		12
3	0		31		59		90		120		151		181		212		243		273		304		334		365
4																									
5	Month:								6																
6	Day:								1																
7									152																
8																									
9																									

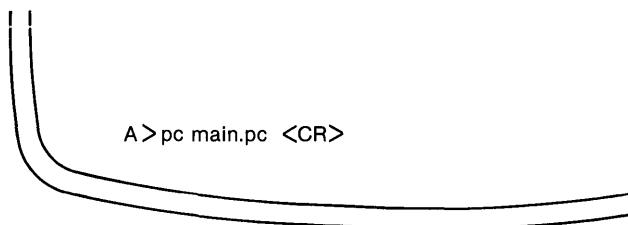
The action of these two spreadsheet files is a simple one. The main information spreadsheet 'depends upon' the computation spreadsheet to compute the day of the year whenever it is supplied with a 'date'. Thus, when we enter a date in the main information spreadsheet, it is automatically referenced by variables in the computation spreadsheet, which performs the calculation. The answer is then referenced by the main information spreadsheet.

This simple operation has numerous applications. For example, it can be used in conjunction with Accounts Receivable or Accounts Payable Programs, or Loan Interest Calculation Programs—in fact, with any program in which 'aging' is involved, and a calculation depends upon a 'date' being represented as a single number.

The above two files have been created for this tutorial and are present on your distribution disk. They are: 'compute.pc' and 'main.pc'.

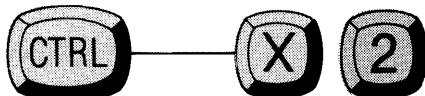
Steps:

1. Begin Perfect Calc by calling up the **first** file in the chain of dependent files:



Perfect Calc first reads the spreadsheet 'main.pc' followed immediately by 'compute.pc'. When these two read operations are complete, the file 'compute.pc' will be displayed to the screen.

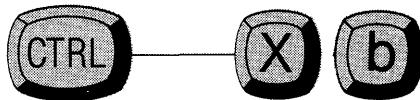
2. Move the cursor to approximately entry position 'a12' and split the window by typing the CREATE TWO WINDOWS Command:



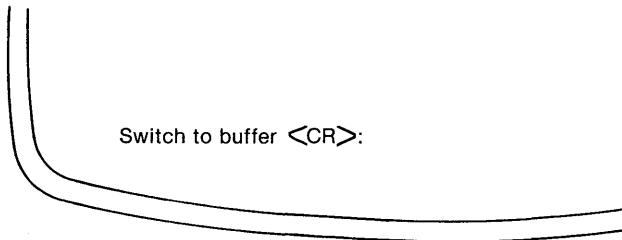
(Do not 'synchronize' windows.)

Perfect Calc splits the screen in half. Each window is displaying a portion of the file 'compute.pc' located in buffer 'compute'. The cursor is positioned in the bottom window.

3. Type the SWITCH BUFFERS command:



Perfect Calc responds with the message:



Switch to buffer <CR>

4. Enter the name of the buffer containing the main information program which 'depends on' the file 'compute.pc':

Switch to buffer: main <CR>

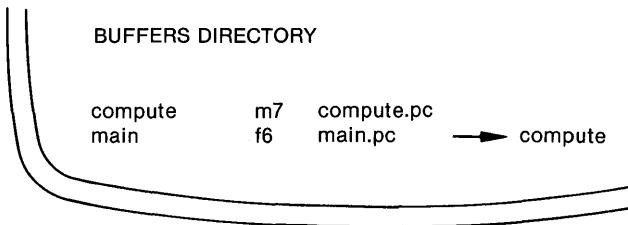
Perfect Calc switches the bottom window to the buffer holding the 'main.pc' spreadsheet.

Your screen should now look like this:

	a		b		c		d		e		f		g
1		JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC
2	0	1	2	3	4	5	6	7	8	9	10	11	12
3	0	31	59	90	120	151	181	212	243	273	304	334	365
4													
5	Month:		6										
6	Day:			1									
7		152											

	a		b		c		d		e		f		g
1													
2													
3	Month:								6				
4	Day:									1			
5													
6	Number of Day of the Year									152			
7													

5. Type the BUFFERS DIRECTORY Command (Control----X, Control----B) to see a listing of the buffers holding these two spreadsheets. It should look like this:



This tells us that spreadsheet 'main.pc' is associated with (depends on) spreadsheet 'compute.pc'. In any recalculation 'compute.pc' will be recalculated **before** 'main.pc'.

-
6. Let us examine the various formulas which function in these two spreadsheets, at the same time tracing the order of recalculation between them. Move the cursor to the top window and examine the two formulas in positions 'c5' and 'c6':

FORMULA: c5 = main[d3]

FORMULA: c6 = main[d4]

Any recalculation will begin in spreadsheet 'compute.pc' with these formulas, which tell Perfect Calc to read input data from positions 'd3' and 'd4' in spreadsheet 'main.pc'. Following this action, recalculation proceeds to position 'a7' of 'compute.pc' which contains yet another formula:

FORMULA: a7 = lookup(c5-1, a2:m2) + c6

This is a simple 'lookup' function. It instructs Perfect Calc to go to the table in 'compute.pc' and search through the range of entries 'a2' through 'm2', finding the entry which corresponds to **one less** than whatever value is in 'c5'; further, it tells Perfect Calc to take whatever number is below this entry and add to it the value contained in 'c6'. This will prove to be the correct day of the year specified by the date originally referenced from positions 'd3' and 'd4' of spreadsheet 'main.pc'. Following this, the order of recalculation passes to the dependent spreadsheet, 'main.pc'.

Entry 'd6' is the only formula in 'main.pc' which needs recalculating:

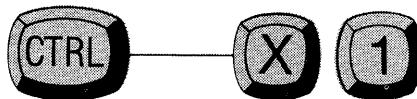
FORMULA: d6 = compute[a7]

This instructs Perfect Calc to retrieve the result from position 'a7' of 'compute.pc', and to post it to position 'd6' of 'main.pc'. When this has been done, the recalculation of the two spreadsheets is complete.

7. Practice using these associated spreadsheets. Enter various 'months' and 'days' into positions 'd3' and 'd4' of 'main.pc'.

As you enter each pair of values, notice how Perfect Calc immediately recalculates both spreadsheets and posts the result to position 'd6' of 'main.pc'. Can you visually follow the order of recalculation?

8. With the cursor in the bottom window, give the CREATE ONE WINDOW command:



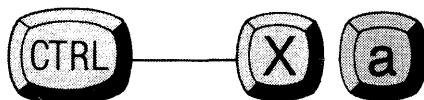
Perfect Calc immediately makes 'main.pc' the only spreadsheet being viewed.

9. Continue to enter sample dates.

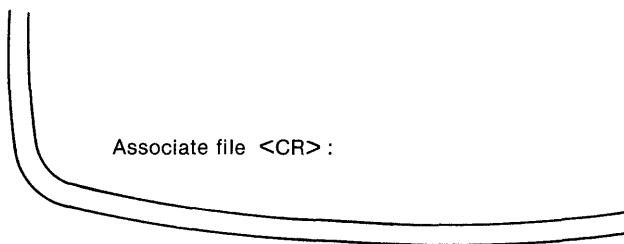
Perfect Calc continues to compute values. Even though spreadsheet 'compute.pc' is not being viewed on the screen, it still remains in an active buffer and in dependent 'association' with the current spreadsheet 'main.pc'.

Unlinking Spreadsheets

10. Finally, suppose that we wish to end the associative dependency of the two spreadsheets. Place the cursor in the 'main' or dependent spreadsheet (i.e. 'main.pc'), and type the ASSOCIATE FILE command:



Perfect Calc responds with the message:



11. Type a carriage return, <CR>, without entering a spreadsheet name.

Perfect Calc unlinks the two files, erasing the original associative command notation that was appended to the end of 'main.pc' linking it to 'compute.pc'.

This concludes our tutorial on 'Associated spreadsheets'.

END OF TUTORIAL

PART III

SPREADSHEET ANALYSIS PROGRAMS

In This Section

- Personal and Home Use Programs
- Small Business Applications
- Spectrum of Applications

This section contains Perfect Calc's preformatted Spreadsheet Analysis Programs, which are provided to help you begin using Perfect Calc immediately. With these programs there is no need for trial and error in trying to figure out what kinds of data you will need, or how it shall be arranged, and what formulas will correctly manipulate that data. All this has been predetermined for each spreadsheet.

Of course, the spreadsheets are of a general nature, and may require modification to meet your particular needs. But this is easily done using the command procedures presented earlier in this manual.

The spreadsheets are grouped in three categories: Personal and Home Use, Small Business Applications, and a spectrum of Scientific and Professional Applications. In the following pages each spreadsheet is presented accompanied by an illustration and a short discussion of its nature and use. It is beyond the scope of this User's Guide to present a tutorial for each spreadsheet, and so to become completely familiar with the spreadsheets which you want to use, you are encouraged to simply call the spreadsheet to the screen, and begin experimenting.

All data entry positions are appropriately labeled and should be relatively self-explanatory. You can use the SEARCH for next unlocked entry command (Control----S) to move the cursor directly to the entry boxes that require data from you. Formulas can be examined by moving the cursor to the various positions and viewing the formula as it will be displayed in the Prompt Line.

Several of the programs involve the use of more than one spreadsheet file, all of which must be present in active buffers when the program is being used.

The best way to become familiar with Perfect Calc is to begin using it. The spreadsheet programs provided in this section are designed to get you started and to illustrate the kinds of applications suited to Perfect Calc. **It must be emphasized that these programs are provided for illustrative purposes only. Perfect Software, Inc. cannot be responsible for assisting you in implementing them.** If you have questions in applying these programs you will need to consult an accountant or other professional who is familiar with your particular needs.

Chapter XIV

PERSONAL AND HOME USE PROGRAMS

The Family Budget Program

Filename: 'fambudgt.pc'

Here is a straightforward program that will analyze all aspects of your household budget. The program consists of a single spreadsheet file. As the illustration shows, it contains two principal sections: Income and Expenditures, each of which is broken into a number of subcategories that will accommodate a variety of family financial activities.

Lines within the categories can be inserted or deleted, and the formulas computing the various totals will automatically expand or contract to accommodate your modifications. The formulas for each category are replicated to each of the 12 months.

FAMILY BUDGET PROGRAM
 Copyright, 1982, Perfect Software, Inc.

