

MICROPOWER SERIES

Multiplan[™] for the Commodore 64[®]

Michael V. Laric/M. Ronald Stiff



*Step-by-step instructions help
you master Multiplan[™] quickly.*

*Includes over fifty applications
for the home and business.*

*Covers the important features of
the Multiplan[™] programs.*

Multiplan™ for the Commodore 64® Laric / Stiff



MultiplanTM
for the Commodore 64[®]

MICROPOWER SERIES

MultiplanTM
for the Commodore 64[®]

Michael V. Laric/M. Ronald Stiff

University of Baltimore

University of Baltimore

wcb

Wm. C. Brown Publishers
Dubuque, Iowa

Consulting Editor:
Edouard J. Desautels
University of Wisconsin—Madison

MICROPOWER SERIES

Multiplan® is a registered trademark owned by Microsoft Corporation.
Commodore® is a registered trademark of Commodore Business Machines.

Cover photo by Bob Coyle.

Copyright © 1985 by Wm. C. Brown Publishers. All rights reserved

Library of Congress Catalog Card Number: 84-72013

Book

ISBN 0-697-00424-4

Book and Diskette

ISBN 0-697-00425-2

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the publisher.

Printed in the United States of America
10 9 8 7 6 5 4 3 2 1

Contents

Preface ix

- 1** **Why Use Multiplan? 1**
- A Brief Example 2*
 - What about Big Problems? 4*
 - Coming Up 5*
 - Summary 5*
- 2** **Getting Acquainted with the Commodore 64 6**
- Physical Components of a Commodore 64 6*
 - Using Diskettes 7*
 - Some Preliminaries 8*
 - Operation of the Commodore 64 8*
 - Using Files 10*
 - Formatting Diskettes 11*
 - Creating a Multiplan Work Disk 12*
 - Summary 13*
- 3** **Using Multiplan: Some Preliminaries 14**
- Entering Multiplan 14*
 - Keyboard Use 18*
 - The Command Lines 20*
 - The Message Line 21*
 - The Status Line 21*
 - Getting Help on Line 22*
 - Writing on the Worksheet 23*
 - Leaving Multiplan 23*
 - Returning to the Commodore 64's DOS 23*
 - Going from Commodore 64's DOS to Multiplan 24*
 - Other Commands 24*
 - Saving a Worksheet 25*

<i>Retrieving a Worksheet</i>	25
<i>Panic Button</i>	26
<i>Using the RETURN Key</i>	26
<i>Summary</i>	27

4 *Solving a Simple Problem* 28

<i>Text and Numbers</i>	28
<i>Using the Cursor Arrow Keys</i>	28
<i>Having Multiplan Compute</i>	31
<i>Printing Worksheets</i>	32
<i>There Must Be a Better Way: The Copy Command</i>	34
<i>Estimating the Value of an IRA</i>	37
<i>Summary</i>	40

5 *Using Functions, Names, and Coping with Change* 42

<i>A Teacher's Gradebook</i>	42
<i>Using Functions</i>	44
<i>Using Names</i>	46
<i>Coping with Change</i>	48
<i>Deleting Rows or Columns</i>	48
<i>Inserting Rows or Columns</i>	49
<i>Adding a Column</i>	51
<i>Planning for Change</i>	52
<i>Erasing Errors</i>	53
<i>Correcting Errors: The Edit Command</i>	54
<i>The Move Command</i>	56
<i>Summary</i>	56

6 *What If It Won't Fit on the Screen?* 58

<i>Window: Scrolling through the Worksheet</i>	58
<i>Printing Long Worksheets</i>	59
<i>Printing Wide Worksheets</i>	60
<i>A Note on Printing More Columns Using Compressed Print</i>	62
<i>Sorting Labels and Numbers</i>	62
<i>Freezing Titles: The Window Command</i>	64
<i>Multiplan in Stereo: Link Screens</i>	66
<i>Summary</i>	67

7 *A Picture from Multiplan* 68

<i>Getting Graphical Output</i>	68
<i>Changing Column Widths</i>	71
<i>Long Titles</i>	72
<i>Error Flag</i>	73
<i>Scaling for Graphs</i>	73
<i>MAX and MIN</i>	73
<i>IRAs Revisited</i>	75
<i>Summary</i>	78

8	More Complex Calculations	79
	<i>Mathematical and Engineering Functions</i>	80
	<i>Trigonometric Functions</i>	81
	<i>Other Functions</i>	81
	<i>Using Multiplan as a Programmable Printing Calculator</i>	82
	<i>Printing Formulas</i>	83
	<i>Faster Computing</i>	83
	<i>String Functions</i>	83
	<i>Logical Functions</i>	84
	<i>Other Logical Functions</i>	86
	<i>NPV, Net Present Value Function</i>	86
	<i>LOOKUP</i>	87
	<i>Summary</i>	89

9	Controlling Formats and Options	91
	<i>Global Formats</i>	91
	<i>Changing Global Formats: The F D Command</i>	92
	<i>Protecting Entries: The Lock Command</i>	95
	<i>Changing Cell Formats: The F C Command</i>	96
	<i>The Options Command</i>	96
	<i>Summary</i>	99

10	Using Disk Files	101
	<i>Reading Your Disk Directory</i>	101
	<i>Saving: The T S Command</i>	102
	<i>Print Files</i>	103
	<i>Using Multiplan Data in Other Programs: The SYLK® (SYmbolic LinK) File Format</i>	103
	<i>Reading Data From VisiCalc® Programs into Multiplan</i>	104
	<i>Creating Multiplan Programs: The eXternal Command</i>	105
	<i>Deleting Files</i>	111
	<i>Duplicate File Names</i>	111
	<i>Summary</i>	111

11	Personal Finance Case Studies	113
	<i>Case Study A: Expense Log</i>	113
	<i>Dollar Format</i>	114
	<i>Another Approach</i>	115
	<i>Expense Log with Categories: Drawing Lines</i>	117
	<i>Summary</i>	118
	<i>Case Study B: Portfolio Evaluation</i>	118
	<i>Portfolio Evaluation with Gains and Losses</i>	121
	<i>Copy Revisited</i>	123
	<i>Avoiding Retyping Data</i>	123
	<i>Another Variation</i>	124

<i>Summary</i>	125
<i>Case Study C: Computing Net Worth</i>	125
<i>Assets</i>	126
<i>Liabilities</i>	127
<i>Summary</i>	130

12	<i>Business Case Studies</i>	131
	<i>Case Study A: Sales Projection</i>	131
	<i>Formatting the Report</i>	135
	<i>Another Approach</i>	136
	<i>Summary</i>	137
	<i>Case Study B: Interest Computations</i>	137
	<i>Monthly Payments</i>	137
	<i>Interest Tables</i>	138
	<i>Multiyear Interest Tables</i>	140
	<i>Summary</i>	141
	<i>Case Study C: An Income Statement Template Using</i> <i>“eXternal”</i>	141
	<i>Summary</i>	146

13	<i>When to Avoid Multiplan</i>	147
	<i>Other Problems</i>	148
	<i>Other Multiplan Versions</i>	148

14	<i>Summary of Multiplan Commands</i>	149
	<i>Format Commands</i>	149
	<i>Mathematical and Engineering Functions</i>	150
	<i>Trigonometric Functions</i>	151
	<i>String Functions</i>	151
	<i>Other Functions</i>	151
	<i>Transfer Commands</i>	151
	<i>Printing Commands</i>	152

15	<i>Using the Optional Diskette</i>	153
	<i>Loading a Worksheet</i>	153
	<i>Diskette Content</i>	155

	<i>Special Characters (Symbols) Appearing in</i> <i>Multiplan</i>	156
--	--	-----

	<i>Index</i>	157
--	---------------------	-----

Preface

This book is written for owners of the Commodore 64 computer interested in using Multiplan or those owners who are considering the purchase of Multiplan for their Commodore 64; likewise, for people who have seen or used Multiplan at work or on an expensive computer, who may be considering buying the Commodore 64 in order to use Multiplan at home.

Multiplan®, a second generation spreadsheet program, makes the Commodore 64 as easy to use as a calculator, yet gives you far more power than a calculator. This book shows you how to use Multiplan effectively. It is designed so those with no previous experience with computers, as well as experienced users, can learn to use Multiplan effectively. Even experienced users may find new ways of doing things, as well as ideas for new applications.

The first chapter sketches the types of uses in which Multiplan excels. The next chapter shows you how to start using the Commodore 64 computer. Then we start looking at the specifics of using Multiplan. Although it is easier to read the chapters in order, you can often skip ahead to look into some feature of special interest to you. The table of contents and a comprehensive index will help you find your way.

Most of Multiplan's features are presented while solving a sequence of realistic problems, such as evaluating the accumulation in an IRA (Individual Retirement Account). Some of these problems are solved a second time, so you can better appreciate the contrast between approaches.

Almost everyone has to come to grips with numbers, lots of them, at one time or another. Multiplan is such a powerful assistant in helping you manage numeric information that you should seriously consider equipping yourself with this tool. Perhaps this book will help you decide.

Although the book contains detailed instructions on how to do everything shown, an optional diskette may be used to reduce the typing which would otherwise be required. See Chapter 15 for further information. It is assumed that you have Multiplan for the Commodore 64 computer. *If not, this book will give you a good idea of the things you can do with Multiplan.*

1

Why Use Multiplan?

Electronic computers have been in use for more than 30 years. Why is it that Multiplan and similar programs are among the best selling computer software packages since the beginning of the computer age? These programs offer a friendly and straightforward set of commands that allows you to manipulate complex formulas and data. The tables you create can be easily changed by the computer following your commands. The greatest appeal of these programs is how easily they can be applied to a wide variety of problems from personal finances to real estate calculations to business planning analysis.

What is it about Multiplan that sets it apart from ordinary computer programming languages? There can be no doubt that any solution Multiplan can produce can also be produced by writing a computer program to generate that solution. As a matter of fact, computer programs using languages such as BASIC, COBOL, or PASCAL can be written to solve most anything. However, you need to become an expert programmer before you can even approach the solutions which Multiplan can provide. After all it took many programmers, many man-years to write Multiplan.

When you use Multiplan, you *don't* have to write a program. You enter the relevant data, by either typing it directly or reading it in electronically. There is no way to avoid data entry, no matter what you are using. You then specify how the data items are related by writing formulas, which are in essence a program. Then you specify what answers you want calculated. At this point, the computer expert might object: "This is the same as writing a program." Theoretically that may be true; in practice, it makes all the difference in the world. Writing computer programs can be very tricky and time-consuming, and that is after you have invested your time and effort in learning the programming language.

When you solve a new problem using a computer, you usually have to write a program to solve that problem using one of the programming languages we mentioned earlier. That usually means you also have to use some kind of editing program just to prepare the program you need. Then you may have to prepare a data file, with the help of the editing program.

The data file is considered *input* to the program you wrote, and the results produced by your program are called its *output*. We often depict the input as flowing into the program which has been placed into the computer's memory, and show the results produced by your program as flowing out from it, as we see in Figure 1.1.



Figure 1.1 Input, program, and output relationships

After you examine the output of your program, you may come to one of several conclusions:

1. Great—let's stop computing.
2. Oops!—there must be a "bug" in the program; try to find it and fix it.
3. It's ok, but what if . . . ?

Case 3 might involve wondering what would happen if some data item had a slightly larger value, or what would happen if the formula in the program was just a little different.

With Multiplan, you are more likely to get the first conclusion quickly. "Bugs" are much less probable, simply because Multiplan won't let you request many ridiculous computations. Multiplan really shines in the "what if" situation. You simply change the desired number or formula, and you immediately see the consequences. You don't have to fool with an editing program, or even request that your program be rerun.

With the conventional approach, the input data is laid out for the convenience of the computer program (or whoever wrote the program). The actual layout of the input data may have little or no spatial relationship to the results. With Multiplan, you begin by putting the numbers where they should be at all times. Then you decide where the results should appear, in relationship to the other input. If you should change your mind, it is a simple and painless matter to move things around. If you then decide to change a number, you simply locate its old value where you expect to find it (not at some strange location chosen for a program's convenience). As soon as you change that number, all the other outputs depending on that number immediately change.

A Brief Example

The following simple example will give you a better idea of the difference between solving a problem using Multiplan versus solving a problem using the conventional computer approach.

Suppose you were a budding author and had just had your manuscript accepted by a publisher. The publisher might have proposed that you be paid royalties based upon the following sliding schedule:

- Earn 5% of selling price, for the first 3,000 copies.
- Earn 7% of selling price, for the next 4,000 copies.
- Earn 8% of selling price, for the next 5,000 copies.
- Earn 10% of selling price, for all additional copies.

With Multiplan, you would lay out the essential data in the form of a table, as we see here in Table 1.1.

Table 1.1 Raw Data

Rate	Copies
5	3,000
7	4,000
8	5,000
10	?

You would then make an educated guess as to what number to use for the ? in Table 1.1. You might like to project your potential income (before taxes) if the book sold, say 50,000 copies, assuming that it sells for \$1 per copy. So the ? is replaced by 50,000-(3,000+4,000+5,000), giving us \$38,000. You then ask Multiplan to work out the product (rate multiplied by copies) for each line, which results in Table 1.2, since our rate figures are actually percentages.

Table 1.2 Projected Income

Rate	Copies	Income
5	3,000	150
7	4,000	280
8	5,000	400
10	38,000	3,800

Of course, what you really want is a running total, so you ask Multiplan to produce another column, headed "Total," which is to show, for each row, the sum of the current and all preceding income figures. That being done, you now see the results in Table 1.3.

Table 1.3 Projected Total Income

Rate	Copies	Income	Total
5	3,000	150	150
7	4,000	280	430
8	5,000	400	830
10	38,000	3,800	4,630

This may seem like much ado about nothing at this point. You could have done the same thing with a calculator, or merely with paper and pencil. True, but having gone this far, you can now begin the "what if . . ." phase. As the budding author, you might want to negotiate a better royalty schedule.

So you begin by thinking, "What if my first book is not a terrific success? I should ask for a higher percentage for the first few thousand sold, just to be safe." Suppose you settled on 6, 8, and 9, in place of the 5, 7, and 8 that was offered; you don't want to be greedy. You can now proceed to replace the 5, 7, and 8 by the 6, 8, and 9 and, 10 and behold, you immediately see the consequences, as shown in Table 1.4.

Table 1.4 Projected Total Using New Rates

Rate	Copies	Income	Total
6	3,000	180	180
8	4,000	320	500
9	5,000	450	950
10	38,000	3,800	4,750

You might think about these totals for a while, and because you are now convinced you have a best-seller on your hands, perhaps it would be better to focus on negotiating a better top rate than the 10 offered. After all, the other rates will make little difference if you sell 100,000 copies. So you change the 10 to 20 (dreamer) and you decide to increase the 38,000 by 50,000. The results are shown in Table 1.5.

Table 1.5 New Projected Totals

Rate	Copies	Income	Total
6	3,000	180	180
8	4,000	320	500
9	5,000	450	950
20	88,000	17,600	18,550

If you were in the publisher's shoes, you would also like to be able to juggle figures. As the author, you have much less experience with the consequences of manipulating either the rates, the sales thresholds, or the number of steps in the schedule. The publishers can probably do it mentally; you could use a little help.

You could, of course, have written a computer program to perform these calculations. The data file for this program could very well look like the following list (using the first set of rates and copies):

```
5,3000,7,4000,8,5000,10,38000
```

which is awkward to read and change. It might have looked like

```
05 3000 or      05070810
07 4000          03000 04000 05000 38000
08 5000
1038000
```

where you must place leading zeroes or leading blanks in the data. This is because the programming language may require input to be in specific columns, with precise spacing between these numbers, depending upon the programming language used in writing the program. You could easily provide data which looks correct to the naked eye, but which the program takes to be 10 times larger (or 10 times smaller) than what you had in mind.

The data file

```
5  3000
7  4000
8  5000
10 38000
```

seems to have the right numbers, even though the alignment for the 4000 and the 10 is a little sloppy. Some computer programs would interpret this 4000 as if you had written 40,000 because the actual position of the number on the line was critical. This is much less of a problem with Multiplan. You immediately see what Multiplan thinks you just typed. If it is not what you intended, you can change it right away.

What about Big Problems?

Suppose you had a really big problem to solve. Could Multiplan handle it? Suppose you were going to be a real estate tycoon and were working out a 10 year projected statement of cash flow. Such a statement is a table with at least 10 columns of numeric data and some 33 rows, depending upon how much detail you want to include in projecting your cash disbursements. Then you also need some labels to keep track of things. If each column is to hold numbers as large as 8 digits, and we leave a little space between columns for ease of reading, the cash flow table will be about 120 characters wide. Since most computer video display terminals (VDTs) can display only 24 or 25 rows of 80 characters at one time, you would be hard pressed to squeeze all of the cash flow table onto the VDT screen at once.

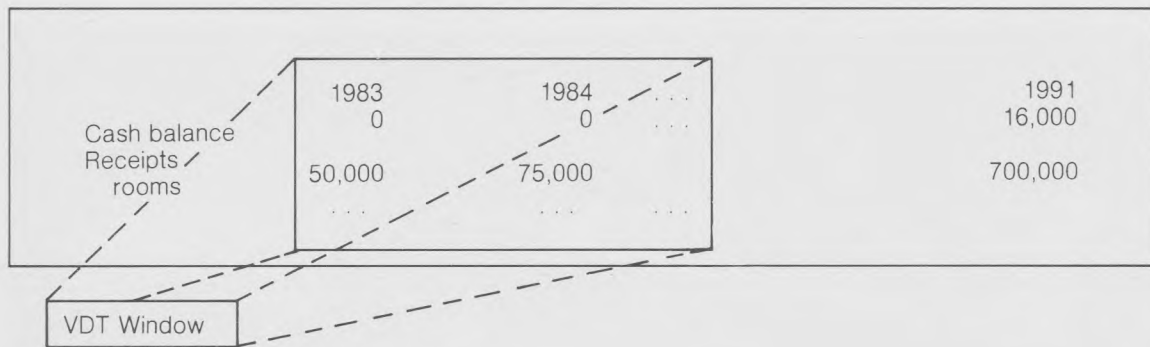


Figure 1.2 VDT Window into a Large Table

With Multiplan, you can construct and display tables with as many as 63 columns (each 8 characters wide) and 255 rows, and you can “browse” over parts of the table very easily. If the whole table won’t fit on your screen, Multiplan treats your VDT screen as if it were a “window.” Your screen window lets you see any part of the table you wish getting as much of it as will fit on your screen at one time. Trying to do this with a conventional computer program would be far more difficult. Figure 1.2 illustrates the idea of a window.

Coming Up

Specifics on using Multiplan: How do you tell Multiplan to do something? What if you make a mistake? What if you are in the middle of filling out a worksheet and have to leave suddenly: How can you save worksheets and recall them later? These and many other features of Multiplan will be described, discussed, and illustrated as we proceed. It goes without saying that you will learn more, faster, and better if you can be using Multiplan and trying the things we are discussing. We hope, however, to provide sufficiently detailed examples so that you can follow what is going on even if you don’t have immediate access to a computer equipped with Multiplan.

Summary

□ Multiplan is data oriented, very much like a calculator. You begin with your own raw data, laying it out on what amounts to an electronic worksheet and using your VDT’s screen as an easily erasable worksheet. Then you begin specifying the relationships between your data and the desired results. You build up to the desired end product in a step-by-step fashion, seeing the results at every step. You always see your input data in the natural spatial relationship intended with respect to any computed results.

□ Problem solving with computers, using the conventional approach, is program oriented rather than data oriented. Most people who are not computer experts feel more at ease with the data they know well than with unfamiliar computer programming languages. Most people are familiar with the everyday use of a simple worksheet. Multiplan combines the ease of using a calculator and the familiarity of a worksheet with the power of a computer. It follows that most people will find that Multiplan provides a natural, user-friendly way to make the computer work.

2

Getting Acquainted with the Commodore 64

This chapter deals with the operation of Multiplan on the Commodore 64. No previous experience with computers is required to use Multiplan with it. Here the rudiments of turning the computer on and setting it up so you can use Multiplan effectively will be described. If you are already familiar with this computer, you might want merely to skim most of this chapter and proceed to the next (you should read the section on preparing a Multiplan working diskette, to make it possible to manipulate large files and store them without having to swap diskettes).

Physical Components of a Commodore 64

Every Commodore 64 consists of a central processing unit (CPU) built into the keyboard. It can be hooked to your television screen or another display device. In order to use Multiplan, a diskette drive is also required. The computer can have optional devices, such as a printer, as well as a second disk drive (which is also called a floppy disk drive, or simply a drive).

Figure 2.1 shows the printer, a floppy disk drive, and the keyboard. The Commodore 64's disk drive, the VIC 1541, also referred to as the 1541, can hold a standard five-inch diskette. Each diskette can record approximately 175,000 characters of information (174,848 to be exact). Some of these characters may be used to represent computer programs, such as Multiplan, and some of them are used to represent your data. The word "byte" is often used in place of the word "character"; for our purposes, these words are equivalent.

The Commodore's disk drive is designed so that you can add additional devices to it. Up to five disk drives and a printer can be attached to your computer via the 1541 disk drive. If you have two 1541 disk drives on the Commodore 64, you can have immediate access to almost one-third of a million characters of information. The Commodore 64 can also be attached to the 4040 dual disk drive or other drive. In this book we will assume that your computer has one disk drive, the 1541. We refer to this drive as drive 8. This is the device number affixed to it in the factory. Should your system have a different drive, refer to your User's Manual for that drive.

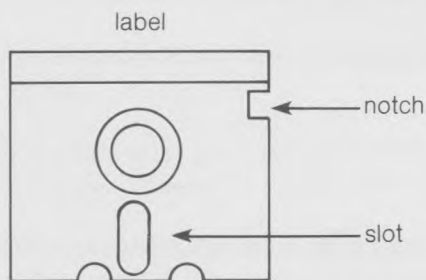
If you should add a second drive we recommend that you place it to the right of drive 8. Moreover you would have to have your dealer change its designation to drive 9. Occasionally we mention a second drive and will refer to it as drive 9. You may, of course, already have the Commodore's Dual-Drives, in which case, the above is known to you as drives 1 and 0.



Figure 2.1 The Commodore 64. (Courtesy of Commodore Electronics Limited)

Using Diskettes

Diskettes are normally stored in protective jackets or sleeves, and they should be kept in an upright position. When you are about to use a diskette you should remove its protective jacket. This will reveal a rigid black, square protective envelope. The envelope has an oval slot through which you can see the diskette. *Never* touch this open area in the square envelope. The disk drive's read and write mechanism uses this opening to record on, and read from, the diskette. The envelope is shown below.



If you see a small notch on the edge to the right of the label, the information on that diskette can be changed and new information added. In such a case, we say that this diskette is "write-enabled." If, on the other hand, you see a piece of tape in the same position covering the notch, then the computer will refrain from changing any information on that diskette; the diskette is said to be "write-protected."

If you are inserting your one and only Multiplan diskette into the diskette drive, make sure its write-protect notch is covered. That will prevent accidental overwriting of your master copy. Your Multiplan diskette is then identified as being "write-protected."

When inserting the diskette into the disk drive, the label side should be facing up and away from the drive's door; the edge furthest from the label must be inserted into the drive first. Gently push the diskette through the drive door slot. Then, push the drive door down.

Each Commodore 1541 disk drive has two indicator lights, a green one and a red one. The green light is lit as soon as you turn the drive on. It will have a steady glow under normal conditions. If the green light blinks, it means that something is wrong; check your manual. Some common errors include: no diskette in the drive, diskette positioned wrong, attempting to write on a write-protected diskette, or

trying to read a file which does not exist. Normally, the green will stay lit until the power is turned off. Always turn the drive on before inserting a diskette. Likewise, take the diskette out *before* turning it off.

The red indicator light is located just below the disk drive slot. This red indicator light will glow whenever the disk drive is being used by the computer. You should not attempt to insert or extract a disk when this light is glowing. Doing so could destroy the diskette and damage the disk drive. If it seems that the indicator light won't ever stop glowing, it would be better to turn off the computer to perform this operation.

Some Preliminaries

Multiplan is very easy to start up and use, but we are in a situation similar to the one faced by the purchaser of unpainted furniture. We have some one-time setup work to do (e.g., sanding, priming, and painting) before we can use the furniture. Before we can proceed with this work, which we need to do only once, we must learn to use another tool, the Commodore 64 Disk Operating System (DOS), which is stored on a chip in the 1541 disk drive. The 1541 user manual has nearly 60 pages. Don't be alarmed! We can discuss what we need to know about DOS in just a few pages.

The computer program managing all the computing resources used on or by the Commodore 64 is called the Disk Operating System (DOS). We will use it to prepare blank diskettes so they can be used to store and recall Multiplan worksheets; this operation is called *formatting*. We will also discuss a few more DOS facilities that will make our use of Multiplan more productive.

Operation of the Commodore 64

We will walk through the steps required to start computing, and we will discuss the purpose of these steps in case you are wondering about what is going on.

1. Turn on the display monitor or TV. Next, turn on the Commodore 64 by pushing the recessed on/off switch up. It is located on the right side of the system unit towards the rear.

As soon as the above is done, your display screen should greet you with a message similar to this:

```
**** COMMODORE 64 BASIC V2 ****
64K RAM SYSTEM 38911 BASIC BYTES FREE
READY.
```

Immediately below the "READY." display you will notice a small blinking square, called a cursor. This indicates that the computer is ready for your instructions and also shows where the next character will be typed.

2. At this point you should turn on the disk drive (drives) (a recessed switch is in the back, opposite the drive's opening). The two indicator lights will glow, and the red light will stop glowing after a few seconds.

Figure 2.2 shows you the keyboard we will be working with. It is very much like a typewriter keyboard, plus a few extra keys that will prove to be very useful as we go on.

Try typing something, such as `abcd`; as soon as you type it, it appears on the screen, but nothing else happens. Next, follow it with a press of the RETURN key, which is located on the right-hand side. The RETURN key is the large key beneath RESTORE. As soon as you press RETURN, what you typed is displayed on the screen. Notice that the Commodore 64 instantly responds with a message of its own, followed by the "READY." sign which we observed earlier. Your Commodore 64 evaluates



Figure 2.2 Commodore 64 keyboard

everything you type as soon as you press RETURN. If what you typed has meaning for your Commodore 64, the computer will do what you asked—print, calculate, display data, or some other operation. If what you typed does not mean anything, the Commodore will indicate that by

```
?SYNTAX  ERROR
READY.
```

This means that what you typed does not match the Commodore 64's vocabulary. Alternatively, if you want to tell the computer to PRINT (that is to display whatever follows it) as in:

```
PRINT 443+8790
```

the computer will evaluate the sum and display the answer 9233, followed by its READY. prompt. On the other hand, if you made a mistake in the order in which you keyed PRINT so that PRITN 443+8790 was keyed in, the Commodore 64 will display the ?SYNTAX ERROR message.

You can correct an error while typing a response by pushing the INST/DEL key located on the top row of the keyboard, immediately to the right of the key marked with CLR/HOME. Do not confuse this with the left-arrow key, which is in the left-hand corner of the top row. If you push the INST/DEL key it will erase the last character you typed. You can then type the correct information.

You have to push the RETURN key to cause your computer to examine your response, but as a general rule no special character will be displayed at the point you pushed the RETURN key. (Note: Whenever you see the word RETURN included among other items to be typed, simply push the RETURN key—do not type the 6 letters R, E, T, U, R, and N.)

The "READY." prompt indicates that your Commodore 64 is waiting for you to issue a command. When you type in a statement or a command line (always end it by pushing RETURN), the Commodore 64 will perform the command and indicate it is ready for more work by displaying another "READY." prompt.

Using Files

As mentioned earlier, the 1541 disk drive has an "operating system" which manages operations and computing resources. The 1541 disk operating system (DOS, for short) provides an umbrella under which other computer programs such as Multiplan can function.

Almost anything you do with DOS involves using "files." A file is a body of information to which you have assigned a name and which you have recorded on a diskette. DOS keeps track of which files are where by maintaining a directory of file names on each diskette. The diskette's directory contains the names of all files on that diskette and other information about these files. You can examine a diskette's directory by placing it in drive 8, closing the drive door, and typing the DOS command

```
LOAD"$",8
```

in response to the "READY." prompt we saw earlier. The screen will display the following 3 lines:

```
SEARCHING FOR $
LOADING
READY.
```

The directory is now in the Commodore's memory. By loading it you copied the directory's contents from the diskette to the computer's memory. To view it on the screen, type

```
LIST
```

A typical directory of a master Multiplan diskette is shown below:

```
0 "Multiplan 1.06 " RP 2A
1 "MP" PRG
81 "MP.CODE" SEQ
26 "MP.DATA" SEQ
45 "MP." PRG
2 "MP/COPY" PRG
1 "ASCII TO C64" PRG
1 "C64 TO ASCII" PRG
4 "MP.COPY" PRG
97 "MP.SYS" REL
152 "MP.HLP" REL
9 "MPC" PRG
224 BLOCKS FREE
```

This directory shows the eleven files listed on the distribution diskette, the one sold by HesWare, the company which licensed Multiplan from Microsoft so as to adapt it for the Commodore 64. The first (top) row of the directory shows the name of the diskette and its ID number. This line is always displayed in reverse video (i.e., the characters appear in background color within squares which take on the color of the characters which are not in reverse video mode). The directory also shows the amount of remaining free space in blocks (on the last line) and Multiplan's version number (1.06 in the above example) on the first line. The 2A identifies the version of DOS on the disk drive used to format the diskette.

A diskette can hold as many as 144 files provided their sizes do not exceed a total of 160,000 bytes. If you are running out of space on a diskette, look for the largest file you no longer need and erase or delete it. We will see how Multiplan lets us delete files which you no longer require. You can also delete files using the DOS command, S, for SCRATCH. For instance, typing

```
OPEN 1,8,15
PRINT#1, "S0: EXAMPLE.MSG" (Note zero following S.)
CLOSE1
```

would "erase" file EXAMPLE.MSG and free space on the diskette. Every once in a while it is a good idea to examine a diskette's directory to see if the diskette is filling up, and perhaps delete files that are no longer needed. Be careful, however, not to erase your Multiplan programs. KEEPING THE WRITE-PROTECT TAB WOULD PREVENT THAT.

If you accidentally erase any of the Multiplan programs you will appreciate not having used the original program disk, which can now be used to run Multiplan while the damaged diskette is mailed back to HesWare for replacement. IT IS A GOOD IDEA TO IMMEDIATELY MAIL IN THE REGISTRATION CARD AS WELL AS A REQUEST FOR THE BACKUP COPY AS SOON AS YOU OPEN THE MULTIPLAN PACKAGE. That way you will not be stuck without the Multiplan Program in case of accidental erasure.

You can print a diskette's directory by turning your printer's POWER switch on. After doing this, you must open a channel to it by using the OPEN statement. The printer is device number 4 (just as the disk drive was assigned the number 8). So you type the DOS command

```
OPEN 1,4
```

which allows you to send all print commands to the printer as in

```
PRINT1, "SEE JANE RUN"
```

The printer would print SEE JANE RUN and advance one line.

To get a directory listing, you must first load the directory, by typing

```
LOAD"$",8
```

if you have been prompted with "READY." Remember that every DOS command line must be ended by pushing the RETURN key. Your directory is now in the computer's memory. Next you must open the printer by typing

```
OPEN 1,4  
CMD 1  
LIST
```

If nothing is printed, your printer is not set up properly for the Commodore 64. Check your *Commodore 64 User's Manual*, the printer's manual as well as the interface's manual for details.

Formatting Diskettes

Formatting is an operation which permits you to prepare blank diskettes so they can be used to save Multiplan worksheets and other files for repeated use. Formatting diskettes is done using DOS. When a diskette is formatted, it gets a name and a number. The diskette ID (name and number) is loaded into the Commodore 64's memory. Whenever a diskette is accessed (to load files or save them) the number in memory is matched with the one in the diskette's directory. Only if the two match will the operation be carried out. This way the Commodore 64 assures you proper disk management and guards against mistakes.

You can format a diskette by using the following commands:

```
OPEN 1,8,15  
PRINT#1, "NEW:DISKNAME,ID"  
CLOSE1
```

You cannot format a diskette if it is write-protected. Furthermore, the process of formatting a diskette will erase any information that was previously recorded on it. The format program takes about a minute or two to complete. It will prompt you when it finishes.

Creating a Multiplan Work Disk

The original disk supplied with Multiplan is not intended to be used as the working copy of the Multiplan program. You will need more space on the diskette than is available on the master copy which came with the program. In order to match Multiplan to your Commodore 64's capabilities, you must carefully follow the instructions provided by the HesWare Company in the *Multiplan User's Guide and Reference Manual*, which is supplied with the program. We shall first explain the steps and then list them here.

The master diskette contains several programs, in addition to the Multiplan programs. The additional programs include a copy utility (MP/COPY), two conversion programs (from ASCII to C64 and from C64 to ASCII), as well as a screen color program (MPC). The roles of these will become apparent as we proceed.

At this point, we will need only the copy utility, which will be used to copy two Multiplan files, *MP.HLP* and *MP.SYS*, to working diskettes. Once this is completed, the master diskette will only be used to load the Multiplan program to the computer's memory. After this is done, we can replace the master diskette with the working diskette, which contains the *MP.HLP* and *MP.SYS* files. This will allow us to store and retrieve files as well as use Multiplan's on-screen HELP without having to swap diskettes.

To accomplish all that, we need to go through several steps. First, you should format several diskettes. As we saw earlier this is done by typing the following lines:

```
OPEN 1,8,15
PRINT#1,"N0:DATADISK,ID"           (Follow with RETURN.)
CLOSE1
```

Second, we must copy the two files called *MP.SYS* and *MP.HLP* to your newly formatted work diskette. You should insert the master diskette (the one containing Multiplan) into the drive, close the door, and type

```
LOAD"MP/COPY",8                     (Follow with RETURN.)
```

The next time you see the **READY.** prompt, type

```
RUN                                  (Follow with RETURN.)
```

The program will prompt you for the name of the file you wish to copy. Type

```
MP.HLP                               (Follow with RETURN.)
```

and watch for the message:

```
INSERT SOURCE DISKETTE . PRESS RETURN
```

Since the source diskette is already in the drive, pressing **RETURN** is all that is required. The copy utility will take over and read the *MP.HLP* program off the diskette. The Commodore 64 will display dots on the screen as it reads. Once this is done you will be prompted to remove the source diskette and insert the destination diskette. This time insert the formatted diskette you just made and press **RETURN**. The copy utility will display a message followed by the now familiar dots on the Commodore 64's screen.

Once the writing is done, **READY.** will be displayed. Repeat the process and copy the *MP.SYS* program. After this is done you will have prepared the stage for us to begin using Multiplan. It will be both pleasant and useful.

Summary

We have had a brief look at the Commodore 64 system hardware and its disk operating system, DOS.

Starting the Commodore 64 involves:

1. pushing the POWER switches of the computer, disk drive, and the television or display device to the ON position.
2. inserting the correct diskette, oval slot first, label side facing up.
3. responding to the Commodore 64's "READY." prompt by typing any valid DOS command and pushing the RETURN key.

Use of the Commodore 64 revolves around the creation and processing of entities known as files. We will see how to create, process, and delete them when we begin using Multiplan. When in the disk operating mode we can find out which files are present on a diskette by typing the DOS command

LOAD "\$",8 (Follow with RETURN.)

This loads the directory to memory. To view it we must type

LIST (Follow with RETURN.)

We can obtain a printout of a diskette's directory by using the command

OPEN 1,4
CMD 1
LIST (Follow with RETURN.)
CLOSE1

An installed work copy of the Multiplan disk should be prepared using the directions in the *Multiplan User's Guide and Reference Manual*. This will provide you with a disk that is self-loading.

New diskettes must be initialized before they can be used to store files. This is done using the following commands

OPEN 1,8,15
PRINT#1,"N0:DATADISK,ID" (Follow with RETURN.)
CLOSE1

3

Using Multiplan: Some Preliminaries

Before embarking on solving complete problems using Multiplan, it is better to deal with a few basic concepts and operations which will be used over and over again. It is better to deal with these sooner rather than later. Entering Multiplan and keyboard use are among them.

Entering Multiplan

There are two ways to load Multiplan. The first is by loading Multiplan itself (MP), and the second is by loading an auxiliary program called Multiplan Color (MPC). Both are on your Multiplan master diskette. The MPC program is useful only if you have a color monitor or television and if you don't like the colors shown on your display. (The Commodore 64 uses a cyan border, a blue inner field, and light blue letters or text.) We found white letters to be better since it was easier to read the text. We will first explain how to enter Multiplan directly. If you have a color display, you may wish to use MPC. We shall describe both methods.

The following steps describe the first method: loading Multiplan directly. They are almost the same ones we used in Chapter 2 to load the directory. The main difference is that we should insert the Multiplan diskette and this time load Multiplan.

The steps involved in loading Multiplan are as follows:

1. Turn on the power switches for the computer, disk drive, television (monitor), and, if you have a printer, turn its power on. If the computer was on, turn it off then on again.
2. When you see the "READY." prompt, type

```
LOAD"MP",8  
then press RETURN.
```

3. The Commodore 64 will respond with:

```
SEARCHING FOR MP  
LOADING  
READY.
```

4. At this point you should respond with RUN then press RETURN.

After a pause, a fresh Multiplan worksheet will be displayed. *Now replace the MULTIPLAN diskette with the WORKING DISKETTE you formatted and onto which you copied MP.HLP and MP.SYS. See Chapter 2 for details.*

To use MPC, the steps are similar. Repeat step 1, but in step 2 load MPC rather than MP. This will allow you to select the colors which will be used on the display screen. Once you have made this selection, MPC will load Multiplan for you. Specifically, the steps involved in loading Multiplan using MPC are

1. Turn on the power switches for the computer, disk drive, television (or monitor), and, if you have a printer, turn its power on. If the computer was on, turn it off then on again.
2. When you see the "READY." prompt, type

LOAD" MPC",8

then press RETURN.

3. The Commodore 64 will respond with:

```
SEARCHING FOR MPC
LOADING
READY.
```

4. At this point you should respond with RUN then press RETURN.
5. The screen will be filled with lines reading:

```
MULTIPLAN FROM: HESWARE
MULTIPLAN FROM: HESWARE
MULTIPLAN FROM: HESWARE
```

This will fill the screen, go through a color change demonstration and display the next screen:

MULTIPLAN SCREEN COLOR
SELECTION MENU

1 BLACK	9 ORANGE
2 WHITE	10 BROWN
3 RED	11 LIGHT RED
4 CYAN	12 GRAY 1
5 PURPLE	13 GRAY 2
6 GREEN	14 LIGHT GREEN
7 BLUE	15 LIGHT BLUE
8 YELLOW	16 GRAY 3

PRESS SPACE TO CHANGE BACKGROUND COLOR
PRESS RETURN TO LOCK IN YOUR CHOICE

Follow the instructions by going through the different colors, and lock in the one you feel best suits your monitor or television. (We found either black or blue to be the best with white text, which is selected in step 6.) Press RETURN after having selected the right color.

6. Now the screen will remain the same, but the last two lines will be replaced by:

PRESS SPACE TO CHANGE BORDER COLOR
PRESS RETURN TO LOCK IN YOUR CHOICE

This will then change to prompt you about the text color:

ENTER THE NUMBER OF THE COLOR TEXT YOU
WANT AND PRESS RETURN
TEXT COLOR?

The program allows you to change your mind even after you have made a selection. Having made a selection and after pressing RETURN you will see the next prompt:

IF YOU WISH TO CHANGE ANY OF THE COLORS
PRESS SPACE
IF YOU ARE SATISFIED WITH YOUR SELECTION PRESS RETURN

Assuming you like your selection, press RETURN. After a pause, a fresh Multiplan worksheet will be displayed. *Now replace the MULTIPLAN diskette with the WORKING DISKETTE you formatted and to which you copied MP.HLP and MP.SYS. See Chapter 2 for details.*

The blank Multiplan worksheet displayed looks like the following:

```
      #1          1          2          3          4
      1 <cell pointer cursor>
      2
      3
      4
      5
      6
      . . .
      . . .
      . . .
      19
```

You will see numbers from 1 to 19 displayed on your screen. Multiplan uses a very large cursor, the big white rectangle you see in the top left corner (under the column headed 1, in row 1), just after you load Multiplan. The cell pointer cursor indicates where on the worksheet a number you type will be placed.

You can move the cell pointer cursor by using the arrow keys which are located on the bottom row, right-hand side. They are marked CRSR (short for cursor) and each has two arrows. There is a key which controls horizontal movement, and a key controlling vertical movement. The very last key lets you move right, or left (with the help of the SHIFT key):

- Move the cell pointer cursor to the right one column.
- ← Move the cell pointer cursor to the left one column.

The key to the left of the horizontal cursor key allows you to move up (with the SHIFT key) and down:

- ↑ Move the cell pointer cursor up one row.
- ↓ Move the cell pointer cursor down one row.

The CLR/HOME key, located next to the INST/DEL key, can move the cell pointer cursor to position R1C1 (upper left-hand corner) from anywhere in the spreadsheet. The Home key can be thought of as a cell pointer cursor arrow key. Pushing it moves the cell pointer cursor immediately back to the top left of the worksheet. With a fresh worksheet this left-hand side cell is at position R1C1.

Multiplan also provides two alternate methods for moving the cell pointer cursor. The first uses the control key, the second uses the Goto command. The first is done by pressing the control key (the large key on the left-hand side labeled CTRL, next to the Q key) at the same time as one of four specific keyboard keys arranged in a diamond shape on the keyboard.

Control with e or E: Move cell pointer cursor up.
Control with d or D: Move cell pointer cursor to the right.
Control with s or S: Move cell pointer cursor to the left.
Control with x or X: Move cell pointer cursor down.

These cell pointer cursor control keys are sometimes called a “diamond” and can be used to move the cell pointer cursor. Additionally, many of the better selling word processing programs make use of a similar diamond to move their cursor, which makes it easier to become comfortable with both Multiplan and these word processors. Finally, some people are more comfortable with using the alphabetic keys to move the cell pointer cursor. A little experimentation will allow you to discover which you prefer. However, for convenience we will refer only to the arrow keys in this book.

If you hold any cursor key down you will keep moving the cell pointer cursor repeatedly. If you should happen to move the cell pointer cursor right, beyond column 4, you will see the missing part of the worksheet come into view. As column 5 appears on the right, column 1 will disappear from the left. You could go as far right as column 63, the last column.

You can return to the original screen appearance by repeated use of the arrow cursor motion key ←. However, you may find it more convenient to use what is known as the Goto command.

The letter G (the first letter of the command “Goto”) is a Multiplan command to be followed by the coordinates of the worksheet cell on which you wish to place the cell pointer cursor. The coordinates are specified by designating the row number, immediately followed by the column number. Thus, when you push “G”, the command lines (the first two lines below the 20th row of the worksheet) change to show you the subcommands of the Goto command. It now reads:

GOTO: Name Row-col Window

You can also activate this command by moving the command cursor to the Goto command (so that it is positioned over the command, on the first line) and then pressing RETURN. The same prompt will appear. Multiplan provides two ways to get the command cursor to the Goto command. You can press the space bar and watch the command cursor move. You can also use F1, preprogrammed by Multiplan to move the cursor. Either of the two command cursor moving methods will move it over to the Goto command and generate the above prompt, as soon as you press RETURN.

Keying in R (the first letter of the subcommand Row-col) or moving the cursor one place to the right and following with RETURN generates the next prompt, which now displays the options of the subcommand:

```
GOTO  row:           column:
Enter a number
```

These row and column options (also called option fields in Multiplan) will show the current cell pointer cursor location following the colon. Keying in a row number, followed by RETURN, indicates to Multiplan which row you wish to go to. The column address must also be indicated, and to do that you must tab to the column field.

To do that you must press the F1 key which is used by Multiplan as a [TAB] key. The F1 key is located on the extreme top right, just to the right of the standard keyboard. Let's try that and key in 1 for the column number. Follow that with a press of RETURN and watch how the cell pointer cursor immediately returns to the top left corner of the worksheet (also called window) and causes the top-left part of the worksheet to be displayed. If you were near the bottom row (say 250), you could press either of two other function keys, also located to the right of the keyboard. Specifically, you could use F5 (page up) or F7 (page left) to return to R1C1.

In this case, you could have used the Home key instead. Home always returns the cell pointer cursor to the top, left-most corner of the screen. Since this is the first screen, also called page, R1C1 is its top left cell. The Goto command lets you specify where you want to place the cell pointer cursor. Your screen's view of the worksheet will be adjusted so the cell pointer cursor always remains visible.

If you look down the worksheet, by pushing the cursor-down arrow key ↓ often enough, you will get to see the bottom row, which is row 255. You could also get there directly by typing

G	(Starts GOTO command.)
R	(Selects the row subcommand.)
255	(Indicates row to jump to, given current column.)
RETURN	(Uses current column address, activates command.)

Have we skipped the column designation? Yes, indeed. Multiplan is trying to anticipate your needs and provide default answers for the various options. Whenever we like the options presented by Multiplan, all we need to do is respond with RETURN, as if we keyed in the default option. As we go along, many such occasions will arise.

Moving the cell pointer cursor so that some information disappears from view as other information comes into view is called *scrolling*. Moving the cell pointer cursor down far enough causes a vertical scroll, with the screen acting as a window into the 255-row by 63-column worksheet. You can scroll up or down, left or right, as desired, using the cursor arrow keys. Scrolling does not ever cause you to lose any information. Information may disappear from view, but it will be recalled if you scroll back to it. You can return to R1C1 by using either the Goto command, repeated presses of cursor arrow keys, one press of the CLR/HOME key, or the combination CTRL and S keys.

Keyboard Use

Many keys take on a special meaning in Multiplan, as you probably noted above. It is a good idea to familiarize yourself with the Commodore 64 keyboard. You may avoid problems in the future by spending a few minutes to learn which keys, or key combinations, have a special use in Multiplan.

The “>” key should be clearly distinguished from the cursor arrow key →. Since the “>” shares the keycap with the period “.”, you must also push the SHIFT key to evoke the “>”. Many keys have arrows of one kind or another etched on them. The CRSR keys, the keys to the right of the right-hand side SHIFT key, have two arrows each on them. As noted earlier, we use the SHIFT key to activate the top arrow direction. There are two big SHIFT keys, one on each side of the board. We need these to get at symbols for typing “(” or “)”, when we start using arithmetic expressions. We will frequently use the INST/DEL key, which we met earlier in this chapter (at times referred to as *Backspace*) and, of course, the large key on the right-hand side of the second row is our frequently used RETURN key.

The function keys (F1 through F8) on the right side of the keyboard allow you to move quickly (and in that capacity they are called action keys). Some function keys allow you to perform quick editing operations (and are called edit keys). To avoid confusion, we will list the meanings of the action keys here and postpone explanation of several edit keys until Chapter 5. Don't worry, we will use and mention all the keys so that you will become comfortable with using them.

The function key F1 lets you tab to the next command, subcommand, subcommand's field, etc. (*Relax*, we will be explaining each of these terms later in this chapter.) You can use the CTRL and I, or CTRL and A key combinations to do the same thing. That means that you have a choice of three key presses to tab to where you want to be. Throughout this book we shall use the [TAB] designation to indicate a tab operation.

F2 is an editing key. To use F2, which shares the F1 key, you must press the combination SHIFT and F1. This lets you move to the word on the left, when in edit mode. (We will see what that means later.) F8 (which also requires SHIFT, and shares the F7 key) has the opposite editing function: word to the right.

F3 is the Multiplan “delete” key. It lets you delete whatever is covered by the command or edit cursor at the moment. You should use it with care, if you wish to avoid excessive retyping.

F4 is similar to F2, in that it is an editing key, as well as one requiring a key pressing combination (with SHIFT). When SHIFT and F3 (with which F4 shares a key) are pressed, F4 is activated, moving the edit cursor one character to the left. F6 (which also requires SHIFT, and shares the F5 key) has the opposite editing function: character to the right.



Figure 3.1 Commodore 64 Keyboard (Courtesy Commodore Electronics Limited)

Function keys F5 and F7 allow you to move faster. F5 moves one page up, whereas F7 moves you one page left. F5 shares the key with F6 while F7 shares its key with F8. Both functions, (i.e., page up and page left) can be performed by alternate key combinations. To move up one page, press the combination CTRL and R, release them, then follow with a press of the up arrow key. To move one page to the left (for which F7 is one alternative) you press the CTRL and R combination, release them and follow with the left arrow key press.

That may sound confusing at first, but it will prove rather useful; there are no function keys to move you to the page down or to the page right positions. This is where the CTRL-R combination with the appropriate arrow key comes in handy. Page right with CTRL-R followed by the right arrow, and page left with CTRL-R followed by the left arrow.

There are several more key combinations worth noting. CTRL and F lets you move to the next unlocked cell (we will see what that means later). This can be also accomplished by pressing the SHIFT and INST/DEL key. This was referred to earlier as the backspace key. The exclamation mark (!) forces Multiplan to recalculate values, which will be extremely valuable with complicated worksheets. This key is accessed with SHIFT and the ! (which shares the number 1 key) combination.

The CTRL and C combination serves as an “escape” key, and will frequently get you out of trouble. The key on the left of the third row, marked *RUN STOP* serves the same purpose. This is the key to push whenever in trouble: it stops every Multiplan command and cancels it. Both alternatives for cancelling commands are identical, and we shall refer to them as Cancel in this book.

The last key which you will, no doubt, find most helpful is the question mark (?). Pressing this after any command generates a command SPECIFIC HELP SCREEN. Depending on the command highlighted on the screen just before you ask for help, the help screen can be used to assist and explain everything on the screen at the moment. Note that there is also a GENERAL HELP SCREEN, which can be accessed by using the HELP command. This too will be addressed later.

Figure 3.1 identifies the keys we have been discussing. Notice the diamond arrangement of the keys E,S,D, and X. To recap, four function groups can be performed.

1. Action: Up, Down, Left and Right (arrow keys); Next window (CTRL-W); Next unlocked cell (CTRL-F or INST)
2. Edit: Delete (F3 or CTRL-Y); Character left and right (F4 and F6, respectively); Word left and right, (F2 and F8); Backspace (DEL)
3. Execute: Do command (RETURN); Next command (Space bar or F1); Tab to next field (F1 or CTRL-A or CTRL-I) Help (?); Recalculate (!); Cancel/Escape (CTRL-C or RUN STOP); Previous command (DEL)
4. Window: Up, Down, Left and Right (CTRL-R, release and follow with appropriate arrow key); Home (HOME)

The Command Lines

Examine the following screen display, which should look similar to yours:

```

#1          1          2          3          4
 1
 2
  . . .
  . . .
  . . .
18
19
COMMAND: Alpha Blank Copy Del Edit Form
          Go Help Ins Lock Move Name Opt Print
          Quit Sort Transfer Value Window Xtern
Select option or type command letter
R1C1                                     100% TEMP

```

The first three lines, immediately following the worksheet itself, are the command lines. The three lines are preceded by the word **COMMAND** and a total of 20 commands. You can type the first letter of the desired command to access that command. An alternative method is to move the command cursor over the command and, once it is there, follow with **RETURN**. The command cursor can be moved by either the space bar or the F1 function key. These move the command cursor to the command on the right, across the first line, and then across the second line. The **DEL** key moves the command cursor to the left. The alternate methods of moving the cursor are identical in their result so you can choose whichever one you prefer.

Once the **RETURN** key has been pushed, or the first letter of the command's name pressed, the command line changes and displays the command you requested and the subcommand alternatives. For example, try to enter the Delete command; you may use the F1 key, or the space bar followed by **RETURN**, or simply push the letter **D**. Now watch the command line. It changed to:

```
DELETE: Row Column
```

As you might suspect by now, Multiplan always positions the cursor over the first option, in fact, if row deletion was what you needed, go ahead and press **RETURN**, or "**R**". You are next prompted by Multiplan to select from a set of options. If you selected **Row**, followed by **RETURN**, the command line now changes to the next prompt, which displays the following options:

```

DELETE ROW # of rows:  1
                starting at:  1
                between columns:  1          and:  63

```

Again, you should select the relevant row numbers and follow with **RETURN**. Note again that the cursor is over the number 1 in the first field. To use this default, if you like the rest of the options, press **RETURN** and the command will be executed. If, however, you want to delete a single row, be-

ginning at row 9, you would have to use a [TAB] key (press either F1 or CTRL-I combination or CTRL-A combination). Pressing [TAB] will move you to the next field, e.g., from "# of rows:" to "starting at:". Now you could enter the number 9 and follow with RETURN if you like the rest of the options on the next two fields.

If you want to delete only a section of a row, you would have to enter [TAB] to get to the next field ("between columns:") and follow with the column range you want to delete. The same rationale applies to all commands. Each can be activated by pressing the first letter. Each has several options displayed in fields. To get from one field to the next you need a [TAB] key. Some of the commands have subcommands activated like the original commands (e.g., by keying in the first letter or moving the cursor over the subcommand and pressing RETURN). Don't be overwhelmed, we shall continue and explain each step as we come to it.

Let us go back to the DELETE command for a moment. What if you wanted to delete a column? Let's try again. First press one Cancel key. (Remember, that means either a CTRL-C combination or a press of the RUN STOP key.) Watch how the command line reverts to its original position, displaying the 20 commands. Now move the cursor to the right using the F1 key. Watch the cursor land over the Delete command and press RETURN. You now see the next prompt asking whether you wish to delete a row or a column. Next use F1 to move the cursor one place to the right from Row to Column. Follow with RETURN. The prompt now reads:

```
DELETE COLUMN # of columns: 1
      starting at: 1
      Between rows: 1           and: 255
```

Enter a few numbers to delete one or more columns or press Cancel to revert to the original menu. The same procedure applies to all the commands entered on the command line.

The Message Line

The next line just below the command lines is the message line. When no commands are used it displays:

```
Select option or type command letter
```

As soon as a command letter is pressed, it changes to guide you further. For example, after executing the Delete command, the command line reverts to displaying the subcommands while the message line retains the same message (e.g., select option or type letter). After you select row or column deletion, it follows with:

```
Enter a number
```

Additional information is displayed at other times.

The Status Line

Look at the bottom line of the worksheet. This is the status line. The status line, the last row, shows the location of the current active cell (the one highlighted by the cell pointer cursor). Now press the ↓ arrow and follow with several additional presses. With each push, the cell pointing cursor moves down one row. If you wish to move in another direction simply press the appropriate arrow key. You can find your location directly from the status line rather than trying to estimate it from the worksheet. Try pushing arrow keys and watch the current active cell change as the cell pointer cursor moves. Now change direction and move the cell pointer cursor again. Watch the new address of the cell pointer cursor being displayed on the status line. Multiplan calls this address "reference."

If the current active cell is empty, nothing else is displayed. If the cell is not empty, the status line will display its current contents, e.g., whatever was keyed in at an earlier time, either values, formulas, or labels (Alpha).

Values or formulas = (Contents of current active cell for values and formulas.)

Labels (Alpha) = (Contents of current active cell for labels.)

If you have any text or labels material in the cell, it will be displayed between quotation marks. If you have numbers they will be displayed as they appear in the cell. As with the cell's address, when you move to a different cell, the cell contents area will change to reflect the new contents.

If you have an empty worksheet, the contents area is blank and the next piece of information indicates the amount of memory (in percentages) still available for your worksheet. A new worksheet usually shows "100% Free", which is the space available to Multiplan when you start with a fresh worksheet.

This percentage does not tell you how much space you have remaining in absolute terms. Multiplan, however, has provided you with an easy way to find that out as well. Select the Options command (type O, or use F1 to tab to Options) and follow with RETURN. The amount of memory available will be displayed on the message line, as soon as you press the RETURN key once more. The amount of workspace available to you will be about 21KB. In addition to the amount of memory, Multiplan also displays its version number:

Multiplan Ver 1.06 Bytes: 21044

Important *From now on, we will usually display only the top nine rows of a worksheet, since most of our examples will fit in that sized worksheet. We will learn how to handle very large worksheets later. On the other hand, we will display eight columns of information, per spreadsheet, so that you can better understand the more complex relationships. Since your screen only displays four columns, you will have to press CTRL-R, release it, then press the right arrow to see the four columns on the right (5 through 8).*

Getting Help on Line

As you may have noticed, and as we mentioned earlier, Multiplan has a Help command on the command line. Try using it. There are two ways to access Multiplan's on-line Help facility:

1. For information on a specific command on the on-line Help text, move to the command menu (press Cancel), then move to the desired command (with keys F1 and DEL or the space bar) and press the question mark. You may press the question mark (SHIFT and ?) key at any time. The Help text that appears applies to the command active at the time you asked for help. When you resume your work (by pressing "R" for RESUME), it will be exactly as you left it.
2. To access all the Help information from the command menu, press the "H", for Help, or position the command cursor over the Help command from the command menu. The entire on-line Help facility is now available, and you can read it page by page. Press N for "Next" or P for "Previous" to move through the Help screens.

Multiplan's Help addresses several areas: Applications, Commands, Editing, Formulas, and Keyboard. If you asked for the Applications Help, your screen will show:

Applications

**The following table provides
an index to the commands and help
topics applying to common questions.**

If you need to: You should consult:

Enter data:	Value, Lock, Help Keyboard
Enter text:	Alpha

These are just the top few lines. Try to access Help and familiarize yourself with it. It will be handy at times, especially when you cannot recall an operation, option, or command.

Writing on the Worksheet

With the cell pointer cursor situated at position R1C1, try typing in a number such as 345 or -765. While you are typing it, no change takes place in cell R1C1, but you see what you are typing displayed below the worksheet, near the bottom right of the screen. This is on the entry line.

If you made an error in typing, and notice it right away, you can correct it by erasing characters. Use the Backspace key (the one labeled INST/DEL) to erase characters that have just been typed. *Do not* try to erase characters by using the cursor arrows: this will simply move you to other cells!

The information displayed on the entry line will be copied into the location the cell pointer cursor is sitting on when you push the RETURN key. To move the cell pointer cursor one cell in the direction of your choice, you must instruct the computer to do so by pushing the cursor arrow of the direction of your choice—either down the column or to the right. You can achieve the same effect and save yourself a keystroke by pressing the cursor arrow key directly. This will both put your entry into the desired cell, as well as move the cell pointer cursor to the next cell location.

Once a workspace entry has been set, you can change it in two ways. For the time being, you can do it by typing a new value and pushing RETURN, provided the cell pointer cursor was positioned at the desired entry. The edit command, which we will see in Chapter 5, provides a nicer way to correct errors or make changes.

Try typing some numbers into location R1C1.

1. Be sure the cell pointer cursor is in position R1C1 either by pushing the cursor arrow keys, or by typing

G R 1 [TAB] 1 RETURN

2. Type the desired value (perhaps correcting with the Backspace key while typing).
3. Push RETURN and see the new value at position R1C1.
4. Watch the cell pointer cursor move to a new location after you push the correct cursor arrow key and observe the change on your status line. Note that Multiplan will not let you scroll "off" the screen (up or left) when in cell R1C1.

Leaving Multiplan

If you want to stop using Multiplan, you can just turn the computer off. If you had anything of value on the screen, you could print it by using the Print command. (More on the Print command in Chapter 4.) If you want to save what is on the screen for later use, then keep reading until we discuss saving worksheets.

Returning to the Commodore 64's DOS

You can leave Multiplan and return to Commodore 64's DOS by using the Quit command. As you probably recall by now, you can get there by pressing the space bar or F1 or by typing Q (for Quit). Whenever you are using Multiplan, the current status of your dialogue is reflected in the command line. It will now read:

QUIT:

and the message line would read:

Enter Y to confirm

If you type Y (for Yes), you will see your worksheet disappear, and in a second or two, you will see the DOS prompt line:

```
**** COMMODORE 64 BASIC V2 ****
64K RAM SYSTEM 38911 BASIC BYTES FREE
READY.
```

Going from Commodore 64's DOS to Multiplan

If you are using DOS and you decide to switch to Multiplan, it is not necessary to turn the computer off. Respond to a "READY." DOS prompt by inserting the Multiplan diskette into drive 8, and typing the load command

```
LOAD"MP",8 (Follow with RETURN.)
```

or

```
LOAD"MPC",8 (Follow with RETURN.)
```

After a few seconds, you can type in RUN, and follow the procedure described at the beginning of this chapter. In a few additional seconds you will see a fresh Multiplan worksheet on your screen.

Other Commands

The meanings of all of the letters and symbols displayed on your command lines are shown below. Each one of these commands will be discussed one at a time as we proceed.

A	Alpha	Used to enter text or labels.
B	Blank	Erase value/label.
C	Copy	Copy one or more cells to another location.
D	Delete	Delete a partial row or column.
E	Edit	Modify an entry without retyping it.
F	Format	Formats an entry.
G	Goto	Allows jumping to any cell on the worksheet.
H	Help	Provides on-screen help to the user.
I	Insert	Inserts a row or column.
L	Lock	Protects cells from unintentional change.
M	Move	Moves one or more rows or columns.
N	Name	Allows naming a cell or a group of cells.
O	Options	Turns certain Multiplan features on or off.
P	Print	Prints all (or segment) of worksheet.
Q	Quit	Exits Multiplan, discards all data not saved.
S	Sort	Sorts worksheet rows in desired sequence.
T	Transfer	Moves files from disk to memory and vice versa, as well as renaming and deleting files.
V	Value	Used to enter a number or a formula.
W	Window	Splits screen operations, displays several worksheets.
X	Xternal	Defines relationships between current worksheet and other worksheet files.

Saving a Worksheet

Suppose we have just returned to DOS and we immediately reload Multiplan. When the worksheet is displayed, our numbers have disappeared. That was why we were asked earlier to confirm if we wanted to leave Multiplan. Leaving Multiplan means losing what we have typed onto the worksheet, unless we take care to save a copy. You can save a copy of a worksheet on your Multiplan disk by leaving it in drive 8, **WHICH WE DO NOT RECOMMEND**, or, if you wish to save it on some other diskette, by placing a formatted diskette with the MP.HLP and MP.SYS files on it into drive 8 or 9 and issuing the Transfer command we just saw. When it gives you the subcommands' menu:

TRANSFER Load Save Clear Delete Options Rename

you can ask to save a copy of your worksheet on the diskette by keying in the letter s or S. You can move the cursor to the right one space (it is over the first option, Load, when this prompt appears), using either the cursor arrow or F1 key. The next prompt will display:

TRANSFER SAVE filename: TEMP

Multiplan will display the current worksheet's name, and await your response. If you wish to retain the name displayed, simply press RETURN. If you wish to rename the file, respond by typing the file name with up to eight characters (Multiplan will accept file names in either capital or lowercase letters). This will direct the file to disk drive 8.

If you inadvertently provide a name already in use when attempting to save a worksheet you will see the prompt:

Overwrite existing file?

If you respond with Y or y, the old file will be replaced by your worksheet. If you enter N or n, you will be allowed to change your file name to something new.

Retrieving a Worksheet

Suppose we have just loaded Multiplan. Once again we get a blank worksheet. We can retrieve the worksheet we called TEMP by issuing the Transfer command we just saw. When it gives you the prompt:

TRANSFER Load Save Clear Delete Options Rename

the cursor will cover the first option, Load. Press RETURN; the next prompt will display:

TRANSFER LOAD filename:

If you respond with the name of your file TEMP, you will see the worksheet appear before you in an instant. If you forgot your file name, press the right arrow key to view the directory of files on your logged drive (normally 8).

Once the directory is displayed, you can use the cursor arrow keys to move the cursor covering the top left-hand side file name. The cursor will move from one file name to another in the direction of the cursor arrow key. When the cursor covers the desired file name, press RETURN, and it will be loaded onto the worksheet.

If you misspelled TEMP, say as TAMP, Multiplan will be unable to load it and you will see the following:

Enter Y to retry access to filename

If you press *y* before correcting the misspelled name, you will see the same message again. You may prefer to view the directory to avoid misspellings. You may respond to the above prompt by pressing either the left or right arrow. This will result in the diskette's directory on the display. You will return to Multiplan by pressing any key.

Panic Button

What if we typed a command and did not get the prompt we expected? You can use one of two "Esc" keys which will stop whatever operation was in progress and provide you with a blank prompt line so you can type in a fresh command or a new entry value. Either the RUN STOP or the combination CTRL-C will serve this purpose.

Using the RETURN Key

Is there any rhyme or reason with regard to using the RETURN key? Fortunately there are some simple rules that will help reduce any confusion. Whenever Multiplan can reasonably anticipate that your entry or response is complete, you do not push the RETURN key. For instance, if you are trying to use a command, you can do so by typing a single character "A" whenever the command line is ready. Without pushing the RETURN key, you immediately see the prompt line:

Alpha:

If you had typed "AONE" without spacing between the "A" and the "ONE," the preceding prompt would be immediately overwritten by the following:

Alpha: ONE

This might happen so rapidly that you might not even have noticed the earlier prompt. If you typed "Q" you would see the prompt

Quit:

Responding with the letter Y (or *y*) will return you to DOS. You could have typed "QY" all at once and gotten the same effect. You may wish to do it in two stages; "Q" first, then "Y" next, so that you have a final chance to reconsider. Should you change your mind and wish to cancel the Quit, you can do so by pushing a Cancel key, instead of Y.

In all of the above situations, we did not use the RETURN key even once. The specific situations we were in made it possible for Multiplan to correctly anticipate what we were up to. That is not the case in the following situations. When we type

G R 1

Multiplan does not know whether we intend a "Goto Row 1" or whether we are simply pausing before finishing the entry, say as in

G R 11

Even then, Multiplan does not know whether we intend a "Goto R 11" or whether we will continue with

G R 111

At this point, Multiplan should know that we cannot go any further (there are only 255 possible row numbers). However, to be consistent, each use of the "Goto" requires that you push RETURN or some other special key.

The rule can be made even more widely applicable: Whenever you are providing a coordinate by typing in the column and row coordinates, you must signal that it is complete by using some special key. So far we have been using RETURN for this purpose. Later we will see what other special keys could be used to our advantage.

In the following chapters we will be typing information that varies in length. These items may be numbers, such as 12345, or names or labels such as September or Date. It follows that since Multiplan can't reasonably be expected to know what we have in mind, the completion of each entry of a variable-length item requires a signal from us. That signal will often be provided by using the RETURN key, as we saw in the Goto situation.

Summary

- Enter Multiplan by turning the system on, and inserting the Multiplan diskette into the drive. Type LOAD"MP",8 and follow with RETURN. When the computer has loaded the program, type RUN and press RETURN.
- Initiate a command by using the F1 and DEL keys followed by RETURN, *or* by pressing the key for the first letter of the command.
- Leave Multiplan by typing Q Y.
- Save a worksheet (*before* leaving Multiplan) by inserting a formatted working diskette (*see Chapter 2 for instructions*) in the drive. Then type T S and provide a new file name directed to either drive ,8 (,8:filename) or to drive ,9 (,9:filename).
- Retrieve a saved worksheet by typing T L and providing an existing file name, or viewing the directory on either drive.
- Position the cell pointer cursor, either by using the cursor arrow keys, the keyboard "diamond" (control e,s,d,x), or by typing the G for "Goto" followed by the row and column coordinates of the desired entry.
- Correct an entry using the Backspace key, provided the entry is still on the entry line.
- Correct a worksheet entry by positioning the cell pointer cursor over it and retyping the entire correct entry, followed by RETURN.
- Use the RUN STOP or CTRL-C combination as a Cancel key if you change your mind and wish to discontinue a particular operation.

4

Solving a Simple Problem

In this chapter we will begin working with labels, simple formulas, printing, erasing (clearing) the screen, and a very powerful technique called Copy. To make things easier for us, we will reexamine a familiar problem, the author-income problem from Chapter 1. Then, just to be certain that these new ideas are well understood, we will apply them to estimating an IRA's value.

Text and Numbers

As you begin typing an entry, Multiplan attempts to classify the entry into one of three categories:

1. A number or formula.
2. A command (begins with A,B, . . . W,X, on the command line)
3. A label or text (must begin with the Alpha command).

A value is a number or an arithmetic expression. It begins with a numeric digit or any one of the following characters: +, =, -, (, or a period. A label or text, also called a title, must begin with the Alpha command. Multiplan will record the text the way you type it. Some typical labels as you would enter them are Rate, Total, Cost.

Using the Cursor Arrow Keys

We will continue to use the following symbols to represent the cursor arrow keys:

→	Move right.
←	Move left.
↑	Move up.
↓	Move down.

Try typing the labels RATE, COPIES, and INCOME (you may use either capital or lowercase letters) at positions R1C1, R1C2, and R1C3. Using the technique we saw in Chapter 3, you would type

G	(Starts the Goto command.)
R	(Selects the row subcommand.)
1 [TAB]	(Omit 1 if the cursor was already at R1C1.)
1 RETURN	(Omit 1 if the cursor was already at R1C1, or if you use the arrow keys.)

RETURN (Signals to Multiplan that you are beginning to enter labels. You could have pressed A to get the same result.)

RATE RETURN

→ (Moves cursor to R1C2.)

COPIES RETURN

→

INCOME RETURN

Notice how the cursor jumps to the next cell in the direction of the cursor arrow after you press the cursor arrow. If you forget the Alpha command before labels, you will get an error message on the lower left-hand side of the screen. Reenter the label following a RETURN, for Alpha first, or you will end up in a command you do not want. Let's clear the screen for the next examples.

We know that returning to DOS and reentering Multiplan will clear our screen. There is a much faster way; type

T (For Transfer.)

and see the command line change to display six subcommands:

**TRANSFER: Load Save
Clear Delete Options Rename**

Move the cursor two spaces to the right, so it covers the Clear option, then press RETURN. See the new prompt:

**TRANSFER CLEAR:
Enter Y to confirm**

If you type Y, the screen will be blanked out or cleared and a fresh worksheet will be displayed. All the previous information you had keyed in will be lost. Clearing the screen has no effect on worksheets that have been saved on diskettes. If you change your mind before typing the Y, you can get back to the worksheet by typing N (or n) or pressing any key except y or Y.

We had the following on the screen:

#	1	2	3	4	5	6	7
1	RATE	COPIES	INCOME				
2							
3							
4							
5							
6							
7							
8							
9							

Pressing T C Y clears the whole worksheet; in particular, it leaves us with the labels of row 1 erased. The cursor will always be left in position R1C1 after you clear the screen with the TCY command. Now we can retype our labels a little faster by using the cursor arrow keys, omitting the RETURN, so that the cursor will jump in this direction following entry of the labels:

RETURN	(Cursor was over Alpha, starts text.)
RATE →	(Enters RATE into R1C1, moves cursor to R1C2.)
COPIES →	(Enters COPIES to R1C2, moves cursor.)
INCOME →	(Enters INCOME to R1C3, moves cursor.)

This leaves the cursor at R1C4, since RATE went into R1C1, COPIES went into R1C2, and INCOME went into R1C3. You don't have to go from left to right; the following would work just as well, but the cursor must be directed toward the left.

G	(Starts Goto command.)
R	(Selects row subcommand.)
1	(Enters row 1 as target.)
[TAB]	(Moves to next field, column option.)
3	(Selects column 3.)
RETURN	(Activates command.)
RETURN	(Starts Alpha.)
INCOME ←	(Enters INCOME, moves cursor to R1C2.)
COPIES	
←	
RATE	
RETURN	

This places INCOME in R1C3, COPIES in R1C2, and RATE in R1C1, as desired. Notice that when RATE is copied into position R1C1, and you are trying to force the cursor left of R1C1, by using the ←-key, nothing bad happens. Position R1C1 is filled in with the label RATE, and the cursor stays at position R1C1. You will find that Multiplan is very forgiving in those situations. Moreover, to alert you, an audible ring will sound. This audible ring will sound whenever you try to force Multiplan to do something it cannot do. Don't be intimidated; there is a way to turn this alarm off. More on that in later chapters.

Let us now proceed to type in the data from Table 1.1. We want the rates 5, 7, 8, and 10 in column 1. Position the cursor at R2C1, either by using the G command or by pushing the cursor arrow keys. Once there, type

V	(Starts the Value command.)
5 ↓	(Enters number, moves cell.)
7 ↓	
8 ↓	
10 RETURN	

That takes care of column 1. You are now at location R6C1. To get to location R2C2, key in (we'll skip the explanation here)

G	
R	
2	
[TAB]	(If you made a mistake entering the row number, tab again to return to the row field.)
2	
RETURN	

Then type

V	(Start values.)
3,000 RETURN	

The prompt and message line will show you that such a number is unacceptable by displaying "Error in formula" on the lower left-hand side of the screen. You have to leave the comma out. So type

3000 ↓

You are ready to type the 4000 in row 3. Row 4 can have 5000 keyed in. Similarly, you can type 38000 in row 5. Follow the 38000 with a RETURN to return to the command line. Our table now looks like:

#	1	2	3	4	5	6	7
1	DATE	COPIES	INCOME				
2		5	3000				
3		7	4000				
4		8	5000				
5		10	38000				
6							
7							
8							
9							

<C4A>

Note Whenever you see a notation like <C4A> at the bottom right of a worksheet, it identifies the worksheet as one which is available on the optional diskette. This particular worksheet has the name <C4A>, as shown within the <. . .> braces. So by typing T, L, C4A RETURN, you could load this worksheet from the optional diskette if you prefer not to construct it following the detailed steps we have just been through, or if you want to examine the contents of any other worksheet in the book.

We have our titles in place and the raw data has been entered. Now we want the computer to do some work for us.

Having Multiplan Compute

We would like to get the rate at R2C1 multiplied by the number of copies at R2C2, and have the result stored at R2C3. We can do this by first placing the cursor where we want the result to be placed; here we can type

G R 2 [TAB] 3 RETURN

or use the cursor arrow to move the cursor to R2C3. Then we type in an arithmetic expression depicting the formula we just described. The rate at R2C1 is represented by the coordinates R2C1. You use an asterisk * to indicate multiplication. So we can type

V
R2C1*R2C2 RETURN

As soon as we do that, the blank entry at R2C3 is filled in with the number 15000, which is the product of 5 (from R2C1) multiplied by 3000 (from R2C2). This is not quite what we wanted. You can't expect Multiplan to read your mind; it does not know that you are using percentages for your rates. So we should revise our formula slightly. Using parentheses to group parts of the formula, we can type

V
(R2C1*R2C2) .01 RETURN

and you will be surprised to see that it is rejected. All multiplications have to be spelled out using an *. So we try again, typing

(R2C1*R2C2)*.01 RETURN

and we immediately see the result at R2C3: 150. We could have typed (R2C1*R2C2)/100 (the / means division). You can also use + and - for addition and subtraction. The symbol ^ is used for exponentiation, as in 2^3, for 2 raised to the power 3.