FAMILY BUDGET	JAN	FEB	DEC
FAMILY INCOME	----	----	----
Source			
Wages and Salary of:			
Husband	0	0	0
Wife	0	0	0
Other	0	0	0
Earnings from:			
Consulting	0	0	0
Rental Property	0	0	0
Other	0	0	0
Total Income	0	0	0
EXPENDITURES			
Fixed Expenses			
Rent or Mortgage	0	0	0
Car Payment	0	0	0
Insurance:			
Auto	0	0	0
Home	0	0	0
Health	0	0	0
Life	0	0	0
Loans:			
Auto	0	0	0
Appliances	0	0	0
Other	0	0	0
Sub total (F.E.)	0	0	0

Variable Expenses

Utilities:

Electric	0	0	0
Water	0	0	0
Gas	0	0	0
Phone	0	0	0
Cable Television	0	0	0
Food	0	0	0
Clothing	0	0	0
Transportation	0	0	0
Furniture	0	0	0
Entertainment	0	0	0
Donations	0	0	0
Subscriptions	0	0	0
Miscellaneous	0	0	0
<hr/> Sub total (V.E.)	<hr/> 0	<hr/> 0	<hr/>	<hr/> 0
TOTAL EXPENSES	0	0	0
SAVINGS or DEFICIT	0	0	0
CUMULATIVE SAVINGS	0	0	0

The Financial Net Worth Program

Filename: 'networth.pc'

This single file program will evaluate your financial net worth. Assets are entered in one column, liabilities in the other. Perfect Calc automatically calculates totals, at the end deducting liabilities from assets, giving a figure for 'Total Net Worth'.

Although lines for new categories can be inserted, be careful when deleting. Only delete using the DELETE ENTRY command (Control----D), since a line deletion would perhaps result in an unintentional loss of entries in an adjacent column.

PERSONAL FINANCIAL NET WORTH STATEMENT Copyright, 1982, Perfect Software, Inc.

We recommend you use the Manual recalculation MODE when entering data onto this spreadsheet (Control----X, m {select m}).

Remember, when entering data on this spreadsheet you may use the SEARCH for next unlocked field command (Control----S) to go to the next field for data input.

ASSETS	LIABILITIES
Liquid Assets	Current Bills Due
Cash on Hand	MasterCard
Checking Accounts	American Express
Savings Accounts	Other Credit Cards
Money Market Funds	Medical Bills
Other	Medical Insurance
Subtotal	0 Rent
	Utilities
Long Term Assets	Home Insurance
Retirement Benefit	Auto Insurance
Savings Bonds	Life Insurance
Annuities	Medical Insurance
Cash Value of Life Insurance	Tuition
Other	Other
Subtotal	0 Subtotal 0

Investments		Outstanding Loans	
Stocks		Home Mortgage	
Bonds		Installment Loans	
Options		Automobile Loan	
Mutual Funds		Education Loans	
Other		Home Improvement Loan	
Subtotal	0	Furniture Loan	
		Other Loans	
		Subtotal	0
Durable Assets		Taxes Due	
Home		Federal Income	
Other Real Estate		State and City	
Furniture		Real Estate	
Appliances		Assessments	
Automobiles		Other	
Clothing		Subtotal	0
Jewelry, Furs, etc.			
Collectibles			
Hobby Equipment			
Sporting Equipment			
Other			
Subtotal	0		
Other Assets			
Business Interests			
Money Owed You			
Tax Refunds Due			
Other			
Subtotal	0		
Total Current Assets	0	Total Liabilities	0
Total Net Worth	0		

The Check Register Program

Filename: 'check.pc'

If you have trouble keeping your checkbook balanced, you will want to begin using this single file program right away. As you record checks that you have written, it computes your checkbook balance, taking account of deposits and checks which have not yet cleared the bank. It automatically creates a sequential listing of check numbers, beginning with the initial number that you enter. It computes a 'Reconciliation Balance' that shows the difference between your checkbook records and the statements which you receive from the bank.

Notations of deposit are easily inserted using the OPEN LINE command (Control----O).

CHECK REGISTER PROGRAM
Copyright, 1982, Perfect Software, Inc.

Enter beginning balance in d12.

Enter beginning check number in a15.

Remember, Control---S will take the cursor directly to these locations.

Deposits can be recorded by opening a line with the Control---O command and entering deposit information on the new line.

Your balance is displayed at the end of the program. Press Escape...> to go to the end.

Beginning Balance = 0

CHECK #	Date	Description	Paid	Deposit
1				
2				
3				
.				
.				
48				
49				
50				

Records Balance 0
(Deposits less checks written)

Reconciliation Balance 0
(Deposits less checks cleared)

The Individual Tax Return Analysis Programs

Filenames: 'tax.pc'
 'scheda.pc'
 'schedx.pc'

The tax analysis programs are a group of three associated spreadsheet files that will assist you in computing your personal income tax. The spreadsheets reproduce the entries and calculations required on IRS forms '1040' (Individual Tax Return), and 'A' (Itemized Deductions).

Only schedule X for single individuals has been included in this example file. To use this tax program, simply call the files into active buffers and begin entering your tax return data. Perfect Calc automatically calculates the necessary relationships and values. When you have finished entering and calculating your tax, print the files to hardcopy and transcribe the data to an official IRS tax form.

Important Note: You will need to show your completed tax form to a professional tax consultant to be sure everything is correct.

1040 INDIVIDUAL TAX RETURN—1981 READ INSTRUCTIONS BEFORE USING

Copyright, 1982, Perfect Software, Inc.

This Spreadsheet file automatically reads in the following files:

shceda.pc		
schedx.pc		
6 Exemptions (col. b = "Error" if 0)	1	1
Income		
7 Wages, salaries, tips, etc.		0
8a Interest (less All-Savers excl.)	0	—
8b Dividends	0	—
8c Total	0	—
8d Exclusion	0	—
8e Taxable Interest & Dividends		0
9 Tax refunds		0
10 Alimony		0
11 Business income		0
12 Capital gain		0
13 40% capital gain distr.		0
14 Supplemental gains		0
15 Pensions & annuities		0
16 Other pensions and annuities		0
17 Rents, royalties, . . .		0

18	Farm income	0
19	Unemployment compensation	0
20	Other income	0
21	Total income	0
Adjustments to Income		
22	Moving Expense	0
23	Employee business expense	0
24	IRA payments	0
25	Keogh payments	0
26	Interest penalty	0
27	Alimony	0
28	Disability income	0
29	Other adjustments	0
30	Total adjustments	0
31	Adjusted Gross Income	0
Tax Computation		
32	Itemized deductions	0
33	\$1000 per exemption	1,000.00
34	Taxable income	(1,000.00)
Note: If you must use Tax Table, enter tax from Tax Table in "40c". Tax (from Sched. X, Y or Z)		
35	Tax (includes Rate Red. Credit)	0
36	Additional taxes	0
37	Total taxes	0
Credits		
38	Political contributions	0
39	Elderly	0
40	Child care	0
41	Investment	0
42	Foreign tax	0
43	Work incentive	0
44	Jobs	0
45	Residential energy	0
46	Total credits	0
47	Balance	0
Other Taxes		
48	Self-employment	0
49	Minimum	0
50	Alternative minimum	0
51	Add'l FICA	0
52	Tax on an IRA	0
53	Advance EIC payments	0
54	Total tax	0

Payments

55	Federal income tax withheld	0	—
56	1981 estimated tax payments	0	—
57	Earned income credit	0	—
58	Form 4868	0	—
59	Excess FICA	0	—
60	Fuels tax	0	—
61	Regulated investment co.	0	—
62	Total payments	0	0

Refund or Balance Due

63	Amount OVERPAID	0
64	REFUNDED TO YOU	0
65	Applied to 1982 tax	
66	BALANCE DUE	0

SCHEDULE A—ITEMIZED DEDUCTIONS

Copyright, 1982, Perfect Software, Inc.

This Spreadsheet is used with the "tax.pc" spreadsheet file.

Medical and Dental Expenses

Medical insurance premiums (total)	0
1 One-half of premiums (to \$150)	0
2 Medicine and drugs	0
3 1 % of Form 1040, line 31	0
4 Line 2 - line 3	0
5 Balance of insurance premiums	0
6 Other medical and dental expenses	0
7 Total	0
8 3 % of Form 1040, line 31	0
9 Line 7 - line 8	0
10 Total	0

Taxes

11 State and local income	0
12 Real estate	0
13 General sales	0
14 Personal property	0
15 Other	0
16 Total	0

Interest expense

17 Home mortgage	0
18 Credit card	0
19 Other	0
20 Total	0

Contributions

21 Cash contributions	0
22 Other than cash	0
23 Carryover	0
24 Total	0

Casualty or Theft Losses

25 Loss amount	0
26 Insurance reimbursement	0
27 Line 25 - line 26	0
28 smaller of \$100 or line 27	0

29 Total 0

Miscellaneous Deductions

30 Union dues, tax prep. fee	0
31 Other	0
32 Total	0
39 Total Itemized Deductions	0
40 Zero Bracket Amount	0

41 Allowed Ded'n's (to 1040 line 32b) 0

Tax Schedule X

SUPPORTING TAX SCHEDULE X

Copyright, 1982, Perfect Software, Inc.

This Spreadsheet file is used with the "tax.pc" spreadsheet file.

Filing Status:	Single
Taxpayer Zero Bracket:	2300.00
Interest exclusion:	200.00
Adjusted Gross Income:	0.00

Tax Schedule X

Chapter XV

SMALL BUSINESS APPLICATIONS

The Income Statement Program

Filename: 'income.pc'

- 'costgood.pc'
- 'gadmexpn.pc'
- 'sellexpn.pc'

This four-file program will compute an Income Statement for your business. The file 'income.pc' is the master file to which the remaining programs post their results. When you read in the 'income.pc' file, all the others are read in automatically.

'Costgood.pc' computes the cost of materials needed to produce the item (or perhaps service) that your business sells. You will perhaps have to modify the categories on this spreadsheet to suit your particular needs.

'Gadmexpn.pc' computes the cost of administering your business: management and employee salaries, telephones, postal service, travel, etc.

'Sellexpn.pc' computes the cost of selling the product: advertising, marketing, salesmen's wages, brochures, etc.

To use the program call the four files into separate active buffers and begin entering data as appropriate. The results tallied by the three 'expense' spreadsheets will post automatically to 'income.pc', which you can then print to hard-copy or to a file for further editing with Perfect Writer. 'Error!' values which may initially be present in the spreadsheets will disappear as you begin entering data.

INCOME STATEMENT

Copyright, 1982, Perfect Software, Inc.