Our table now shows:

#1	1	2	3	4	5	6	7
1	RATE	COPIES	INCOME				
2		5	3000	150			
3		7	4000				
4		8	5000				
5		10	38000				
6							
7							
8							

We can now finish specifying the relationships for the items of row 3. If your cursor is not already at R3C3, send it there by typing

```
G R 3 [TAB] 3 RETURN V
(R3C1*R3C2)*.01 RETURN
```

Similarly, we can specify the formula for R4C3

```
G R 4 [TAB] 3 RETURN V
(R4C1*R4C2)*.01 RETURN
```

and R5C3 is computed using $(R5C1 * R5C2) * .01$. Our table now looks like:

#1	1	2	3	4	5	6	7
1	RATE	COPIES	INCOME				
2		5	3000	150			
3		7	4000	280			
4		8	5000	400			
5		10	38000	3800			
6							
7							
8							
9							

<C4B>

This is such an achievement that we want to record it for posterity. How do we print this table?

Printing Worksheets

The Print command is used to print worksheets. Before you type it though, verify that your printer is on, and, if it has an on-line/off-line switch, that it is in the on-line position. Type

```
P (Initiate Print.)
```

See the command line change to display the subcommands:

```
PRINT Printer File Margins Options
```

Select Printer by pressing RETURN. The use of the other subcommands and options will be discussed later. If nothing is printed, the printer is not set up correctly. If the printer is not available, is not set up properly, or is turned off, you will see a message:

Printer error

If we typed Margins in response to Print, rather than Printer, we would see the following:

```
PRINT MARGINS: left 5          top: 6
                print width: 70  print length: 54
                page length: 66
```

These are the default settings. Unless you change them, the output will be printed on the paper according to the above specifications, which assume 8.5- by 11-inch paper.

Suppose you wish to print only a portion of the worksheet. In this case you would select the Options subcommand, following the Print command. You will see the prompt:

```
PRINT OPTIONS: area:
                setup:
                formulas: Yes (No)   row-col: Yes (No)
Enter reference to cell(s)
```

Let us choose a group of cells, say

```
R1C1:R7C5          (R1C1 defines the top-left corner and R7C5 defines the bottom-
                    right corner.)
```

This gets typed immediately following the "area:" option. Leaving the other options for later discussion, we should get a printout like this:

RATE	COPIES	INCOME
5	3000	150
7	4000	280
8	5000	400
10	38000	3800

We can now reap the fruit of our efforts by seeing new results computed immediately, merely by typing in either a new rate value, or a new copies value, or both. Since we are also interested in the total income, we should add a new column entitled TOTAL. Type

```
G
R
1 [TAB]
4 RETURN
RETURN          (Start Alpha.)
TOTAL
↓
```

Your cursor should be at location R2C4, move it there. We want location R2C4 to be computed using the total income to date; this is simply the income figure found at location R2C3. So try typing

```
V
R2C3 RETURN
```

Our worksheet now has the following appearance:

#1	1	2	3	4	5	6	7
1	RATE	COPIES	INCOME	TOTAL			
2		5	3000	150	150		
3		7	4000	280			
4		8	5000	400			
5		10	38000	3800			
6							
7							
8							
9							

The formula for the next entry in the TOTAL's column, R3C4, is the sum of the current income R3C3, plus the previous total R2C4. So we can type

```
G R 3
[TAB] 4
RETURN          (To place the cursor at R3C4.)
V              (To start the Value command.)
R3C3+R2C4 ↓    (Formula for R3C4, move cursor.)
```

As we are typing each formula, we see its result as soon as we push the RETURN key (or cursor arrow key) which causes the formula to be both recorded and evaluated. You can examine a previous formula by moving the cursor to its location. The formula will be displayed on the status line of the worksheet, below the command and message lines.

Proceeding in this fashion our table looks as follows:

#1	1	2	3	4	5	6	7
1	RATE	COPIES	INCOME	TOTAL			
2		5	3000	150	150		
3		7	4000	280	430		
4		8	5000	400	830		
5		10	38000	3800	4630		
6							
7							
8							
9							

<C4C>

We can now proceed to change any of the rate or copies entries, and the consequences will be immediately computed and displayed. You should not, of course, try to change the entries in the income or totals columns, unless you want to change the formulas.

There Must Be a Better Way: The Copy Command

You don't mind typing in some formulas, if there are only a few involved. But even then it's very easy to make a mistake which you might not notice until it is too late. Some mistakes will be detected by Multiplan. Others will not. If we had typed in the formula R3C3+R2C4 for location R4C4, instead of R4C3+R3C4, Multiplan would have no way of knowing that we had made a mistake.

So we have two needs: reducing unnecessary typing, and reducing the chance of errors. The Copy command addresses itself to both of these needs.

When we were defining the INCOME formulas, we had in mind:

Put (R2C1*R2C2)*.01 in R2C3.
Put (R3C1*R3C2)*.01 in R3C3.
Put (R4C1*R4C2)*.01 in R4C3.
Put (R5C1*R5C2)*.01 in R5C3.

You would imagine there must be some way of saying: "My formula is (Xm*Yn)*.01, and I want it copied into locations R2C3 through R5C3, replacing the "Y" and "X" each time with the corresponding row and column numbers." This is almost, but not quite, how it is done in Multiplan.

Multiplan addresses cells in one of three ways. We have seen the first one, known as absolute reference, just a paragraph ago. However, when the need to copy arises, we need a more general reference, called relative reference. A relative reference describes cell locations (addresses) in terms of their relationship to the active cell. This relationship is defined by the arithmetic operators "+" and "-". A + means right or down. A - means left or up. Let us give an example to clarify what we mean.

In our income example we needed to put (R2C1*R2C2)*.01 into cell R2C3. Using the receiving cell as the anchor (the current active cell), Multiplan has to move two columns to the left (to R2C1), take the value of that cell, and hold it for a moment. Next Multiplan needs to move one column to the left (of the active cell) and multiply that value (from R2C2) times the value it took earlier from cell R2C1. The product still has to be multiplied by .01 and the result recorded in the active cell (the one into which the formula is inserted).

Let us see how Multiplan records these relative references. Although it seems very complicated, relax. Multiplan does most of the work. All you have to do is just point to the right cell. Let us do it step by step. First you move into the active cell. You can either use arrow keys or type

G R 2 [TAB] 3	
RETURN	(Goto the active cell.)
V	(Activate Value.)
←	(Move cell pointer one cell to the left. WATCH the relative cell reference change to RC[-1].)
←	(You arrived at your destination. NOTICE relative reference RC[-2].)
*	(Asterisk, for multiplication. WATCH the cursor move back to active cell.)
←	(Move to left, WATCH relative reference change to RC[-1].)
*	(Multiply, WATCH cursor move back.)
.01	(End of formula.)
RETURN	(Activate formula, enter into cell.)

You now see the active cell display 150, and the contents of the cell (bottom line) are shown as:

RC[-2]*RC[-1]*.01

In this case you don't need parentheses (as shown earlier) since we are only dealing with multiplication. It is, however, advisable to use parentheses for ease of reading. The formula in cell R2C3 can now be duplicated; we call on the copy command

C

You will see the three subcommands:

COPY: Right Down From
Select option or type command letter

You want to copy one thing, the formula located at R2C3 into the next few cells below it. Enter D (for down) and see the next prompt:

```
COPY DOWN  number of cells:          starting at: R2C3
Enter a number
```

Notice how Multiplan anticipates your needs ("starting at:" is already right where you wish). Since you want the formula copied into the next three cells, enter the number 3. Since the cursor is at R2C3, you need only push RETURN. If the cursor were elsewhere, you would have to press [TAB] or F1 to get to the next field (starting at:), and there could type R2C3. There is no need to do it here since Multiplan correctly anticipated our needs.

Now, just sit back and watch Multiplan copy into locations R3C3, R4C3, and R5C3. The default number of cells to copy into is the number of cells used the last time the copy command was activated. The way of specifying this request is to respond to the "number of cells:" option by pressing RETURN.

The formula will be entered in R3C3, R4C3, and R5C3. Remarkably, copying is a very powerful tool. You can use it for all kinds of things. We will be looking at other uses of it later. For the present, how could we have used it to also generate the formulas for the TOTAL income? Recall that these formulas and their intended homes were

```
Put R2C3 in R2C4.
Put R3C3+R2C4 in R3C4.
Put R4C3+R3C4 in R4C4.
Put R5C3+R4C4 in R5C4.
```

It seems we can't use the formula from R2C4 as the target for copying. However, we certainly can duplicate the formula from cell R3C4. If we want to duplicate we need relative references which would be

```
Put RC[-1]+R[-1]C into location RC.
```

RC, of course, denotes the current active cell. We now start

```
G R 3 [TAB] 4 RETURN      (Goto the active cell.)
V                          (Activate formula.)
←                          (Move one cell to the left. WATCH the relative cell
                           reference.)
+                          (WATCH cursor return to RC.)
↑                          (Move to the empty cell before RC.)
RETURN                    (Complete command, insert formula.)
```

Then we can copy it into locations R4C4 and R5C4 by making sure that we are at location R3C4 and by typing

```
C          (Invoke the Copy command.)
D          (Select the copy down option.)
2          (Replaces the default 3 from the previous copy command with 2.)
RETURN    (Selects current cell, R3C4, as source.)
```

What we have seen here involves replicating from one row to others. We can also copy from one column to others, as well as perform some rather complex copying operations; we will be doing so soon. For the time being, let us summarize briefly the relative addresses and their meanings.

```
R          Current row.
C          Current column.
RC         Current active cell.
```

R[+n]	The row n rows <i>below</i> R.
C[+n]	The column n columns to the <i>right</i> of C.
R[-n]	The row n rows <i>above</i> R.
C[-n]	The column n columns to the <i>left</i> of C.
RC[-1]	A single cell, just to the left of the current cell.
R[-1]C	A single cell, just above the current cell.

Estimating the Value of an IRA

Recent changes in the tax laws now permit any wage earner to participate in a tax-sheltered Individual Retirement Account (IRA). A wage earner can contribute a maximum of \$2,000 per year to such an account; a married wage earner whose spouse is not a wage earner can contribute up to \$2,250 per year. These accounts are attractive in that the contributions may be deducted from one's income, thereby deferring taxes on them (and their earnings) until retirement. Each bank, savings and loan, and other approved institution can provide a different kind of IRA investment, ranging from guaranteed rate certificates to equity funds. For that reason, it is generally not possible to say what the payout on an IRA will be; rates may change from time to time. But can you project what the payout might be, based on some clearly stated assumptions?

We can set up a worksheet as follows. An assumed yearly rate of return and its label "Rate," can be placed in positions R1C1 and R2C1. We can create a "Years" column in column 1. Column 2 starting in row 3 can be used to record the anticipated annual contribution. We should plan to make it early in each calendar year, to get the full benefit of interest earnings. The next column, column 3, can display the total accumulation earned to date.

Our worksheet begins as follows:

#1	1	2	3	4	5	6	7
1	Rate		12				
2	Year	Contr	Value				
3	1	2000					
4							

We can use the copy command to fill in the numbers 2, 3, . . . , for as many years as we wish. Suppose we only set it up here for 20 years.

G R 4 [TAB] 1 RETURN	
V	(Starts formula.)
↑	(Enters relative reference R[-1]C.)
+	(Formula for R4C1, cursor to RC.)
1	(Completes formula.)
RETURN	(Enters formula at R4C1.)
C	(For Copy.)
D	(Copy R4C1's formula, downward.)
18	(Copy into R5C1, R6C1, . . . , R22C1.)
RETURN	(Use current cell as source.)

Assuming that we want to deposit \$2000 in each year, we can copy that amount through the remainder of column 2 as follows

G R 3 [TAB] 2 RETURN	
C	
D	(From R3C2.)
19	(Into R4C2, R5C2, . . . , R22C2.)
RETURN	(Use current cell as source.)

All that remains is to provide a formula for the desired accumulation. At an interest rate i , a year's deposit d would be worth $d + (i*d)$ after one year. If the interest rate i is given as a whole number rather than a fraction (i.e., 12% instead of .12), then we can correct the formula to read $d*(1+i/100)$. The formula for the first year would then be $(R3C2)*(1+(R1C2/100))$. Then the formula for the second year would take year 1's results (in R3C3) and compound that with year 2's contribution (from R4C1), using the formula $(R4C2+R3C3)*(1+(R1C2/100))$. Subsequent years follow a similar pattern.

$(R3C2)*(1+(R1C2/100))$ for R3C3.
 $(R4C2+R3C3)*(1+(R1C2/100))$ for R4C3.
 $(R5C2+R4C3)*(1+(R1C2/100))$ for R5C3.
 \vdots
 $(R22C2+R21C2)*(1+(R1C2/100))$ for R22C3.

Now it is a simple matter to type in the formula for R3C3 and copy the formula from R4C3, into R5C3, R6C3, . . . , R22C3. However, we must be careful since we want the reference to the rate shown in R1C2 not to be adjusted as we copy from R4C3 to R22C3. This cell reference would have to be absolute, rather than relative, to ensure that.

We start of course by keying in the first year's accumulation. Next, we enter the formula for years 2 through 20, using relative references and copying them. So we type

G R 3 [TAB] 3 RETURN	(Goto R3C3.)
V	(Starts formula.)
←	(Enters relative reference RC[-1].)
*	(Multiplication sign.)
((Opens first parentheses.)
1+	(Numeral.)
((Opens second parentheses.)
R1C2	(Absolute reference to cell with rate.)
/	(Division sign.)
100	(Numeral, convert rate to percentage.)
)	(Close second parentheses.)
)	(Close first parentheses.)
↓	(Enter formula, move to R4C3.)
((Start formula with first parentheses.)
←	(Enters relative reference RC[-1].)
+	(Addition, cell pointer returns to RC.)
↑	(Enters relative reference R[-1]C.)
)	(Closes parentheses, pointer to RC.)
*	(Multiplication sign.)
((Opens second parentheses.)
1	(Numeral.)
+	(Addition sign.)
((Opens third parentheses.)
R1C2	(Absolute reference to rate cell.)
/	(Division sign.)
100	(Numeral, convert rate to percentage.)
)	(Close third parentheses.)
)	(Close second parentheses.)
RETURN	(Enters formula at R4C3.)
C	(For Copy.)
D	(Copy R4C3's formula, downward.)
19	(Copy into R5C3, R6C3, . . . , R22C3.)
RETURN	(Use current cell as source.)

This leads to the following IRA table (except for some format changes that can be handled later).

#1	1	2	3	4	5	6	7
1	Rate		12				
2	Year	Contr	Value				
3		1	2000	2240			
4		2	2000	4748.8			
5		3	2000	7558.656			
6		4	2000	10705.69			<C4D>

You can now easily evaluate the impact of a new rate, simply by placing it in position R1C2.

You can make it easier to compare the effect of different rates by having their consequences placed side by side. Suppose we use column 5 to display the values computed using a different rate. We can set up column 5 by typing

```

G R 1 [TAB] 5 RETURN      (Goto R1C5.)
14                          (Enters new rate.)
↓                            (Enters rate and moves down to R2C5.)
Value                       (Activates Alpha, title of column.)
↓                            (Moves down to R3C5.)
V                            (Activates formula.)
←
←
←                            (Enters relative reference RC[-3].)
*                            (Multiplication sign.)
(                            (Opens first parentheses.)
1                            (Numeral.)
+                            (Addition sign.)
(                            (Opens second parentheses.)
R1C5                        (Absolute reference to cell with rate.)
/                            (Division sign.)
100                         (Numeral, convert rate to percentage.)
)                            (Close second parentheses.)
)                            (Close first parentheses.)
↓                            (Enters formula, moves to R4C5.)
(                            (Initiates Formula, opens parentheses.)
←
←
←                            (Enters relative reference RC[-3]C.)
+                            (Addition, cursor to RC.)
↑                            (Enters relative formula R[-1]C.)
)                            (Closes parentheses.)
*                            (Multiplication sign.)
(                            (Opens second parentheses.)
1                            (Numeral.)
+                            (Addition sign.)
(                            (Opens third parentheses.)
R1C5                        (Absolute reference to cell with rate.)
/                            (Division sign.)
100                         (Numeral, convert rate to percentage.)
)                            (Close third parentheses.)
)                            (Close second parentheses.)
)                            (Close first parentheses.)
RETURN                      (Enters formula at R4C5.)
C                            (For Copy.)

```

D (Copy R4C5's formula, downward.)
 19 (Copy into R5C5, R6C5, . . . , R22C5.)
 RETURN (Use current cell as source.)

We now have the following table:

#1	1	2	3	4	5	6	7
1	Rate		12			14	
2	Year	Contr	Value		Value		
3		1	2000	2240		2280	
4		2	2000	4748.8		4879.2	
5		3	2000	7558.656		7842.288	
6		4	2000	10705.69		11220.208	
7		5	2000	14230.38		15071.037	
8		6	2000	18178.02		19460.983	<C4E>

Note how easy it is to make comparisons now. Of course there are many other factors to be taken into account in setting up an IRA; getting a good rate of return is just one of these factors.

Summary

- You can complete entries and advance the cursor to the next location by using the cursor arrow keys.
- Numbers and formulas begin with digits or the special characters +, -, (, and the period. They can also begin with cell locations, provided you first pressed V(alue).
- Labels and text begin with an Alpha character. It must be preceded with a RETURN (if the cursor was over Alpha) or A. At this point Multiplan will continue to enter text until you enter another number or value.
- The Clear-the-Screen command is TC. It provides a prompt and expects a Y response.
- Print worksheets using the P(rint) command. Multiplan answers with the subcommands, the first of which is printer. Make sure that your printer is set up and turned on. Simply pressing RETURN activates this first subcommand.
- Copying: type C and respond to several subcommands:

COPY: Right Down From

Selecting any one of the three will give you several other options, the first of which requests the number of cells. We specify the number and follow by pressing [TAB], if we wish to change the default option in the next field. If we do not wish to change the default options in the second field, responding with RETURN, rather than [TAB], activates copy.

Multiplan requires relative references which have the following meaning:

R	Current row.
C	Current column.
RC	Current active cell.
R[+n]	The row n rows <i>below</i> R.
C[+n]	The column n columns to the <i>right</i> of C.
R[-n]	The row n rows <i>above</i> R.
C[-n]	The column n columns to the <i>left</i> of C.
RC[-1]	A single cell, just to the left of the current cell.
R[-1]C	A single cell, just above the current cell.

The above are generated automatically, by pointing the cell pointer to the appropriate cells.

You can get the directory of files on your diskette by pressing an arrow key following a Transfer Load command. (This command prompts you for file name; if you do not recall the file name, press an arrow key.)

5

Using Functions, Names, and Coping with Change

In this chapter we will introduce the use of some very handy tools known as functions. You may already be familiar with the use of high-powered functions on a calculator. They can save us a lot of time and effort. In order to better appreciate them, and also to consolidate the techniques we saw in Chapter 4, we will first work out a new problem with the techniques we have in hand. Then we will redo the problem using the functions we will soon discuss.

We will also examine another handy tool: names. Finally, we shall examine how best to cope with problems whose boundaries are not completely predetermined. How should you set things up when you don't know in advance exactly how many rows or columns you may need at a future time? This happens frequently and when dealing with a paper worksheet you do a lot of erasing or cutting and pasting. Fortunately, this has been anticipated by Multiplan and we will see how to cope with it. But first let us look at a new problem.

A Teacher's Gradebook

Most teachers' gradebooks are handwritten, and they have the following appearance:

Student	Homework1	Homework2	Test1	. . .	Total	Average
Alan	95	100	80		850	85
Bernice	85	95	73		820	82
Charles	60	0	50		400	40
. . .						
Fred	80	60	70		650	65

You might prefer to think of this example in other ways. A farmer can see it as recording dairy production by head and by month. A sales manager sees it as a monthly sales record for each salesperson. All you have to do is change the names and the titles.

To keep our example down to a manageable size, let us have a class of five students, with the following entries:

Student	HWK1	HWK2	TST1	TST2
Alan	95	100	80	85
Bernice	85	95	73	94
Charles	60	0	50	66
Fred	80	60	70	78
Mary	82	80	75	86

If we enter these labels, names, and numbers on the Multiplan worksheet, beginning in the top left corner, we get the following assignment of rows and columns:

#1	1	2	3	4	5	6
	1 STUDENT	HWK1	HWK2	TST1	TST2	
2	Alan	95	100	80	85	
3	Bernice	85	95	73	94	
4	Charles	60	0	50	66	
5	Fred	80	60	70	78	
6	Mary	82	80	75	86	
7						<C5A>

We can use column 6 for the desired Totals and column 7 for the Average. After typing these labels in the usual way

```
G R 1 [TAB] 6 RETURN
RETURN TOTAL
→
V
AV
ENTER
```

Then we should type the formula $R2C2+R2C3+R2C4+R2C5$ and enter that in position R2C6. Similarly the row 3 total is computed using $R3C2+R3C3+R3C4+R3C5$. Since this pattern applies to all of the totals, we can save some effort by using the Copy command. Begin by typing

G R 2 [TAB] 6 RETURN	(Position the cursor at R2C6.)
V	(Value initiated.)
← ← ← ←	(Enter relative reference.)
+	
← ← ←	(Second relative reference.)
+	
← ←	(Third reference.)
+	
← RETURN	(Last reference, entry complete.)
C D	(Copy Down initiated.)
4 RETURN	(Replicate into R3C6 to R6C6.)

This generates the formulas for the totals for rows 2 through 6 in column 6, and simultaneously calculates and displays the totals, as in the following figure:

#1	1	2	3	4	5	6	7
	1 STUDENT	HWK1	HWK2	TST1	TST2	TOTAL	
2	Alan	95	100	80	85		
3	Bernice	85	95	73	94		
4	Charles	60	0	50	66		
5	Fred	80	60	70	78		
6	Mary	82	80	75	86		
7							<C5B>

How can we compute the averages for column 7? The average of a set of numbers is defined as the sum divided by the number of items. The average grade for a particular student can then be calculated using the total we just computed and dividing that total by the number of items involved, which in this case is just 4.

The formulas we need are:

For R2C7, we need R2C6/4.

For R3C7, we need R3C6/4.

For R4C7, we need R4C6/4.

For R5C7, we need R5C6/4.

For R6C7, we need R6C6/4.

Once again we can use replication

G R 2 [TAB] 7 RETURN

(Position cursor at R2C7.)

V

(Initiate Value.)

← / 4 RETURN

(Enter formula for R2C7.)

C D 4

(Request Copy Down, 4 cells.)

RETURN

(Activate copy of R2C7.)

Our table now:

#1	1	2	3	4	5	6	7
1	STUDENT	HWK1	HWK2	TST1	TST2	TOTAL	AV
2	Alan	95	100	80	85	360	90
3	Bernice	85	95	73	94	347	86.75
4	Charles	60	0	50	66	176	44
5	Fred	80	60	70	78	288	72
6	Mary	82	80	75	86	323	80.75
7							<C5C>

If you are typing this into a Multiplan worksheet as you are reading this, the averages Multiplan calculates won't be rounded, appearing exactly as shown here. In a later section we will see how we can get Multiplan to round off numbers to avoid cluttering up your display. If you can't wait to see, look up Format in the index, or read Chapter 9.

We are ready now to redo this problem using functions.

Using Functions

Most of us have used functions of one kind or another, perhaps in high school or college. They have many different appearances. Square root is a function. So are the trigonometric functions with such names as sine and cosine. Some things we don't usually think of as functions can be very useful if we modify our point of view.

The two functions which we will use here are:

SUM	Compute the sum or total.
AVERAGE	Compute the average.

When you use a function, you have to specify what number the function is to use to compute. The items it computes are called its *arguments*. Such functions as square root and sine have only one argument or number. The SUM and AVERAGE can have many arguments, and these are presented as an *argument list*.

You can use SUM by providing it with a list of one or more arguments, as in

SUM(R2C3)

In this case, SUM has only one argument, the coordinate R2C3, and the result will be the same as if we had typed R2C3 as a formula. If you type

SUM(R2:7C3) RETURN

this will compute R2C3+R3C3+R4C3+R5C3+R6C3+R7C3. You can, if you wish, include other items in the list.

SUM(R5:9C4,100,R10C3:8)

will add the items from rows 5 through 9 of column 4, the number 100, then the items from columns 3 through 8 of row 10. The argument list can include such expressions as 2*(R1C5-R1C4), and so on. The use of AVERAGE is similar to the use of SUM.

AVERAGE(R3:15C2)

will compute the average value of the items in rows 3 through 15 of column 2, but this average will include only the nonblank numeric values. In the case of a teacher's gradebook, this means that the student who missed a test due to illness and presumably received a blank grade won't be penalized; all blank grades will be ignored.

Now let us redo the gradebook, using SUM and AVERAGE. To get totals in rows R2C6 through R6C6, type

G R 2 [TAB] 6 RETURN	(Position cursor at R2C6.)
V	
SUM	(Initiate function.)
(← ← ← ←	(Inserts list of arguments.)
: ←) RETURN	(Completes list, activates a relative reference. See contents sum(rc[-4]:rc[-1]) . Copy, next:)
C D 4	(Copy Down 4 cells.)
RETURN	(Activates command.)

This gives us the table:

#1	1	2	3	4	5	6	7
1	STUDENT	HWK1	HWK2	TST1	TST2	TOTAL	
2	Alan	95	100	80	85	360	
3	Bernice	85	95	73	94	347	
4	Charles	60	0	50	66	176	
5	Fred	80	60	70	78	288	
6	Mary	82	80	75	86	323	
7							<C5D>

Similarly, we can get the averages for rows R2C7 through R6C7 by typing

G R 2 [TAB] 7 RETURN	(Position cursor at R2C7.)
V	
AVERAGE	(Initiate function.)
(← ← ← ←	(Inserts list of arguments.)
: ← ←) RETURN	(Completes list, activates a relative reference. See contents average (rc[-5]:rc[-2]) .)
C D 4	(Copy Down 4 cells.)
RETURN	(Activates command.)

which leaves us with the following table:

#1	1	2	3	4	5	6	7
	STUDENT	HWK1	HWK2	TST1	TST2	TOTAL	AV
2	Alan	95	100	80	85	360	90
3	Bernice	85	95	73	94	347	86.75
4	Charles	60	0	50	66	176	44
5	Fred	80	60	70	78	288	72
6	Mary	82	80	75	86	323	80.75
7							

<C5E>

It would be very nice if, as the grades for the latest homework or test are entered, the gradebook would compute the class average for that homework or test. All we have to provide is a formula for a new bottom row, which we will label Av, for Average. We can provide the label and the formula, by typing

G R 7 [TAB] 1 RETURN	(Get to R7C1.)
RETURN Av RETURN →	(Title in, cursor moved to R7C2.)
V AVERAGE	(Initiate Formula.)
((Start list of arguments.)
↑↑↑↑↑	(Enter first argument.)
:	(Through.)
↑)	(Formula for R7C2.)
RETURN	(Activate it in R7C2.)
C RETURN	(Copy Right.)
5	(Begin replicating it.)
RETURN	(Activate Copy.)

Our table now looks like:

#1	1	2	3	4	5	6	7
	STUDENT	HWK1	HWK2	TST1	TST2	TOTAL	AV
2	Alan	95	100	80	85	360	90
3	Bernice	85	95	73	94	347	86.75
4	Charles	60	60	50	66	176	44
5	Fred	80	60	70	78	288	72
6	Mary	82	80	75	86	323	80.75
7	Av	80.4	67	69.6	81.8	298.8	74.7
8							

<C5F>

We will examine some of the other useful functions provided by Multiplan later. We will now consider the next major topic of this chapter.

Using Names

In order to use the name function, it is essential that the use of cell addresses, both relative and absolute references, be understood. Let us use the gradebook example and use the column titles as names for the columns. We will assume that table C5B is entered.

You invoke the Name command by going to R1C1 and typing

N

You will see the prompt:

```
NAME: define name: STUDENT
      to refer to: R1C1
```

```
Enter name
R1C1      "STUDENT"          99% Free      Multiplan: c5f
```

As soon as we are in R1C1, Multiplan responds to our command with several default answers, portrayed in the prompt above. Thus, the contents of RC (our active cell) are displayed following "NAME: define name:". Since we agreed to use the column heading as a name for the column, we need only check the other fields in the prompt.

The second field "to refer to:" shows R1C1. We want only C1 there. So we type

```
[TAB]                (To next field.)
C1 RETURN            (Enters the name for column 1.)
→                  (Moves to R1C2, our next name cell.)
```

Now we are ready to name column 2, HWK1. We repeat the above procedure and type

```
N
```

and then we see the prompt as we have before, but wait. The next field ("to refer to:") now shows C2 (!). Again, Multiplan anticipated our request and shows the column as the target area for the name. We need only press RETURN, and move to the next cell we wish named. Similarly, we can name HWK2, TST1, and so on by entering

```
N RETURN →
```

for each column. From now on we can use the names in lieu of cell references. In a like manner, we can define row names and use them as well. An example will help.

Suppose we wish to average every student's grade, just as we did earlier in the chapter. Recall that we had to type in

```
G R 2 [TAB] 7 RETURN    (Position cursor at R2C7.)
V
AVERAGE                (Initiate function.)
( ← ← ← ← ←           (Inserts list of arguments.)
: ← ←) RETURN          (Completes list, activates a relative reference.)
C D 4                  (Copy Down 4 cells.)
RETURN                 (Activates command.)
```

With the row names we can now simplify matters. Let us use the student's name as the row name. We type

```
G R 2 [TAB] 1 RETURN
N [TAB] R2C2:5 RETURN ↓
N RETURN ↓
N RETURN ↓
N RETURN ↓
N RETURN ↓
N RETURN ↓
```

Each name now refers to its row of grades. We are ready to enter the students' averages. Type

```
G R 2 [TAB] 7 RETURN
V
AVERAGE(Alan) RETURN ↓
V
AVERAGE(Bernice) RETURN ↓
. . .
AVERAGE(Mary) RETURN
```

Now that may not seem like a great difference at first, but it may be much closer to our everyday use of English, and, more importantly, it is closer to the way we think!

As you probably noticed, you can name whole or partial rows, individual cells, and entire windows. Multiplan will store your names along with their definitions so you never have to worry about remembering them or the exact references.

Should you ever forget the names, Multiplan will help; simply key in the now familiar N and follow with either

← (left arrow,) or → (right arrow,)

and you will see the list of names. You can use the above keys to move among them, just as you did moving through the diskette directory. The → moves "forward", ← moves "backward", whereas HOME goes to the first name on the list, while END moves you to the last one.

Coping With Change

What if part way through the school year a new student joins the class? Does the teacher have to rewrite all the gradebook formulas? Similarly, the teacher may not know exactly how many assignments or tests will be given during the school year. So how can things be set up so that changing the number of rows or columns can be done with a minimum of effort?

Deleting Rows or Columns

Having convinced ourselves that the formulas we are using to compute the average grades are correct, we might decide to simplify our gradebook by erasing the column headed TOTAL. How can we erase or delete an entire column (or row)? Typing

D (For the delete command.)

produces the prompt:

```
Row Column
```

You can respond with an R or RETURN to delete a row, with a C to delete a column, or push Run Stop if you don't want to delete anything. If you type R the prompt will be:

```
DELETE ROW # of rows: 1
starting at: current row
between rows: 1 and: 63
```

Pressing RETURN without a row number will delete the row the cursor currently occupies. Entering a specific row number will delete that row. Note that the default answer (underlined above) is 1 row, starting at the current cell's row.

Likewise, if we placed the cursor anywhere within column 6, say at R1C6

G R 1 [TAB] 6 RETURN

D (Delete.)

C

RETURN (Current column to be deleted.)

then column 6 will disappear, and all columns to its right will move over to fill in the empty space, leaving us with the table as it appears in the next section.

The deletion of a row has a similar effect. Rows below the deleted row will move to close the gap. In all cases (deletion of a row or column), formulas which used to refer to rows (or columns) which are moved due to a row (or column) deletion will be adjusted so the results remain correct.

Inserting Rows or Columns

Suppose we have a version of our gradebook in which the TOTAL column has been deleted, since we don't really need it now. Furthermore, suppose the second homework, HWK2, had just been graded and recorded:

#1	1	2	3	4	5	6
	STUDENT	HWK1	HWK2	TST1	TST2	AV
2	Alan	95	100			97.5
3	Bernice	85	95			90
4	Charles	60	0			30
5	Fred	80	60			70
6	Mary	82	80			81
7	Av	80.4	67			73.7
8						<C5G>

A new student, Doris, has just joined the class. Suppose her name is to be entered between Charles and Fred. We have

Row 4 Charles . . .

Row 5 Fred . . .

We want

Row 4 Charles . . .

Row 5 Doris . . .

Row 6 Fred . . .

We can insert a row by using the I (Insert) command. Selecting the I command requests insertion, and the I command will prompt you:

Row Column

Here we will type R for row insertion. Multiplan will prompt:

INSERT ROW # of rows:

before row:

between columns: and:

A blank row will be inserted at the specified row, pushing the previous contents of the entire row down one row. All rows below it will be adjusted to reflect this change.

As an alternative to entering the row number you could have located the cursor on the row just below where the new row is desired. For example, if you wish to insert a row above row 5, place the cursor on row 5 before you use the IR command. Similarly, if you want a new column to the left of an existing column, place the cursor on the existing column before you type IC. If you find you made a mistake, you can delete the new row or the new column by typing

D
R RETURN (Delete row the cursor is on.)

or

D
C RETURN (Delete the column the cursor is on.)

In our gradebook, if we place the cursor on row 5 and type IR RETURN, we will see a blank line appear in row 5 as the older rows 5, 6, etc., are pushed down and renumbered 6, 7, etc. Notice that the averages have not been affected.

#	1	2	3	4	5	6
	STUDENT	HWK1	HWK2	TST1	TST2	AV
1	Alan		95	100		97.5
2	Bernice		85	95		90
3	Charles		60	0		30
4						
5	Fred		80	60		70
6	Mary		82	80		81
7	Av		80.4	67		73.7
8						
9						<C5H>

We can now type in the new student's name, Doris, at R5C1. If we position the cursor at R8C2, we can examine the formula saved to calculate the average for column B. It appears on the prompt line as:

AVERAGE(R[-6]C:R[-1]C)

But if you look back a few pages, we entered it as

AVERAGE(R[-5]C:R[-1]C)

How did the R[-5]C suddenly change into a R[-6]C? If we look at the formulas for R8C3 and R8C4, those too have been changed. What is going on here?

In the process of inserting a row, if the new row falls into the range of some formula, that formula will be updated to include the new row, and it will also continue to include the old rows it used to cover. That inclusion encompasses those rows which had to be assigned new row numbers due to the row insertion. So Fred and Mary, from old rows 5 and 6, are still included in the computation for Av, using their new row numbers 6 and 7. The electronic worksheet provided by Multiplan behaves just as a paper worksheet should if you squeezed in a new row.

Notice that since AVERAGE ignores blank entries, Doris's blank grades for HWK1 and HWK2 have no effect on the averages for the rest of the class.

Caution

You may wish to circle this paragraph because it makes an important point about the consequences of an insertion. You must insert a row into (INTO) the range of a formula, not merely add a row to one end or the other of a formula's range, if you want the formula to be automatically adjusted to include the new row. So if we had inserted Doris's entry in front of old row 7, the formula would not have been adjusted. We will reconsider this situation in a few moments.

Adding a Column

Suppose we had our gradebook as follows:

#	1	2	3	4	5	6
1	STUDENT	HWK1	HWK2	TST1	TST2	AV
2	Alan	95	100			97.5
3	Bernice	85	95			90
4	Charles	60	0			30
5	Doris					
6	Fred	80	60			70
7	Mary	82	80			81
8	Av	80.4	67			73.7
9						<C5I>
10						

However, we really wish it had an extra column, as in:

#	1	2	3	4	5	6	7
1	STUDENT	HWK1	HWK2	HWK3	TST1	TST2	AV
2	Alan	95	100				97.5
3	Bernice	85	95				90
4	Charles	60	0				30
5	Doris						
6	Fred	80	60				70
7	Mary	82	80				81
8	Av	80.4	67				73.7
9							<C5J>

We have finished grading homework number 3, and wish to enter it as HWK3, between HWK2 in column 3 and TST1 in column 4. We could begin by placing the cursor on column 4 (any row in column 4 will do). That ensures that the new column will be inserted immediately "in front of" column 4, which is to say, immediately to its left. We should now type

```
I  
C RETURN (Insert a column.)
```

which will give us the desired additional column. Having graded homework number 3, we can enter its title HWK3 and the grades 95, 80, 54, 88, 64, and 90 in column 4, by typing

```
G R 1 [TAB] 4 RETURN
RETURN HWK3
↓
95 ↓
80 ↓
54 ↓
88 ↓
64 ↓
90 RETURN
```

```
(Cursor at R1C4.)
(Enter the label HWK3.)
(Move cursor to R2C4.)
```

As we type in the new grades, the row averages for all of the students are updated, except for Doris. We have to provide a formula for this new row's average if we want it to be computed. Similarly, having entered the new column 4, for homework number 3, we have to provide the formula for that column's average if we want it to be computed. In both cases we can either type in the desired formula or position the cursor on the corresponding formula for an older row or column, and replicate it. Using this latter approach, for instance, we could copy the formula from R8C3 to R8C4 by typing

```
G R 8 [TAB] 3          RETURN (Position over old formula.)
C RETURN 1            (Replicate it.)
RETURN                (Use the formula at R8C3.)
```

You could of course simply have typed

```
G R 8 [TAB] 4 RETURN
AVERAGE(R2:7C4) RETURN
```

Using either approach, you can also provide the formula for the average for R5C7. *WARNING:* If you use the second approach and directly enter the formula, you will not be able to copy the contents of the cell at a later time because only *relative reference* type formulas can be copied. (Look at the contents of R8C4 using both approaches to confirm this.) The result in either case is:

#	1	2	3	4	5	6	7
	STUDENT	HWK1	HWK2	HWK3	TST1	TST2	AV
2	Alan		95	100	95		96.66667
3	Bernice		85	95	80		86.66667
4	Charles		60	0	54		38
5	Doris				88		88
6	Fred		80	60	64		68
7	Mary		82	80	90		84
8	Av		80.4	67	78.5		75.3
9							<C5K>

Planning for Change

If you were setting up a new gradebook, you could anticipate having to add new students, or recording more tests or assignments than originally anticipated, either by setting up dummy entries or entries which will always retain their relative positions.

The first approach goes as follows. For student names, you could have two dummy entries, one called "first," the other called "last." Then you could insert students as needed, in any row between these two, and be confident all column formulas will be adjusted accordingly. Similarly, you can safely designate a column to record the grades for a final exam, even if you never use it. Any extra tests or assignments would be recorded to its left. Your gradebook could look like:

#1	1	2	3	4	5	6	7
1	STUDENT	HWK1	HWK2	HWK3	TST1	Final	AV
2	first						
3							
4							
5							
6							
7							
8	last						
9	Av						

The formula for the row averages can be defined for each of rows 2 through 8, using the Copy command. Similarly, the formula for the column averages can be defined for each of columns 2 through 7, using the same command. For your convenience, they are repeated here.

```
G R 9 [TAB] 2 RETURN          (Move cursor to R9C2.)
V
AVERAGE
(↑↑↑↑↑↑↑
:                               (Move cursor back to R9C2.)
↑) RETURN                     (Completes command.)
C RETURN 6                     (Replicate.)
RETURN                         (Use R9C2 as source.)
```

That takes care of the column averages.

Note that **#DIV/0!** will appear in any cell where average is used and no numbers appear in the cells. (The # “pound” sign indicates a possible error.) This is because the Average function has no data and an attempt to divide by zero is illegal (not defined in math). There is no need to worry about this, since as soon as you enter numbers into the columns to be averaged, the correct average will appear. The row averages are generated by typing

```
G R 2 [TAB] 7 RETURN          (Cursor to R2C7.)
V
AVERAGE
(←←←←←
:←) RETURN
C D 6
RETURN
```

What if you make an error when you type in a value? The next section provides some relief.

Erasing Errors

If you type in an incorrect value, or simply change your mind, you can replace any value (or label) by positioning the cursor over the desired item and typing in the new alpha or value entry. Unfortunately, that does not let you erase anything. You might think: “I can simply replace the old value by the number 0.” That won’t usually be satisfactory, for two reasons: (1) a zero will be displayed and/or printed, and it may look awful (think of the times you were not paying attention, and typed something into location R1C1, and you could not get rid of it) and, (2) a zero counts when the AVERAGE function is being used, whereas a blank entry does not count—that could change your results. How then do you erase something?

You could try DC RETURN; that would delete the entire column the cursor was sitting on. Similarly, if you try DR RETURN, that will delete the entire row the cursor was sitting on. These actions may be a little drastic. If you want only to erase (blank out) one item, place the cursor on it and type

B	(Blank.)
RETURN	(Contents of current cell.)

The B command erases the information at the location the cursor is sitting on if you push RETURN or any of the cursor keys. If you change your mind and wish not to erase the item you have selected, then push the Run Stop key before pushing the RETURN key.

You can also blank out a partial row or column of cells by specifying a range. For example, if we wished to blank out all data in column 4 of our gradebook we would type

B R2:7C4 RETURN

This would leave the average calculations which would show an #DIV/0! (recall that the pound “#” sign indicates an error) for column 4 since there is now no data in this column. We could even blank out all the scores in our gradebook by specifying a range from the upper left cell to the lower right cell of the block that we want to blank. This command would be

B R2C2:R8C4 RETURN

Correcting Errors: The Edit Command

Selecting the E command activates the edit command. Without this command, if you wish to change a label, a value, or a formula, you have to retype the whole thing, even if the new version differs from the old in just one or two characters. Using the Edit command makes it possible to retrieve a label, a value, or a formula, change any desired part or parts, and replace the old version with the new one. Let us look at a simple example.

Suppose you had earlier typed the value 12345 in location R2C2, and you now want to change it to 912345. You can proceed as follows

G R 2 [TAB] 2 RETURN	
E	(Invoke the edit command.)

You will see the prompt:

EDIT: 12345< > (< > here represents the edit cursor.)

If you enter the location of any cell other than R2C2, the contents of that cell will be placed on the entry line. This data can be either edited or returned to cell R2C2.

Having placed the cursor at R2C2 (using the cursor arrows, or a G R 2 [TAB] 2), when you type the E, the item at the current location, R2C2 is copied onto the entry line at the bottom of the worksheet on the left. The entry line will show a small cursor just to the right of the right-most character of the item on the edit line. Here in the text we will use the pair of symbols “<” and “>” to embrace the character selected by the edit cursor.

The edit cursor is moved with the function keys F4, F6, F2, and F8. In Chapter 3 we introduced some of the function keys as “action” keys. Let us review them here; the edit keys are:

F2	Word left.
F8	Word right.
F4	Character left.
F6	Character right.

When editing you move the cursor to the character that you want to edit using the above keys. You can delete characters by pressing either the F3 key (or the combination CTRL and Y) once the cursor is over the character. If the cursor covers a word (for example in a label) pressing F3 or CTRL-Y combination will delete the entire word. Blanks may be inserted to the left of the cursor by using the space bar. You can enter characters over the resulting blanks or leave them as blanks. Now move the cursor back and forth using the function keys, ending over the number 1, on the left side. You now see:

<1>2345 (Edit cursor over the "1")

Type in a 9, thus producing:

9<1>2345

You may want to practice deleting characters, creating spaces and filling spaces by using the four function keys. Also observe that the DEL key may be used to delete characters located to the left of the cursor. When you are satisfied with the results, pressing the RETURN key will copy what is on the entry line into the active cell.

# N/A	(Data not available.)
#NAME?	(Name not defined.)
#NUM!	(Overflow or illegal operation.)
#DIV/0!	(You are trying to divide by 0.)
#REF!	(Reference to a nonexistent cell.)
#NULL!	(Reference to an intersection which is disjointed.)
#VALUE	(Wrong type of value used.)

When you are editing labels or text, they will appear on the entry line within quotation marks. These quotation marks must be left at the beginning and end of your label or text, since this is the way Multiplan distinguishes between the two. You can change formulas and values to text simply by entering quotation marks at their ends. Note that Multiplan will no longer treat these formulas as such; they will become labels.

The rules for the use of the edit command are:

1. Use F2, F8, F4 and F6 to position the edit cursor.
2. Use the space bar to insert a space to the left of the cursor.
3. Use DEL to delete the character at the cursor.
4. Pushing RETURN terminates the edit process, copying the item from the edit entry line back into a worksheet location.
5. Pushing any other symbolic key (letters, digits, arithmetic operators, etc.) replaces the corresponding character to the left of the edit cursor.
6. Quotation marks (“”) distinguish between labels or text and values or formulas. If editing text, leave these as they are; do not eliminate the quotation marks unless you wish to convert the text to a formula. You must, of course, be sure that this new formula conforms to all the rules governing formulas in Multiplan.

One more example should suffice. Suppose you had earlier typed a formula in position R1C1, R1C2*R1C3+1/R1C4, and later realized that you intended the 1 to be added to R1C3 before the multiplication by R1C2 occurs. So you have to insert parentheses, which transforms the formula R1C2*R1C3+1/R1C4 into R1C2*(R1C3+1)/R1C4. Instead of retyping the formula (what if it were part of a very long formula!), let us edit it. Type

G R 1 [TAB] 5 RETURN
E

and see $R1C2 * R1C3 + 1 / R1C4 < >$ on the edit entry line. Move the cursor so it sits on the R of the R1C3, by pushing F4 eleven times. Insert a "(" . This leaves the cursor to the right of the "(" we just typed. Our edit entry line should be showing $R1C2 * (<R>1C3 + 1 / R1C4$. We need the matching ")" between the pair 1 and /, so we must move the cursor right six places, to sit on the /. Typing F6, 6 times does this. Again we insert the missing ")" . This leaves us with $R1C1 * (R1C3 + 1) < / > R1C4$. Since this is exactly what we want, press RETURN. You do not have to return the cursor to the right-most position to enter the edited entry. Pushing RETURN at any place on the edit line will place the entire entry into the active cell.

Do not hesitate to use the Edit command from now on; with practice doing so will become second nature for you.

The Move Command

What if you had just inserted a row for a new student named Susan? After having typed in her name and recent grades, you notice that you inserted the line at the wrong place! You could of course delete the row, then insert a new row at the correct place, then retype the information you just deleted. Somehow computers are supposed to simplify work, not complicate it—there must be a better way.

You could try moving the line to the desired position, with the help of the M command. It prompts you very simply:

Row Column

If you were at location R1C1 when selecting R RETURN for row, it prompts you with:

```
MOVE ROW from row: 1
                to before row: 1
                # of rows: 1
Enter a number
```

Having entered the row number (or RETURN if we want the default), we need only enter the desired target row number, and the number of rows to be moved. At this point pushing RETURN will execute the command.

So to move an entry from, say, row 5 to row 7, type

```
M                (Move command.)
R
5                (Specify from row 5.)
[TAB]
8 RETURN        (Specify before row 8.)
```

Similar considerations apply to moving a column to the right or left.

Summary

We have reviewed the simple uses of Copy to copy the formulas of one cell to one or more cells. Copy assures us that the current relative and absolute references are included in the copied formulas.

We have introduced the use of two functions:

```
SUM(argument list)
AVERAGE(argument list)
```

Deletion of a row is requested using DR. If you followed it with a RETURN, the row the cursor was sitting on will be deleted, and any formula the row was included in will be adjusted to compensate for the row's deletion.

Deletion of a column is requested using DC, and the effect on formulas is similar to that when deleting rows.

Insertion of rows is performed by positioning the cursor on an existing row, and typing IR to create a blank row in front of (above) the older row.

Insertion of columns uses IC, and the new column will be placed left of the column the cursor was on.

Insertion causes formulas to be adjusted to include the new rows or columns, provided they were inserted within the range of a formula; specifically, formulas will not be adjusted if you insert a row or column in front of the first row or column of a formula's range. Similarly, inserting a row or column after the last row or column of a formula's range will leave the formula unchanged.

Erasing (blinking out) an entry is done by positioning the cursor on it and typing B. Alternatively, B can be followed by the range of the cells for a row, column, or block of cells that you wish to erase.

Editing an entry instead of retyping the entire entry is possible by using the E command. Edit allows insertions and deletions, one character at a time.

Moving a row (or a column) is accomplished by using the M command.

6

What If It Won't Fit on the Screen?

We will now examine those features of Multiplan that permit you to manipulate much larger worksheets than you can display on your screen. How you can do this, and how you can print out worksheets that require wider paper than your printer can handle, will provide the substance of this chapter.

Window: Scrolling Through the Worksheet

It bears repeating that Multiplan makes it possible for smaller computers, such as you are using, to do things that are conspicuous by their absence on much larger computers. The fact is that few large computers provide the features of Multiplan in such an easy-to-use fashion. This is particularly applicable to what we are about to discuss.

As you recall, the Multiplan worksheet can extend from column 1 to column 63 and from rows 1 to 255. Suppose you filled out a worksheet with a row of eleven labels, as you would obtain by typing

```
G R 1 [TAB] 1 RETURN
RETURN Name →
H1 →
H2 → .
T1 →
T2 →
. . .
T8 RETURN
```

At the point where the label T4 has been typed, the label "Name" disappears from view, leaving your screen as follows:

#	1	2	3	4	5	6	7	8
1	H1	H2	T1	T2	T3	T4	T5	
2								

Similarly, if you filled in the 19 rows with names and grades, the titles in row 1 would disappear from your screen as you began entering the items for row 20. Suppose we had the names and grades shown in Table 6.1. We saw some of this earlier, but now it is much larger.

How can we easily work with worksheets that do not fit on the screen? Before you get upset and start blaming yourself for having purchased a Commodore 64 microcomputer instead of buying time for use on a bigger computer, note that the state of affairs with a larger computer is not much different. Your screen size is typical of the screen size used on larger computers. In fact, your Multiplan program

Table 6.1 Large Gradebook

#1	1	2	3	4	5	6	7	8	9	10	11
	Name	H1	H2	T1	T2	T3	T4	T5	T6	T7	T8
2	Alan	95	100	88	55	88	85	79	82	76	83
3	Bernice	85	95	72	88	66	74	80	74	69	78
4	Charles	60	0	84	92	89	78	70	83	84	71
5	Ruth	66	76	75	77	78	79	73	78	77	75
6	Mary	82	80	76	66	76	83	78	80	89	79
7	Fred	80	60	52	98	83	68	79	83	55	73
8	Sara	98	66	88	69	45	88	74	51	66	72
9	Phil	88	71	49	94	77	59	88	77	50	73
10	Patrick	90	87	86	78	75	81	78	90	94	84
11	Sue	66	68	72	74	77	83	90	92	95	80
12	Maria	75	78	67	72	77	80	82	78	85	77
13	Jim	66	69	64	72	77	70	74	78	78	72
14	Jack	44	49	70	65	68	72	76	72	78	66
15	Nicole	66	68	78	62	70	80	78	88	84	75
16	Roger	69	73	80	71	76	90	81	84	88	79
17	Larry	88	81	78	73	77	78	80	79	84	80
18	Adam	56	67	87	72	66	45	69	78	88	70
19	Helene	66	78	88	89	90	88	90	88	94	86
20	Ronald	66	68	87	72	66	78	83	71	77	74
21	Francine	66	78	88	92	88	86	80	85	88	83
22	Eddy	50	52	45	67	76	62	56	55	58	58
23											
24	Average	72	70	75	76	75	77	78	78	79	76

<C6A>

makes your very small computer much more useful than is the case with many of the fancy terminals connected to very large computers.

If you had a very large screen, you probably would make more errors. Having a reasonably sized screen lets you concentrate on the few things the human mind can handle well at one time.

Printing Long Worksheets

If a worksheet won't fit on the screen because it is too long, it still can be easily printed in one operation. Suppose we had a roster with 30 rows, with the names in column 1 and the badge numbers in column 2.

	1	2
	Name	Badge Number
1	Sam	321
2	James	700
...
30	Clouseau	123

To print the worksheet, place the cursor at R1C1 by typing

HOME

To get the print command to work, type

P (For Print.)
RETURN (For Printer.)

This will automatically print your entire worksheet, if it fits on the printer paper. If you wanted just the names printed, without titles or badge numbers, you could change the range

P (For Print.)
O (For Options.)

and see the options displayed:

```
PRINT OPTIONS:  area:                setup:
                 formulas: Yes (No)   row-column numbers: Yes (No)
```

The default response to the area field is the entire size of your worksheet. Should you want to print the names only, all you have to do is enter R1:30C1 like this:

```
PRINT OPTIONS: area:R1:30C1
```

and follow with

```
RETURN  
P (Begin printing.)
```

This will cause column 1 to be printed, starting at row 1, continuing up to row 30.

Printing Wide Worksheets

If your worksheet is too wide to fit on the screen, Multiplan will automatically split it into two or more sections, depending on your printer. If you don't like the way Multiplan split the table for you, you can instruct it to print differently, using a few commands; it depends on the printer you are using and the paper it is printing on (you might be using 8-inch wide paper on a printer capable of using 15-inch wide paper).

Suppose your printer accommodates only 80 character lines. Our example can be printed in two segments. Segment 1 is produced using the commands

```
P (For Print.)  
O (For Options.)  
R1:24C1:7 (Select range.)  
RETURN RETURN (Begin printing.)
```

Segment 2, shown after segment 1, is produced by typing

```
P  
O  
R1:24C7:11  
RETURN RETURN
```

Note that column 7 has been printed in both segments to show where they overlap.

A wider worksheet will also be printed by Multiplan, which will automatically slice it up into the widest slices your printer can handle. You can reconstruct the desired end product with some cutting and pasting or taping of the printout segments. Moreover, you could at times reduce the default column size; in the gradebook example above, we reduced column 2 from 10 to 4, and all the columns from 3 and up to 3. We thus accomplished reduced size without losing student grades. More on changing widths later.

If you expect to be dealing with very large worksheets frequently, consider investing in a printer with a wider print line. The typical print line sizes are 80, 120, and 132 characters. There are larger sizes available, specifically for financial reports. Just because your screen only accommodates 40 char-

Segment 1

1	2	3	4	5	6	7	
1 Name	H1	H2	T1	T2	T3	T4	
2 Alan		95	100	88	55	88	85
3 Bernice		85	95	72	88	66	74
4 Charles		60	0	84	92	89	78
5 Ruth		66	76	75	77	78	79
6 Mary		82	80	76	66	76	83
7 Fred		80	60	52	98	83	68
8 Sara		98	66	88	69	45	88
9 Phil		88	71	49	94	77	59
10 Patrick		90	87	86	78	75	81
11 Sue		66	68	72	74	77	83
12 Maria		75	78	67	72	77	80
13 Jim		66	69	64	72	77	70
14 Jack		44	49	70	65	68	72
15 Nicole		66	68	78	62	70	80
16 Roger		69	73	80	71	76	90
17 Larry		88	81	78	73	77	78
18 Adam		56	67	87	72	66	45
19 Helene		66	78	88	89	90	88
20 Ronald		66	68	87	72	66	78
21 Francine		66	78	88	92	88	86
22 Eddy		50	52	45	67	76	62
23		--	--	--	--	--	--
24 Average		72	70	75	76	75	77

Segment 2

7	8	9	10	11	
1 T4	T5	T6	T7	T8	
2	85	79	82	76	83
3	74	80	74	69	78
4	78	70	83	84	71
5	79	73	78	77	75
6	83	78	80	89	79
7	68	79	83	55	73
8	88	74	51	66	72
9	59	88	77	50	73
10	81	78	90	94	84
11	83	90	92	95	80
12	80	82	78	85	77
13	70	74	78	78	72
14	72	76	72	78	66
15	80	78	88	84	75
16	90	81	84	88	79
17	78	80	79	84	80
18	45	69	78	88	70
19	88	90	88	94	86
20	78	83	71	77	74
21	86	80	85	88	83
22	62	56	55	58	58
23	--	--	--	--	--
24	77	78	78	79	76

acter lines does not mean you can print only 80 character-wide segments. The maximum width of the segment depends on your printer and the commands you use; it does not depend on your screen width. You can also buy a printer with a compressed print mode, that squeezes more letters onto the line. If you have one already, read the next section.

A Note on Printing More Columns Using Compressed Print

If you have a printer like the Epson MX-80 Printer, or some other printer that supports a compressed-print feature, you could use it with Multiplan and the Commodore 64. This allows you to print 132 smaller characters on one line. You can activate this feature when you use the Multiplan program, using the following commands

P (Print.)

At this point the prompt line will respond:

PRINT: Printer File Margins Options

If you respond with

P (For Printer, or press RETURN.)

your printer will immediately begin printing in the manner that has been described in the previous section. Again, this assumes that you have an Epson-like printer capable of compressed print.

With the Print Margins subcommand you can change the Print Length, which will allow you to change the lines per page, and Print Width, the number of characters per line of the printed output. In order to use the compressed print feature of your printer make sure that it is on and type the ^O command in the setup field of the Print Options command.

Note that all printing will now be performed in the compressed mode until you change the setup or until you turn the printer off.

You can resume the normal 80-character-per-line full-sized character mode of printing either by reloading the Multiplan program or by typing the command sequence:

P M	(Print Margins.)
[TAB] [TAB]	(Move to Print Width field.)
80	(Enter desired range.)
RETURN	(Completes selection, returns to Print.)
O	(Options.)
[TAB]	(To Setup field.)
^	(Press the control character required and recognized by your printer.)
R	(Letter R, requesting regular print.)
RETURN	(Completes selection, returns to Print.)
P	(Starts Print.)

If you have a Commodore 1525 printer, the maximum number of columns you can print is 80, and larger worksheets must be printed in sections, as was shown earlier in this chapter.

Sorting Labels and Numbers

Let us go to worksheet C6A (shown as Table 6.1 at the beginning of this chapter). We have the grade-book example with all the students' names and grades. Would it be possible to arrange the grades in a descending or ascending order, so we can see which student did the best on a given assignment or test? Fortunately, Multiplan provides us with this capability. The Sort command allows rearrangements. Let us proceed with an example.

Say we wanted to arrange the students listed in worksheet C6A in an ascending order of performance based on the grade they received in the eighth test, labeled as T8 in column 11. We would position the cell pointer at R2C11 and type

S

and we will see the prompt:

```
SORT by column:11
      between rows: 2   and:2   order:(>)<
```

Note the default answers given by Multiplan. (These options are underlined.) Since we planned ahead and positioned the cell pointer at column 11, Multiplan anticipated our wish.

Responding to the above by

```
[TAB]           (To get to the next field.)
[TAB]           (Move to the next field.)
22             (Enter rows to be sorted.)
```

We must be careful to indicate to Multiplan that the first and last rows are not to be included in the sort. Had this not been done, Multiplan could (and would) sort them as well. This would have caused considerable confusion since the last row, with relative references in the Average function, would have been placed in the middle, and the data would no longer be the correct averages.

When you press RETURN after checking all the responses given, Multiplan would execute the command and produce the following:

#1	1	2	3	4	5	6	7	8	9	10	11	12
	Name	H1	H2	T1	T2	T3	T4	T5	T6	T7	T8	
2	Eddy	50	52	45	67	76	62	56	55	58	58	
3	Jack	44	49	70	65	68	72	76	72	78	66	
4	Adam	56	67	87	72	66	45	69	78	88	70	
5	Charles	60	0	84	92	89	78	70	83	84	71	
6	Sara	98	66	88	69	45	88	74	51	66	72	
7	Jim	66	69	64	72	77	70	74	78	78	72	
8	Phil	88	71	49	94	77	59	88	77	50	73	
9	Fred	80	60	52	98	83	68	79	83	55	73	
10	Ronald	66	68	87	72	66	78	83	71	77	74	
11	Nicole	66	68	78	62	70	80	78	88	84	75	
12	Ruth	66	76	75	77	78	79	73	78	77	75	
13	Maria	75	78	67	72	77	80	82	78	85	77	
14	Bernice	85	95	72	88	66	74	80	74	69	78	
15	Mary	82	80	76	66	76	83	78	80	89	79	
16	Roger	69	73	80	71	76	90	81	84	88	79	
17	Sue	66	68	72	74	77	83	90	92	95	80	
18	Larry	88	81	78	73	77	78	80	79	84	80	
19	Alan	95	100	88	55	88	85	79	82	76	83	
20	Francine	66	78	88	92	88	86	80	85	88	83	
21	Patrick	90	87	86	78	75	81	78	90	94	84	
22	Helene	66	78	88	89	90	88	90	88	94	86	
23												
24	Average	72	70	75	76	75	77	78	78	79	76	
25												<C6B>
26												

If you wish to sort alpha characters, simply follow the above procedures, sorting column 1 (Name) this time.

Freezing Titles: The Window Command

When you are working with a problem so large that not all of it fits on the screen at one time, it is very easy to make a mistake. As you scroll far over to the right, information disappears from the left of the screen. Similarly, if you scroll down far enough, information at the top of the screen disappears. Suppose we have the following on our screen:

#1	1	2	3	4	5	6	7	8	
	1 Name	H1	H2	T1	T2	T3	T4	T5	
2	Alan		95	100	88	55	88	85	79
3	Bernice		85	95	72	88	66	74	80
4	Charles		60	60	84	92	89	78	70
5	Ruth		66	76	75	77	78	79	73

Suppose we now move the cursor from its position on column 1 over+ to the right, so that we can see column 8. If we do that, either by pushing the cursor arrow key → or by typing

```
G R 1 [TAB] 8 RETURN
```

then column 1 will disappear from view. The names in column 1 have been forced off the screen. This can be a very annoying situation. How can we avoid it?

The W command (W for Window) allows you to fix (or freeze) several rows or several columns, or both, in place. Suppose we wanted to ensure that the names in our example would always be on the screen? Typing

```
G R 2 [TAB] 2 RETURN
W                               (For Window.)
```

generates the prompt:

```
WINDOW: Split Border Close Link
Select option or type command letter
```

Responding with an S (S for Split) or RETURN displays the next prompt:

```
WINDOW SPLIT: Horizontal Vertical Titles
```

Responding with a V for vertical generates the next response:

```
WINDOW SPLIT VERTICAL  at column:2
                        linked:Yes (No)
```

Pressing enter at this point fixes those columns which are at and to the left of the cursor in the cell it presently occupies. In effect, the active cell now becomes the upper left corner of a new window. If you change your mind, type

```
W C           (Initiate Window, Close WINDOW #___)
```

The C (C for Close) undoes any previous use of this command. So, in order to freeze the names of column 1 in place, we can type

```
G R 2 [TAB] 2 RETURN
W S                               (Window Split.)
V RETURN                          (Vertical, activate.)
```

Now, if we scroll far enough to the right, instead of column 1 being forced off the screen, column 2 will be forced off instead, as shown in the next worksheet:

#1	1	#2	3	4	5	6	7
1	Name	1	HWK2	TST1	TST2	TOTAL	AV
2	Alan	2	100	80	85	360	
3	Bernice	3	95	73	94	347	
4	Charles	4	0	50	66	176	
5	Fred	5	60	70	78	288	
6	Mary	6	80	75	86	323	

Notice how column 2 HWK1 was forced off the screen. In the same fashion, we could have elected to freeze the labels of row 1 by typing

```
G R 2 [TAB] 2 RETURN
W S                (Window Split.)
H RETURN          (Horizontal, activate.)
```

Then all rows at or above the cursor position are frozen in place. So now, no matter how far down we scroll, row 1 will not disappear, and our worksheet will now look like:

#1	1	2	3	4	5	6
1	Name	HWK1	HWK2	TST1	TST2	TOTAL
#2	1	2	3	4	5	6
3	Bernice	85	95	73	94	347
4	Charles	60	0	50	66	176
5	Fred	80	60	70	78	288
6	Mary	82	80	75	86	323

Notice that row 2 (with Alan's grades) is temporarily off the screen. We now see as far down as row 21.

You can fix more than one row or column in place. How many rows or columns are fixed in place is determined by the position of the cursor when you type the W S command. As we shall see, this feature becomes extremely useful. You can fix both a set of rows and a set of columns in place, simultaneously. Note, however, that you cannot ask for too many window splits since you are limited by the size of your screen. Attempting to split more columns or rows than are visible on the screen will result in an error message: "Window will not fit." You can activate this command by typing

```
G R 2 [TAB] 2 RETURN
W S                (Window Split.)
T RETURN          (Title, use default to activate.)
```

Then all rows at or above the current cursor position and all columns at and to the left of the cursor will be fixed in place. This actually creates 4 windows: horizontal titles, vertical titles, and a new active screen with its vertical titles frozen. These are numbered #1, #2, #3, and #4, respectively; this is presented in the following table where both row and column 2 are hidden:

#1	1	#2	3	4	5	6	7
1	Name	HWK2	TST1	TST2	TOTAL	AV	
#4		#3					
3	Bernice	95	73	94	347		
4	Charles	0	50	66	176		
5	Fred	60	70	78	288		
6	Mary	80	75	86	323		
7							

As we stated earlier, you can override any consequence of a WS command by typing

W C (Close Window #__, nullify previous W S request.)

Windows must be closed one at a time by entering their number in response to the prompt:

WINDOW CLOSE window number :

However, all windows can be closed at one time by using

W C RETURN

Multiplan in Stereo: Link Screens

If you think of your screen as a window into a large workspace, you might like to have a larger window. The next best thing would be to have a pair of smaller windows. In fact, a pair of smaller windows may be even better than a single larger window. Multiplan will allow as many as eight different windows to be defined, linked, and used.

With Multiplan, you can set up many windows which will allow you to concentrate on those parts of your worksheet of immediate concern. You retain the usual freedom to browse, scroll, change values and labels, etc. You can invoke this capability by typing the now familiar command:

W S	(Window Split.)
V	(Vertical, or H for Horizontal.)
RC	(Enter desired # of R or C split.)
RETURN	(Vertical, activate.)

Typing H lets you split your screen horizontally at the current cursor position. Typing V gives you a vertically split screen at the current cursor position. You can split your screen only one way at a time vertically, horizontally, or both, simultaneously. However, you can split the screen further by initiating the W command again. Thus, you can split it vertically, next split it horizontally, and follow this with a third split, if need be. If you change your mind about which way to split it or where to split it, you can return to the normal single screen by typing C in response to another W request. Each W C closes the active window so that if you have a number of windows, you need to execute several W C commands.

Once you have split your screen, you can use any of the usual commands. If you change a value on one screen, then any values depending on it will change on all screens. Since you still have only one cursor to work with, typing CTRL-W will select the next screen. So you can type CTRL-W to move the cursor from one screen to the other.

The position the cursor had on one screen is saved when you type CTRL-W, so the next time you type CTRL-W the cursor will return to the same position it had previously.

You might prefer not having the cursor positions in each of your screens change independently. You might prefer the opposite, as you move the cursor around in one screen, that typing F1 would place the cursor on the corresponding row or column of the other screen. If so, you can select what is called synchronized scrolling by typing

W S	(Synchronized scrolling.)
H	(Horizontal.)
[TAB]	(Move to Link, on next field.)
Y	(Link activated.)
RETURN	

Notice in this case that the bottom window loses its column numbers, which are now read from the left-most window.

You can disable the synchronized scrolling by typing the command over, and after tabbing to the link field, selecting the No option. The *synchronized scrolling* mode selected by the W S T command is automatically selected when you initially use Title to split the screen. You cannot unlink the windows with the Window Link command. You must use the Window Close command.

You can save a file even when you are using split screens. You do so using the same TS command we saw earlier. All values will be saved, not just the visible ones.

If you try printing a file while using split screens, you probably won't print what you are looking at. For instance, suppose you have your screen split vertically, as shown below:

#1	1	2	3	#2	6	7	8
1				1			
2				2			
3				3			
4				4			
5				5			
6				6			
7				7			
...				...			
19				19			

If you try to print the segment defined by the corners R1C1 and R10C70, by typing

```
P
RETURN
```

your printout will include columns 4 and 5, even though they were not visible on your screen when you set out to do this printing. The Print command works on the single-screen interpretation of your worksheet. You can, however, use the Shift-PrtSc keys to capture the image of a split screen just as if you had a camera. You will, of course, also capture the worksheet row and column labels 1, 2, 3, . . . , 1, 2, 3, etc.

Summary

- Printing is performed using the P command. This command expects to receive answers to a series of prompts. The first asks if you want to print on the Printer or onto Disk (the File subcommand). Normally you will enter P or RETURN to generate a printed output.
- Titles or other information can be frozen in place using the Window command. We can lock into place all rows above a given row, using W S H. Similarly, all columns to the left of a given column can be locked into view using W S V. Frozen rows or columns can be unfrozen using W C.
- Split screen viewing can be requested using the same window command, followed with either vertical, horizontal (or both, using Title) split. The single cursor can be made to jump from one window to the other by pressing the F1 function key. A maximum of nine windows can be defined using successive window commands. Split screens can be unsplit by typing W C.

7

A Picture from Multiplan

Sometimes a picture conveys a result more clearly than a set of numbers might. You can direct Multiplan to produce some simple images or graphics, which is what we will do in this chapter. You can display these graphics on your screen; if you have a printer, you can print these graphics.

Getting Graphical Output

Some computers have elaborate equipment to produce very sophisticated graphical output. Multiplan assumes that all you have on your computer is an ordinary alphanumeric video display, and if you want graphics printed, an ordinary alphanumeric printer. That being the case, the kind of graphical output you can produce is the same kind you could produce on an ordinary typewriter: horizontal bar charts, also known as histograms. For instance, given the following data (Table 7.1):

Table 7.1 Data for Graphic Display

#1	1	2	3	4	5
1	YEAR	PRECIP			
2	1970		5		
3	1971		4		
4	1972		5		
5	1973		3		
6	1974		7		
7	1975		1		
8	1976		4		

We can arrange to produce one of the next two tables (either Table 7.2 or 7.3).

How can we get the results shown in Table 7.2? The F command (F for Format) allows you to specify a format for any desired position on the worksheet, and that format will affect the appearance of any value displayed in that position. Note the use of the word *appearance* in the earlier sentence. Underlying values will not be changed, so you can always change your mind about which format to use without worrying about losing anything. Typing F generates the prompt line:

FORMAT: Cells Default Options Width

Table 7.2 Graphic Display Alternative 1

#1	1	2	3	4	5
1	YEAR	PRECIP			
2	1970	*****			
3	1971	****			
4	1972	*****			
5	1973	***			
6	1974	*****			
7	1975	*			
8	1976	****			

Table 7.3 Graphic Display Alternative 2

#1	1	2	3	4	5
1	YEAR	PRECIP			
2	1970		5 *****		
3	1971		4 ****		
4	1972		5 *****		
5	1973		3 ***		
6	1974		7 *****		
7	1975		1 *		
8	1976		4 ****		

After selecting one of these, say C, the prompt line shows:

```

FORMAT  cells:  alignment:(Def)C G L R -
        code:(Def)Cont Exp Fix Gen Int $ * % -
        # of decimals:
    
```

Enter reference to cell(s)

The cells field defines the worksheet area to be affected by the Format command. The second field, alignment, highlights the various display alternatives. The format field allows changes to the displayed format. If you do not wish to alter the alignment, tab directly to the Format field. If you have a certain alignment designated and you wish to retain it, yet you wish to alter the format, enter the hyphen from the alignment field. Conversely, if you wish to change the alignment, but not the format, select the hyphen from the format field. Each of the options is described below:

The Alignment Field Shows

- Def Aligns as specified in the Format Default command.
- Ctr Centers cell entries.
- Gen (General) Aligns text left, numbers right.
- Left Aligns cell entries to the left.
- Right Aligns cell entries to the right.
- Does not change alignment.

The Format Field Shows

- Def Format as specified by the Format Default command.
- Cont Continues long text across column border, if next cell empty and also formatted "Cont".

Exp	(Scientific notation) Displays numbers as a decimal times a power of ten. Number of digits as specified in "# of decimals."
Fix	Rounds to and displays number of decimals specified in "# of decimals" field.
Gen	(General) Displays numbers in the most appropriate form considering the size of the cell and of the number.
Int	Numbers will be shown as integers after rounding.
\$	Displays numbers with leading dollar sign and 2 decimal places (negative numbers enclosed in parentheses).
*	(Bar Graph) Displays as many asterisks as the value of the number.
%	(Percent) Displays the number times 100 followed by a percent sign.
=	Does not change format.

The Third Field (# of Decimals) Shows

of decimals: This last field, (# of decimals) is used in the Exp, Fix and % formats above. The maximum number of decimals is 15.

We will, of course, examine and discuss each of these format commands as we go along. (See Chapter 9 for in-depth treatment.) For the time being we shall concentrate on graphs.

Selecting the * (asterisk) response will cause the worksheet position at which the cursor is currently situated to display zero, one, or more asterisks, depending on the rounded value at that position. If the value is less than one, then no asterisks will appear. The number of asterisks displayed is determined by truncating the value (for display purposes only; the underlying value remains intact). That simply means any fractional part of a number will be ignored.

Asterisks will be displayed up to the width of the column less one character. If the position is not wide enough to display the appropriate number of asterisks, the column will be filled with ##### to indicate an overflow. The column must be made wider to allow complete display of the correct number of asterisks. We will soon see how we can enlarge (or shrink) the width of our columns. In the meantime, note that the default width of a column is ten characters. When displaying numbers or asterisks, one of the ten character positions will always be left blank so that you can see where the columns are separated.