This Spreadsheet file automatically reads in the following files:

costgood.pc (Cost of Goods Statement)
 gadmexpn.pc (General and Administrative Expense Statement)
 sellexpn.pc (Sales Expense Statement)

We recommend using manual recalculation mode for this file.

	PERCENT
Total Sales	0.00 100.00
Less Cost of Goods Sold	0 0
(This data is supplied from the 'costgood.pc' file.)	
Gross Profit on Sales	0 0
Less Operating Expenses	
Selling Expense	0
(This data comes from the 'sellexpn.pc' file.)	
General Expense	0
(This data comes from the 'gadmexpn.pc' file.)	
Subtotal	0 0
Net Income from Operations	0 0
Other Income & Expenses	
Royalties	0
Dividends	0
Sale of Fixed Assets	0
Other	0
Subtotal	0
Interest & Debt Expense	0
Net Addition to Income	0 0
Net Income Before Taxes	0 0
Less Estimated Income Tax	0 0
Net Income After Taxes	0 0

COST OF GOODS STATEMENT

Copyright, 1982, Perfect Software, Inc.

The information from this statement is posted to the Income Statement file 'income.pc'.

Direct Material Expenditures	
Materials Inventory	0
Purchases	0
Less for Returns	0

	0
Materials Available	0
Less Materials Inventory	
Direct Materials Used	0
Direct Labor Costs	0
Fixed Operation Expenses	0
Salaries	0
Rent	0
Payroll Taxes	0
Utilities	0
Phone	0
Postage	0
Supplies	0
Depreciation—Equipment	0
Depreciation—Building	0
Maintenance	0
Insurance	0
Services	0
Other	0

Total Fixed Expenses	0
Total Production Costs	0
Work in Progress Inventory	0

Less In Progress Inventory	0
Cost of Goods Produced	0
Inventory of Finished Goods	0

Less Finished Goods Inventory	0
Total of Cost of Goods Sold	0

SELLING EXPENSES

Copyright, 1982, Perfect Software, Inc.

The total from this statement is posted to the Income Statement file 'income.pc'.

Salesperson Wages	0
Sales Commissions	0
Supplies	0
Advertising	0
Marketing	0
Brochures	0
Telephone	0
Postage	0
Payroll Taxes	0
Travel	0
Entertainment	0
Other	0
<hr/> TOTAL	<hr/> 0

GENERAL AND ADMINISTRATIVE EXPENSES

Copyright, 1982, Perfect Software, Inc.

The data in this file is posted to the Income Statement file 'income.pc'.

NOTE: If you alter this file be sure to correct file references in the 'income.pc' file.

Executive Salaries	0
Management Salaries	0
Employee Salaries	0
Payroll Taxes	0
Supplies	0
Telephone	0
Postage	0
Travel	0
Entertainment	0
Other	0
<hr/> TOTAL	<hr/> 0

The Cashflow Assessment Program

Filename: 'cashflow.pc'

This program will quickly give you an accurate picture of the cash position of your company or business. You enter data values in the appropriate locations for the individual months, and Perfect Calc calculates both monthly and yearly totals.

Lines can be deleted or inserted within the two sections, and the formulas which calculate the totals for the section will automatically adjust to accommodate the change. Remember, however, not to delete the first or last lines of any regions used in formulas.

CASH FLOW ASSESSMENT
 Copyright, 1982, Perfect Software, Inc.
 *** All figures based on projections***

	Jan.	Feb.	Dec.	YEARLY TOTAL
Cash Flow In					
Sales					0
Services					0
Accounts Receivables					0
Contract Accounts					0
Other					0
Total Cash Flow In	0	0	0	0
Cash Flow Out					0
Rent					0
Utilities					0
Phone					0
Payroll					0
Supplies					0
Insurance					0
Postage					0
Advertising					0
Maintenance					0
Other					0
Total Cash Flow Out	0	0	0	0
Monthly Cash Position	0	0	0	
WORKING CAPITAL	0	0		

The Accounts Receivable Worksheet

Filename: 'arecmult.pc'
 'arentry1.pc'

This multiple-file analysis program computes accounts receivable for a small business. The file 'arecmult.pc' is the master file to which individual accounts receivable are posted by 'arentry1.pc'. This subsidiary file can be duplicated and used repeatedly to enter the various individual accounts. (When saving this file, use the WRITE FILE command, saving to successively numbered filenames, e.g. 'arentry2.pc', 'arentry3.pc', etc. Also, be sure to modify the references in the 'arecmult.pc' file to reflect the added entry file.)

The program computes the date of invoice, and an 'aging' function automatically flags delinquent accounts. The invoice number, Net Collectable days, and amount, are automatically posted to the master 'arecmult.pc' spreadsheet. However, labels are not posted, and you must transfer Name of Account to the master spreadsheet.

ACCOUNTS RECEIVABLE WORKSHEET
 Copyright, 1982, Perfect Software, Inc.

Today's Date (enter MN:DY:YR) → 1 1 82

This program automatically reads in the following entry files:

arentry1.pc arentry3.pc arentry2.pc

Switch to these files to enter the appropriate data. We recommend using manual recalculation mode for this file. Also, you will need to press the recalculation key "!" twice because of forward references.

ACCOUNTS RECEIVABLE SUMMARY STATEMENT

L	a	t	e	Name of Account	Invoice Number	Amount	Date of Invoice	Days Net	Date Due	Age of Invoice (Days)
							MN:DY:YR		MN:DY:YR	

The Accounts Receivable Entry Sheet

ACCOUNTS RECEIVABLE ENTRY ARENTRY1 (See 'arecmult.pc')
Copyright, 1982, Perfect Software, Inc.

Invoice Number: 0 Date: MN: DY: YR:
Net Collectable (days): 0 1/ 1/82

Name:
Organization:
Address:
Address:
City/ST/Zip:

Quantity	Item	Unit Price	Price
0		0	0
TOTAL:			0

The Accounts Payable Worksheet

Filename: 'apaymult.pc'
 'apentry1.pc'

This multiple-file analysis program computes accounts payable for a small business, and functions almost identically to the accounts receivable program. The file 'apaymult.pc' is the master file to which, individual accounts payable are posted by 'apentry1.pc'. As in accounts receivable this subsidiary file can be duplicated and used repeatedly to enter the various individual accounts. (When saving this file, use the WRITE FILE command, saving to successively numbered file names, e.g. 'apentry2.pc', 'apentry3.pc', etc. Also, be sure to modify the references in the 'apaymult.pc' file to reflect the new entry file.)

ACCOUNTS PAYABLE WORKSHEET
 Copyright, 1982, Perfect Software, Inc.

Today's Date (enter MN:DY:YR) → 1 1 82

This program automatically reads in the following entry files:
 apentry1.pc apentry2.pc apentry3.pc

Switch to these files to enter the appropriate data. We recommend using manual recalculation mode for this file. Also, you will need to press the recalculation key "!" twice because of forward references.

ACCOUNTS PAYABLE SUMMARY STATEMENT

D	Name of Vendor	Purchase Order Number	Amount Due	Date of P.O. MN:DY:YR:	Days Net	Date Due MN:DY:YR:	Age of Pur- chase Order (Days)
0		0	0	1 1 82	0	1 1 82	
0		0	0	1 1 82	0	1 1 82	
0		0	0	1 1 82	0	1 1 82	
Totals		0					

The Accounts Payable Entry Sheet

ACCOUNTS PAYABLE ENTRY APENTRY1

Copyright, 1982, Perfect Software, Inc.

Purchase Order Number:
Net Due (days):

0
0

Date:
MN: DY: YR:
1/ 1/82

Name:
Organization:
Address:
Address:
City/ST/Zip:

Quantity	Item	Unit Price	Price
		0	0
	TOTAL:		0

The Payroll Analysis Program

Filename: 'payroll.pc'
 'payfacts.pc'

This two-file program shows how Perfect Calc can be used to manage the payroll for a small business. Employee rates are entered in the spreadsheet 'payfacts.pc'. This information is then posted to 'payroll.pc' where calculations are performed to determine tax, gross pay, net pay, FICA deductions, etc. Labels do not transfer between spreadsheets, and therefore you will have to enter the Name of Employee to the master spreadsheet 'payroll.pc'. Be careful to keep employee entries in the same relative positions on both spreadsheets.

PAYROLL ANALYSIS WORKSHEET

Copyright, 1982, Perfect Software, Inc.

 This program automatically reads in the 'payfacts.pc' file needed by this file. This file uses forward referencing, therefore you will need to press the recalculation key (!) twice when updating the file.

PAYROLL SUMMARY STATEMENT (weekly)

Name of Employee	Hourly Rate	Hours	O.T. Hours	Federal Tax	FICA	Other Deduct.	Gross Pay	Net Pay
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0
Totals		0	0	0	0	0	0	0

The Payroll Fact Sheet

PAYROLL FACT SHEET

Copyright, 1982, Perfect Software, Inc.

This file is used with 'payroll.pc'.

FICA rate: 0.06100

Name of Employee	Social Security Number	Hourly Rate	Single	(yes = 1) Married	Exemptions
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0

Chapter XVI

SPECTRUM OF APPLICATIONS

The Stock Portfolio Evaluator Program

Filename: 'stock.pc'

If you own stocks, you will find this program invaluable in managing your portfolio. It is structured to accommodate any quantity of stock purchases. As you add new stocks, you can insert lines for them in the spreadsheet. The formulas computing the totals will automatically adjust to accommodate the change (although you will have to copy and replicate to the new lines the commands which compute column values: i.e. Commission, Yield, Capital Gain/Loss).

The Real Estate Analysis Program

Filename: 'reanal.pc'

This file analyzes a real estate investment over an eight year period. You enter the appreciation (as a percentage), the value of the property, and the down payment. The program computes the dollar gain, percentage gain, and the rate of return. Be sure to examine the formulas used by the program to understand the assumptions it is based on.

A REAL ESTATE ANALYSIS PROGRAM

Copyright, 1982, Perfect Software, Inc.

When entering data use the SEARCH next unlocked field command to determine where to enter data (Control---S).

Year	Start	1	2	8
Appreciation	0		.		
Purchase Price	0	0	0	0
Down Payment	0				
Loan Amount	0				
PITI + maintenance	0				
Rental Income	0	0	0	0
Cash Flow	0	0	0	0
<hr/>					
- selling costs	0	0		0
- cash flow	0	0		0
<hr/>					
\$ gain	0	0		0
% gain					
rate of return					

The CHI-Square Analysis Program

Filename: 'science.pc'

This single file program is a statistical tool to examine relationships between two nominal variables. It is beyond the scope of a brief discussion to illustrate the complex applications of this program. Normally it is useful only to those professionals familiar with it, and is included here to illustrate how Perfect Calc can be used for statistical analysis.