To obtain the result in Table 7.3, we should begin with a worksheet which has the appearance as shown in C7A. We would like asterisks in place of the numbers of column 2 (that is the only choice you get, asterisks or nothing). So let us move the cursor to R2C2 and request the * format, by typing

```
G R 2 [TAB] 2 RETURN
F                               (Format entry.)
RETURN                          (Use Cell option, could also type C.)
C2                               (Request Column 2 formatting.)
[TAB]                           (To Alignment field.)
[TAB]                           (To Format field.)
*
RETURN
```

This will result in:

#	1	2	3	4	5
1	YEAR	PRECIP			
2	1970		5		
3	1971		4		
4	1972		5		
5	1973		3		
6	1974		7		
7	1975		1		
8	1976		4		

<C7A>

This will enter the asterisks; it will also erase your stored values. If you prefer restoring the usual numeric display, then you can restore the default format which is in effect when you start with a new worksheet. Typing

```
G R 2 [TAB] 2 RETURN
F                               (Format entry.)
RETURN                          (Use Cell option, could also type C.)
C2                               (Request Column 2 formatting.)
[TAB]                            (To Alignment field.)
[TAB]                            (To Format field.)
D                               (Set Default format.)
RETURN
```

restores the original display format.
Table 7.3 can be obtained by typing

```
G R 2 [TAB] 3 RETURN
F C                               (Format column #3.)
3
[TAB]
L                               (Ask for Left.)
[TAB]
*                               (Ask for * format.)
RETURN                          (Use * format.)
V                               (Start Formula.)
←                               (Use R2C2 as data source.)
RETURN                          (Activate R2C2 as the formula for R2C3.)
C D                               (Use R2C3 as source.)
7 RETURN
```

We immediately see the results as shown in Table 7.3, which is repeated here:

#1	1	2	3	4	5
1	YEAR	PRECIP			
2	1970		5 *****		
3	1971		4 ****		
4	1972		5 *****		
5	1973		3 ***		
6	1974		7 *****		
7	1975		1 *		
8	1976		4 ****		<C7B>

It is worth repeating that using the F display format does not change any values you have recorded. As you move the cursor over any positions with asterisks, you can see the original value intact and displayed on the status line below your worksheet.

Changing Column Widths

The standard column width is 8 characters. You can change the width of your columns by using the Format command F. It will prompt you:

```
FORMAT: Cells Default Options Width
```

Respond with W, and see the next prompt:

```
FORMAT WIDTH in chars or d(efault): d
           column:           through:
```

Enter [TAB] followed by a number, then push RETURN, if you like the rest of the defaults.

Selecting the D response lets you change the width of all columns in your worksheet. You type a D followed by W and the desired width, and push RETURN. If you wanted your columns to be 15 characters wide, instead of the normal 8 characters wide, type

```
F D
W 15
RETURN
```

The narrowest column width you can request is 3 characters wide; this lets you see columns 1 through 25, simultaneously. The widest column width is 32 characters. If you ask for the 32-column width, you will only see one column on your screen.

You can also change the width of a single column by using FW, which will prompt you for a column width and range. Respond with the desired column width followed by a RETURN. Type F and see:

```
FORMAT: Cells Default Options Width
```

After selecting W, you will see:

```
FORMAT WIDTH in chars or d(efault):d
           column:1           through:1
```

Enter the desired number of characters, followed by RETURN. If you desire, each column in your worksheet can be formatted for a different width.

Long Titles

If you have titles wider than your column widths, they will not be displayed past the existing width. To get the title displayed on the first row, first 10 columns, you need the command:

```
G R 1 [TAB] 1
RETURN
F C           (To activate.)
R1C1:10      (Enter desired range.)
[TAB] [TAB]  (Move to Format Codes field.)
C RETURN     (Activates the Continuous command.)
```

Now your titles will extend across adjacent columns until they encounter a column with an entry in it. When you delete a column including any part of the title except the beginning, Multiplan will not delete any part of your title. The title will be extended across all remaining cells which were not used.

This particular feature is extremely useful when entering titles extending throughout several cells. You should be certain to leave the columns adjacent to the beginning of the title blank. The Continuous command is usually applied to rows.

Error Flag

If you should happen to see a string of pound signs, as in:

```
#####
```

fill up a worksheet position, this indicates that the value at this position cannot be displayed because the column is too narrow. It has no effect on any calculations, but it serves as a reminder that perhaps you should increase the column width, say to the value *n*, by typing F D W *n*.

Scaling for Graphs

Not all tables of numeric information can lead to a manageable bar graph. For instance, suppose you had the following table:

#1	1	2	3	4	5
1	Date	Market			
2	Q1	990			
3	Q2	940			
4	Q3	890			
5	Q4	920			<C7C>

You would not want to have bars with over 800 asterisks printed. For openers, you could set up a worksheet that would omit the first 850 asterisks from each line. But this is still not manageable because the “spread” or “range” in this example is 100 (the largest number is 990 and the smallest is 890). Suppose you wanted to display no more than 20 asterisks per line. You could begin by setting the column width to 21, by typing

```
F W
21 [TAB]
3 [TAB]
3
RETURN
```

(If you want as many as 20 characters to be displayed, the column must be 20+1 characters wide.)

You want either to stretch or squeeze the numbers you are working with so the display can fit in the space you have designated. This process of stretching or squeezing is called *scaling*. Here you want to find the largest value (the maximum value) and the smallest value (the minimum value) so you can know what the range, or spread, is. Since you expect to save this worksheet and change the quarterly market values, you need a formula that finds the maximum and minimum values for you. Once again, we get to use functions.

MAX and MIN

The function MAX(argument list) will choose the largest value associated with the items in its argument list. Similarly, MIN(argument list) will select the smallest value.

In our market example, we can use column 3 to store computed values which will drive our graphical display. Prior to writing down a formula, let us take a closer look at the numbers we are dealing with: 990, 940, 890, and 920.

The largest number here is 990. The smallest is 890. So the range (largest–smallest) is 990–890, or 100. We don’t want to display as many as 100 asterisks, so let us choose a squeezing or scaling factor, say 10. For every increase in value of 10, we will display 1 asterisk. In order to give the bar graph an

appearance of being the top part of a full-size chart, let us add in a bias so we have a solid base to look at. A bias of 5 would not be unreasonable here. We can see what that would do to each of our numbers:

Value	Bias + Spread/Scale	= Result
990	$\rightarrow 5 + (990 - \text{MIN})/10 \rightarrow 5 + 100/10$	= 15
940	$\rightarrow 5 + (940 - \text{MIN})/10 \rightarrow 5 + 50/10$	= 10
890	$\rightarrow 5 + (890 - \text{MIN})/10 \rightarrow 5 + 0/10$	= 5
920	$\rightarrow 5 + (920 - \text{MIN})/10 \rightarrow 5 + 30/10$	= 8

The corresponding formula is

bias + (current value-min)/scale factor.

If we agree to use a bias of 5 and a scale factor of 10, this leads us to the formula

$5 + (\text{current value} - \text{min})/10$

If we plan to use column 3 for the bar graph, we can begin by selecting the * format for that column

F	(Format entry.)
RETURN	(Use Cell option, could also type C.)
C3	(Request column 3 formatting.)
[TAB]	(To Alignment field.)
L	(Request Left display.)
[TAB]	(To Format field.)
*	(Set * format.)
RETURN	

We must not forget to enlarge the column width from its normal size of 9 characters to, for example, 21 characters. We can do this by typing

F W
21 RETURN

Then we can define the formula for position R2C3. Note that to copy that formula into R3C2 to R5C2, care should be used in using relative and absolute references. We type

G R 2 [TAB] 3 RETURN	(Goto R2C3.)
5 +	
(← -	RC[-1] -
MIN (R2:5C2)	
) / 10	(Complete formula.)

We can now replicate both the format and the formula we placed in R2C3.

G R 2 [TAB] 3 RETURN	
C D 3	
RETURN	(Copy from R2C3.)

The fruits of our labor are shown below:

#1	1	2	3	4	5
	1 Date	Market			
2	Q1	990	*****		
3	Q2	940	*****		
4	Q3	890	*****		
5	Q4	920	*****		<C7D>

You could make it easier to modify this display for other values by making the bias (currently equal to 5) and the scaling factor (currently 0.1, the inverse of 10) variables instead of constants.

Since no columns beyond column 3 are being used, you could use position R1C4 to store the bias, and position R2C4 to store the scaling factor. Then the formula would become

$$R1C4 + R2C4 * (R2C2 - \text{MIN}(R2C2:R5C2))$$

It is now easy to try out various scaling factors and biases without retyping lengthy formulas. You need only change the scaling factor in R2C4 and change the bias in R1C4.

There is an easier method yet. Suppose you named column 1 as Date, column 2 as Market, column 3 as Graph, and columns 4 and 5 as Bias and Scale, respectively. Assume further that we have just entered these names. The formula would be as simple as

```
G R 2 [TAB] 4 RETURN
V
bias + scale*(market-min(market))
```

You could now choose whichever of the three graphing methods and formulas that makes you happy.

IRAs Revisited

An IRA chart can be prepared using a slight variation of the IRA formula we saw back in Chapter 4. Once again we will use yearly interest rates, with interest computed once per year. If you want to compute your interest quarterly, monthly, or otherwise, see Case Study B (Chapter 12) dealing with interest calculations. This IRA chart gives us the opportunity to use several of the Multiplan facilities in combination. We will build this worksheet one step at a time.

The last thing we will do is adjust the column width to allow room for a graph. It is easy to change the column width at any time, and trying to build a worksheet while using wide columns means you have to do a lot of scrolling for no good purpose. So let us begin by laying out the labels we want, and a few initial values, to obtain the following worksheet.

#1	1	2	3	4	5	6	7
1				IRA	GRAPH IN		
2	RATE	CONTRIB	YEAR	VALUE	\$20,000		
3	12	2000	1				
4		?	?		?		
5		?	?		?		
6		?	?		?		<C7E>

Question marks appear where we want Multiplan to fill in the results. Instead of replicating the \$2,000 per year contribution directly, we can place the formula R[-1]C in position R4C2, then replicate it into R5C2, R6C2, up to R32C2 so that each year's contribution is a copy of the preceding year's entry. That way, we could easily assess the impact of reducing the contribution from any point on. The typing for column entries is

```
G R 4 [TAB] 2 RETURN
V
↑
RETURN
C D 29
RETURN
```

We can fill in the Year column, column 3, as follows

```
G R 4 [TAB] 3 RETURN
V
↑
+
1
RETURN
C D (29 remained from previous copy.)
RETURN
```

For column 4 we have

```
G R 3 [TAB] 4 RETURN
( ← ← (Enters "RC[-2]").)
) * (Closes reference, multiplies.)
( 1 + (Starts second argument.)
( R3C1 (Uses absolute reference of rate.)
/ 100 (Converts to percentage.)
) )
RETURN (Activates formula.)
```

Since we want the first year's contribution and interest added to the second year's contribution, we must add R3C4 to R4C2, in the following year as well as in the years after that. We type

```
G R 4 [TAB] 4 RETURN
( ← ← (Enters "RC[-2]").)
+ ↑ (Enters "R[-1]C".)
) * (Closes reference, multiplies.)
( 1 + (Starts second argument.)
( R3C1 (Uses absolute reference of rate.)
/ 100 (Converts to percentage.)
) )
RETURN (Activates formula.)
C D 28 (Copy Down.)
RETURN
```

Getting a graph into column 5 goes as follows. We first format the column, then we can replicate the formulas of column 4 into column 5. Then we assign the very simple formula IRA/20000 to position R3C5 and copy it down 29 rows. Here is the formatting.

```
G R 3 [TAB] 5 RETURN
F C (Format Cells.)
C5 (Column 5.)
[TAB] L (Justify Left.)
[TAB] (Move to Format.)
* RETURN (Request display of *'s.)
```

This completes the formatting. Now type the formula

```
V
IF
( IRA /
20000
<1,0,INT
(IRA /
20000 ) )
```

The number 20,000 is used to shrink or scale our dollar values into numbers small enough to be represented by no more than 20 to 30 asterisks each. Thus each asterisk represents an increment of \$20,000. The IF function was used in the above formula. A detailed explanation of this and other functions appears in the following chapter (Chapter 8).

No asterisks are displayed until we reach year 7, when the accumulation with interest first exceeds \$20,000. When we reach year 21, and from then on, we will only see 9 #####, indicating an overflow, since all values from year 21 onward exceed \$180,000 and the default column width is 10 columns with 9 columns showing data.

Table 7.4 Graph of Your IRA

#1	1	2	3	4	5
1	RATE	CONTRIB	YEAR	IRA VALUE	GRAPH IN \$20,000s
2	12	2000	1	2240	
3		2000	2	4748.8	
4		2000	3	7558.656	
5		2000	4	10705.695	
6		2000	5	14230.378	
7		2000	6	18178.023	
8		2000	7	22599.386	*
9		2000	8	27551.313	*
10		2000	9	33097.47	*
11		2000	10	39309.167	*
12		2000	11	46266.267	**
13		2000	12	54058.219	**
14		2000	13	62785.205	***
15		2000	14	72559.429	***
16		2000	15	83506.561	****
17		2000	16	95767.348	****
18		2000	17	109499.43	*****
19		2000	18	124879.36	*****
20		2000	19	142104.88	*****
21		2000	20	161397.47	*****
22		2000	21	183005.17	*****
23		2000	22	207205.79	*****
24		2000	23	234310.48	*****
25		2000	24	264667.74	*****
26		2000	25	298667.87	*****
27		2000	26	336748.01	*****
28		2000	27	379397.77	*****
29		2000	28	427165.51	*****
30		2000	29	480665.37	*****
31		2000	30	540585.21	*****
32		2000			
33					

You can now set the column width as you wish. Let's set column 5 to 30 characters wide

```
F W  
30 RETURN
```

This will generate the graph you see in Table 7.4. Obviously, you will be able to see it only partially on the screen.

To be able to see as far down as year 30, we run the risk of having our top rows scrolled off the top of the screen. We can avoid that by freezing these titles into place using the WT command we saw in Chapter 6.

Summary

The F C [TAB] L [TAB] * RETURN command displays as many asterisks as are indicated by the integer part of the value associated with the current cell. The integer value is obtained by truncating a copy of the position's value. The original value is not changed; only the display is affected.

You can "turn off" the effect of the above (*) command by replacing the * with D(efault): F C [TAB] D [TAB] D.

Column widths can be changed as desired by typing F W <width>, when <width> is the number that indicates the desired column width; it must be at least 3 and no more than 32. If you wish to change the width of several columns, tab to the next field in the F W command and change the defaults accordingly.

is displayed when a column is too narrow to properly display the value it contains.

The MAX (argument list) and MIN (argument list) functions select the largest and smallest values in their respective argument lists.

8

More Complex Calculations

We have seen some of the functions supported by Multiplan, such as AVERAGE, SUM, MIN, and MAX. There are several other built-in functions provided. This chapter introduces many of them.

Some of these new functions are very simple; for instance,

COUNT(argument list)

generates the number of numeric entries in the locations or the values in its list. You could use it to find out how many students took a particular exam, or how many customers have sent in a payment toward their outstanding balances.

If you had in row 1, beginning at R1C1

#1	1	2	3	4	5	6
1	12		24		13	

COUNT(R1C1:5) would return the value 3.

All the functions we have seen so far can examine several items in their argument lists. The next set of functions allows only a single item as their argument. This item may be a worksheet coordinate; it can also be a formula.

The first of these functions is ABS. ABS(arg) returns the absolute value of its argument. That is, it strips away the arithmetic sign, so that its result is always positive.

Example: for

#1	1	2	3	4	5	6
1	77	-33	0			

ABS(R1C1) returns 77

ABS(R1C2) returns 33

ABS(R1C3) returns 0

The function INT(arg) ignores the fraction part of the value associated with its argument "arg", and it returns the integer part of the argument as its result.

Example: for

# 1	1	2	3	4	5	6
1	2.1	2.9	-3.8			
INT(R1C1)	returns 2					
INT(R1C2)	returns 2					
INT(R1C3)	returns -4					

The function SQRT(arg) returns the square root of its argument if the argument value is not negative.

Example: for

# 1	1	2	3	4	5	6
1	0	4	-4	2		
SQRT(R1C1)	returns 0					
SQRT(R1C2)	returns 2					
SQRT(R1C3)	returns #NUM!					
SQRT(R1C4)	returns 1.414214					
SQRT(ABS(R1C3)+1)	returns 2.236068					

The square root of the value stored at R1C3, -4, is not defined, since that value is negative. The result "#NUM!" or "#DIV/0!" will be displayed whenever impossible computations are requested.

Mathematical and Engineering Functions

Many of the functions used in mathematical and engineering problems are available in Multiplan. These functions are

EXP(arg)	Returns the arg power of the natural number e.
Example:	EXP(1) returns the value 2.718282 ("e" to the power 1).
LOG10(arg)	Returns the base 10 logarithm of its argument. N must be > 0
Example:	LOG10(100) returns the value 2; LOG(-100) returns #NUM! error message.
LN(arg)	Returns the base e logarithm, or natural log of its argument.
Example:	LN(100) returns 4.605170. LN(2.718282) returns 1.
INT(arg)	Returns the largest integer that is less than or equal to n.
Example:	INT(154) returns 154; INT(-33.99) returns -34; INT(7.8) returns 7.
MAX(list)	Returns the largest of the values referred to. If list contains no number values, returns 0.
Example:	MAX(45,678,786,342) returns 786.
MIN(list)	Returns the smallest of two or more values.
Example:	MIN(45,678,786,342) returns 45.
MOD(n,m)	Returns the remainder of the integer division n/m.
Example:	MOD(3,2) returns 1; MOD(3,-2) returns -1.
ROUND(n,m)	Returns n rounded to m decimal places. If m < 0 rounding is carried to 1+ABS(m) decimal places.
Example:	ROUND(45.987655,5) returns 45.98766, ROUND(991,-2) returns 1000.
SIGN(n)	Returns -1 if n<0, 0 if n=0, 1 otherwise.
Example:	SIGN(-456) returns -1; SIGN(456) returns 1; SIGN(0) returns 0.

Trigonometric Functions

The usual complement of trigonometric functions is supported by Multiplan. For each of these, the argument is expressed in radians. Recall that 2π radians equals 360 degrees. The names used for these functions are

ATAN SIN COS TAN

These names refer to the arc-tangent, sine, cosine, and tangent functions. The arguments must be expressed in radians.

Other Functions

There are three additional functions which have no arguments, and several which do:

COLUMN NA ROW ISERROR() ISNA() PI() TRUE() FALSE()

COLUMN returns the number of the column containing a formula with COLUMN. For example, if in column 4 you type

```
G R 1 [TAB] 4 RETURN
V
12+COLUMN()
RETURN
```

you will see the number 16 in R1C4. You could also use the function to generate a sequence of numbers.

ISERROR(n) returns True if (and only if) n is an error value; otherwise, it will return the value False. You can use ISERROR to test very complicated sequences of formulas. The error value in n could be any kind of error, resulting from a variety of sources. #N/A, #VALUE!, #REF!, #DIV/0!, #NUM!, #NAME?, AND #NULL! are the error messages which will generate a true response. Otherwise, false will be returned.

The function NA() should be used when data is not available. That alerts formulas that may be referencing empty positions to ignore them. For instance, if you were using AVERAGE(R1:9C1) at R10C1, and had not yet filled in any values for R1C1 through R9C1, then position R10C1 would display the result #N/A! because Multiplan alerts you to the fact that there are undefined names in R1:R9C1. So filling unused positions with the "value" NA() would cause results which depend on these positions to be displayed as "#N/A", which is somewhat less alarming than seeing the message "#NAME!" In short, NA() returns the value #N/A. ISNA(n), on the other hand, returns True if (and only if) n is an #N/A value.

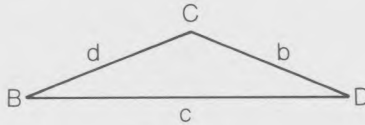
Suppose the coordinates R1C45, R2C45, and R3C45 are each supposed to depend on a value at R1C46, among other values. You can verify this dependency experimentally by giving R1C46 the value NA(), which will generate a string of #N/A through all the positions depending on the value at R1C46. In particular, positions R1C45, R2C45, and R3C45 should display the result #N/A. If they do not, you know something is wrong. Of course this check does not prove you used the correct formula; it only shows that you did or did not reference R1C46 in providing the formulas for R1C45, R2C45, and R3C45.

The function PI returns the value 3.141593 which you know under the name pi (pronounced pie) in the famous formula $2 * \pi * r = \text{circumference}$, for a circle with radius r.

Using Multiplan as a Programmable Printing Calculator

There are calculations that you may be performing using a calculator that could be performed using Multiplan. Why consider using Multiplan instead of a calculator? It might be helpful to get a hard-copy record you may not get from a nonprinting calculator. Also, it may be easier for you to show another person how to enter the data using Multiplan than might be the case with some calculators.

For instance, suppose you often need to compute triangular relationships. Specifically, you may need to know how long a side of a triangle is, given the lengths of its other two sides and their opposing angles. Using the names b, c, and d to stand for the lengths of sides b, c, and d, and letting the opposing angles be designated as B, C, and D,



we can use the relationship

$$d = b * \text{COS}(C) + c * \text{COS}(B)$$

to find the length of side a, given the lengths b and c, and the angles B and C. We can set up a worksheet as follows:

#1	1	2	3	4	5
1	Side		3	3	
2	Angle	1.05	1.05		
3					<C8A>

for a triangle with sides b and c of length 3, and opposing angles B and C of 60 degrees. Here we are entering the angles in radian measure and have converted them from degrees to radians by multiplying the number of degrees by pi/180, which is .0174533. We can now enter the formula for the length of side d, in position R1C4.

```
G R 1 [TAB] 4 RETURN
R1C2 * COS(R2C3) + ( R1C3 * COS(R2C2) ) RETURN
```

The spaces in the formula will be ignored if you type them; they are present here for ease of reading.

We can build in the conversion from degree measure to radian measure by including it in the formula for R1C4. The formula would then read

```
R1C2 * COS(R2C3 * PI()/180) + ( R1C3 * COS(R2C2 * PI()/180) )
```

You could replace the expression PI/180 by its numeric value, if you prefer, .0174533, which leads to the following worksheet:

#1	1	2	3	4	5
1	Side		3	3	3
2	Angle	60	60		
3					<C8B>

This lets us express the angles in degrees. The previous worksheet expected angles to be given in radians.

Printing Formulas

When you save a worksheet using the T S command, the formulas for that worksheet are saved with it. So far, the only way you can see an old formula is to load the worksheet with which it was saved by using the T L command. Then you can place the cursor over the position associated with the desired formula and see it displayed on the Entry Content Line below your worksheet. Is it possible to have formulas printed for future reference? Yes, you can do this by typing P O which will prompt you:

```
PRINT OPTIONS: area:          setup:
                formulas: Yes (No)      row-col: Yes (No)
```

If you tab two times to the "formulas": field and enter Y or y, Multiplan can then print all the formulas.

```
#1      1          2          3          4          5          6          7
1 Side          3          3          R1C2 * COS(R2C3 * PI()/180)+ ( R1C3 *
                                           COS(R2C2 * PI()/180) )
```

Faster Computing

You have become accustomed to getting immediate recalculation of the current worksheet entries whenever you change any entry. There are situations where you would prefer to have the recalculation deferred until you had changed several entries. For instance, in the triangle computation we just looked at, you usually don't want the result for side d until you have typed in all the data. In some larger problems, a recalculation may take several seconds, and you can't be typing in new entries during that time. So you could save time by having the recalculation postponed. How do you do this?

Typing O (Options) generates the following response:

```
OPTIONS recal: (Yes) No      mute: Yes (No)
      iteration: Yes (No)    completion test at:
```

In this case typing N or n in the first field will delay all recalculations until you wish to have the recalculations performed. In this mode, you can type as many entries as you wish without any calculations being done. When you finally want the worksheet recalculated, you request this action by typing an "=".

You can resume the usual automatic recalculation mode by typing O (Options), which again generates the following response:

```
OPTIONS recal: Yes (No)      mute: Yes (No)
      iteration: Yes (No)    completion test at:
```

except this time you will type Y or y (for yes).

String Functions

Multiplan provides us with several string functions which help in reducing typing. These include

DOLLAR(n)	Converts text string showing the value n in \$ format.
Example:	DOLLAR(3.872) returns \$3.87; DOLLAR(-1) returns \$1.00; DOLLAR(.563) returns \$0.56.
FIXED(n,m)	Converts the value to text containing n in Fix format with m decimals. When m=0 converts to Int format.
Example:	FIXED(345687,3) returns 345.687. Though same as INT(), FIXED(-34.9,0) returns -34(!).

LEN(t)	Returns the length of the text (t) in characters.
Example:	LEN(Multiplan) returns the value 9; also LEN(Personal Computer) returns 17 (16 plus 1 space).
MID(t,n,m)	Returns m characters starting at beginning (n) of text t.
Example:	MID("COMPUTER",0,7) returns the text "COMPUTE"; MID("FFFFFFDDCBA",INT(grade/10),1) would be useful in the gradebook example. If the value of the grade is 50, then INT(50/10) would return 5, returning the 5th character "F".
REPT(t,n)	Reproduces text t, n times. Can be used to create bar graphs.
Example:	REPT("#", grade), entered into a group of cells which parallel a column named grade, will produce a bar chart consisting of "#". Be sure to make the column sufficiently wide.
VALUE(t)	If text t contains the representation of a numeric constant, its value will be returned.
Example:	If you had a column called date, with dates entered in the form: 8/9/83, then the function VALUE(MID(DATE,3,1)) would return the value 9.

Logical Functions

The functions we have been using either produce a numeric result or one of the error messages. You can use IF, TRUE, FALSE, OR, AND, and NOT to construct logical comparisons. We will examine IF in this section. The general form of the IF

IF (conditional value, alternative value 1, alternative value 2)

The IF always has exactly three arguments. The first argument is the conditional value, used as a standard, which should turn out to be TRUE or FALSE. The other two arguments, "value 1" and "value 2," are alternative outcomes. When the "conditional value" turns out to be TRUE, then IF produces the value of "value 1." When the "conditional value" turns out to be FALSE, IF returns the value of "value 2." Values 1 and 2 can take on either text or arithmetic values. So

IF(X1,R1C1,R1C2)	produces R1C1's value when X1 is TRUE
IF(X1,R1C1,R1C2)	produces R1C2's value when X1 is FALSE

You can think of an IF(condition,first,second) being read as: "If this condition is true, then use the first thing, else use the second thing." The "first" and "second" things can be as complicated as you want them to be, as long as they make sense as arithmetic expressions, if in fact they are numeric. So we could have the following:

IF(R1C12 < R4C11+1, R1C34, SUM(R1:R15))

Each side of the comparison operator "<" is evaluated, then the two values are compared, and if the value of R1C12 is less than the value of R4C11 plus 1, the IF selects R1C34's value. Otherwise it will evaluate the last argument SUM(R1:R15) and produce its value.

The comparison operators we can use are

Symbol	Name.
<	Less than.
>	Greater than.
=	Equal to.
>=	Greater than or equal to.
<=	Less than or equal to.
<>	Not equal to.

You can now handle such complex situations as—an employee works no more than 40 hours a week at the normal pay rate; for hours in excess of 40 hours per week the normal rate is increased by a specified factor. Suppose the factor we use for overtime pay is 1.5, and we have a pay formula which can be expressed as

$$\text{hours} * \text{rate} + (\text{if hours} > 40, \text{multiply extra hours by } .5)$$

Let us build upon this example by using a small business, with 9 employees and different pay rates as shown next:

#	1	2	3	4	5	6
	Name	Hours	Rate	Pay		
2	Joe	40	\$5.00			
3	Sam	50	\$5.00			
4	Mary	55	\$5.00			
5	Dora	35	\$7.00			
6	Bill	50	\$4.50			
7	Harry	34	\$5.50			
8	Joan	50	\$6.00			
9	Mike	45.5	\$5.00			
10	Roger	52	\$5.50			
11						
12			Total			<C8C>

As a Multiplan formula, using the data portrayed in the coordinates we see above, we can express the pay for position R2C4 as

$$RC[-2]*RC[-1]+IF(RC[-2]<40,0,(RC[-2]-40)*RC[-1]*.5)$$

Alternatively, using names of columns, we can restate the formula

$$HOURS*RATE+IF(HOURS<=40,0,(HOURS-40)*RATE*.5)$$

After entering this formula in position R2C4, we can copy it

```
G R 2 [TAB] 4 RETURN
C D 9
RETURN
```

Then we can obtain the total pay for R12C4.

```
G R 12 [TAB] 4 RETURN
V
SUM({:                               (R[-1]C:
↑↑
↑↑↑
↑↑↑
↑
)
RETURN
```

All we have to add is the dollar format, and provide for columns that are sufficiently wide to show the data. Let us first enter the dollar format

```
F C
R1:12C3:5
[TAB] [TAB]
$
RETURN
```

which produces this table:

#1	1	2	3	4	5	6
	1	Name	Hours	Rate	Pay	
	2	Joe	40	\$5.00	\$200.00	
	3	Sam	50	\$5.00	\$275.00	
	4	Mary	55	\$5.00	\$312.50	
	5	Dora	35	\$7.00	\$245.00	
	6	Bill	50	\$4.50	\$247.50	
	7	Harry	34	\$5.50	\$187.00	
	8	Joan	50	\$6.00	\$330.00	
	9	Mike	45.5	\$5.00	\$241.25	
	10	Roger	52	\$5.50	\$319.00	
	11					
	12		Total	\$2357.25		<C8D>

What if, in using IF, its first argument does not have a logical value? If the argument's value is NA(), then the result of the IF is also #N/A! Any other nonlogical value will produce the appropriate error message.

Other Logical Functions

The other logical functions are NOT, OR, AND, TRUE, and FALSE. The NOT needs a single logical argument. It returns these values:

Use	Result
NOT(TRUE)	1
NOT(FALSE)	0

So if you have the formula NOT(R1C1 < R1C2), the result should be "1" whenever R1C1 is greater than or equal to R1C2, because ">=" is the complement or opposite of "<".

The two logical functions OR and AND each allow you to specify two logical arguments. OR(larg1, larg2) returns "1" if one or both of its logical arguments larg1, larg2 is "TRUE." AND(larg1, larg2) returns "0" unless both of its logical arguments are "TRUE."

The TRUE and FALSE functions can also be used for testing in complex formulas. The function FALSE returns the logical value false, while true returns the logical value true. You can combine as many logical functions as you wish—as long as they make sense.

NPV: Net Present Value Function

The time value of money has always been an important consideration. This is even more so when double-digit interest rates are the norm. The NPV function calculates the present value of a series of future amounts FA1, FA2, . . . , FA_n, for years 1, 2, . . . , n, at an assumed rate of return *i*. The general form in using NPV is

NPV(rate *i*, range of locations for future amounts)

The assumed rate of return i is specified as a decimal number; so 10% is written as .10. Let us examine a few simple cases. They are simple enough to be mentally verified. If we assume a rate of return of 10% per year (compounded annually), we should expect that a payment of \$110 one year from now should be worth \$100 today. We can verify this by typing

```
G R 1 [TAB] 1 RETURN
110                               (year 1 amount in R1C1)
→
.10                               (rate in R1C2)
→
NPV(R1C2, R1C1:R1C1)           (result in R1C3)
RETURN
```

Suppose we expected payments of \$110 at the end of year one and \$121 at the end of year two. What is the present value of such a series of payments if we continue to assume the rate of return is 10%? If we modify our formula in R1C3 to include both the locations for the values 110 and 120, we should see

	1	2	3
1	110	.1	199.17355
2	121		

on the worksheet, where the R1C3 formula is NPV(R1C2,R1C1:R2C1).

You can place the list of future amounts in any desired row or column. The assumed rate of return uses simple annual discounting. Since the principal use of NPV involves estimating present value based on an educated guess as to what average interest rates will prevail for one or more years, there is not much point to using fancier interest compounding formulas. For a range of values v_1, v_2, \dots, v_n if we let d be $1 + \text{rate of return}$, the net present value is calculated using the formula

$$v_1/(d)^1 + v_2/(d)^2 + \dots + v_n/(d)^n$$

LOOKUP

The LOOKUP function lets you select the i 'th item from a table based upon a search argument that need not have the value " i ." The table consists of two adjacent rows (or two adjacent columns). The top row (or left column) is called the search list. The row below (or column on the right) provides the search values. You provide LOOKUP with two arguments as in

LOOKUP(arg, table).

The first argument "arg" is used to examine the search list. The second argument, "table," provides the range which indicates the first and last locations of the search list. For example,

#1	1	2
1	Search	Search
2	List	Values
3	10	31
4	15	2
5	25	-3
6	45	14

The "table" we have here fills R3:6C1 and R3:6C2. If we want to use it with LOOKUP, we will specify the range R3:6C1 for LOOKUP's second argument. Whatever value is provided for LOOKUP's first argument, the "search argument" is used to search the "search list." If we call the search argument x , then if x is less than the smallest search list value (here, the smallest search list value is 10), LOOKUP

will return the value #N/A! If x has the value 10, or greater than 10, but less than the next search list value 15, LOOKUP returns the result chosen from the corresponding search value position next to the 10. In this case a value of x between 10 and 15, possibly equal to 10, but less than 15, will have LOOKUP return the search value from row 3, which is the number 31.

In the general case, where we have a search list such as this:

$$l_1, l_2, l_3 \dots, l_n$$

the use of LOOKUP requires that these values be in ascending order. Since only the first of any duplicated search list items would ever be used, as a general rule we should have our search list values obey the relationship

$$l_1 < l_2 < \dots < l_n$$

In such a case, if we set up the table

Search List	Search Values
l_1	v_1
l_2	v_2
...	...
l_n	v_n

then LOOKUP(x, table) would operate as follows:

if $x < l_1$	then return NA
if $l_1 \leq x < l_2$	then return v_1
if $l_2 \leq x < l_3$	then return v_2
...	
if $l_n \leq x$	then return v_n
if x is blank	then return #N/A
if x is na()	then return #N/A

If the search list includes blank values, they are ignored unless the last part of the table is blank. In such a case, LOOKUP returns the value 0 if the search argument exceeds the last nonblank entry in the search list. We can settle many uncertainties by examining the results shown in the next worksheet. The search arguments used in column 4 lead to the results shown in the corresponding locations in column 5.

The formula for position R2C5 is LOOKUP(args,R3:6C1:2). We named column 4 "args," and after inserting the formula in cell R2C5, we copied it down. The result is:

#1	1	2	3	4	5	6
1	search	search				
2	list	values		-9	#N/A	
3		10	31	10	31	
4		15	2	12	31	
5		25	-3	15	2	
6		45	14	16	2	
7				24	2	
8				25	-3	
11				44	-3	
10				45	14	
11				100	14	
12				75	14	
13						

<C8E>

In these examples we had our search table set up in adjacent columns. You can set them up in adjacent rows if you prefer. When a search table is set up in adjacent rows, the top row holds the search list with the smallest values on the left. The row immediately beneath the search list row will be expected to have the search values.

Summary

- The following functions were discussed:

COUNT ABS INT SQRT

- These mathematical and engineering functions were discussed:

EXP LOG10 LN
 SIN COS TAN
 ASIN ACOS ATAN

- These special functions were introduced:

ERROR useful in testing
 NA() to designate "Not Available" data
 PI() provides the constant 3.1415. . .

- The error message "ERROR" is displayed when attempting undefined arithmetic calculations.

- The Options command can stop the automatic recalculation mode. By typing N into the "recalc:" field, you set this in effect. New entries will not cause any recalculations until you request them by typing an F4.

- Asking for the Options command and typing Y into the "recalc:" field forces reversion to the normal automatic recalculation mode.

- The Print Options command prints the worksheet with formulas.

- The following functions were introduced:

Comparison operators <
 >
 =
 < =
 > =
 <>

Logical functions IF(logical-arg,value1,value2)
 NOT(logical-arg)
 OR(list of logical-args)
 AND(list of logical-args)
 FALSE(logical value False)
 TRUE(logical value True)

String functions

DOLLAR(n) shows value n in \$ format.

FIXED(n,m) converts value to text.

LEN(t) returns length of the text.

MID(t,n,m) returns m characters.

REPT(t,n) create bar graphs.

VALUE(t) numeric value is returned.

- Finally, the net present value and lookup functions NPV and LOOKUP were presented.

9

Controlling Formats and Options

The visual disposition of items on a worksheet is controlled by the use of format commands. We have been ignoring this aspect of Multiplan until now, because we concentrated on substance rather than form. However, it is important that the results you are producing look very good so as to lend them credibility and avoid misperceptions, misinterpretations, or misunderstandings. Likewise, the Options command allows you to use advanced functions such as iteration, cell protection, and recalculations. We begin with Format.

When we dealt with the gradebook example back in Chapter 5, some of the average grades were displayed with fraction parts as shown in the following worksheet:

#1	1	2	3	4	5	6	7
	Name	HWK1	HWK2	HWK3	TST1	TST2	AV
2	Alan		95	100	95		96.66667
3	Bernice		85	95	80		86.66667
4	Charles		60	0	54		38
5	Doris				88		88
6	Fred		80	60	64		68
7	Mary		82	80	90		84
8	Av		80.4	67	78.5		76.88889
9							<C9A>

Two things stand out; (1) the titles do not seem to fit over the columns as well as they should, and (2) some of the calculated averages (e.g., 96.66667) stick out like a sore thumb. Why does Multiplan present the kind of display it does? How can we make the actual worksheet more presentable? This leads us to consider the Format command.