CHI-SQUARE ANALYSIS
Copyright, 1982, Perfect Software, Inc.

Independent Variable

Dependent Variable

Chi-square 0 d.f. = 8.00

0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

The Professional Fee Analysis Program

Filename: 'profee.pc'

If you are an independent professional who is not making as much money as you should, perhaps it is because you have never really determined what you should be charging for your services. This program enables you to systematically determine how much you should be getting paid. All you do is enter the data, and Perfect Calc does the rest.

Lines can be inserted within ranges for other categories if you need them. The formulas which calculate the totals will automatically adjust to accommodate the insertion.

DETERMINING FEE TO CHARGE FOR PROFESSIONAL SERVICES

Copyright, 1982, Perfect Software, Inc.

Yearly Salary	0	You supply the salary you
Days worked per month	0	desire to achieve.
Days worked per year	0	
Desired Percent Profit	0	You supply the percent profit you want.

	Monthly	Yearly
Operation Overhead		
Office Expenses	0	0
Rent	0	0
Secretary's Wage	0	0
Postage	0	0
Telephone	0	0
Utilities	0	0
Supplies	0	0
Other	0	0
Subtotal	0	0
Support Services		
Insurance	0	0
Marketing	0	0
Legal	0	0
Advertising	0	0
Accounting	0	0
Other	0	0
Subtotal	0	0
Additional Expenses		
Automobile	0	0
Entertainment	0	0
Travel	0	0
Vacation	0	0
Miscellaneous	0	0
Subtotal	0	0
TOTAL EXPENSES	0	0
Daily Overhead	0	
Daily Salary	0	
Daily Profit	0	
Required Billing Rate	0	

Appendix A INSTALLING 'PERFECT CALC'

System Hardware Requirements

Perfect Calc requires a computer with at least 56k of memory and at least 75k of disk storage space. To take full advantage of its virtual memory architecture, at least 140k of disk space is recommended.

System Files

The Perfect Calc distribution diskette you received contains the following system files:

PC.COM The Perfect Calc main program.

PC.OVL The overlay, help, and terminal information file.

PCCONFIG.COM The configuration program.

PC.SWP The virtual memory swap file.

You will want to create a Perfect Calc system diskette that has, at least, PC.COM, PC.OVL and PC.SWP on it. Use this diskette along with a data file diskette containing either your data files, the prestructural programs or the lessons.

Back-up Diskettes

The first step in installing Perfect Calc is to make a copy, or 'backup', of the original Perfect Calc diskette, one that will function as 'master working disk'. To do this:

- Transfer your operating system to the backup diskette, again, using an appropriate utility, (e.g. 'Sysgen', 'Copy', etc.). This will enable your working diskette to 'boot up'.
- Copy the above Perfect Calc system files to a new diskette using an appropriate utility, (e.g. 'PIP', 'COPY', etc.).

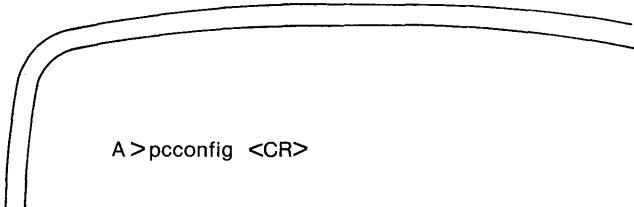
Backups are important for two reasons: First, if you make a mistake while configuring Perfect Calc, your backup diskette allows you to recover quickly with no damage to your original Perfect Calc disk. Second, if you are using Perfect Calc a great deal, the information on the original diskette could become distorted and thus unusable. Having once made a backup copy, put the original away in some safe place, and always use the backup copy as your principal working diskette when using Perfect Calc.

Configuring Perfect Calc

Your dealer may have provided you with a copy of Perfect Calc that is already preconfigured for your computer. If this is the case, then most of the steps in this installation procedure will not apply to you, with the exception of one: you may have to describe one or two operating characteristics of your printer to Perfect Calc. This is a very simple step, and we will indicate it to you in the course of configuration.

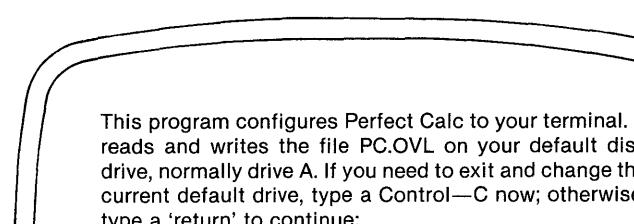
Steps:

1. Turn on your computer and mount your Perfect Calc working disk on drive A. When your control system boots up and presents you with a command prompt indicating that it is ready to receive a command, type 'pcconfig', followed by a carriage return:



A > pcconfig <CR>

Perfect Calc's configuration program will come on-line, displaying the following message:

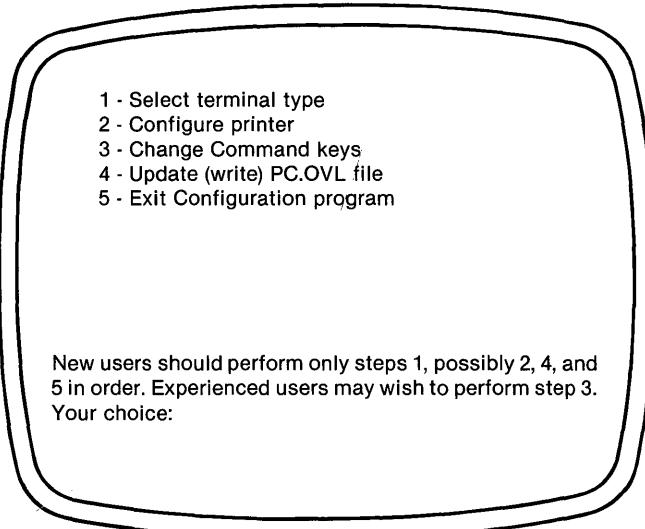


This program configures Perfect Calc to your terminal. It reads and writes the file PC.OVL on your default disk drive, normally drive A. If you need to exit and change the current default drive, type a Control-C now; otherwise, type a 'return' to continue:

The file 'PC.OVL' is the file which will hold all of the technical information regarding your computer that will be generated during configuration. (It also contains the Help Menu and several other command functions). Following configuration, PC.OVL will be updated by Perfect Calc, and thereafter must **always** be accessible to the other system programs in order for Perfect Calc to function correctly.

2. Since your working diskette already occupies the default drive (drive A), simply type a carriage return, <CR>.

Perfect Calc immediately scrolls the screen to present the Main Menu of the Configuration Program:

- 
- 1 - Select terminal type
 - 2 - Configure printer
 - 3 - Change Command keys
 - 4 - Update (write) PC.OVL file
 - 5 - Exit Configuration program

New users should perform only steps 1, possibly 2, 4, and 5 in order. Experienced users may wish to perform step 3.
Your choice:

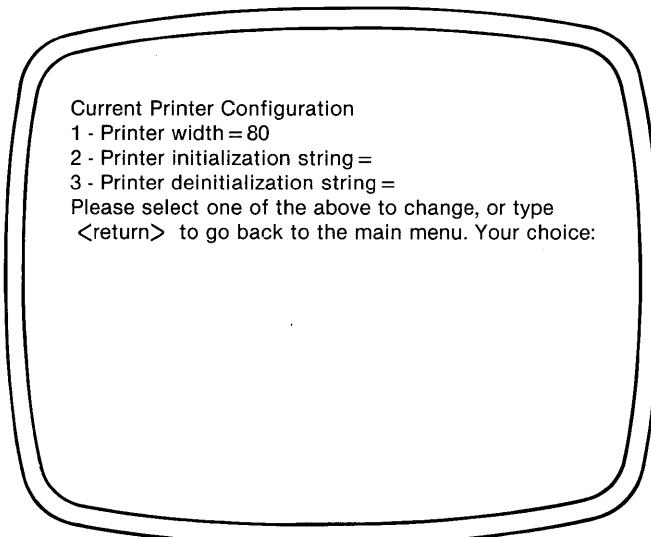
The options on this menu represent the four possible steps which you may use in configuring Perfect Calc. They function as follows:

1. SELECT TERMINAL TYPE

Select this option if Perfect Calc has not already been configured to your computer. You will be presented with a list of various computer terminals. Selecting yours from this list automatically tells Perfect Calc all about your computer. If your computer is not listed, you must select the terminal option 'Other'. Perfect Calc will then ask a variety of questions (unavoidably technical in nature) regarding your computer. You may be able to answer these using your computer manual, but if not, you should consult a qualified computer technician or see your computer dealer for help.

2. CONFIGURE PRINTER

This printer configuration option should be selected following completion of option 1 (or selected first, if Perfect Calc is already configured to your computer). Perfect Calc will present the display:



The Printer Width

The default printer width is initially set at 80 columns. If your printer will print a wider page than 80 columns, e.g. 132, select this option and supply the new value. Perfect Calc returns you to this menu, to allow you the chance of selecting the remaining options.

Perfect Writer

If you want to use your Perfect Calc data files with Perfect Writer, then you will need to set the printer width to 70. The disadvantage with this selection is that Perfect Calc will not print an exact screen image, but will print parts of the spreadsheet which fit into 70 character wide segments.

The Printer Initialization String

An 'initialization string' is a code that tells your printer 'how' to print—i.e. whether or not to compact characters, whether or not to form the characters into a particular shape, size, etc. Initialization strings are commonly used with dot-matrix printers which possess a variety of printing capabilities. The initialization strings which a printer will take are usually unique to that printer, and thus specified in the manual which accompanies the printer.

Perfect Calc will insert the initialization string which you supply, at the beginning of each printed text file. As the first string which the printer encounters when printing a file, it causes the printer to print according to whatever character format the initialization string specifies.

A 'deinitialization string' may also be specified, which Perfect Calc will insert at the end of a file, in order to 'shut off' the particular printing characteristics specified by the initialization string. Deinitialization strings should also be found in the manual accompanying the printer.

Initially, these two options are set with 'null' strings, which will cause the printer to print in whatever default mode it was configured to by its manufacturer.

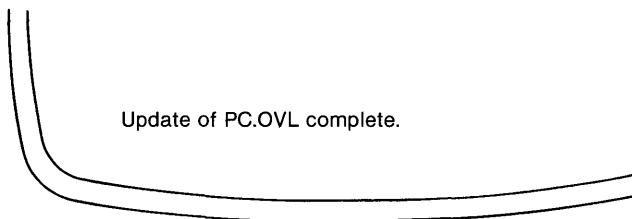
When you have selected and changed the options you wish, type a carriage return to return to the main configuration menu.

3. CHANGE COMMAND KEYS

This option allows you to change most of the command keys. Unless you are an experienced user, we recommend against selecting this option, at least until you have gained some familiarity with Perfect Calc's command structure.

4. UPDATE (WRITE) 'PC.OVL' FILE

If you have made any changes in any of the preceding configuration options (selecting a terminal type, configuring the printer, or changing command keys), you **must** select this option. This option writes the new configuration information to the PC.OVL file, which the other Perfect Calc system programs will reference while operating with your computer and printer. The information is vital to the correct functioning of Perfect Calc, and therefore must contain the most recent updates which you have made. When Perfect Calc has finished updating the PC.OVL file, it will respond with the message:

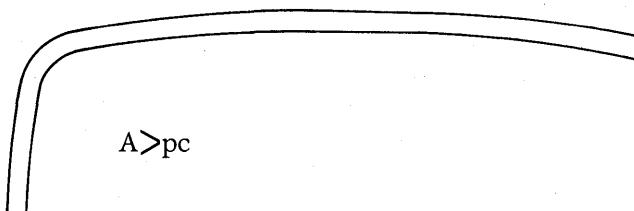


5. EXIT CONFIGURATION PROGRAM

Use this final option to exit the configuration program.

Testing

When you have returned to the disk operating system, type the letters 'pc', following the system command prompt:



Perfect Calc should display to the screen a matrix of columns labeled from 'a' to 'h', and lines numbered from '1' to '23'. A nine-character wide cursor (displayed in inverse video, as an underline, or as facing angle brackets, < >), will be positioned at 'a1'. Some status information will appear at the lower righthand edge of the screen. If you do not see a display similar to the following illustration, you should perform reconfiguration again.

A hand-drawn illustration of a computer monitor screen displaying a table. The table has five columns labeled 'a', 'b', 'c', 'd', and 'e'. The rows are numbered 1 through 5. The 'b' column contains numerical values: 350.00, 250.00, 125.00, etc. An arrow points from the word 'cursor' to the value '350.00' in the second row, second column. The table is enclosed in a rectangular border.

	a	b	c	d	e
1	Jan.	Feb.	Mar.	April	
2	350.00	350.00	350.00	350.00	
3	250.00	250.00	250.00	250.00	
4	125.00	125.00	125.00	125.00	
5					

Note: Once the configuration is complete, the program 'pcconfig' is no longer necessary and can be deleted from the working disk to give you more room for data storage. Later, if you decide to change any of the configuration parameters, the file can be copied again from the original Perfect Calc distribution diskettes, in order to update the PC.OVL file.

Final Note: You may wish at this time to make a further backup copy of the working diskette you have just configured, and which you will use as your actual working copy. This way, should your working copy become damaged, you will not have to preconfigure Perfect Calc to your computer again.

Appendix B

STRUCTURE OF PERFECT CALC DATA FILES

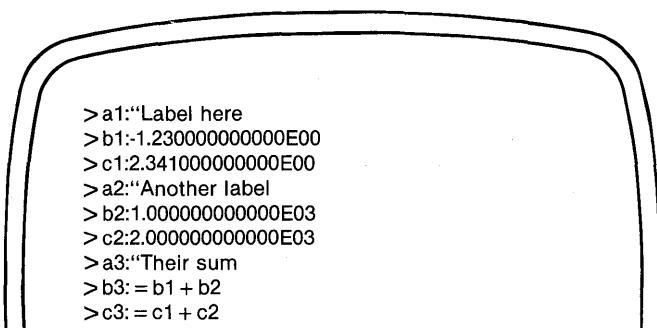
There is no special format for Perfect Calc files as in other spreadsheet programs. Perfect Calc data files are merely listings of the exact keystroke sequences that are used when entering the data on the spreadsheet. This representation of Perfect Calc files allows for a great deal of flexibility in manipulating the file outside of the Perfect Calc environment. You may edit Perfect Calc data files using Perfect Writer, adding or deleting elements as you wish. As well, you can easily transmit Perfect Calc data files over modem lines to other users as you would any other text file.

Consider a sample spreadsheet to see exactly how a data file is structured:

The diagram illustrates the structure of a Perfect Calc data file. It features a header row with three columns labeled 'a', 'b', and 'c' separated by double vertical bars. Below this is a data row with four columns. The first column contains row numbers 1, 2, and 3. The second column contains labels: 'Label here', 'Another label', and 'Their sum'. The third column contains numerical values: '-1.23', '1000.00', and '998.77'. The fourth column contains numerical values: '23.41', '2000.00', and '2023.41'. The entire structure is enclosed within a rounded rectangular border.

	a		b		c
1	Label here		-1.23		23.41
2	Another label		1000.00		2000.00
3	Their sum		998.77		2023.41

This simple spreadsheet would automatically be saved by Perfect Calc as follows:



```
>a1:"Label here  
>b1:-1.230000000000E00  
>c1:2.341000000000E00  
>a2:"Another label  
>b2:1.000000000000E03  
>c2:2.000000000000E03  
>a3:"Their sum  
>b3:= b1 + b2  
>c3:= c1 + c2
```

As you can see, each line consists of a '>' (which means GOTO), followed by a pair of entry coordinates, followed by a colon (which terminates the entry coordinates specification), followed by a value (label, number, or formula), followed by a carriage return, <CR>, which terminates the entry. Typing these entries into a file using Perfect Writer would result in the same spreadsheet placement as if the file had been constructed using Perfect Calc. Each line is independent and may be changed or deleted with your editor as you wish.

Most Control, Escape and Extended commands do not appear in data files. Those that do are typically the Set Width and Set Display commands. These appear as '\d' and '\w' respectively. In this case '>' is available as a synonym for Control---X (or Extended). However, although Perfect Calc does not routinely save its data files using non printable control characters, you are free to enter any Perfect Calc command sequence into the data file even if it contains control characters. See your Perfect Writer manual for further instructions.

Appendix C

TIPS ON USING PERFECT CALC EFFICIENTLY

As you become proficient with Perfect Calc, you will probably want to know something about how it functions internally. In this appendix we discuss some of the more important technical aspects of the Perfect Calc program, suggesting some operating hints to make your use of Perfect Calc more efficient and profitable.

Overlays

Perfect Calc uses a data management technique called 'overlays' to fit its approximately 75k of program code into a 64k or smaller computer. An overlay is simply a portion of the program that is placed on a disk file separate from the main program. Often several overlays may be stored in a single file, as is the case in the file 'PC.OVL'. Within the main program there is a portion of code which is always present in memory, called the 'resident' code. In addition, there is also a fixed window in memory for an overlay of code. If a segment of code is needed which is not present within this window, the 'resident' code goes to a disk file and retrieves the desired code, copying it into the window, 'overlaying' the code already there, with the code for functions that are needed.

By now you are familiar with those times when, given a command, Perfect Calc pauses, as if thinking about what to do. Actually Perfect Calc is loading an overlay into machine memory, which contains the functions needed to complete the command.

Perfect Calc's commands are divided roughly among four overlays according to their functional groupings. For example, Perfect Calc is activated with the 'I/O' overlay resident, since the most common operation used when beginning Perfect Calc is retrieving a data file. Also contained in this overlay are several spreadsheet formatting commands, since formatting is also performed toward the beginning of a Perfect Calc session.

Another overlay is the 'replication' overlay, which contains the commands for copying data into and yanking it back out of the Save Buffer. Also included here are the line/column insertion and deletion commands.

The third overlay contains primarily those commands which implement Perfect Calc's split screen capabilities.

The fourth overlay is used primarily for manipulating Perfect Calc's multiple buffers.

Many frequently used commands, such as data entry and simple cursor commands, are not in any overlay at all, but are held continuously in resident memory. Thus, there is never any waiting involved when these commands are used.

As may already be apparent to you, this functional grouping of commands into overlays can be exploited to minimize the amount of waiting involved in swapping overlays to and from disk. For example, many times you will wish to activate Perfect Calc just to perform a quick analysis. If this does not involve replication, split-windows, or the use of multiple buffers, Perfect Calc may never have to swap an overlay at all. If you use only one of these functional groups, such as replication, you may need to wait for swaps only twice: once when you perform the first replication command, and later when you save your data before exiting.

Data Space Management: 'Virtual Memory'

Today, memory space limitations loom large, and most microcomputer programs exist as a tradeoff between command capabilities and machine power. In Perfect Calc, for example, extra command functions mean less space for data, and Perfect Calc must use disk storage (via overlays) to compensate for a lack of sufficient 'RAM' memory in the computer to store command functions. But even with this, there is not really sufficient storage in machine memory for the data files which you will generate.

Although only 20K or less of user RAM is actually available for data storage on 64K machines, Perfect Calc allows you to use up to 64K bytes of data space during any spreadsheet editing session. This difference in storage capability is made up using a program structure called virtual memory, a technique closely related to the system of overlays discussed previously.

Basically, it works like this: Just as Perfect Calc's overlay management system continuously transfers overlays of computer code back and forth between disk and machine memory, so does the virtual memory segment of Perfect Calc, in a process called 'swapping', continuously read and write portions of spreadsheet data to and from disk. The system keeps track of the data that you are currently using, and transfers to disk only that which it thinks you will not be needing right away. Swapping takes a little bit of time, but not much. Overall you will find that, even if you have multiple buffers filled with spreadsheets, if you keep your attention focused on one of these buffers at a time, very little disk swapping will actually occur.

Perfect Calc uses its virtual memory data space in two ways: On the one hand Perfect Calc uses it to store the spreadsheet entry itself. Each time you type an entry, Perfect Calc reserves storage for that entry in its virtual memory. Some entries take a little memory to store, some take a lot. Second, Perfect Calc stores information regarding the location of the entries in the spreadsheet. This information is known as a 'screen pointer', and indicates where the entry at a particular screen position is located in virtual memory.