Global Formats

Multiplan normally tries to display each number and each calculated result to its full accuracy within the column width allowed. The default column width is 8 characters. Whenever you are displaying a number, one of those characters is always reserved for a blank character, so numbers of up to eight digits will be displayed if it takes eight digits to represent the best approximation. You can, of course, increase or decrease the column width using the F W (width command), in which case more or fewer

digits will be displayed. Similarly, for text, as much of the text as can fit in the given column (including all character positions) will be displayed, *but*:

1. Numbers and formulas will be right-adjusted.
2. Text will be left-adjusted.

We see that clearly in column 1: the names line up nicely with the column label "Name"; all of them are left-adjusted, by default. The column 2 label, "HWK1," is left-adjusted, but the numbers in column 2 are right-adjusted, all by default. When we get to the calculated results, some of them look "nice," because they only need two digits to be represented exactly. For instance, Mary's average grade of 84 comes from $(82+80+90)/3$, or $252/3$, which is exactly 84. On the other hand, Fred's grade does not come out as a whole number, so it is displayed in seven digits plus a decimal point. Some of the column averages come out as whole numbers and others (e.g., 80.4 for column 2) are exact if displayed with an extra digit.

What we have just described, along with its ramifications, is the Default Format which is in effect when you start with a blank sheet. What other formats are there and how do we invoke them?

Changing Global Formats: The F D Command

Changes in format which affect the entire worksheet are entered with the F(ormat) command, followed by the D(efault) subcommand. We used F earlier to get at F W to change column widths. Recall the prompt generated by F is:

```
FORMAT: Cells Default Options Width
```

Selecting D will generate the next prompt:

```
FORMAT DEFAULT cells width
```

Selecting C for cells will generate the following:

```
FORMAT DEFAULT CELLS alignment: C (Gen) L R
  format code: Cont Exp Fix(Gen)Int $ * %
  # of decimals: 0
```

If we then select Int, for Integer, in the second field, all numbers on display will be rounded to the nearest integer. Note the phrase "all numbers on display." The underlying values are not affected. You will continue to compute with full accuracy, but you see only a display to the accuracy you select. If we then typed

```
F D
C [TAB]
I RETURN
```

our worksheet would now look as shown on page 93 (C9B).

This looks much better, and it also makes much more sense. When the raw data is only accurate to the nearest unit, what sense does it make to report averages accurate to the nearest tenth, hundredth, or thousandth?

#1	1	2	3	4	5	6	7
1	Name	HWK1	HWK2	HWK3	TST1	TST2	AV
2	Alan	95	100	95			97
3	Bernice	85	95	80			87
4	Charles	60	0	54			38
5	Doris			88			88
6	Fred	80	60	64			68
7	Mary	82	80	90			84
8	Av	80	67	79			77
9							<C9B>

Now, how do we get our labels to fit more nicely? Entering F D R selects the right-adjust option. This will force all items not otherwise being controlled to be right-adjusted in their respective columns. So our worksheet now looks like:

#1	1	2	3	4	5	6	7
1	Name	HWK1	HWK2	HWK3	TST1	TST2	AV
2	Alan	95	100	95			97
3	Bernice	85	95	80			87
4	Charles	60	0	54			38
5	Doris			88			88
6	Fred	80	60	64			68
7	Mary	82	80	90			84
8	Av	80	67	79			77
9							<C9C>

What are the meanings of the remaining Default Format selections? The ones we saw earlier are included as well. Note that we selected F D(efault) followed by C(ells) to get this prompt.

The Alignment Field Shows

Def	(Default) Aligns as specified by the Default command.
Ctr	(Center) Aligns all entries in the middle.
Gen	(General) Aligns all text left, all numbers right.
Left	Aligns all entries to the left.
Right	Aligns all entries to the right.
-	Leaves the alignment field unchanged.

The Format Field Shows

Def	Format as specified by the Format Default command.
Cont	Continues long text (over 9 characters long) across column border, if next cell is empty and also formatted with this option ("Cont").
Exp	(Exponent) Displays numbers as a decimal times a power of ten. Number of digits as specified in the next field, the "# of decimals."
Fix	Rounds to and displays number of decimals specified in "# of decimals" field.
Gen	(General) Displays numbers in the most appropriate form considering the size of the cell and of the number.
Int	Numbers will be shown as integers after rounding.
\$	Displays numbers with leading dollar sign and 2 decimal places. (Negative numbers in parentheses)
*	(Bar Graph) Displays as many asterisks as the value of the number.
%	(Percent) Displays the number times 100 followed by a percent sign.
-	Does not change the format field.

Table 9.1 Data, as Centered by Multiplan

#1	1	2	3	4	5
1	1		a		
2	12		ab		
3	123		abc		
4	1234		abcd		
5	12345		abcde		
6	123456		abcdef		
7	1234567		abcdefg		
8	12345678		abcdefgh		
9	1.2E+08		abcdefgh		
10	1.2E+09		abcdefgh		
11			abcdefgh		
12					
13					

The Third Field (# of Decimals) Shows

of decimals: This is used in the Exp, Fix and % formats above. The maximum number of decimals is 15.

The first field we encounter is the alignment field, and it deals with how the text and/or numbers are displayed within the 8 character default format. We have discussed the Right command from the alignment field; Left for Left-adjust text is the opposite of R for Right-adjust. The G for General-format was discussed earlier (labels left-adjusted up to 8 alpha characters per column of 8, numbers right-adjusted to 7 digit accuracy). The C, for Center, centers all entries in the worksheet, leaving spaces, if possible, on both sides of an entry. Note how the C option displays numerical and text data in columns 1 (numeric) and 3 (alpha) in Table 9.1. The hyphen option in the alignment field leaves the format selections untouched and allows you to move on to the next field.

In the Format field, we have commands which affect the display of our data rather than (or in addition to) where in the cell it will be placed. The display in the cell does not affect the stored data. We discussed the Int option. The grades in our gradebook which we changed with the Int command will still be recorded by Multiplan as natural (rather than integer) numerals. If you wish to test this, simply request the next option—G. This option makes Multiplan display data in the best possible way. Note how in Table 9.1 column 1 changes over to scientific notation as soon as the number of digits in our data exceeds eight. If we request the Int format in this case, cell R1C10 will display “#####.” The Exp option converts the entire display to scientific notation. The Fix option rounds all data to the nearest decimal requested in the third field (# of decimals) and converts the display to this new format. The % option displays all data in percentages followed by the “%” sign. In effect it requests Multiplan to multiply all entries by 100.

The * format has the same effect as the one we saw when discussing graphical output in chapter 7. As you recall, it caused the value at an entry to be rounded to an integer number, and that number of asterisks to be displayed up to the column width capacity. If we typed

F D C [TAB] *

all our numbers would be replaced by *****. Note that as soon as more than eight asterisks are required, the cell will be filled with eight #####.

The \$ format will display any and all numbers as money, that is with a \$ sign and two decimal points, so 80, 83.8, and 79.06667 are displayed as \$80.00, \$83.80, and \$79.07, respectively. The dollar format is convenient for data entry of any financial transactions, as you do not have to key in the “.00”

for exact dollar amounts such as \$123. Of course you should not be typing the \$ with the 123, since a \$ will be displayed. The E format displays all the cells specified in scientific notation, regardless of the number's magnitude.

The Cont(inuous) option allows you to devote a number of cells to a long title, as we saw in Chapter 6.

Protecting Entries: The Lock Command

After grades have been entered into the gradebook, you may want to make sure they cannot be accidentally changed. This can be accomplished by protecting the cells for grades that have already been entered.

Recalling our gradebook (C9C), we would like to protect from R2C2 to R9C4. To do this enter

L

See the prompt:

```
LOCK: Cells Formulas
Select option or type command letter
```

At this point, select C (you could also press RETURN, in lieu of C) for Cells and see the next prompt:

```
LOCK cells:
status: Locked(Unlocked)
```

At this point, you can enter a cell, a partial or full column, a row, or a block. To protect the grades enter

```
R2:8C2:7
[TAB]
L
RETURN
```

The protected area will now be protected, which means you cannot enter data in this block or change it with the commands Blank, Copy, eXternal, Load, or Delete. If you try to alter the cell's contents, Multiplan will sound its bell, and the message line will display:

Locked cell may not be changed

You can, however, remove the protection by entering the same sequence and choosing the unlock option. Try unprotecting a part of your gradebook. Type

```
G R 2 [TAB] 2 RETURN
L C
R2:9C2:4 RETURN
[TAB] U
RETURN
```

The lock command is very useful if you have other people entering data into your worksheet and you want to minimize the errors they might make when entering data into existing cells. Moreover, if you have a worksheet laid out to facilitate preparing, say, monthly reports, you could lock all the cells which do not change from month to month. When someone else or you are ready to update or insert

the monthly data, you can use the CTRL-F or INST key to tab from one unlocked cell to the next unlocked cell. That way you don't even have to remember which cells are locked and which are not. You would also, no doubt, save a fair amount of time.

The CTRL-F or INST keys are valuable when working with templates. Templates are stored worksheets which you use at different times with different numbers. For example, the monthly reports mentioned in the previous paragraph would be an example of a template. Returning to the gradebook example, a teacher will save the teacher's gradebook as a template, without names and grades, to use with other classes. The CTRL-F or INST keys enhance the automatic movement of the cursor by making the cursor skip over locked cells. Using the class gradebook as an example, if the first two columns (HWK1 and HWK2) have been protected, you can tab directly from the first column (1) to the next unprotected column (4), thus saving several keystrokes. This saving becomes substantial when large amounts of data are to be entered.

The Lock command has one more option; recall that the first prompt to the Lock command was:

```
LOCK: Cells Formulas
```

if you respond with the F subcommand, you would see the next prompt:

```
LOCK FORMULAS:  
Enter Y to confirm
```

If you entered Y or y, all cells containing text or formulas would be locked. This subcommand protects all the values (!) that result from the formulas and all text (alpha) formats. No numbers will be locked. Use the previous subcommand (L C) to lock numbers.

Changing Cell Formats: The F C Command

We can change the format of an individual cell, row, column or block by typing F followed by C. The prompt generated will be similar to the one we saw earlier in this chapter, when we discussed global format changes:

```
FORMAT cells:          align:(Def)C G L R -  
code:(Def)Cont Exp Fix Gen Int $ * % -  
# of decimals:  
Enter reference to cell or group of cells
```

Notice that the only change is addition of a new field "cells:" which allows you to specify which block, column, row, or cell is to be affected by your format changes. The "alignment:" field, the "format:" field, and the "# of decimals:" field were moved from fields 1, 2, and 3 to 2, 3, and 4, respectively. All else remains the same.

Clearly the above implies that Multiplan can handle a variety of formats, displays, and even graphs on the same worksheet. For additional options, keep reading.

The Options Command

An important command which lets you control a different set of tasks is the Options command. We met the Options command in Chapter 8, where the recalculation option was discussed. Recall that after typing

O

you see the prompt:

```
OPTIONS recal: Yes (No)      mute: Yes(No)
        iteration: Yes(No)
        completion test at:
```

Each of the fields serves as a toggle command (an on/off command) setting some Multiplan feature either on or off. We met the "recalc:" field in Chapter 8. The first time you enter this command, a set of parentheses around the default currently in effect is displayed. When you start with a fresh worksheet, the (Yes) option in the recalculation field is set on. To turn it off, you had to press n or N followed with RETURN, and now no recalculations will be performed until you instruct Multiplan to do so by pressing ! (Shift and 1).

The next option "mute:" controls the audible beeper sounded off Multiplan whenever you try to do something illegal. The sound will be audible only if your monitor has audio capabilities, as in a TV set. Some monitors do not have a speaker built in and will not produce a sound.

Thus, for example, if you try to scroll to the left past column 1, a beep will sound. If you try to write into a protected cell, a beep will sound. The "mute:" field is turned off, and you can turn it on by tabbing to it and pressing y or Y.

After you press RETURN, you will both activate the command, as well as gain an additional piece of information: Multiplan will display below the worksheet, on the message line:

```
Multiplan 1.06      Bytes 21044
```

This will give you the version number, the serial number, and the total amount of memory (equivalent to 100%) available to you. This is important to keep in mind when working with large spreadsheets or spreadsheets with complex formulas or numbers. These tend to use up memory rather quickly.

There are two more option fields in this command. The "iteration" option has a Yes (No) response, with the No serving as the default response. When "Yes" is selected, the recalculation of worksheets with circular references will be repeated until a completion test (see below) is satisfied.

A special function, DELTA(), can be used with the iteration option. It returns the maximum change in the values of cells from one iteration to another. By referring to DELTA() from the worksheet more than once, the maximum change in a particular portion of the worksheet may be calculated.

The last option field, "completion test at" specifies the location of a formula for completion test. Iteration in the worksheet will stop when this formula evaluates to TRUE. If the cell which is supposed to contain a "completion test" formula is empty, the formula DELTA()<0.001 will be used.

An example may be useful at this point. Say a salesperson is trying to calculate his (or her) income. In addition to a salary, that salesperson is to receive a 2% commission out of net sales. Since the net sales cannot be calculated until this very commission is subtracted from gross sales, we have a problem with a circular reference. We will need the iteration function.

We set up Table 9.2 as follows

```
G R 1 [TAB] 1 RETURN
RETURN ITEM ↓
GROSS ↓
COM (2%) ↓
NET RETURN
G R 3 [TAB] 2 RETURN
V
↓ * .02 RETURN ↓
V
↑ ↑ -↑ RETURN
```

We will see:

Table 9.2 Commission Calculation Template

#1	1	2	3	4
1	ITEM			
2	GROSS			
3	COM (2%)	0		
4	NET	0		
5				
6				<C9D>

We are ready to enter the gross sales amount. We need to enter that data into R2C2. Let us try a \$40,000.— gross sales:

```
G R 2 [TAB] 2 RETURN
40000 RETURN
```

We will see:

Table 9.3 Commission Calculation Without Iteration

#1	1	2	3	4
1	ITEM			
2	GROSS	40000		
3	COM (2%)	0		
4	NET	0		
5				
6				

Notice what happened—no calculation took place. We must invoke the iteration option. We type

```
O [TAB] [TAB] Y Enter
```

This invokes the iteration option, and, in less than a minute, we get the data shown in Table 9.4:

Table 9.4 Commission Calculation With Iteration

#1	1	2	3	4
1	ITEM			
2	GROSS	40000		
3	COM (2%)	784.31373		
4	NET	39215.686		
5				
6				<C9E>

Summary

- The F D command permits you to select one of the following:

The Alignment Field Shows

Def	(Default) Aligns as specified by the Default command.
Ctr	(Center) Aligns all entries in the middle.
Gen	(General) Aligns all text left, all numbers right.
Left	Aligns all entries to the left.
Right	Aligns all entries to the right.
—	Leaves the alignment field unchanged.

The Format Field Shows

Def	Format as specified by the Format Default command.
Cont	Continues long text (over 9 characters long) across column border, if next cell is empty and also formatted with this option ("Cont").
Exp	(Exponent) Displays numbers as a decimal times a power of ten. Number of digits as specified in the next field, the "# of decimals."
Fix	Rounds to and displays number of decimals specified in "# of decimals" field.
Gen	(General) Displays numbers in the most appropriate form considering the size of the cell and of the number.
Int	Numbers will be shown as integers after rounding.
\$	Displays numbers with leading dollar sign and 2 decimal places (negative numbers in parentheses).
*	(Bar Graph) Displays as many asterisks as the value of the number.
%	(Percent) Displays the number times 100 followed by a percent sign.
—	Does not change the format field.

The Third Field (# of Decimals) Shows

of decimals: This is used in the Exp, Fix and % formats above. The maximum number of decimals is 15.

The default format in effect when you begin is G; it forces labels to be left-adjusted, and numbers are right-adjusted, displayed to full accuracy within the column width provided. The default format selections are mutually exclusive, as are the default alignment selections.

The F C command also permits you to select from the same list and have the selected format apply to the current row, column, block, or cell. The F C selection followed by D(efault) causes that position's format to revert to whatever global format is in effect.

Blanking an entry will not erase a format assigned using F.

Formats may be replicated. Replicating a format will also replicate the source position's value. It is faster to replicate a column with its format, then blanking its values, than it is to format a new column.

Saving a worksheet before any formats have been assigned makes it easy to assign formats without worrying about destroying any values or formulas.

The L [TAB] L command permits you to protect a cell, a partial row or column, or a block of cells. Data cannot be entered into a protected cell.

- The L [TAB] U command removes protection from a cell, partial row or column, or a block of cells.
- The O (options) command permits you to set several options either on or off. These include
 - Recalculation: (Yes) No—allows you to halt recalculations until desired. You request recalculation with !.
 - Mute: Yes (No)—allows you to turn off the audible alarm which Multiplan sets off whenever you make an error.
 - Iteration: Yes (No)—lets you specify worksheets with circular references.

Note that the command cursor is in the left-hand side, top corner of the directory when you first see it. In the directory reproduced above, the cursor is over the file "C4D". To get the cursor to move to that place, press ↓ three times. The cursor should now be over file C4D. At this point, if you want to load this file, simply press RETURN. In seconds, the file will be loaded into your worksheet. If the file is not a Multiplan worksheet, you will get a "File Format Error" message and nothing will be loaded.

Multiplan will let you set other default drives by using the Transfer Options Setup command sequence. To change the logged drive, you type T (for Transfer) followed by O (for Options).

TRANSFER OPTIONS

mode: Normal Symbolic Other

setup:

Select option

This prompt allows you to enter another disk drive by moving onto the Setup field. This drive will remain the one referenced by Transfer commands until it is changed or until you leave Multiplan. If you have files on another disk drive, enter its specification. Generally this will be for the drive on the right (if you have two drives), and you will enter 9, followed by comma (,) into the Setup field, followed with RETURN. Now, to see the directory of files on 9, you simply press a direction key in response to the file name request. Thus, for example:

T	(Transfer initiated.)
L	(For Load.)
←	(Any direction key would do, here.)

If you have the second drive (for your needs) logged, its directory will now be displayed. To load the appropriate file, use the direction keys until the command cursor is over the desired file name. Now load the file by pressing RETURN. If you had setup drive 9, and wanted to save on drive 8, you can specify 8 before the File name.

Saving: The TS Command

The command TS will save your current worksheet to disk for later use. If you have finished preparing the file and keyed in T followed by S, Multiplan will ask you for a file name. You may use the default name which Multiplan supplies with the prompt. The default name is the active file's name; press RETURN if this is what you wish, or respond with the name under which you wish to save the worksheet. If the worksheet is a new one and was started without a name, Multiplan provides a default name—TEMP and will refer to it as such until you rename it.

If you enter a new file name which already exists in the file directory, Multiplan will give you the prompt:

Overwrite existing file (Y/N)?

When you are entering a revision of a currently saved worksheet, you probably want to replace the old worksheet with the new one. To do this enter Y. *Warning:* This will write over the existing file on the disk and you will lose the old file. Make sure you want to do this before you save the file. If you have any doubt about the value of the old file, use the N command. This will retain your old file name for the file currently on disk, while allowing you to use a different name for the new file which you are about to save. For example, if your old file was IRA and you had a currently updated version in your computer's memory, you might wish to save the old version. Using the N answer to the prompt above, you could now use the TR command which would allow you to rename the current worksheet under a new name. You would produce the following files on your disk (be careful to use file names that help you to remember what they mean):

IRA2	(Your newly edited worksheet.)
IRA	(Your old, unedited version of the worksheet.)

Print Files

You may want to use your Multiplan worksheet files with another program, such as a word processor or database program. If you entered your spreadsheet into a word processor, you could prepare attractive looking reports of the results of your efforts. Most of the tables in this book were prepared this way. Fortunately, Multiplan allows you to save a file in a format that many other programs can use. This is accomplished through the File option of the Print command. Type

P (Print.)
F (on File, on the logged diskette.)

As a result of these commands, Multiplan will ask you for a file name; see the prompt:

PRINT on file:

It is a good idea to use a noncurrent file name since Multiplan cannot use these files as regular worksheet files. Should you accidentally give a file name which already exists on the diskette, Multiplan will prompt you with the now familiar prompt:

Overwrite existing file (Y/N)?

It is good practice to differentiate print files from regular worksheets with either a special character or symbol in the file name. Alternatively you might wish to use a file extension (suffix) on names of print files, "PRN," for example. NOTE: This file is no longer in Multiplan Format and cannot be read by Multiplan. Be careful not to overwrite your only copy of a worksheet when saving a print Format file.

If you have been routinely printing your row and column numbers, they too will be printed unless you have made certain these numbers were deleted. In this case, you might want to delete these borders using the Options subcommand of the Print command. You can also print out the formulas using the same command before saving a print file. Be careful, however, to make sure that the columns are wide enough to show the complete formula.

Using Multiplan Data in Other Programs: The SYLK® (SYmbolic LiNK) File Format

After committing yourself to Multiplan, you may discover another program that allows you to do things you are not able to do with Multiplan. A good example of this is a database program that will conveniently let you save student grades for years on a single disk and access any individual student. Most of us have asked a former teacher for a recommendation hoping that he or she remembers who we are. A database program will remember, but how can we use it without reentering the data?

SYLK, for symbolic link, was designed to allow you to take Multiplan worksheets and convert them to other formats. As a result you can create files from your worksheets that can be used in a variety of other programs, including database programs and other spreadsheet programs that will accept either "comma separated" files, "DIF" files or SYLK files.

To activate the SYLK file format and obtain a SYLK format, you first load the file (or create it) the way you normally process worksheets. Once ready, you must type the following:

T (Transfer.)
O (Options.)
S (Symbolic, you could also press F10 and RETURN.)
RETURN (Sets Transfer Save to save in SYLK Format.)
T (Transfer.)
S (For Save, or L if you wish to Load.)

Note: As in the print on file example, be sure to differentiate your file names with an appropriate suffix: "SLK" for example. (If you fail to do this you may accidentally overwrite your Multiplan files. While SYLK files can be retrieved, it will take some time. . . .) After typing the file name, activate the command with

RETURN

You now have a worksheet in the SYLK format which can be used on other programs. *REMEMBER to turn the Transfer Options command back to Normal* as soon as you complete your work with SYLK files.

Reading Data From VisiCalc® Programs into Multiplan

Returning to the gradebook example, suppose that you had discovered Multiplan after you had started your gradebook on VisiCalc. You are very impressed with Multiplan, but you have hundreds of students in your classes and really don't feel like keying all these names and numbers into Multiplan. All is not lost yet.

As an example, look at the gradebook at the beginning of Chapter 9. You might have it entered in your computer in the VisiCalc format. Of course, there would be hundreds of students.

The VisiCalc gradebook might look like:

	A	B	C	D
1	Alan	95	100	90
2	Bernice	79	95	90
3	Charles	83	0	95
4	Fred	82	80	79
5	Mary	80	60	88
6				

While it looks very much like a Multiplan table (other than the column letters), VisiCalc tables are stored in a different file format. To convert the students' grades to a Multiplan file enter the choice

T O (Transfer Options.)
O (For Other, in response to the menu.)
RETURN

You will be back at the command menu as soon as you pressed RETURN above. Now go back to Transfer Load. Type

T L

Multiplan asks for a file name. Respond with your program's file name, which could be GRADE.VC. (VC is the suffix which VisiCalc automatically adds to its file names.)

Multiplan will start reading your file and indicate to you that it is doing that by displaying:

Now reading line #

The line number will change, and after all lines are read Multiplan will recalculate all the formulas it just converted. This, too, will be indicated by the message:

Cells to calculate #

As soon as this is done, a Multiplan worksheet is displayed:

#1	1	2	3	4
1	Alan	95	100	90
2	Bernice	79	95	90
3	Charles	83	0	95
4	Fred	82	80	79
5	Mary	80	60	88

This is a perfectly normal looking Multiplan file and if you had hundreds of students you would probably be overjoyed. You can now use your Multiplan program to keep your students' grades. You didn't even have to reenter the data. You don't even have to enter the column headings nor any average and total formulas you need.

Creating Multiplan Programs: The eXternal Command

The eXternal command (X) allows you to perform a series of complex operations on several worksheets using only a single worksheet. This is of great value when a repetitive job is called for regularly; for example, updating inventories, or transferring monthly data to a cumulative year-to-date database. To use the X command, you first create the series of links you need.

Multiplan allows you to develop separate worksheets which can draw data from one another. These are related to each other in a hierarchy specifying which worksheet supports another and which are dependent on a source. Supporting sheets are the worksheets supplying data to other sheets. Dependent sheets receive data from other worksheets.

The creation and use of external links between worksheets is a powerful technique when using Multiplan. For example, we can create dependent worksheets that contain the titles and structure or logic of the worksheet without specific variable data in any of the cells. Each person can enter his or her own data directly or from previously constructed worksheets. We can develop an IRA income statement worksheet that calculates the income we expect to have based on the monthly budget for the year. Let us follow this example using the IRA table from Chapter 4. (Don't load this worksheet into memory yet.)

#1	1	2	3	4	5	6
1	RATE	12				
2	YEAR	CONTRIB	VALUE			
3	1	2000	2240			
4	2	2000	4748.8			
5	3	2000	7558.656			
6	4	2000	10705.695			
7	5	2000	14230.378			
8	6	2000	18178.023			<C4D>
9	7	2000	22599.386			
10	8	2000	27551.313			
11	9	2000	33097.47			
12	10	2000	39309.167			
13	11	2000	46266.267			
14	12	2000	54058.219			
15	13	2000	62785.205			
16	14	2000	72559.429			
17	15	2000	83506.561			
18	16	2000	95767.348			
19	17	2000	109499.43			
20	18	2000	124879.36			

Let us assume that instead of putting \$2,000 per year into our IRA, we will put in whatever savings are available to us at the end of the year. We would like column 2 above to reflect our yearly total savings, rather than a flat \$2,000.00.

We begin by constructing a new worksheet which shows months in the first column and gross income in the second, followed by taxes, net income, expenses, and, finally, savings in column 6. (The results are shown on page 107.)

```
T C Y
RETURN MONTHS →
GROSS →
TAXES →
INCOME →
EXPENDITURES →
SAVINGS RETURN
```

Since the EXPENDITURES column will not display the entire title, we may wish to increase the width of column 5 to, say, 13, columns:

```
G R 1 [TAB] 5 RETURN
F W 13
```

We now return to R2C1 and begin entering the months from January in R2C1 through December in R13C1.

```
G R 2 [TAB] 1 RETURN
RETURN JANUARY ↓
FEBRUARY ↓
      ( . . . )
NOVEMBER ↓
DECEMBER ↓
```

(Continue in that fashion.)

You should now be at R14C1, just below the last month. It would be nice to draw a line through the worksheet to separate the columns from the total. We are still in the alpha mode and can type a dashed line:

RETURN

How can we extend this line across all columns to column 6? Notice that the line does not appear past the second column. Do we have to do it again? Fortunately, since nothing is planned for this row, we can simply request a continuous display:

```
F C
R14C1:14
[TAB] [TAB]
C
RETURN
```

(Format Column.)
(Row 14 only.)
(Move to Format field.)
(Ask for Continuous.)

Now the line extends across the table, which should look like this:

#1	1	2	3	4	5	6
	MONTH	GROSS	TAXES	INCOME	EXPENDITURES	SAVINGS
2	JANUARY					
3	FEBRUARY					
4	MARCH					
5	APRIL					
6	MAY					
7	JUNE					
8	JULY					
9	AUGUST					
10	SEPTEMBER					
11	OCTOBER					
12	NOVEMBER					
13	DECEMBER					
14	-----					
15						
16						

All that is left is adding a "TOTAL:" in R1C15, just below the line. Once we do that, we can name each column, using its heading as name. So we type

```
G R 1 [TAB] 1 RETURN
N [TAB] C1 RETURN → (Name C1 Months.)
N RETURN → (Name C2 Gross.)
... (Repeat for C3, C4, C5.)
...
N RETURN (Name C6, Savings.)
```

Now we are ready to fill the table. We will make a few simplifying assumptions:

1. Total annual income will be divided into equal monthly payments.
2. Taxes in column 3 will be 25% of the monthly gross (shown in 2).
3. Income (column 4) was the difference between these two (columns 2-3=4).
4. Expenditures are placed in column 4.
5. Savings are the difference between income and expenditures (columns 4-5=6).

A column's total is calculated by adding the 12 rows above the total. These calculations are shown step by step. (Note that changing one entry, namely the annual salary (at R15C2), will change the entire table.)

We now turn to typing the table's contents:

```
G R 2 [TAB] 2 RETURN
V R15C2/12 RETURN
C D 11 RETURN
```

This completes the first column, entering a 0 to each cell. If you wish to test it, merely go to R15C2 and enter an annual salary. For our example, \$32,000 will generate a monthly stream of \$2,666.67. Column 3 is filled with

```
G R 2 [TAB] 3 RETURN
V .25 * GROSS RETURN
C D RETURN
```

We are ready for the income column. Go to R2C4 and insert the formula:

```
V
GROSS-TAXES RETURN
C
D RETURN
```

Expenditures are entered as shown in C10A. Savings are calculated by entering

```
G R 2 [TAB] 6 RETURN
V
INCOME—EXPENDITURES
C D
RETURN
```

Now enter the totals, beginning with column 3:

```
G R 15 [TAB] 3 RETURN
V SUM
(↑↑. . . ↑ (Up to R2, 13 times.)
:↑↑) (Up to R13, 2 times.)
RETURN
C R
3 RETURN
```

With the totals in, name the cell which portrays Total Savings. Place the cursor over it by typing:

```
G R 15 [TAB] 6 RETURN
N TOTSAV
[TAB] R15C6
RETURN
```

That will register cell R15C6 as TOTSAV. The last thing we want to do is control the format, after providing for larger columns, so as to allow display of all digits:

```
F D W 10 RETURN
F D RETURN [TAB] $ RETURN
```

At this point, if everything is entered correctly, the screen will become very active, and the worksheet will appear on the screen:

#1	1	2	3	4	5	6
	1 MONTH	GROSS	TAXES	INCOME	EXPENDITURES	SAVINGS
2	JANUARY	\$2666.67	\$666.67	\$2000.00	\$1600.00	\$400.00
3	FEBRUARY	\$2666.67	\$666.67	\$2000.00	\$1700.00	\$300.00
4	MARCH	\$2666.67	\$666.67	\$2000.00	\$1800.00	\$200.00
5	APRIL	\$2666.67	\$666.67	\$2000.00	\$1900.00	\$100.00
6	MAY	\$2666.67	\$666.67	\$2000.00	\$2000.00	\$0.00
7	JUNE	\$2666.67	\$666.67	\$2000.00	\$2100.00	(\$100.00)
8	JULY	\$2666.67	\$666.67	\$2000.00	\$2200.00	(\$200.00)
9	AUGUST	\$2666.67	\$666.67	\$2000.00	\$1800.00	\$200.00
10	SEPTEMBER	\$2666.67	\$666.67	\$2000.00	\$1960.00	\$40.00
11	OCTOBER	\$2666.67	\$666.67	\$2000.00	\$1600.00	\$400.00
12	NOVEMBER	\$2666.67	\$666.67	\$2000.00	\$1400.00	\$600.00
13	DECEMBER	\$2666.67	\$666.67	\$2000.00	\$2000.00	\$0.00
14	-----					
15	TOTAL	\$32000.00	\$8000.00	\$24000.00	\$22060.00	\$1940.00
16						
17						<C10A>

If you don't have the optional diskette, you should save this table. The table should be saved using

```
T S
C10A
RETURN
```

At this point we have completed the support table and are ready to deal with the dependent table. Recall that the IRA table shown earlier had a column for the yearly contribution. Let us now load this table and change the first year's contribution to . . . you guessed it: TOTSAV. **Note:** The IRA worksheet must be on the same diskette as its supporting tables (in our case, drive 8.).

Now we are ready to use the eXternal command; first we must blank column 2, below the titles

```
B
R3:22C2
RETURN
```

Notice how column 2 fills with blanks while column 3 (our IRA income) fills with zeroes. Now we type

```
X RETURN (eXternal, Copy.)
```

See the prompt:

```
EXTERNAL COPY from sheet:      name:
                             to: R1C1      linked:(Yes)No
Enter filename
```

We respond with the supporting worksheet's file name:

```
B:C10A (Enter supporting worksheet's name.)
[TAB] (Tab to "name:").
TOTSAV (Name cells which need copying.)
```

If your cursor is still over R3C2 in the dependent (IRA) table, type RETURN. Otherwise, tab to the next field of the eXternal Copy command and type in R3C2 followed by RETURN. At this point, your IRA table is linked to the supporting table. The TOTSAV is entered into R3C2, and the IRA income is recalculated.

Notice that you could easily copy cell R3C2 into the cell below it, and copy that down the column:
G R 4 [TAB] 2 RETURN

```
V
↑
RETURN
C D
18 RETURN
```

This will result in:

#1	1	2	3	4	5
1	RATE	12			
#2					
2	YEAR	CONTRIB	VALUE		
3	1	1940	2172.8		
4	2	1940	4606.336		
5	3	1940	7331.8963		
6	4	1940	10384.524		
7	5	1940	13803.467		
8	6	1940	17632.683		
9	7	1940	21921.405		
10	8	1940	26724.773		
11	9	1940	32104.546		
12	10	1940	38129.892		<C10B>
13			
14					
15			
22	20	1940	156556.55		

Notice also that thus far we invested a fair amount of work but have not yet reaped the benefit. Well, imagine what happens if the person in our example expects a raise, say to \$36,000 per year. How will this affect the calculations?

You may have guessed it. All that is required is a change in the annual salary of the supporting table, C10A. As soon as this new salary is entered, the monthly budget is recalculated. This new worksheet must be saved as C10A, overwriting the previous C10A. (Note: Alternatively, you can use another eXternal subcommand "use" which is explained below.) Now, the minute you load the IRA table C10B the new total savings figure is already in it, and the entire IRA table has been recalculated. (Note: If you try this new salary, the total savings will exceed \$2,000 which is the maximum per "nonjoint" account.)

This exercise may seem like a lot of work for moving one cell value between worksheets. When there are several numbers to be transferred between worksheets, eXternal can prove very useful, a real time saver, as we shall see in Chapter 12.

The eXternal command provides you with two additional subcommands. When you type X, Multiplan prompts you with:

EXTERNAL: Copy List Use

Selecting the List subcommand will automatically result in Multiplan's displaying a list of all the worksheets which are linked to your current worksheet. If there are none, that fact will be displayed in lieu of the list.

```
No sheets support TEMP
No sheets depend on TEMP
Press any key to redraw screen
```

The Use subcommand allows you to rename the supporting or dependent worksheets. When you press

```
X U
```

you will see the prompt:

```
EXTERNAL USE filename:
                instead of:
Enter filename
```

Entering a file name here would let you rename any and all worksheets. This saves removing links and then redefining them.

Deleting Files

You can delete a file using the TD command. It will prompt you by requesting a file name or by asking you to press a direction key if you want to see the file directory. If you press any direction key, you will get the directions for the directory. You can move within this directory using the direction keys. When the cursor reaches the desired name, press RETURN to delete it. Alternatively, if you entered a file name in response to the prompt, that file will be deleted from your disk, and you won't be able to use it again.

Duplicate File Names

You can have a file called TEST (produced using TS) and a PRN-formatted file called TEST.PRN (produced using PF). They will have had the same name except for the suffix .PRN attached to one name.

You can delete whichever one of the two files with similar names you wish by typing the file's full name (e.g., TEST or TEST.PRN).

Summary

- TS stores all information necessary to construct a worksheet including formulas and formats in a Multiplan file.
- An attempt is made to prevent you from accidentally destroying an existing file by overwriting it with a new file. If you assign an existing file's name to be used while attempting to save a worksheet, you will be asked to confirm that you wish to use that name.
- PF stores labels and values just as they would appear on a printout generated using a PP command. The file created using PF should have the suffix PRN attached to its name. It can no longer be used by Multiplan. It can, however, be used by other programs, especially word processing programs.

- Loading files: You can load a Multiplan format file using T L. You can't load a file which is in the PRN format.
- The Transfer Symbolic command provides a useful method of converting files from other programs into the Multiplan spreadsheet format. Symbolic also allows us to convert Multiplan files into forms that can be used by many other programs.
- The eXternal command, X, provides the ability to create linkages between Multiplan worksheets. Data from supporting worksheets can be read into dependent worksheets using the eXternal subcommands.
- TD is used to delete files.

11

Personal Finance Case Studies

This chapter consists of three case studies. They will provide you with an opportunity to review many of the things we have been discussing while developing some useful applications for your personal finances. We will also see some new features of Multiplan.