Screen Pointer Space

Initially, all of Perfect Calc's virtual memory is free. Moving the cursor around the spreadsheet does not use up any storage. However, when you type an entry, Perfect Calc reserves two bytes of storage within the screen matrix, for the item and furthermore two bytes for each location **above and to the left of it**—i.e., those locations which fall within the rectangular boundaries of the spreadsheet which the entry defines. Perfect Calc assumes that you eventually intend to fill up these locations with values and thus initializes their screen pointers for you. (Notice that if Perfect Calc were to initially have all possible spreadsheet locations set up this way, $52 \times 255 \times 2 = 26,500$ bytes would be required. Instead, Perfect Calc reserves locations only according to your usage of the spreadsheet.)

From this you can infer that it is better to keep your spreadsheets as rectangular as possible, in order to minimize use of screen pointer storage. For example, suppose you have 100 entries to be entered into a spreadsheet. Normally, these 100 entries will be arranged in a matrix which 'makes sense' to you. But suppose in this instance that the problem is to configure them in such a way as to conserve as much pointer space as possible.

If you use a matrix of 100 lines x 1 column, the array of screen pointers will require just 200 bytes of virtual memory storage ($100 \times 1 \times 2 = 200$ bytes), the minimum possible storage. Similarly, if you arrange the entries as two lines of 50 elements each, the result would also be 200 bytes. In fact, any perfectly rectangular matrix would net the same minimal usage of screen pointer storage ($25 \times 4, 5 \times 20$, etc.). However, if you chose to store the first 50 elements in your first line (positions 'a1' through 'X1'), and the second 50 elements in the first column (positions 'a2' through 'a51'), you would use $51 \times 50 \times 2 = 5100$ bytes of storage, or 25.5 times more than necessary! There will be times when non-rectangular spreadsheets must be used, but you should keep this characteristic of Perfect Calc in mind when choosing spreadsheet arrangements.

Entry Contents Storage

As for the storage of entries in virtual memory, the following rules of thumb can be used to determine how much storage a given entry will require:

- Labels require the number of characters they contain plus three bytes. For example, a 10-character label will require about 13 bytes.
- Numbers require 12 bytes of storage, irrespective of their magnitude or display characteristics.
- Formulas, being a combination of labels (character strings) and numbers (their values), require about 12 bytes plus the number of characters they contain. Since some formula compaction is performed, this latter rule is not precise. However, one can estimate that the formula $b2 + (c5 \cdot 2.50)$ will require about $15 + 12 = 27$ bytes to store.

The implication of this is that labels and numbers typically require less storage than do formulas. When storage is approaching 64K and getting scarce, one can try replacing as many formulas with pure numbers as is possible, thus freeing storage space, while retaining information.

Whenever a line or column of entries is deleted, some storage is freed. However, the storage given back is not the entire amount that is used by the entries in the line or column. This is because all deletions are saved temporarily in the Save Buffer. The memory the line or column uses cannot be given back until it has been replaced in the Save Buffer by yet another deletion! Only then is the storage space it occupied given back as useable memory.

Hints on the Use of Virtual Memory

When constructing very large spreadsheets, you will find that having to swap data back and forth to disk can take a considerable amount of time. This can be a burdensome thing when recalculating the spreadsheet, since all of it must be swapped from disk to machine memory and back again for the recalculation.

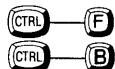
The first, and perhaps obvious, method of handling this problem is to create smaller spreadsheets. This is sometimes easy to do, since large spreadsheets are usually just conglomerations of many smaller ones which can be split into separate buffers. Recalculations usually involve only the current buffer, and as long as you do not reference or associate with a spreadsheet in another buffer, it remains on disk and does not need to be swapped to memory for recalculation. Each time you switch buffers, you may have to wait a moment for the proper data to be loaded into main memory, but this is far more acceptable than having to wait long periods for a single recalculation of your only (and very large) spreadsheet.

If your spreadsheet cannot be broken up into smaller pieces (it really is a ten-year plan by months), then you should consider changing modes of recalculation so that Perfect Calc will recalculate only defined regions of the spreadsheet. (Use the CHANGE RECALCULATION MODE command (Control----X, m), selecting option 'r'.) You will possibly find that 90 percent or more of your spreadsheet is not affected by carefully defined, regional recalculations.

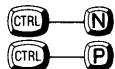
Finally, if nothing else works, remember that you are already using four to five times the memory available with other spreadsheet calculators in the 8080-family of microprocessors, and you might just need to consider buying another, more powerful computer.

Appendix D **COMMAND SUMMARY**

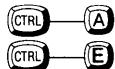
MOVING THE CURSOR



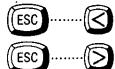
FORWARD COLUMN Command
BACKWARD COLUMN Command



NEXT LINE Command
PREVIOUS LINE Command



BEGINNING OF LINE Command
END OF LINE Command



TOP OF COLUMN Command
BOTTOM OF COLUMN Command

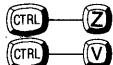
>

GOTO Command

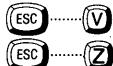


SEARCH to next unlocked formula or number.

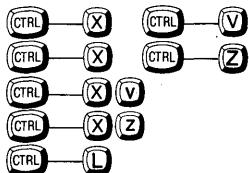
MOVING THE SCREEN



PAGE UP Command
PAGE DOWN Command



PAGE RIGHT Command
PAGE LEFT Command



PAGE DOWN 'Other Window'
PAGE UP 'Other Window'
PAGE RIGHT 'Other Window'
PAGE LEFT 'Other Window'
Redraw window, centering cursor's line.
(or Escape . . . Control---L)†

† for alternate command set

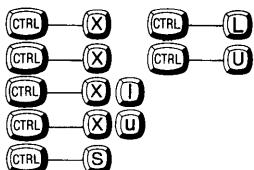
ENTERING DATA AND FORMULAS



Begin formula



Force label



LOCK CURRENT FORMULA Command
UNLOCK CURRENT FORMULA Command
LOCK FORMULAS IN REGION Command
UNLOCK FORMULAS IN REGION Command
SEARCH FORMULA Command

FORMULA OPERATORS

Arithmetic Operators

\wedge	Raise to the power of
*	Multiplication
/	Division
+	Addition
-	Subtraction

Relational Operators

<	Less than
\leq	Less than or equal to
=	Equal to
\neq	Not equal to
\geq	Greater than or equal to
>	Greater than

FORMULA BUILT-IN FUNCTIONS

Arithmetic Functions

abs(x)	Absolute value
avg(x)	Average or mean
count(range)	The count of non-blank entries in a range.
exp(x)	Computes e (the base of the natural logarithm) to the power of x.
int(x)	Finds the largest integer not exceeding x.
ln(x)	Computes the natural logarithm (in base e) of x.
log(x)	Computes the log (in base 10) of x.
lookup(x, range)	Searches for 'x' in the range specified, and returns the value held in the entry position adjacent to 'x'.
max(range)	Finds the maximum value in a range.
min(range)	Finds the minimum value in a range.
npv(x, range)	Computes the net present value across the cash flow range specified, discounted at a rate specified by 'x'.
sqrt(x)	Computes the square root of x.
sum(range)	Computes the sum of a range of values.

Logical Functions

and(x,y)	Returns 'true' (the value 1) if both x and y are 'true' (i.e., x and y are both non-zero). Otherwise returns 'false' (the value 0).
or(x,y)	Returns 'true' (the value 1) if either x or y are 'true' (i.e., either x or y is non-zero). Otherwise returns 'false' (the value 0).
if(x,y,z)	Returns y, if x is 'true' (i.e. non-zero); otherwise returns z.
not(x)	Returns 'true' (the value 1) if x is 'false' (i.e., 0). Otherwise returns 'false' (the value 0).

Trigonometric Functions

atan(x)	Computes the arc tangent (inverse tangent) of x. (x must be in radians.)
cos(x)	Computes the cosine of x. (x must be in radians.)
sin(x)	Computes the sine of x. (x must be in radians.)

DEFINING REGIONS

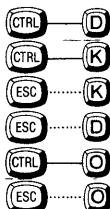


Set mark at current position. (After setting the mark, moving the cursor defines the region as the area between the mark and cursor.)



Exchange Cursor and Mark Command

DELETING & INSERTING



DELETE ENTRY Command

DELETE LINE Command (or Control----C)†

DELETE COLUMN Command (or Escape----C)†

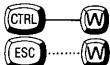
DELETE REGION Command

OPEN LINE Command

OPEN COLUMN Command

COPYING & MOVING

The COPY Commands



The COPY ENTRY Command

The COPY REGION Command

The YANKBACK Commands



Copies an entry, line, column, or region from the Save Buffer to the spreadsheet, **at**, or **beginning at**, the position of the cursor, overwriting whatever data is already present.



Copies an entry, line, or column (but **not** a region) from the Save Buffer throughout a defined region. Used in replicating formulas.

† for alternate command set

ABSOLUTE vs. RELATIVE VARIABLES

When yanking from the Save Buffer, you are asked whether each variable in a formula is relative to its position. Possible responses:



Yes, the relative variable should be adjusted for position.



Repeat previous answer for **every** variable in the Save Buffer (default = relative; i.e. 'yes')

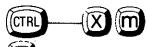


Start the questions over from the first variable.

any other key

No, the non-relative variable should be copied with no change.

RECALCULATION MODES



CHANGE RECALCULATION MODE Command



Recalculate formulas by **lines** (this is the normal default mode).



Recalculate formulas by **columns**.



Recalculate the entire spreadsheet **automatically** whenever the value of a number or a formula entry is changed (this is the normal default mode).

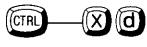


Recalculate the spreadsheet only upon a **manual** request to do so (i.e., when an 'exclamation point' (!) is typed).



Recalculate only the currently defined **region**.

ELEMENT DISPLAY FORMATS



CHANGE DISPLAY FORMAT Command (Global, line or column), followed by one of these choices:

0-13

Causes numeric values to be displayed showing this number of decimal places.



Causes numeric values to be displayed in scientific notation.



Causes numeric values to be displayed in graphic notation.

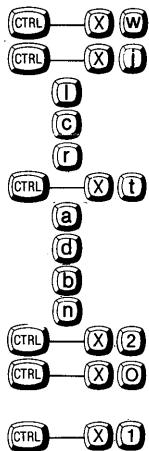


Causes formulas to be displayed at their entry positions.



Causes numeric values to display in a dollars-and-cents format. Also uses commas to indicate thousands.