Case Study A: Expense Log

You would like to keep track of your major expenses. How can Multiplan help you in this task? If you were doing it by hand on a paper worksheet, it might look like this:

Date	Item	Cost	Total
1/20	battery	55.98	55.98
1/21	suit	133.00	188.98
2/3	desk	250.00	438.98

The Total column is the running total of expenses incurred to date. You can set up a worksheet by typing the titles in row 1:

```
G R 1 [TAB] 1 RETURN
RETURN
Date →
Item →
Cost →
Total RETURN
```

Then you can try entering the first item into row 2:

```
G R 2 [TAB] 1 RETURN
1/20 →
```

Surprise! The date "1/20" for January 20 does not appear in position R2C2. What you see is the number .05 which, upon reflection, you connect with the arithmetic expression 1/20 (1 divided by 20). When you were typing 1/20 into position R2C1, Multiplan interpreted it as a value because it starts with a number and did not start with an "A" (Alpha). Out of desperation, you might try changing the entry to Jan 20. That is not necessary. You can simply type the character "A" (or RETURN) followed

by your numbers. This conditions Multiplan to treat the remainder of the entry as text, instead of taking it as a value. We can reenter our date 1/20 into R2C1 by typing the sequence:

```
G R 2 [TAB] RETURN
RETURN 1/20
RETURN
```

After you enter the new date, follow with the rest of the information, as well as the other expense items. We are ready to name each column; we press HOME to get to R1C1 and type

```
N [TAB]
C1 RETURN →
N RETURN →
N RETURN →
N RETURN
```

This should establish Date as the name for column 1, Item as the name for column 2, and so on, through column 4.

We want the worksheet to maintain a running total for us in column 4. So the running totals can be computed as follows:

```
Use Cost          for R2C4.
Use Cost + R[-1]C for R3C4.
Use Cost + R[-1]C for R4C4.
```

The worksheet can be completed as follows:

```
G R 2 [TAB] 4 RETURN
V
COST ↓ RETURN
V
COST+↑
RETURN
C D 1 RETURN
```

Dollar Format

In keeping track of expenses all the values are in dollars and cents. That being the case, we can save some typing by letting Multiplan fill in zeroes for any missing cents if we use the F D [TAB] \$ RETURN command. This will cause numeric entries in all positions to adopt the \$ format, which always displays numbers in the form \$\$\$\$\$.¢ ¢.

Type FDC [TAB] \$ RETURN. From then on, typing a value such as 250 causes it to be displayed as \$250.00. Typing a value 34.6 causes it to be displayed as \$34.60. If you were working with mills (one tenth of a cent; a common measure when speaking of property tax rates) and you typed 12.342, the value \$12.34 would be displayed. As always, the higher precision numbers you type will be used in all calculations even though the results may be displayed only to two decimal points of accuracy, at your request.

We can fill in the values and description for the remaining two items, and our worksheet now has the appearance:

# 1	1	2	3	4	5	6
	Date	Item	Cost	Total		
2	1/20	battery	\$55.98	\$55.98		
3	1/21	suit	\$133.00	\$188.98		
4	2/3	desk	\$250.00	\$438.98		
5						
6			<C11A>			
7						

If we now want to add a new expense, say a lamp for \$45.00 purchased on 2/15, we can type this information and finish the line by copying the running-total formula:

```
G R 5 [TAB] 1 RETURN
RETURN 2/15 →
lamp →
45→ RETURN ↑           (We want to copy from R4C4.)
C D 1                   (Copy.)
RETURN
```

Our worksheet now has the following appearance:

#1	1	2	3	4	5	6
1	Date	Item	Cost	Total		
2	1/20	battery	\$55.98	\$55.98		
3	1/21	suit	\$133.00	\$188.98		
4	2/3	desk	\$250.00	\$438.98		
5	2/15	lamp	\$45.00	\$483.98		
6						
7				<C11B>		
8						

Another Approach

If you do not use the preceding expense log very often, you might forget the steps involved in copying the running-total formula as you add new items to the log. We will examine another approach that is easier to use. It involves eliminating column 4 which displays the running total, and having two extra rows at the bottom, one of which displays the grand total by keying in:

```
G R 6 [TAB] 2 RETURN
RETURN Total → RETURN
```

The only formula we need is the one for position R6C3; since we are there we need only:

```
V
SUM(↑ :
↑↑↑↑) RETURN
```

Our table now shows:

#1	1	2	3	4	5	6
1	Date	Item	Cost			
2	1/20	battery	\$55.98			
3	1/21	suit	\$133.00			
4	2/3	desk	\$250.00			
5						
6		TOTAL	\$438.98			
7				<C11C>		

Now as we wish to add new expense items, we can insert a new row just above the blank row and the formula for the total will be automatically adjusted to include the new cost we just inserted. So if we were using this latest worksheet and wanted to log the expense for the lamp we saw earlier, we could type:

I R [TAB] 5 RETURN (Push it down by Inserting a new Row.)
 G R 5 [TAB] 1 RETURN
 RETURN 2/15 → lamp → 45.00 RETURN

Our worksheet now looks as follows:

#1	1	2	3	4	5	6
1	Date	Item	Cost			
2	1/20	battery	\$55.98			
3	1/21	suit	\$133.00			
4	2/3	desk	\$250.00			
5	2/15	lamp	\$45.00			
6						
7		TOTAL	\$483.98			
8						
9				<C11D>		

In the previous version of the worksheet where we kept a running total, why couldn't we use the same technique, that of inserting a row, and get a similar effect?

Let us reexamine the earlier worksheet:

#1	1	2	3	4	5	6
1	Date	Item	Cost	Total		
2	1/20	battery	\$55.98	\$55.98		
3	1/21	suit	\$133.00	\$188.98		
4	2/3	desk	\$250.00	\$438.98		
5						
6						
7					<C11E>	
8						

We could have created this worksheet with a totals row, as shown below:

#1	1	2	3	4	5	6
1	Date	Item	Cost	Total		
2	1/20	battery	\$55.98	\$55.98		
3	1/21	suit	\$133.00	\$188.98		
4	2/3	desk	\$250.00	\$438.98		
5		TOTAL	\$438.98			
6						
7					<C11F>	
8						

But if we use the I R (Insert Row) command to add a new item between rows 3 and 4, we would get the following result after we filled in the entries for the new item, in the new row 5:

Note that we can type any combination of characters and repeat it. We could use abcd or 123. You may want to stop this string of characters before they cross a particular cell. Recall that the continuous format option will stop the display at a nonempty cell. Before you create the long string, go to the last cell in which you want the string and enter the label in that cell. To stop the repeating signs in the above example, use the following commands:

```
R5C4 RETURN
RETURN -+-+--+-+-- RETURN    (Create a nonempty cell.)
```

This will stop the continuous display of the string you typed into R1C1 at R5C3.

Summary

□ We have seen two approaches to setting up a worksheet for use in a situation in which you will expect to be adding new items at the bottom (or new columns to the right). It is somewhat simpler to use Row Insertion (I R) for this purpose if you set up your worksheet anticipating this.

□ F D \$ is the Global Format Dollar command which lets you enter dollars and cents amounts without typing the right-most zeroes for cents. The values will always be displayed in the form \$\$\$. . . \$\$\$.¢ ¢, provided the column width is adequate.

□ To repeat labels, provide one or more characters which will be duplicated to fill the current cell location by using the alpha mode, and typing in the desired string. To get the string of characters displayed horizontally (past the cell) and across your worksheet, you should format the row using the F C range (or row#) [TAB] [TAB] C RETURN command. This will allow you to fill the range or row with the desired string until a nonempty cell is encountered.

Case Study B: Portfolio Evaluation

From time to time, you may wish to see where you stand in terms of your current investment holdings. Suppose you owned five shares of the ABC company, 100 shares of WOW, and 20 shares of XYZ. You could set up the following worksheet:

Stock	Shares	Price	Worth
ABC	5	110	550
WOW	100	45	4500
XYZ	20	75	1500
		Total	6550

Periodically you can check the current stock prices and update those that have changed. Suppose WOW goes up to 65. Then it would be nice to get an update by typing

```
G R 3 [TAB] 3 RETURN
65 RETURN
```

which will change the 45 into a 65 and simultaneously update the Worth entry to 6500 and the bottom Total value to 8550.

We can set up such a worksheet as follows. First we can enter the titles in row 1 beginning at position R1C1:

```
G R 1 [TAB] 1 RETURN
RETURN Stock → Shares → Price → Worth RETURN
```

Then enter the ABC stock information in row 2 by typing

```
G R 2 [TAB] 1 RETURN
RETURN ABC → 5 → 110 RETURN
```

Similarly, we can enter the WOW and XYZ information in rows 3 and 4 by typing

```
G R 3 [TAB] 1 RETURN
RETURN WOW → 100 → 45 RETURN
G R 4 [TAB] 1 RETURN
RETURN XYZ → 20 → 75 RETURN
```

We can create a dashed line to separate the bottom Totals row from the other rows by typing

```
G R 5 [TAB] 4 RETURN
RETURN ----- RETURN
G R 5 [TAB] 1 RETURN
F C
R5 [TAB] [TAB] C RETURN
RETURN ----- RETURN
```

Next we name the columns

```
Home
N [TAB] R2:4C1 RETURN →
N RETURN →
N RETURN →
N RETURN
```

Then the formula (Shares * Price) for column 4 is entered

```
G R 2 [TAB] 4 RETURN
V
Shares * Price RETURN
```

and then we can copy down for R3 through 4 in C4

```
C D 2 RETURN
```

Finally, we can enter the label "Total" in R6C3 and request a total for the Worth column by typing

```
G R 6 [TAB] 3 RETURN
RETURN Total → RETURN
V
SUM(Worth) RETURN
```

Our worksheet now has the following appearance:

#1	1	2	3	4	5	6	7
	1 Stock	Shares	Price	Worth			
2	ABC	5	110	550			
3	WOW	100	45	4500			
4	XYZ	20	75	1500			
5	-----						
6			Total	6550			
7							
8							
9						<C11I>	

If we now buy a new stock, say 50 shares of SUPER at \$33, we can insert this stock in front of WOW, which is in row 3

```
G R 3 [TAB] 1 RETURN
I R RETURN
```

This will insert a blank new row, as row 3, pushing down rows 3 through 6. We are now at the new R3C1 and we type

```
RETURN SUPER → 50 → 33 → RETURN
V
```

```
Shares * Price RETURN
```

and now we see:

#1	1	2	3	4	5	6	7
	1 Stock	Shares	Price	Worth		<C11J>	
2	ABC	5	110	550			
3	SUPER	50	33	1650			
4	WOW	100	45	4500			
5	XYZ	20	75	1500			
6	-----						
7			Total	8200			
8							
9							

We could avoid typing in the worth formula each time we add a new stock by copying an existing row and correcting the name and values. For instance, before we introduced SUPER, we could have proceeded as follows:

```
G R 3 [TAB] 1 RETURN
I R RETURN
C F ↑
: 5 RETURN
```

(You can use the default.)
 (Copy From R[-1]C.)
 (Through R[-1]C[+4] into target location starting at R3C1.)

Now you can edit the first three entries (R3C1 through R3C4). It's a toss-up whether you prefer to retype a formula such as Shares * Price RETURN or copy first and then edit the entry. Clearly copying is preferred if you have a more complicated formula. Couldn't we have proceeded in another way? Why not fill column 4 with the Worth formula, right from the beginning. Anticipating that we

won't have more than 50 rows in our worksheet, we could fill R2C4 with a formula, then copy it through positions R3C4, R4C4, . . . , R51C4. This will work, except that since most of the rows will have no data in columns 2 (Shares) or 3 (Price), and the product of two blank positions is 0, you will see a column of zeroes down column 4. If you try to erase them by typing B (to blank out an entry), you will also erase the corresponding formula. So it appears we will make do either with retyping formulas as needed or using the Copy command.

Portfolio Evaluation with Gains and Losses

Suppose we wanted to keep track of our gains or losses on each stock. We need a column headed "Cost," to record our original cost, and one headed "Gain." We can insert the "Cost" column just to the left of the Price column which is in column C. Starting with our earlier worksheet, as shown on the next page, and proceeding to insert a Cost column between the Shares and Price columns, we can type

G R 1 [TAB] 3 RETURN
I C RETURN

(Insert a column before column 3.)

#1	1	2	3	4	5	6	7
	Stock	Shares	Price	Worth		<C11K>	
2	ABC	5	110	550			
3	SUPER	50	33	1650			
4	WOW	100	45	4500			
5	XYZ	20	75	1500			
6	-----						
7			Total	8200			
8							
9							

We can proceed to type in the title "Cost" and fill in the costs for our four stocks in column R1C3; type

G R 1 [TAB] 3 RETURN
RETURN Cost ↓ 100 ↓ 33 ↓ 50 ↓ 60 RETURN

The dashed line fills in automatically. Our worksheet now looks like:

#1	1	2	3	4	5	6	7
	Stock	Shares	Cost	Price	Worth		<C11L>
2	ABC	5	100	110	550		
3	SUPER	50	33	33	1650		
4	WOW	100	50	45	4500		
5	XYZ	20	60	75	1500		
6	-----						
7				Total	8200		
8							
9							

We would like the Gain column to be in R1C6. The gain for a stock is defined as the current worth minus the original cost. For the purpose of this example we will ignore any other expenses, such as broker commissions. The formula

$$\text{worth} - (\text{shares} * \text{share cost})$$

is to be assigned to R2C6 and the name Gain to R1C6 by typing

```
G R 1 [TAB] 6 RETURN
RETURN Gain RETURN
N [TAB]
R2:6C6 RETURN
G R 1 [TAB] 3 RETURN
N [TAB]
RETURN
```

we can now enter the formula into R2C6:

```
G R 2 [TAB] 6 RETURN
V
Worth - ( Shares * Cost)
RETURN
```

Then we can copy rows 3 through 5 by typing

```
C D 3 RETURN
```

and add our line at row 6 with

```
G R 6 [TAB] 6 RETURN
RETURN ----- RETURN
```

Our table now looks like:

#1	1	2	3	4	5	6	7
	Stock	Shares	Cost	Price	Worth	Gain	
2	ABC	5	100	110	550	50	
3	SUPER	50	33	33	1650	0	
4	WOW	100	50	45	4500	-500	
5	XYZ	20	60	75	1500	300	
6	-----						
7				Total	8200		
8							
9							<C11M>

We can ask for a grand total for our gains, in R7C6, by typing

```
G R 7 [TAB] 6 RETURN
V SUM(Gain) RETURN
```

We then obtain the following worksheet:

#1	1	2	3	4	5	6	7
	Stock	Shares	Cost	Price	Worth	Gain	
2	ABC	5	100	110	550	50	
3	SUPER	50	33	33	1650	0	
4	WOW	100	50	45	4500	-500	
5	XYZ	20	60	75	1500	300	
6	-----						
7				Total	8200	-150	
8							
9							<C11N>

Copy Revisited

We have been building up our worksheets one feature at a time so we could more easily follow the manner in which they were constructed. Suppose we knew exactly what we wanted. Couldn't we set up worksheets with less effort?

As a matter of fact, that will often be the case. When you need to copy more than one thing for a single worksheet, it may be possible to copy them all at once. This is the case when you want to copy things that happen to be in adjoining columns (or adjoining rows).

Suppose we were just setting up our Worth and Gain columns in columns 5 and 6

G R 2 [TAB] 5 RETURN

V

Shares * Price

(Formula for Worth, in R2C5.)

→ RETURN

(Advance to R2C6.)

V

Worth - (Shares * Cost)

RETURN

(Gain for R2C6.)

We can now copy both formulas simultaneously into positions R3C5, R4C5, and R5C5 for the Worth formula in R2C5, and into R3C6, R4C6, and R5C6, for the Gain formula in R2C5:

C F

R2C5:6 [TAB]

(Source range is R2C5 to R2C6.)

R3C5 RETURN

(Beginning of target range for row 3.)

When you are copying two or more adjacent items, you must specify the beginning and ending coordinates of the source row (or column). Here it is simply R2C5 and R2C6, both in row 2. From that information Multiplan can determine that you want to copy two things (namely, the things in R2C5 and its immediate neighbor, R2C6).

For the target range you only specify the target range for the first source item. Multiplan will copy the first source item as usual into the specified target range. Then when it comes to copy the next source item, it knows how many times to copy it and which row (or column) to begin in, since that information is implicit in the response you gave to the request for Target Range coordinates. Think of painting into a rectangle:

Top edge, source coordinates.

Left side, target coordinates.

R2C5R2C6

R3C5

.

.

.

R5C5

It takes a little thought to figure out where the bottom edge and the right side of this rectangle are located. That is the mental model you need when trying to copy several things at a time while intending to fill several adjacent columns (or rows).

Avoiding Retyping Data

What if you bought stocks at different times, presumably at different costs? Then you should keep track of each purchase almost as if it were a different stock.

Suppose you had bought shares of ABC on five different occasions at different costs. You would like to enter them in your portfolio worksheet. Suppose you had purchased the following quantities at the indicated prices:

Number	Cost
5	100
5	110
10	101
20	90
25	80

Whenever you care to evaluate your portfolio, you have to enter ABC's current price five times. Suppose today it was \$123 per share. The worksheet might first have looked like:

#1	1	2	3	4	5	6	7
	Stock	Shares	Cost	Price	Worth		
2	ABC	5	100	110	550		
3	ABC	5	110				
4	ABC	10	101				
5	ABC	20	90				
6	ABC	25	80				
7	SUPER	50	33	33	1650		
8	WOW	100	50	45	4500		
9	XYZ	20	60	75	1500		
10	-----						
11				Total	8200		
12							
13				<C110>			

You can update ABC's price of \$123 by typing into positions R2C4, R3C4, . . . , R6C4, by setting the cursor pointing down and typing

```
G R 2 [TAB] 4 RETURN
123 ↓ 123 ↓ 123 ↓ 123 ↓ 123
RETURN
```

or by typing

```
G R 2 [TAB] 4 RETURN
123 RETURN
C D 4 (Copy from R2C4.)
RETURN
```

Another Variation

If you expect to have many instances of a given stock bought at different prices, it is a nuisance to update its current price several times. You could avoid having to do this by typing a simple formula for every Price entry for a given stock, save one.

For instance, the first time you record your holdings of ABC stock bought at 100, with a current value of 123, you would type

```
G R 2 [TAB] 1 RETURN
RETURN ABC → 5 → 100 → 123 →
RETURN
V
Shares * Price RETURN
```

The next time you buy ABC stock, say at 110, you can type

```

I R                               (Insert Row before row 4.)
C F R2C5:6 RETURN
[TAB] R3C5 RETURN
RETURN ABC → 5 → 110 →
RETURN V
Price RETURN
  
```

Then whenever you update ABC's current price (in R2C4), the other ABC entries will reflect the same current price in their column 4 locations. This is how we produced the following worksheet:

#1	1	2	3	4	5	6	7	
	Stock	Shares	Cost	Price	Worth	Gain		
2	ABC	5	100	123	615	115		
3	ABC	5	110	123	615	65		
4	ABC	5	100	123	615	115		
5	ABC	10	101	123	1230	220		
6	ABC	20	90	123	2460	660		
7	ABC	25	80	123	3075	1075		
8	SUPER	50	33	33	1650	0		
9	WOW	100	50	45	4500	-500		
10	XYZ	20	60	75	1500	300		
11	-----							
12				Total	16260	2050		

Summary

Portfolio evaluation uses simple formulas and the SUM function. Adding new stock entries involves using row insertion with the I R command and either retyping a simple formula or using Copy to copy it.

We have seen how several items can be copied with one copy command. This is possible when you wish to copy items which appear in adjacent rows or in adjacent columns.

Case Study C: Computing Net Worth

Net worth is an illusive beast. It keeps changing, even if you remember to include all your relevant assets and liabilities, simply because the values of the items making up your net worth can fluctuate almost daily. The problem of defining net worth is compounded by the differing approaches taken in the accounting community. Some would evaluate assets based on their original cost, while others would evaluate them based on current market value; still others may use replacement cost, and so on. Be that as it may, once you choose an approach to net worth that meets your needs, the same kind of Multiplan worksheet can be used in all cases. So let us proceed.

Your net worth is the sum of all of your assets, minus the sum of all of your liabilities. So we can begin with the simple formula:

$$\text{Assets} - \text{Liabilities} = \text{Net Worth}$$

Assets

We can begin by concentrating first on your assets and worry about the liabilities later on. We might start by listing the major assets' categories. You may wish to refine or change this later.

```
ASSETS
  Cash
    Savings account
    Money fund
  Real estate
    Home
  Securities
    Stocks
    Bonds
  Personal property
    Furniture
    Clothing
    Car
  Long-term
    Insurance
    Pension
-----
TOTAL
```

We can see that if we want to avoid cryptic abbreviations we will need wide enough columns to store these labels. We can begin by requesting at least a 15-character wide column, which allows us to display 15 character labels or 14 digit values in column 1. We can change this width again later if we wish to. Getting a 19-character column is accomplished by typing

```
Home
F W 19 RETURN
```

We can start typing the labels in column 1. In order to enhance the readability, it is nice to provide some indentation of the labels. By typing A followed by two spaces, we will get the desired indentation. For example

```
G R 3 [TAB] 1 RETURN
A  Savings      (A starts Alpha, denotes space.)
```

We can fill in the labels for column 1 by typing

```
G R 1 [TAB] 1 RETURN
A ASSETS ↓ Cash ↓ Savings RETURN . . .
. . .
. . . ↓ Pension RETURN
```

We can fill in the dashed line we want in R17C1 by typing

```
G R 17 [TAB] 1 RETURN
F C R1C1:2
[TAB] [TAB] C RETURN
RETURN- . . . -RETURN (Repeat "-" as needed to fill label.)
```

NOTE: we could also repeat the above by using the REPT(T,n) function in the last line above, as follows:

```
V REPT("-",30)      (Repeat "-" 30 times.)
```

Table 11.1 Net Worth Worksheet

#1	1	2	3	4
1	ASSETS			
2	Cash			
3	Savings	2000		
4	Money fund	12000		
5	Real estate			
6	Home	125000		
7	Securities			
8	Stocks	32500		
9	Bonds	2000		
10	Personal prop			
11	Furniture	5000		
12	Car	1500		
13	Clothing	2000		
14	Long-term			
15	Insurance	1800		
16	Pension	24000		
17	-----			
18	TOTAL	207800	<C11Q>	
19				
20				

The formula for the total can go into position R18C2:

```
G R 18 [TAB] 2 RETURN
V
SUM(↑ :
↑↑. . .↑↑          (17 times, up to "Savings")
RETURN
```

If we subsequently wish to insert any new asset items above row 17, their values will automatically be included in the total. Using I R to insert a new row at or above row 17 (presently occupied by our dashed line) will automatically adjust the argument range in SUM above to include all of the old positions as well as the new one. Our worksheet is shown in Table 11.1 (some numbers have been already entered in column 2).

Liabilities

We can proceed to prepare the liabilities part of the worksheet in the same way. If we choose to have that part to the right of the assets portion, say in columns 3 and 4, we may later have an annoying problem on our hands. Each time you wish to insert a new row for some asset you had forgotten, or each time you need a new row to account for yet another liability, both sides of your worksheet will be affected. When you insert a new row, it cuts across the entire worksheet. You can avoid this problem by setting up your liabilities segment below the assets segment.

It would not matter which columns you chose as long as the first row is below that of the Assets TOTAL row (presently row 18). So let us place the liabilities segment in columns 1 and 2. This has the advantage that we can print the whole worksheet with a single Print command and not have any cutting and pasting to do. Leaving row 19 blank, we can type in the liabilities segment by typing

G R 20 [TAB] 1 RETURN
 RETURN LIABILITIES ↓ Bills ↓ ~~✓~~ ~~✓~~ Charge acct . . .
 . . . ~~✓~~ ~~✓~~ Other ↓
 ----- ↓
 TOTAL RETURN

Here we chose to fill in the dashed line as we entered the labels. The formula for the sum of liabilities can now go into position R36C2.

We can complete our worksheet by repeating the totals at the bottom in rows 38 and 39 and calculating their differences in row 41, to obtain the following worksheet:

#1	1	2	3	4
1	ASSETS			
2	Cash			
3	Savings	2000		
4	Money fund	12000		
5	Real estate			
6	Home	125000		
7	Securities			
8	Stocks	32500		
9	Bonds	2000		
10	Personal prop			
11	Furniture	5000		
12	Car	1500		
13	Clothing	2000		
14	Long-term			
15	Insurance	1800		
16	Pension	24000		
17	-----			
18	TOTAL	207800		
19				
20	LIABILITIES			
21	Bills			
22	Medical	3400		
23	Charge acct	1850		
24	Mortgages			
25	Home	39600		
26	Cottage	15800		
27	Taxes			
28	Federal	6000		
29	State	890		
30	Property	1900		
31	Loan Payments			
32	Car	1200		
33	College	8500		
34	Other	2000		
35	-----			
36	TOTAL	81140		
37				
38	ASSETS	207800		
39	-LIABILITIES	81140		
40	=====			
41	NET WORTH	126660		<C11R>

We typed

```
G R 38 [TAB] 1 RETURN
RETURN ASSETS ↓
-LIABILITIES ↓
RETURN V
F C C2 [TAB] [TAB] C RETURN
REPT(“=”,30)
RETURN (Fill with 30 =’s, for columns 1+2.)
↓
NET WORTH RETURN
```

For the four bottom-most rows of columns 1 and 2

```
G R 38 [TAB] 1 RETURN
R18C2 ↓ (Copy Assets total.)
R36C2 ↓ (Copy Liabilities total at R39C2.)
↓
V
↑ ↑ ↑ - ↑ ↑
RETURN
```

You can now insert new asset lines or new liability lines as desired, using I R to insert a new row immediately above the row you select with the cursor. Similarly, you can remove any line which is no longer needed by typing D R. You can update any values that have changed just by placing the cursor on the outdated value and typing in the new value. The appropriate total and the net-worth figure will automatically be updated.

If you feel you should want to rearrange the worksheet, say by moving the “Charge acct” line, row 23, so that it would precede rather than follow the “Medical” line, row 22, you can do so by using the Move command M. Type

```
M (See prompt R(ow) or C(olumn).)
R (See prompt From? Enter Row number.)
23 (See prompt To? Enter Row number.)
[TAB] (Next field: before column.)
22 RETURN
```

This will remove row 23 and place it in front of old row 22. All rows at or below the new row 22 will be renumbered, and all value references to items within them will be adjusted so the results don't change. We can see the “before” and “after” views of the surrounding rows.

Before moving row 23:

```
21 Bills
22 Medical 3400
23 Charge acct 1850
24 Mortgages
```

After moving row 23 to a position preceding old row 22:

```
21 Bills
22 Charge acct 1850
23 Medical 3400
24 Mortgages
```

Summary

Computing your net worth uses SUM for two subtotals: one for the assets and the other for liabilities. In a worksheet with two essentially independent segments which may vary in size from one month to the next (that is, you may wish to insert new information in one part without changing the other part), it is better to have one segment below the other rather than side by side.

Moving a row (or a column) can be done by using the Move command M R. The Move command makes it possible to rearrange a worksheet without having to start all over again.

12

Business Case Studies

In this chapter you will develop three business applications. These include a sales projection, interest computations, and techniques for developing and using "templates" of worksheets.

Case Study A: Sales Projection

Suppose you are selling cars and trucks and wish to project your future sales based on your past sales. You have been in business for three years, so your dollar sales per year record to date is:

	1978	1979	1980
Cars	42323	51891	65123
Trucks	45671	46128	49088

You would like to calculate the growth rate for the sales, obtaining the average yearly compounded rate. What you would really like almost fills the rest of this page. How do we produce this report?

#1	1	2	3	4	5	6	7
1			-Actual-		Growth		Total
2		1978	1979	1980	Rate	Average	(000's)
3	Cars	42323	51891	65123	24.04	53112	159.34
4	Trucks	45671	46128	49088	3.67	46962	140.89
5	Total	87994	98019	114211	13.93	100075	300.22
6							
7	% Cars	48.10	52.94	57.02	8.88	52.69	158.06
8	% Trucks	51.90	47.06	42.98	-9.00	47.31	141.94
9	Total	100.00	100.00	100.00		100.00	300.00
10		<C12A>					

#1	8	9	10	11	12
1			-Projected-		
2	1981	1982	1983	1984	1985
3	80782	100206	124300	154188	191262
4	50891	52761	54699	56708	58791
5	131673	152966	178999	210896	250053
6					
7	61.35	65.51	69.44	73.11	76.49
8	38.65	34.49	30.56	26.89	23.51
9	100.00	100.00	100.00	100.00	100.00
10					

We will build up this report one step at a time, not worrying about any formatting for the time being. We can begin by building the top half of the report first, starting from the left. We can also defer some of the headings without creating more work for ourselves.

If you start with year 1's sales being S, and let the unknown yearly average interest rate be i, then for year 2 and year 3, you would expect:

Year 2 sales are $S*(1+i)$
 Year 3 sales are (Year 2 sales)* $(1+i)$, or

$S*(1+i)*(1+i)$, or
 $S*(1+i)^2$

We can solve for i since the total increase in sales, T, is known to us, and it must also be equal to

$$\begin{aligned} S*(1+i)^2 - S &= T \\ S*(1+i)^2 &= T + S \\ (1+i)^2 &= (T+S)/S \\ &= (\text{year 3 sales})/S \\ (1+i) &= \text{square root of } (\text{year 3 sales}/S) \end{aligned}$$

We begin to lay out our worksheet as follows:

#1	1	2	3	4	5	6	7
1		1978	1979	1980			
2	Cars	42323	51891	65123			
3	Trucks	45671	46128	49088		<C12B>	
4							
5							

We can rephrase our solution for i in terms of our coordinates. Assuming we want the growth rate for car sales in R2C5, type

```
G R 1 [TAB] 5 RETURN
RETURN Rate ↓ RETURN          (Cursor moves to R2C5.)
(SQRT ( ← /
← ← ← ) -1)
* 100 RETURN                    (At R2C5.)
```

The *100 converts the fraction into a percentage. Then we can copy this for the trucks, and, assuming we want the growth rate for total sales, we can fill in row 4 as follows and then copy the growth rate from R2C5 into R3C5 and R4C5. Copying first would also work, except you might be surprised to see the result #N/A in position R4C5 due to not having any values yet for R4C2.

```
G R 4 [TAB] 1 RETURN
RETURN Total →
=                               (Equal sign, instead of V(alue).)
SUM(↑ ↑ :↑) RETURN            (At R4C2.)
C R 2                           (Now copy the formula at R4C2.)
RETURN                          (Use R4C2 as source.)
```

Now we can copy the rate formula from R2C5 into R3C5 and R4C5:

```
G R 2 [TAB] 5 RETURN
C D 2
RETURN
```

Our worksheet now looks as follows:

#1	1	2	3	4	5	6	7
1		1978	1979	1980	Rate		
2	Cars	42323	51891	65123	24.04492		
3	Trucks	45671	46128	49088	3.67342	<C12C>	
4	Total	87994	98019	114211	13.92720		
5							
6							

We want the average in column 6. The average is the sum of the yearly sales divided by the number of years; we can use the AVERAGE function for this as follows:

```
G R 1 [TAB] 6 RETURN
RETURN AVERAGE ↓
V AVERAGE
( ← ← :
← ← ← ← )
RETURN
C D
2 RETURN
```

The next column is the total sales in thousands of dollars. We can set it up as follows:

```
G R 1 [TAB] 7 RETURN
RETURN (000's) ↓
V SUM
(← ← ← :
← ← ← ← ← )
/ 1000
RETURN
C D
2 RETURN
```

The next five columns are the sales projections we want calculated. Assuming that the average growth rate for sales is maintained and that it compounds yearly, if we have sales S, we expect $S*(1+i)$ next year, and $S*(1+i)*(1+i)$ the following year, and so on. We can set this up for cars, in year 1981, and copy it to the right. Unfortunately, it won't quite work out that way. In some cases we need relative cell references, while in others we need absolute ones. Let us name the columns so we can use them in absolute references:

```
G R 2 [TAB] 2 RETURN
N [TAB]
R3:5C5 RETURN → (Use 1978 for R3:5C2.)
N RETURN → (Use 1979 for R3:5C3.)
N RETURN → (Use 1980 for R3:5C4.)
N RETURN → (Use Rate for R3:5C5.)
N RETURN → (Use Average for R3:5C6.)
```

Let us now provide the projected sales for cars, in position R2C8:

```
G R 2 [TAB] 8 RETURN
← ← ← ← * (Enters reference to R2C4.)
(1 +
← ← ← /100) (Enters reference to R2C5.)
RETURN
```

The rate (in R2C5) is a percentage so we have to divide by 100. If we now try to copy this formula for the next few years of sales projections, we run into a trap. All the years for a given row should use the same growth rate, from R2C5, and we easily get this by entering a relative cell reference. But what of R2C4? Clearly we want that to change, so we must use relative reference . . . well, let us examine this problem with more care.

R[-4] (referencing cell R2C4) from our formula for position R2C8 will change to R[-5] when we copy R2C8 into the next position R2C9. It will now reference R2C5; but R2C5 is not the prior-year sales! Position R2C5 holds a growth rate. We simply have to expect that copying with relative references works well only when you are dealing with adjacent rows or columns. Unfortunately sales for the year 1980 and those projected for 1981 are separated by three other columns.

So let's not copy the R2C8 formula. We can type in the one we need for R2C9, and copy it right:

```
G R 2 [TAB] 9 RETURN
← * (1 +Rate/100) RETURN
C R 3 RETURN
```

You can now copy R2C8 down its column, to define R3C8 and R4C8, but you may be in for a nasty surprise. R4C8 is supposed to be the total projected sales for 1981. You would expect the sales breakdown for that year to add up to the total. But if you compute R4C8 using

$$R[-4]C * (1 + R[-3]C/100)$$

the result is 130,117 which is not the total of 80,782 and 50,891 (131,673). We seem to be off by 1,556, which is about a one percent error. When there are two ways of deriving a result, you might expect some discrepancy, especially when functions such as square root are involved. Calculating the total projected sales figure by using the compounding formula will lead to the totals not agreeing. It would be better in this instance to use a simple sum and avoid the problem. So let us use the compounding formula for the Trucks row and a simple sum in the Totals row. We can fill in column 8 as follows:

```
G R 3 [TAB] 8 RETURN
V
← ← ← ← * (Enters reference to R3C4.)
(1 +
← ← ← /100) (Enters reference to R3C5.)
↓
SUM(↑↑:
↑) RETURN
```

This last entry can be copied for the remaining Totals projections:

```
C R
3 RETURN
```

Then add the years 1981 through 1985 in locations R1C8 to R1C11. We can proceed as before to fill in the remaining projections for truck sales.

```
G R 3 [TAB] 9 RETURN
V
← * (1 +Rate/100) RETURN
C R
3 RETURN
```

We have now completed the top half of our report, save for formatting. The bottom half consists almost entirely of percentages. It is only the two Growth Rate items that break up this nice pattern. That being the case, we can fill in all of these positions as if they were all percentages, dealing with

their relative columns using the Copy command. Then we can go back and patch up the few exceptions. We can start with labels for R6C1, R7C1, and R8C1:

```
G R 6 [TAB] 1 RETURN
RETURN % Cars ↓ % Trucks ↓ Total RETURN
```

Then we can compute percentages for R6C2 and R7C2:

```
G R 6 [TAB] 2 RETURN
V
↑ ↑ ↑
/ ↑
* 100 RETURN
C D
1 RETURN
```

and we can copy these two formulas simultaneously into positions R6C3 through R6C12, and R7C3 through R7C12, respectively. But we should first fill in the bottom formula and then we can copy all three simultaneously.

```
G R 8 [TAB] 2 RETURN
V
SUM(↑ :
↑ 6) RETURN
G R 6 [TAB] 3 RETURN
C F
R6:8C2 RETURN
```

We can now amend positions R6C5, R7C5, and R8C5. Position R8C5 is easy to fix; we just want it blank so we position the cell pointer over it and type

```
B RETURN
```

Positions R6C5 and R7C5 reflect the rate of growth of the sales percentages. We have been grossing more from car sales at the expense of truck sales, so we should expect the first rate to be positive and the next to be negative. We could either copy an existing rate formula from above, also in column 5, or simply type them in again, as we show here:

```
G R 6 [TAB] 5 RETURN
V
(SQRT(← / (RC[-1] / .)
← ← ← ) - 1 ) (RC[-3]) - 1 .)
*100 RETURN (At R6C5.)
C D
1 RETURN
```

Formatting the Report

We have defined everything that needs to be calculated, so we should now set out to make the report more presentable by formatting it. We can examine our screen and see what needs to be done. For the numeric entries, it seems that they should either be displayed as integers (for sales projections) or using two-digit fractions (for percentages). We could start out by forcing all numbers into the I format, since that provides for integers. So we type

```
F D C [TAB] I (Set Format Default Cells Integer.)
```

Having taken care of all the units of cars and trucks, we can give the percentages two decimal place formats by assigning

```
F C
R6:8C2:12
[TAB] [TAB]
F (For Fixed decimal.)
[TAB]
2
RETURN
```

The rates in locations R2C5 to R4C5 still need to be put in the same format. This can be done with

```
F C
R2:4C5
[TAB] [TAB]
F RETURN
```

We need to add the top line of the report

```
I R [TAB] 1 RETURN
G R 1 [TAB] 3 RETURN
RETURN -Actual- RETURN
G R 1 [TAB] 5 RETURN
RETURN Growth RETURN
G R 1 [TAB] 7 RETURN
RETURN Total RETURN
G R 1 [TAB] 9 RETURN
RETURN -Projected- RETURN
```

Finally, to make the column headings easier to read, enter

```
F C
R1:2
[TAB]
R RETURN
```

Now that you have the worksheet in the shape you want, you should save a copy of it, say by typing TS SALESPR, and you can print a copy by using P P. Multiplan will automatically split printing of large tables. If you don't like Multiplan's split you could print it out in two segments, as we did back at the start of this case study. We printed the segment defined by the corners R1:9C1:6, then we printed the segment defined by the corners R1:9C7:12.