SCREEN FORMATS



The CHANGE COLUMN WIDTH Command
JUSTIFY ENTRY Command (Global, line or column)

Left justify

Center

Right justify

FIX TITLES Command

Fix only the titles **across** line 1

Fix only the titles **down** column 'a'

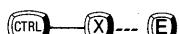
Fix **both** sets of titles

Unlock any previously-fixed titles

CREATE TWO WINDOWS Command

OTHER WINDOW Command (switch cursor to other window)

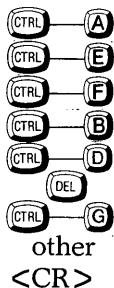
CREATE ONE WINDOW Command



EDITING FORMULA TEXT

EDIT FORMULA Command

Associated strictly with the EDIT FORMULA command are a set of editing commands that are exactly like those used in Perfect Writer. **These commands only work while editing a formula.**



Moves the cursor to the begining of the prompt line.

Moves the cursor to the **end** of the prompt line.

Moves the cursor **forward** one character.

Moves the cursor **backward** one character.

Deletes the character occupied by the cursor.

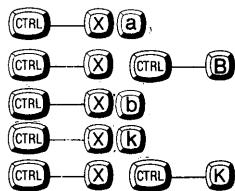
Deletes a character **backwards**.

Go back to the spreadsheet with the original entry.

Insert typed character.

Enter the modifications and return to the spreadsheet.

MULTIPLE BUFFERS AND FILES



- ASSOCIATE FILES Command
BUFFER DIRECTORY Command
SWITCH BUFFER Command
DELETE BUFFER Command
CLEAR MEMORY Command

READING AND WRITING FILES

Save File Commands



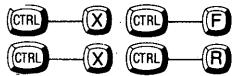
- SAVE FILE Command
WRITE FILE Command

Print File Commands



- PRINT SPREADSHEET Command
PRINT REGION Command

Read File Commands

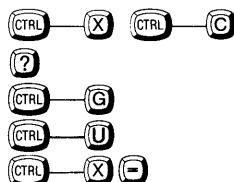
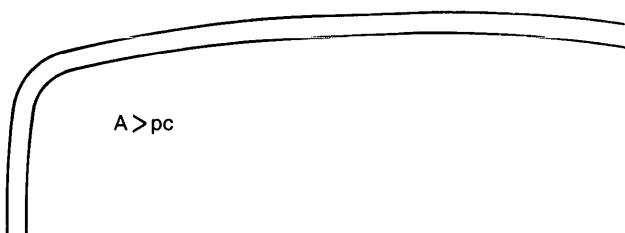


- FIND FILE Command
READ FILE Command

SPECIAL KEYS, EXITING

A>pc

System command invoking Perfect Calc



The QUIT Command

The HELP Command

The CANCEL Command

REPEAT Command

STATUS Command

Appendix E **GLOSSARY**

absolute variable reference

A variable reference which is not dependent upon the spreadsheet location of the formula which contains it (i.e., it will not change, even though its formula may be moved to a new location in the spreadsheet).

bar graph

A display in which numeric values are shown as lines of asterisks (*), such that the length of each line is proportional to the value of the number which it represents. (See **graphic notation**)

blind variable reference

A variable reference to a non-existent entry position. Blind references cause the formula which contains them to compute a value of 'Error!'. Blind references will occur when a line or column referenced by a formula is deleted. A deleted column will be represented by a question mark in the formula variable ('?17'), while a deleted line will be represented by a 'zero' ('h0').

buffer

A temporary working space in computer memory where a spreadsheet file is stored while being edited. Perfect Calc allows a maximum of seven buffers to be activated during any one spreadsheet session.

buffer name

The name assigned to a buffer, usually consisting of the first portion of the filename of whatever spreadsheet is occupying the buffer.

Buffer Directory	A screen listing of those buffers currently holding spreadsheet files, indicating the name of each buffer, the relative size of the spreadsheet it contains, the filename of the spreadsheet, whether the spreadsheet has been saved to disk since it was last modified, and any associated buffers. Invoked by the BUFFERS DIRECTORY command (Control----X, Control----B).
centered	A display format wherein data is centered within its entry box.
circular reference	A situation in which two variables reference each other, either directly or through intervening references, and which, during recalculation, results in faulty values for both.
column	A vertical row of spreadsheet entry boxes, identified by a letter (a - z and A - Z).
coordinates	A designation used to identify the entry boxes of the spreadsheet, consisting of the letter and number of the column and line which intersect to form the entry box.
current column	The column which the cursor is occupying.
current entry	The entry box which the cursor is occupying.
current line	The line which the cursor is occupying.
current window	The window which the cursor is occupying, when the screen has been split into two windows.
cursor	A highlighted box which marks the position on the spreadsheet where data is being entered. Sometimes the cursor is displayed as facing angle brackets, <>, or as an underline.

display format	(see numeric display format)
editing entries	The process of altering a label or formula without having to rewrite it completely; initiated by giving the EDIT ENTRY command.
entry box	The intersection of a line and column on the matrix of the spreadsheet, where data can be entered. Each entry box is identified by a single pair of coordinates.
entry position	(See entry box)
exponentenial notation	(See scientific notation)
file	A space on disk where a single spreadsheet, identified by a unique filename, is stored.
filename	A designation assigned to every spreadsheet stored to disk, consisting of a maximum of eight characters, and ending with the suffix '.pc'. Example, 'ourtaxes.pc'.
fixing titles	Fixing top and left margin titles in place, so that they will remain in view when the screen is shifted.
forcing	Causing Perfect Calc to accept as a label an entry that would otherwise be recognized as a number or a formula.
formula	An arithmetic or logical expression that instructs Perfect Calc to perform an operation upon the numeric data in the spreadsheet. Composed of numbers, variables, operators, and/or functions.
format	(See numeric display format)

forward variable reference	A variable reference to an entry position which, during recalculation, will be updated after the formula which references it has been updated (usually resulting in a faulty value).
function	A pre-structured formula supplied by Perfect Calc which concisely expresses an involved logical or mathematical relationship (e.g., squareroot, cosine(x), etc.).
global	A designation for a command procedure which affects the entire spreadsheet.
graphic notation	A numeric display format in which number values are represented on the screen as asterisks, as in a bar graph.
Help Menu	An on-line directory of Perfect Calc's command procedures, accessible by typing '?' at any point during editing.
left justified	A display format wherein data is displayed beginning at the left margin of its entry box.
line	A horizontal row of spreadsheet entry boxes.
locking formulas	The procedure of locking a formula in place, so that it cannot be moved, edited, deleted, or overwritten, without first being unlocked.
mark	An invisible indicator used by the SET MARK command in establishing the boundaries for a region.
numeric display format	The manner in which numeric values are displayed; specifically, numeric values may be displayed in scientific notation, in graphic notation, in dollars and cents, as either decimals or integers, or as the formulas which compute them.

precedence

The order in which arithmetic operations (multiplication, division, addition, etc.) are performed during the calculation of formulas. Exponentiation (\wedge) is performed first, followed by multiplication (*) and division (/), followed by addition (+) and subtraction (-). Parentheses may be used to change the order of precedence.

operators

The symbols which express the arithmetic relationships existing between the numbers, variables, and functions of a formula (e.g. (\wedge) to the power of; (*) multiplication; (/) division; etc.)

other window

The window which the cursor is **not** occupying when the screen has been split into two windows.

overlay

The procedure of partially overwriting one spreadsheet with another, in order to supply values and data to that spreadsheet for calculation.

Prompt Line

A partially blank line at the bottom of Perfect Calc's window used for displaying labels, numbers, and formulas prior to their being entered into the spreadsheet, as well as system messages and certain status information regarding the current spreadsheet.

range

Any linear or columnar sequence of spreadsheet values.

recalculation

The process of computing the values of the formulas in a spreadsheet.

region

Any linear, columnar, or rectangular (lines and columns) area of the spreadsheet.

relative variable reference	A variable reference which is dependent upon the spreadsheet location of the formula which contains it.
replication	The process of duplicating a formula across a line, column, or region, so that its variable references will accurately reflect the positions of the entry boxes it will occupy.
right justified	A display format wherein data is displayed ending at the right margin of its entry box.
save	The procedure of storing a spreadsheet file to disk for later retrieval and use.
Save Buffer	A buffer space in computer memory in which the most recent deletion is temporarily stored.
scientific notation	A numeric notation which uses an integer or decimal fraction (termed a mantissa) followed by the letter 'e' or 'E' and a positive or negative integer exponent. This notation indicates that the mantissa is multiplied by 10 raised to the power of the exponent. For example, 7.19E + 5 is equivalent to 719,000 (or 7.19 times 10^5).
spreadsheet	The electronic matrix of lines and columns created by Perfect Calc on which data can be entered and calculations performed.
swapping	A program operation connected with Perfect Calc's virtual memory architecture; specifically, the process of continually and automatically exchanging portions of a spreadsheet file to and from disk, thereby utilizing disk storage to enhance computer memory.
synchronized windows	A condition during split-screen editing where in the columns or lines of the two windows will scroll in unison.

value	The number content of an entry box.
variable	A reference to an entry value, expressed as the coordinates of the entry box which contains the value.
variable reference	(See variable)
virtual memory architecture	A design feature of Perfect Calc which allows your computer to work with files larger than its internal memory would otherwise permit.
window	A designation for the computer screen (meaning that it functions much like a window which looks onto a portion of the larger spreadsheet).

INDEX**A**

absolute value {abs (x)} 106; 334
 absolute, variable references 183-185;
 186; 336; 341
 accounts payable 272
 accounts receivable 272
 active buffer 266
 addition operator 101; 333
 adjusting variable references 140
 alternate command set 57; 143; 144
 and, logical function {and (x, y)} 107; 334
 arc cosine 108
 arc sine 108
 arc tangent {atan(x)} 108; 334
 arg: (argument) 53
 arguments, in functions 106
 arithmetic operators 101; 333
 arrow keys 41
 ASCII data files 8
 ASSOCIATE FILES command 265; 338
 associated files 21; 226; 233; 240; 251;
 263; 265
 associated spreadsheets 271
 asterisk 17; 191
 automatic recalculation 103
 average {avg (range)} 106; 334

B

back-up 314
 backspace 69
 BACKWARD COLUMN command 42;
 331
 backward formula references 117
 Bad Formula 99
 bar graph 193; 341
 BEGINNING OF LINE command 45
 beginning Perfect Calc 35
 blanks, in a formula 100
 blind column reference 156
 blind line reference 156
 blind reference 149; 341
 BOTTOM OF COLUMN command 46;
 331

boundaries, spreadsheet 15; 44

buffer 20; 341
BUFFER DIRECTORY command 52; 226;
 266; 338; 342
Buffer in Use! 222
 buffer name 22; 85; 341

C

CANCEL command 32; 69; 339
 Cannot delete current buffer 231
 Cannot put a window there! 241
 center 342
 center, justification 81
CHANGE COLUMN WIDTH command
 79; 86; 194; 337
CHANGE DISPLAY FORMAT command
 190
CHANGE RECALCULATION command
 198; 330
 Check Register Program 288
 Chi-square Analysis Program 309
 circular reference 117; 342
CLEAR MEMORY command 232; 338
 column 342
 column width 13; 72
 also see **CHANGE COLUMN**
WIDTH command
 column, restoring 171
 columns, moving 171; 172
 command characters 36
 command keys 319
 command keys, changing 319
 command messages 31
 command structure 6; 135
 commas with thousands 192
 configuration program 313
 CONTROL key 23; 24
 coordinates 14; 17; 103; 342
COPY 314
 Copy Column command procedure 164;
 179
 copy column, across region 180

COPY ENTRY command 90; 121; 127;
129; 174; 175; 335
copy entry procedure 175
copy entry, into region 176
Copy Line command procedure 161; 177
copy line, across region 178
COPY REGION Command Procedure
181; 335
copying data 160; 174
copying from save buffer 174
cosine {cos(x)} 108; 334
Cost of Goods Statement 297
count function {count (range)} 106; 334
create new buffer 229
CREATE ONE WINDOW command 249;
337
CREATE TWO WINDOWS command 18;
241; 337
cross-referencing 149
current column 342
current entry 342
current line 342
current region 203
current window 16; 342
cursor 16; 342
cursor commands 41
cursor location 85; 203

D

data entry 69
data storage 6; 8; 323
decimal numbers 74; 191
default filename 19
default.pc 19; 22; 36; 229
defining regions 147; 335
delete buffer 230
DELETE BUFFER command 230; 338
DELETE COLUMN command 144; 171;
335
DELETE command 24
DELETE ENTRY command 142; 335
DELETE LINE command 143; 145; 157;
170; 335
delete modified buffers 230
DELETE REGION command 145; 158;
172; 335

deleting 141
deletion commands 142
dependency link 265; 268
DIR command 37
disk drive location 38
display format 190; 336; 343
display notation 74
division 101; 333
dollars and cents format 191
DOS (Disk Operating System) 55
double quote, to force a label 71
drive location 38

E

echoing technique 25
edge-wise cursor commands 43; 44; 62
EDIT ENTRY command 111; 337; 343
editing a formula 111
electronic spreadsheet 3
Elementary Cursor Commands 41; 57; 58
END OF LINE command 44; 45; 331
enter data 69
entry box 343
entry contents storage 329
entry, restoring 169
equal sign 98
equal to {relational operator} 101; 333
error 149; 154; 158; 172; 185
ESCAPE Key 25
EXCHANGE CURSOR & MARK
command 148; 335
exponential 78; 343
exponents 76
extended command 26

F

Family Budget Program 283
file 343
file dependency 264
filename 18; 343
Financial Net Worth Program 286
FIND FILE command 37; 221; 223; 266
fix titles 188; 343
FIX TITLES command 188; 337
forcing a label 71; 333; 343
formula operators 101

- formula, definition of 100; 343
 formula, display 97; 191
FORMULA: 98; 329
 formulas, entering 98; 333
 formulas, using numbers in 100
FORWARD COLUMN command 42; 331
 forward reference 117; 344
 functions 104; 344
 functions, precedence 104
- G**
General and Administrative Expense Statement 298
Global: 80; 82; 344
GO BACK command 32; 69
GOTO command 50; 67; 331
GOTO command, canceling 51
 graphics notation 191; 344
 greater than (relational operator) 101; 333
 greater than or equal to (relational operator) 101; 333
- H**
 hardware requirements 313
Help Command 32; 52; 57; 339
Help Menu 32; 33; 313; 344
 help with command prompts 187
 horizontal split window 239
- I**
if, logical function {if {x, y, z}} 107; 334
Ignore changes this session? 40; 236
 illegal coordinates 51
 illegal numbers 76
 illegal values 112
Income Statement Program 295
 indenting 87
Individual Tax Return Program 289
insert column
 see **OPEN COLUMN** command
insert line
 see **OPEN LINE** command
installation 10
installing Perfect Calc 313
integer function {int{x}} 106; 334
 interdependent spreadsheets 269
- inverse of the natural log {exp(x)} 106
 itemized deductions 292
- J**
JUSTIFY ENTRY command 81; 89; 337
- K**
 key stroke sequence 323
- L**
 label, column width 72
 label, editing a 111
 label, extended beyond column width 72
 label, justification parameter 73; 329
Label: 70
 labels 70
 labels, size of 72
 left justification 81; 344
 legal numbers
 see numbers, legal
 less than 101; 333
 less than or equal to 101; 333
 line 344
 line, restoring 170
 linear region 146
 lines, moving 170; 173
 link 266
 link spreadsheets 263
 location coordinates 17
LOCK FORMULA command 114; 333
LOCK FORMULAS IN REGION
 command 115; 333
 locked 113; 114; 115
 locking formulas 113; 344
log function {log(x)} 106; 334
 logical functions 107; 334
 lookup function {lookup {x, range}} 109;
 334
 lookup column 109
 lookup line 110
- M**
MANUAL RECALCULATION command
 202
 manual, recalculation mode 199; 202
 mark 344

Mark Boundaries Procedure 147
MARK SET command 91; 147; 335
maximum value { max (range) } 107; 195;
 334
memory available to Perfect Calc 203
minimum value { min (range) } 107; 334
mistakes 186
modem 323
moving data 172
moving entries 172
multiple buffer 21
multiple file display 16
multiple spreadsheet display 7
multiple spreadsheets 240
multiple yankback 168
multiplication 101; 333

N
natural logarithm { ln(x) } 106; 334
negative numbers 74
nesting arithmetic operators 102
net present value { npv (x, range) } 107;
 334
NEXT LINE command 43; 331
non-filled entries 105
normalized bar graph 193
not, logical operator { not (x) } 107; 334
not equal to (relational operator) 101
NUMBER: 74
numbers in a formula 100
numbers, definition 74; 75
numbers, justification 82
numbers, legal 75; 76

O
off-screen position 43; 50
one-way dependency 265
OPEN COLUMN command 25; 138; 140;
 335
OPEN LINE command 136; 140; 335
operators 345
operator precedence 102
or, logical function { or (x, 4) } 107; 334
order of dependency 267
order of recalculation 133
other window 241; 345

OTHER WINDOW command 244; 337
other window commands 241
overflow of a number 78
overlays 9; 234; 235; 313; 325; 345
overwriting 149; 169; 172; 175; 179; 181;
 183; 223; 225

P

padding a label 73; 87
PAGE DOWN 'Other Window' command
 245; 332
PAGE DOWN command 48; 332
PAGE LEFT 'Other Window' command
 248; 332
PAGE LEFT command 49; 332
PAGE RIGHT 'Other Window' command
 247; 332
PAGE RIGHT command 49; 332
PAGE UP 'Other Window' command
 246; 332
PAGE UP command 47; 332
page-wise cursor commands 43; 47; 64
parentheses, in a formula 102
Payroll Analysis Program 250; 304
Perfect Writer 2; 8; 41; 208; 210-215; 318;
 323; 324; 337
PIP 314
position of labels 73
position of numbers 81
precedence 102; 345
precision of calculation 77
PREVIOUS LINE command 43; 331
PRINT REGION command 211; 338
PRINT SPREADSHEET command 210;
 338
printer initialization string 318
printer width 318
printer, configure 317
Professional Fee Analysis Program 310
programming language 270
prompt line 17; 69; 70; 345

Q

QUIT command 40; 68; 95; 134; 339

R

raising a number to a power 101; 333
range 104; 345
READ FILE command 224; 235; 338
Reading... 39
Real Estate Analysis Program 308
recalculation 103; 117; 263; 267
recalculation mode 198; 336
rectangular array 105
rectangular region 146
REDRAW SCREEN command 33; 52; 332
reference a value in another file 233; 264
referencing data, between spreadsheet
 buffers 233
region 146; 345
region, defining 147
region, restoring 172
regional recalculation 7; 330
relative or absolute, variable reference
 184; 196; 336; 346
relative...? 127; 184; 186
REPEAT command 53; 59; 139; 172; 339
replicating a formula 119; 121; 127; 167;
 184; 196; 346
replication 326
resident code 325
restoring a column 171
restoring a line 170
restoring a region 159; 172
restoring an entry 169
restoring deletions 169; 173
retrieving a spreadsheet 37
right justification 81; 346

S

save buffer 121; 127; 141; 167; 173; 326
save buffer, current contents 203
SAVE FILE command 95; 232; 237;
 269; 338; 346
scientific notation 74; 77; 78; 191; 346
screen commands 187
screen matrix 328
screen pointer 327
screen pointer space 328
scrolling the other window 244

SEARCH NEXT UNLOCKED ENTRY

116; 331; 333
Selling Expense Statement 298
set mark procedure
 see MARK SET command
sharing data between buffers 233
significant figures 77
sine { sin (x) } 334
split-screen 7; 239; 337
spreadsheet 346
spreadsheet analysis program 9
spreadsheet filenames 18
spreadsheet overlay 234
square root 107; 334
star 17; 191
STATUS command 148; 203; 339
Stock Portfolio Evaluator Program 307
structure of data files 323
subroutines 270
subtraction 101; 333
suffix '.pc' 18
swapping 327; 346
switch buffers 228; 249
SWITCH BUFFERS command 228; 338
synchronize windows 242; 243; 346
SYSGEN 314
system files 313

T

tangent 108
terminal 317
testing installation 320
titles, fixing 188
TOP OF COLUMN command 46; 331
transfer spreadsheet files to Perfect
 Writer 8
turning on Perfect Calc 36
two windows 16

U

underflow of a number 78
unlinking files 279
UNLOCK FORMULA command 114; 333
UNLOCK FORMULA IN REGION
 command 115; 333
unlocked 113; 114

update cycle 268

V

value 347

variable 103; 347

variable references, adjusting 140; 347

vertical split window 239; 241; 242

virtual memory 6; 263; 327; 330

virtual memory architecture 6; 347

W

window 16; 347

WRITE FILE command 22; 207; 269; 338

Y

YANKBACK command 92; 120; 141; 156;

167; 335

yes/no response 31

T.M.

Perfect Software, Inc. 1400 Shattuck Avenue Berkeley, California 94709