Another Approach

If you had to do this sales projection all over again (say you lost the file you just saved), you could in the light of our previous discussion save some typing. By planning ahead, which is always easier after a bit of experience, we can see that if we assigned formats as we defined formulas that were going to be copied, we could make one Copy command do the work of two.

For this report, one could begin by assigning the Format Default Cells [TAB] I since we won't be copying into the actual-sales positions. Then we can assign F C [TAB] [TAB] F [TAB] 2 RETURN formats as we proceed to type percentage formulas.

Summary

- The typing effort in preparing a complicated report can be reduced by careful planning, so that as formulas are copied, the appropriate formats are also copied.
- Care must be taken to ensure that results labeled as “totals” are in fact computed so that they represent correct totals.

Case Study B: Interest Computations

When looking into an investment possibility, you need to have a good idea as to what it might cost you to borrow the money to make the investment. You should know whether or not you could make the monthly payments on a new loan. You would also like to have some idea of how aggressively you should negotiate for a better rate. What difference will a 0.5% or 1% per year difference in the loan rate make to you in terms of monthly payments? You probably have the same problem I have, of not being able to find the interest tables when you need them. It is even more frustrating to find the interest tables that cover every rate from 0.5% to 10% in steps of 0.5% when you need to investigate possibilities in the 16 to 18% range.

In such a case, it is nice to have a few Multiplan worksheets available so you can create the tables when you need them.

Monthly Payments

Suppose you want to borrow amount A, which you wish to repay in n equal monthly payments P, at a monthly interest rate of i%. Your monthly payments are then given by the formula

$$P = A * i / (1 - (1+i)^{-n})$$

The symbolism x^n , which is called exponentiation, is read as “x to the power n.” It means that x is to be multiplied by itself n times. The symbol used by Multiplan to denote exponentiation is \uparrow (the solid up arrow obtained by typing SHIFT and the solid arrow). We can set up a worksheet to compute these monthly payments as follows:

- Use column 1 for the desired amount A.
- Use column 2 for the monthly interest rate i.
- Use column 3 for the loan duration in months.
- Use column 4 to display the resulting monthly payment P.

We can set up titles in row 1:

```
G R 1 [TAB] 1 RETURN
RETURN amount → int → months → payment RETURN
```

Then we can enter the data in row 2. For a \$100 loan at 12% per year, compounded monthly, we would enter 1% (12% divided by 12 months) as the interest rate. If we intended to pay it off in one month, we could fill in row 2 as follows:

```
G R 2 [TAB] 1 RETURN
100 → 1 → 1 RETURN
```

We should now enter the payment formula in position R2C4:

G R 2 [TAB] 4 RETURN	(Place formula in D2.)
=	
← ← ← *	(RC[-3] *)
← ← *	(RC[-2] *)
0.01 /	
(1 - (
(1 + (← ← *	(RC[-2] *)
0.01)) ^ - ← ←))	(0.01)) ^ -RC[-2]))

The worksheet should display the following items:

# 1	1	2	3	4
1	Amount	Int	Months	Payment
2	100	1	10	101
3	<C12D>			
4				

It seems that we have many more parentheses in the D2 formula than we had in the original formula. Let us place them next to each other. The original formula was:

$$P = A * i / (1 - (1+i)^{-n})$$

The translation of this formula for use with Multiplan is:

$$RC[-3]*RC[-2]*0.01/(1-((1+(RC[-2]*0.01)) ^ -RC[-2]))$$

Why do we need so many parentheses? Let us number them in pairs, so we can explain why each pair is required.

$$RC[-3]*RC[-2]*0.01/(1-((1+ (RC[-2]*0.01)) ^ -RC[-2]))$$

1 23 4 43 21

The need for the pair of parentheses numbered 1 is clear; those parentheses come from the original formula. Pairs number 2 and 3 prevent $1 + (RC[-2]*.01)$ from being subtracted from the 1 on its left, until it has been exponentiated by the power $-RC[-2]$ which is enclosed in the fourth pair. It is so easy to make a mistake in typing a lengthy formula that you should always have some very simple test case on hand to verify that the result produced by the formula is reasonable.

In this case, two simple tests can be applied. If you borrowed zero dollars, you should expect the payment to be zero. Setting R2C1 to zero should cause R2C4 to become zero. The other simple test is that for a \$200 loan at 1% per month, payable in one month, you would expect to pay \$200 plus 1% of \$200, for a single payment of \$202. Test by setting R2C1, R2C2, and R2C3 to 200, 1, and 1, respectively; these values should cause R2C4 to display 202.

Interest Tables

You may have noticed that the monthly payment for a loan of amount $2*A$ is exactly twice that for a loan of amount A , given the same terms. So we can simplify things by considering a *unit* loan, where A is always 1. The payment formula reduces to

$$P = i / (1 - (1+i)^{-n})$$

To make things simpler, we can modify this formula so that it can work with annual interest rates and always specify the loan period in years. Since we still want monthly compounding, the revised formula becomes:

$$P = (i/12) / (1 - (1 + (i/12))^{-(n*12)})$$

Suppose you want a table showing the payment schedule for a five-year unit loan, for interest rates 10%, 11%, . . . , 14%. We can enter titles in R1C1 and R1C2:

```
G R 1 [TAB] 1 RETURN
RETURN Rate → Payment RETURN
```

We can enter the desired interest rates in R2C1, R3C1, etc., by typing them in, or, as we show here, by using Copy so that we can then easily change the interest rates by changing the lowest of the rates.

```
G R 2 [TAB] 1 RETURN
10 ↓
=
↑+1 (Compute 10% +1, for R3C1.)
RETURN
C D 3
RETURN
```

The payment formula can be placed in R2C2 and replicated through R3C2, R4C2, . . . , R6C2. **IMPORTANT:** For Multiplan to calculate exponents correctly, the minus sign in the exponent must be inside the parentheses, that is just before the 5 in our formula.

```
G R 2 [TAB] 2 RETURN
.01 *
← / 12 /
( 1 - ( ( 1 +
.01 * ← / 12 )
^ ( - 5 * 12 ) ) )
RETURN
C D 4
RETURN
```

We can see the fruits of our labor below:

#1	1	2	3	4	5	6
	1 Rates	Payment				
2	10	0.021247				
3	11	0.021742				
4	12	0.022244				
5	13	0.022753				
6	14	0.023268				
7			<C12E>			
8						
9						

It would be convenient to be able to easily obtain such a table for periods other than 5 years. This can be achieved by using some position, say R2C3, to hold the desired number of years and then changing the last part of our formula, the (5*12), into (R2C3*12). While we are at it, we can go further, as the next section discusses.

Multiyear Interest Tables

It would be useful to have a table showing the monthly payments for both a range of interest rates and a range of loan periods, as in:

#	1	2	3	4	5	6	7
1	Rate	1	2	3	4	5	
2	10	0.087916	0.046145	0.032267	0.025363	0.021247	
3	11	0.088382	0.046608	0.032739	0.025846	0.021742	
4	12	0.088849	0.047073	0.033214	0.026334	0.022244	
5	13	0.089317	0.047542	0.033694	0.026827	0.022753	
6	14	0.089787	0.048013	0.034178	0.027326	0.023268	
7	<C12F>						
8							
9							

We can build such a table by using our previous formula and making good use of the Copy command. Let us begin by initializing column 1 as follows:

```
G R 2 [TAB] 1 RETURN
RETURN Rate ↓
10 ↓
V
↑+1 RETURN (Place formula in R4C1.)
C D 3 RETURN
```

This sets up the rate column with the values 10, 11, . . . , 14. Merely by changing the 10 in R2C1, say to a 16, our table would then reflect the rates (and corresponding payments) for 16%, 17%, etc. The loan period can similarly be set up in row 2, from R2C2 to R2C6.

```
G R 2 [TAB] 2 RETURN
1 → (Place a 1 in R2C2, advance to R2C3.)
=
←+1 RETURN (Place formula in R2C3.)
C R 3 RETURN
```

Once again it is worth noting that the loan period or term range can be changed from the 1 to 5. We have here merely by changing the first value, the 1, into the starting value for a new range of years. The other term periods will be calculated by repeatedly adding one. If you wish, you can also change the formula for R2C3. It now increases successive values by one; you could change that to steps of 0.5 or any other desired value.

Finally, we can fill in a payment formula. We can proceed as follows:

```
G R 3 [TAB] 2 RETURN
0.01 * ←
/ 12 /
( 1 - ( ( 1 +
( 0.01 * ←
/ 12 ) )
^ - ( ↑
* 12 ) ) )
RETURN
```

It is hard enough to type the parentheses correctly once; imagine how difficult it would be to type this formula once for each of 5 terms and then do that again for each of 5 interest rates. You would type it 25 times to fill in the table we want. Fortunately we need only type it in once (correctly) and then use Copy to create the other 24 copies, suitably altered. We can start by copying the payments formula across row 3:

```
C R 3
RETURN
```

This fills up positions R3C3 to R3C6 with the appropriate payment formulas and displays the results in those positions. We can now replicate all of the formulas in row 3, from positions R3C2 to R3C6, into the adjacent rows 4, 5, on up to row 7, as follows:

```
G R 3 [TAB] 2 RETURN
C F
R3C2:6 [TAB]
R4C2 RETURN
```

Summary

□ We have finished building the multiyear loan payment table we showed at the beginning of this section. It has been constructed so that the interest rate ranges and term ranges can easily be changed. Similarly, you can extend the table to cover many more term periods or a wider range of interest rates, or both, simply by specifying a wider range when replicating the factors involved.

Case Study C: An Income Statement Template Using eXternal

The creation and use of templates is a powerful technique Multiplan offers. Templates are worksheets containing the titles and structure or logic of the worksheet without specific variable data in any of the cells. Templates allow us to enter our own data directly or from previously linked worksheets. We can develop an income statement template that calculates the profits we expect to make based on the sales of cars and trucks developed in Case Study A.

We will first enter the titles. Let's call this template "Projected Pretax Profits for the Next Five Years."

```
T C Y
RETURN PROJECTED PRETAX PROFITS FOR THE NEXT FIVE YEARS
RETURN
F C
R1 [TAB] [TAB]
C RETURN
```

Column 1 will be used for titles for our sales revenues from cars and trucks, the costs of our business, and the expected pretax profits. In order to make enough space for these titles, column 1 must be 18 characters wide.

F W 18 RETURN	(Set column 1 width to 18.)
G R 3 [TAB] 1 RETURN	
RETURN Sales in Dollars ↓	(Go down column 1.)
␣ Cars ↓	(␣ denotes blank space.)
␣ Trucks ↓	
Total Sales ↓ ↓	(Skip row 7.)

Costs in Dollars ↓
 ⌘ ⌘ Cost of Cars ↓
 ⌘ ⌘ Cost of Trucks ↓
 ⌘ ⌘ Commissions ↓
 ⌘ ⌘ Overhead ↓
 Total Costs ↓↓ (Skip row 14.)
 Pretax Profits RETURN

Now we must enter the formulas for our income statement template. We will use column 2 for the first year of our projected income statement.

Assume that cars cost us 70 percent of their retail price and trucks cost 80 percent of their price.

G R 9 [TAB] 2 RETURN
 .7* ↑↑↑↑↑ RETURN ↓
 .8* ↑↑↑↑↑ RETURN (In R10C2.)

Assume that we pay our sales force a 5 percent commission and that we have overhead costs of \$10,000.

G R 11 [TAB] 2 RETURN
 .05 *
 ↑↑↑↑↑
 RETURN ↓
 10000
 RETURN (In R12C2.)

Our costs can be totaled in location R13C2.

G R 13 [TAB] 2 RETURN
 V SUM (↑ :
 ↑↑↑↑)
 RETURN

Profits, the difference between sales and costs, can be entered in R15C2.

G R 15 [TAB] 2 RETURN
 ↑↑ -
 ↑↑ . . . ↑ (9 times, to R6C2.)
 RETURN

Finally, our template will have a better appearance if we enter a few dashes at location R14C2.

G R 14 [TAB] 2 RETURN
 RETURN
 ⌘ ⌘ ⌘ -----
 RETURN

We would also prefer whole dollar amounts, so request

F W 10 RETURN
 F D
 C [TAB]
 \$ RETURN

As a result of our efforts, we have created a template which will permit us to calculate the projected profits for one year if we simply enter the expected sales of cars and trucks. However, we previously had projected sales for five years, and we probably want also to project profits for five years. This is a task for the copy command.

G R 6 [TAB] 2 RETURN

C F

R6:15C2

(Column between rows 6 and 15.)

[TAB]

R6C3:6

(Into columns 3 to 6.)

RETURN

The template should now look like:

#1	1	2	3	4	5	6
1	PROJECTED PRETAX PROFITS FOR THE NEXT FIVE YEARS					
2						
3	Sales in Dollars					
4	Cars					
5	Trucks					
6	Total Sales					
7						
8	Costs in Dollars					
9	Cost of Cars	0	0	0	0	0
10	Cost of Trucks	0	0	0	0	0
11	Commissions	0	0	0	0	0
12	Overhead	10000	10000	10000	10000	10000
13	Total Costs	10000	10000	10000	10000	10000
14	-----					
15	Pretax Profits	-10000	-10000	-10000	-10000	-10000
16						
17	<C12G>					
18						

The template looks rather empty. This is correct. At this point you should make sure that you have saved a copy of the template. If you have the optional diskette for this book, it is saved as C12G. Do not overwrite this file. If you do not have the optional diskette, save this file.

T S

C12G RETURN

Now you can enter values for car and truck sales into the template, and you will see pretax profits calculated. However, this is no different than many of the previous examples in this book in what it does for you. So what's the big deal about templates?

The advantage of Multiplan templates is that we can load data into the template from another Multiplan worksheet. Recall that we already have the projected sales for cars and trucks from 1981 through 1985. Why should we bother to enter them again? Granted there are not too many entries, but there will be times when we might have many more entries or our projections might change, and we might want to recalculate our income statement.

Multiplan has stored our template in the form shown as columns 1 through 4 below. (Columns 1 to 4 may have to be widened to accommodate the complete formulas.) You can get this on your printer by entering

P O (Print Options.)
 R1:15C1:6
 [TAB] [TAB] Y (To "formula:" field, press Y.)
 RETURN P (Printer subcommand.)

You can also print to your file by pressing F(ile) instead of P(rinter) as the last step above. You will see:

```

#1          1          2          3          4          5          6
1 "PROJECTED PRETAX PROFITS FOR THE NEXT FIVE YEARS"
2
3 "Sales in Dollars"
4 "  Cars"
5 "  Trucks"
6 "Total Sales"
7
8 "Costs in Dollars"
9 "  Cost of Cars"      0.7*R[-5]C      0.7*R[-5]C      0.7*R[-5]C
10 "  Cost of Trucks"  0.8*R[-5]C      0.8*R[-5]C      0.8*R[-5]C
11 "  Commissions"    0.5*R[-5]C      0.5*R[-5]C      0.5*R[-5]C
12 "  Overhead"       10000           10000           10000
13 "Total Costs"      SUM(R[-1]C:R[  SUM(R[-1]C:R[  SUM(R[-1]C:R[
14                   "-----"      "-----"      "-----"
15 "Pretax Profits"   R[-9]C-R[-2]C
16
17                                     "<C12G>"
  
```

We can now obtain data to enter into our template. This data is in our file C12A. Load this file, to refresh your memory:

T C Y (Clear worksheet.)
 T L
 C12A

Columns 8 through 12 contain our needed data:

```

#1          8          9          10         11         12         13         14
1          -Projected-
2          1981      1982      1983      1984      1985
3          80782    100206   124300   154188   191262
4          50891    52761    54699    56708    58791
5          131673   152966   178999   210896   250053
6
7          61.35    65.51    69.44    73.11    76.49
8          38.65    34.49    30.56    26.89    23.51
9          100.00   100.00   100.00   100.00   100.00
10
11
  
```

The data we need is at locations R3:5C8:12, so let's write down these references for use when we call on the eXternal command. First, we must clear the worksheet and load the income statement template file back in:

```
T C Y
T L C12G RETURN
```

The years for your projected sales which appear from R3:5C8 through column 12 need to be inserted into columns R4:6C2 through column 6 of our template. We can copy our data there by entering

```
G R 4 [TAB] 2 RETURN
X C
C12A
[TAB]
R3:5C8:12
[TAB]
[TAB]
N RETURN
```

This produces the following results. To make our final display neater, delete the row and column numbers from the final printout. Using the template, we now know our projected pretax profits:

#1	1	2	3	4	5
1	PROJECTED PRETAX PROFITS FOR THE NEXT FIVE YEARS				
2					
3	Sales in Dollars				
4	Cars	\$80781.77	\$100205.68	\$124300.06	\$154187.91
5	Trucks	\$50891.21	\$52760.65	\$54698.77	\$56708.08
6	Total Sales	131672.98	152966.34	\$178998.83	\$210895.99
7					
8	Costs in Dollars				
9	Cost of Cars	\$56547.24	\$70143.98	\$87010.04	\$107931.53
10	Cost of Trucks	\$40712.97	\$42208.52	\$43759.02	\$45366.47
11	Commissions	\$6583.65	\$7648.32	\$8949.94	\$10544.80
12	Overhead	\$10000.00	\$10000.00	\$10000.00	\$10000.00
13	Total Costs	\$113843.85	\$130000.82	\$149719.00	\$173842.80
14	-----				
15	Pretax Profits	\$17829.12	\$22965.52	\$29279.83	\$37053.19
16					
17	<C12H>				

If you want to save the complete table, it should be saved as C12H to avoid overwriting your template. It appears as C12H on the optional diskette.

The creation of the template took some time, and it's probably not worth the effort for a single analysis of our projected income. However, we can save the template and use it again when our projected sales change.

As mentioned, the table can be printed without a border for presentation. Moreover, Multiplan allows you to have the title (row 1 in our case) printed in wider (double font letters) and larger letters. You can, in fact, use different type faces for each column in your table, provided you have an EPSON or other Dot Matrix printer. Consult advanced report generation tips in your Multiplan manual.

PROJECTED RETAX PROFITS FOR THE NEXT FIVE YEARS

Sales in Dollars	1981	1982	1983	1984	1985
Cars	\$80782	\$100206	\$124300	\$154188	\$191262
Trucks	\$50891	\$52761	\$54699	\$56708	\$58791
Total Sales	\$131673	\$152966	\$178999	\$210896	\$250053
Costs in Dollars					
Cost of Cars	\$56547	\$70144	\$87010	\$107932	\$133884
Cost of Trucks	\$40713	\$42209	\$43759	\$45366	\$47033
Commissions	\$6584	\$7648	\$8950	\$10545	\$12503
Overhead	\$10000	\$10000	\$10000	\$10000	\$10000
Total Costs	\$113844	\$130001	\$149719	\$173843	\$203419
	-----	-----	-----	-----	-----
Pretax Profits	\$17829	\$22966	\$29280	\$37053	\$46634

<C12H>

We could have entered our income statement into our projected sales worksheet. This would have produced a fairly complex worksheet, and whenever we wanted to work with only the sales projection or the income statement worksheet, we would be forced to use both. But the real value of templates shows when we don't have to construct them ourselves.

We can obtain already designed templates into which we can enter our own data. Many different Multiplan templates are available, printed in books or articles or on disks. There are templates available for income tax and personal finance and real estate investments. We merely need to enter our own data into these templates to solve our problems, such as tax computations.

Summary

- Templates and eXternal provide a powerful way to develop and store the structure of worksheets. These templates can be used either by entering data directly or from another worksheet. Very complex calculations can be accomplished using templates.
- Templates can also be purchased, therefore saving us the efforts of designing and entering them.

13

When to Avoid Multiplan

Multiplan is a marvelous tool when you are dealing with numbers in a situation where you must “cut and try.” You could, however, get carried away and become so infatuated with this tool that you find yourself creating work rather than reducing it because you are using a good tool for the wrong job. Consider the worksheet we set up in Chapter 11 to maintain an expense log, with expenses divided into two categories:

Date	Item	Category1	Category2
1/23	Battery		55
1/29	Suit	125	

What if we wanted an expense journal with a dozen or more categories? It might look like:

Date	Item	C1	C2	. . .	C12
1/23	Battery		55		
1/29	Suit	125			
2/2	Heat				197
. . .					

It would be a nuisance to have to skip over so many columns to fill in new entries. What if you wanted to set up a journal in four columns, using a code for each category, as in:

Date	Item	Category	Cost
1/23	Battery	B	55
1/29	Suit	A	125
2/2	Heat	L	197
. . .			

There is no way to ask the current version of Multiplan to do the following:

1. If an item has category code A, then add its cost to the Total for Category A.
2. If an item has category code B, then add its cost to the Total for Category B.
3. Similarly for categories C, D,

Could we not get the same effect by using numeric category codes 1, 2, 3, . . . , instead of codes A, B, C, . . . ? Not with this version of Multiplan.

This inability of having Multiplan perform according to “if this situation is in effect, then do that” in all its generality, separates it fundamentally from the high level computer programming languages such as BASIC, COBOL, FORTRAN, PASCAL, etc. They all include the decision-making ability implied by an “if this, then do that” capability. Note that the IF function does not solve this problem. The IF lets you choose one of two values; it does not let you choose where the result will be placed. This inability of Multiplan is both a weakness and a virtue. We can see how it limits, but we should appreciate that not having this ability presents a tremendous simplification. You may already feel that Multiplan has too many rules; you need only examine books which describe programming in such languages as BASIC or FORTRAN, and you will be amazed at how much more complicated life can be.

You can still sometimes get the best of both worlds. Take the example we just saw. Multiplan makes it so easy to fill in the entries that we might want to use Multiplan just to do that: type in expense items. Then we can save the expense worksheet using the SYLK data format. Whenever we need or want a current total for each of the cost categories A, B, C, . . . , or categories 1, 2, 3, . . . , we can use DOS to run a program we have had written for this purpose. This program will process our SYLK formatted file without changing it. The program will display and/or print the category totals we need.

Other Problems

The most serious problem we encounter reflects the way the world works. Most things we deal with require a lot of information. In computer jargon, decision making requires many inputs. Being very specific, much of the data you want to base your decisions on is accumulated in different computer files. You may have an accounts receivable file, an accounts payable file, a customer name-address file, a year-to-date sales file, etc. There is no easy way to extract the desired data from multiple files to send on to a Multiplan worksheet. You would have to find or write a program to do this. The simplest method is for you to do a lot of thinking about what really matters. It has to matter so much that you are willing to type in the required data.

If you can find some way of letting another program provide the data you need for cogitating with Multiplan, or create a program that will extract the data you need from existing files for the same purpose, then you can avoid all the data errors that sneak in due to excessive data transcription, and you can use some of the time saved to do some more thinking, so you can concentrate on making the right decisions.

That may be the best criterion for deciding whether or not some problem is worthy of being attacked with Multiplan. If it is worth your time to type it in, then Multiplan may be the tool to use, and then the search for a program to help with the use of existing data files will certainly be warranted. Should no such program turn up, you can then consider writing one yourself or contracting for one.

Other Multiplan Versions

It is comforting to know that almost everything we have seen in this book applies verbatim to the versions of Multiplan available for other personal computers. Furthermore, as new versions of Multiplan evolve, most probably as supersets of what we have seen, it should not be necessary to unlearn or undo anything. It becomes a matter of expanding the scope of things that possible new versions might let us tackle.

The background you have developed by working thus far should make it easy for you to learn how to use future enhancements. Happy computing.

14

Summary of Multiplan Commands

You can see the commands at the bottom of your Multiplan screen. The meanings of all of the commands displayed are shown below.

A	alpha	Used to enter text or labels.
B	blank	Erases value/label.
C	copy	Copies one or more cells to another location.
D	delete	Deletes a partial row or column.
E	edit	Modifies an entry without retyping it.
F	format	Formats an entry.
G	goto	Allows jumping to any cell on the worksheet.
H	help	Provides on-screen help to the user.
I	insert	Inserts a row or column.
L	lock	Protects cells from unintentional change.
M	move	Moves one or more rows or columns.
N	name	Allows naming a cell or a group of cells.
O	options	Turns certain Multiplan features on or off.
P	print	Prints all (or segment) of worksheet.
Q	quit	Exits Multiplan, discards all data not saved.
S	sort	Sorts worksheet rows in desired sequence.
T	transfer	Moves files from disk to memory and vice versa, as well as renaming and deleting files and clearing the computer's memory.
V	value	Used to enter a number or a formula.
W	window	Splits screen operations, displays several worksheets.
X	external	Defines relationships between current worksheet and other worksheet files.

Format Commands

What are the meanings of the Format selections? The ones we saw in Chapter 9 are included as well. Note that we selected F D(efault) followed by C(ells) to get this listing.

The Alignment Field Shows

Def	(Default) Aligns as specified by the Default command.
Ctr	(Center) Aligns all entries in the middle.
Gen	(General) Aligns all text left, all numbers right.
Left	Aligns all entries to the left.
Right	Aligns all entries to the right.
-	Leaves the alignment field unchanged.

The Format Field Shows

Def	Format as specified by the Format Default command.
Cont	Continues long text (over 9 characters long) across column border, if next cell is empty and also formatted with this option ("Cont").
Exp	(Exponent) Displays numbers as a decimal times a power of ten. Number of digits as specified in the next field, the "# of decimals."
Fix	Rounds to and displays number of decimals specified in "# of decimals" field.
Gen	(General) Displays numbers in the most appropriate form considering the size of the cell and of the number.
Int	Numbers will be shown as integers after rounding.
\$	Displays numbers with leading dollar sign and 2 decimal places. (Negative numbers in parentheses)
*	(Bar Graph) Displays as many asterisks as the value of the number.
%	(Percent) Displays the number times 100 followed by a percent sign.
-	Does not change the format field.

The Third Field (# of Decimals) Shows

# of decimals:	This is used in the Exp, Fix and % formats above. The maximum number of decimals is 15.
----------------	---

Mathematical and Engineering Functions

Many of the functions used in mathematical and engineering problems are available in Multiplan. These functions are

EXP(arg)	Returns the "arg" power of the natural number "e".
LOG10(arg)	Returns the base 10 logarithm of its argument. N must be > 0.
LN(arg)	Returns the base e logarithm, or natural log of its argument.
INT(arg)	Returns the largest integer that is less than or equal to n.
MAX(list)	Returns the largest of the values referred to. If list contains no number values, returns 0.
MIN(list)	Returns the smallest of two or more values.
MOD(n,m)	Returns the remainder of the integer division n/m.
ROUND(n,m)	Returns n rounded to m decimal places. If m < 0 rounding is carried to 1 + ABS(m) decimal places.
SIGN(n)	Returns -1 if n < 0, 0 if n = 0, 1 otherwise.

Trigonometric Functions

The usual complement of trigonometric functions is supported by Multiplan. These functions are

ATAN
SIN
COS
TAN

These refer to the arc-tangent, sine, cosine, and tangent functions. The arguments must be expressed in radians.

String Functions

Multiplan provides us with several string functions which help in reducing typing. These include

DOLLAR(n)	Converts text string showing the value n in \$ format.
FIXED(n,m)	Converts the value to text containing n in Fix format with m decimals. When m=0 converts to Int format.
LEN(t)	Returns the length of text (t) in characters.
MID(t,n,m)	Returns m characters starting at beginning (n) of text t.
REPT(t,n)	Reproduces text t n times. Can be used to create bar graphs.
VALUE(t)	If text t contains the representation of a numeric constant, its value will be returned.

Other Functions

There are several functions which have no arguments. These are

COUNT	Returns number of nonblank cells.
ABS	Absolute value.
INT	Integer value.
SQRT	Square root.
ERROR	Useful in testing for errors.
NA()	To designate "Not Available" data.
PI()	Provides the constant 3.1415 . . .

Transfer Commands

All the commands below are subcommands of the Transfer command.

S	Save a worksheet as a file.
L	Load a file into a worksheet. Both commands will prompt: Enter file name (or ← arrow key for directory).
C	Clear a worksheet.
D	Delete a file.
O	Options—allows loading VisiCalc® format or SYLK® format files.
R	Rename a file.

Printing Commands

- P Printer output request.
- F File will be printed on your diskette, for use by word processing programs.
- M Margins to be printed are specified.
- O Options for aid in and control of printing.

15

Using the Optional Diskette

The optional diskette available with this book's Deluxe edition contains only worksheets. This optional diskette assumes you have already purchased the Multiplan software for the Commodore 64 Computer, or will do so shortly.

If you follow the instructions in this book carefully, you will have constructed many Multiplan worksheets. Some of these you may wish to save for future use (such generally useful worksheets are often called "templates"). In order to reduce the time you might have to spend typing values or labels or formulas, each of the significant worksheets we discuss, as well as those used to demonstrate commands and techniques, are being made available on an optional diskette. The worksheets available on the diskette are identified in the text by the notation "<worksheet name>." Each worksheet has a name of the form

Chapter number Uppercase letter

where the chapter number is always preceded by the letter C. Thus <C4A> identifies the first such worksheet from Chapter 4.

Loading a Worksheet

All the worksheets were saved using the TS command, so each one of them can be loaded using the TL command. Keep in mind that loading a worksheet does not erase those entries on your screen that are not being affected by the new worksheet. So you should precede each TL command with a TCY command to clear the screen before loading a new worksheet.

Commodore's disk drive is referred to as unit 8, by default. If you have some other unit, referred to as say 9, you have to specify the "9,:" as a prefix with every use of the load command, T L. So you would type

9, C4A (load worksheet C4A from drive 9,)

Alternatively, you can request the TO command, tab to the "setup:" field and type 9, followed by RETURN. From now on your default drive will be 9,—until you change it—or quit Multiplan.

For most worksheets the loading is very quick. For a few (e.g., multiyear interest tables), it seems to take a bit longer (about three seconds) even though the resulting worksheet has only some forty entries. Why does it take longer? You have to keep in mind that the file which represents a worksheet

does not contain any calculated results. Those files contain the formulas used to calculate the desired results. So whenever you load a worksheet, all the formulas are evaluated. In the case of multiyear interest tables that represent a substantial amount of computing, and hence, a minute or two more to load.

You must always have a copy of the file MP.SYS on a disk in drive 8. *Remember:* It is advisable to first prepare several diskettes (i.e., format a few) then make a copy of your Multiplan's MP.HLP and MP.SYS files on these formatted diskettes. Once this is done, you can always store the original diskette as soon as you have loaded Multiplan to your Commodore 64.

With only one drive you could have several data files on the working diskette in addition to the MP.HLP and MP.SYS FILES. However, that may not give you enough space. You could, however, copy Multiplan's MP.HLP and MP.SYS files to several diskettes giving them separate ID.

Once the Multiplan program has been loaded, you may take it out of the drive, store it, and replace it with one of your newly prepared MP diskettes. Once loaded, Multiplan only needs the MP.SYS file to function. The other file we advised you to copy is MP.HLP which is useful if you need the on-line help. After you become familiar with Multiplan, you can delete the MP.HLP from your data diskette. This will give you more space for your files.

Even if you have two drives, you may find this useful. For example, you could copy the programs from the optional diskette onto one of your MP disks. You can use the utility programs designed to let you copy from one drive to another.

Having done that, you will have a working copy of the optional diskette complete with the Multiplan program. This is often handy even for those with two drives because you could follow an example from the optional diskette, modify it to your liking, then save the personalized version onto another diskette.

To load worksheet C12H from drive 8, you would type

```
T
L
C12H
```

Whether you are using drive 8 or 9, if you followed these directions carefully, you should see the information displayed within a few seconds. (To aid you, the worksheet is reproduced below).

Each worksheet includes its name in some location, so you can be certain that you loaded what you wanted to load. The name will usually appear on the top right or the bottom left of the worksheet.

#1	1	2	3	4	5
1	PROJECTED PRETAX PROFITS FOR THE NEXT FIVE YEARS				
2					
3	Sales in Dollars				
4	Cars	\$80781.77	\$100205.68	\$124300.06	\$154187.91
5	Trucks	\$50891.21	\$52760.65	\$54698.77	\$56708.08
6	Total Sales	\$131672.98	\$152966.34	\$178998.83	\$210895.99
7					
8	Costs in Dollars				
9	Cost of Cars	\$56547.24	\$70143.98	\$87010.04	\$107931.53
10	Cost of Trucks	\$40712.97	\$42208.52	\$43759.02	\$45366.47
11	Commissions	\$6583.65	\$7648.32	\$8949.94	\$10544.80
12	Overhead	\$10000.00	\$10000.00	\$10000.00	\$10000.00
13	Total Costs	\$113843.85	\$130000.82	\$149719.00	\$173842.80
14		----	----	----	----
15	Pretax Profits	\$17829.12	\$22965.52	\$29279.83	\$37053.19
16					
17	<C12HG>				
18					

You might want to duplicate the optional diskette, just in case. Follow the same procedures outlined in Chapter 2, to first “format” a blank diskette, then “backup” an information-bearing diskette onto the freshly formatted diskette.

Diskette Content

All the worksheets of general use are included (e.g., IRA, portfolio evaluation, net worth, sales projection, interest calculations, etc.). The worksheets on the optional diskette and the subject matter are:

C4A-C	Author-income.
C4D-E	IRA.
C5A-K	Gradebook-alias-Production/Sales records.
C6A	Handling big worksheets.
C7A-B	Precipitation graphs.
C7C-D	Market graphs.
C7E-F	IRA graph.
C8A-B	Using trig functions.
C8C-D	Using IF for a payroll.
C8E	Using Lookup to search a table.
C9A-C	Gradebook-alias-Production/Sales records.
C10A-B	Using eXternal to link Home Budget and IRA.
C11A-H	Case study A (expense log).
C11I-P	Case study B (portfolio evaluation).
C11Q-R	Case study C (net worth).
C12A-C	Case study A (sales projection).
C12D-F	Case study B (interest calculations).
C12G-H	Case study C (income statement template).

The optional diskette may be helpful for those of you who can't spare the time to do any avoidable typing or those of you who will do anything to avoid typing. Again, happy computing!

Special Characters (Symbols) Appearing in Multiplan

Arrows (cursor)

- ↓, 16, 28
- , 16, 18, 28
- ←, 16, 17, 28
- ↑, 16, 28

Asterisk *

- Multiplication, 31
- Rounded Value, 70, 94

Caret ^

- Raised to the Power, 31

Colon :

- Through, 46

Comparison Operators

- > Greater Than, 18, 84
- > = Greater Than or Equal to, 84
- < Less Than, 84
- < = Less Than or Equal to, 84
- < > Not Equal to, 84

Dollar \$, 70, 94–95, 118

Equal = , 70, 84

Exclamation Point !

- Recalculation, 97

Minus Sign –

- Does Not Change Alignment, 69
- Subtraction, 117

Parentheses ()

- Coding Arithmetic Expressions, 18

Percent Sign %

- Number Times One Hundred, 70, 94

Plus Sign +

- Addition, 31
- Copy, 35

Pound Sign

- Error Flag, 73
- Number of Decimals, 70, 94
- Overflow, 70
- Possible Error, 53–55

Quotation Marks “ ”

- To Distinguish Between Labels/Text or Values/Formulas, 55

Slash /

- Division, 31

Space ¢, 126

Index

A

ABS, 79
Absolute
 Reference, 38–39
 Reference Key, 54
 Value (ABS), 79
Action
 Function, 20
 Keys, 18, 21–22, 24
Active
 Cell, 21, 35
 Command, 35
Adding (column), 51
Addition, 31
Address, 21–22, 46
Alignment, 66
ALPHA, 20–21, 24, 26, 28–30, 40
AND, 86
Applications
 Cash Flow, 4
 Computing Net Worth, 125–29
 Assets, 126–29
 Liabilities, 127–29
 Gradebook, 42–56
 Interest Computations, 137–41
 IRA (Individual Retirement Account), 28, 37–40,
 75–77, 105–11
 Personal Expense Log, 113–18
 Portfolio Evaluation, 118–25
 Projected Income, 2–4
 Sales Projections, 131–37
Applications (HELP), 22
Argument List, 44–45
Arguments
 AVERAGE, 44–45, 48, 50, 56
 COUNT, 79, 89
 LOOKUP, 87–89, 90
 MAX, 73–78
 MIN, 73–74, 78
 SUM, 44–45, 56
Arithmetic Expression. *See* Value
 Parentheses with, 18
Arrows. *See* Symbols
Asterisk. *See* Symbols
ATAN, 81, 89, 151

AVERAGE (arg), 5, 44–45, 48, 56
 Zero, Dividing by (#DIV/0!), 53
Averages, 53

B

Backspace (INST/DEL), 18
Bar Chart, 68–71
BASIC, 1, 140
BLANK, 53–55
Byte, 6

C

Calculator, 82
Cancel
 CTRL-C, 19, 27
 RUN STOP, 19, 27
Caret. *See* Symbol
Cash Flow, 4
Categories, 117–18
Cell
 Contents, 21
 Pointer, 16–17
 References, 35
 Unlocking of, 95–96
Central Processing Unit (CPU), 6, 14
Chart, Bar, 68–71
Clear-the-Screen, 40
COBOL, 1, 140
Colon. *See* Symbols
Columns
 Address of, 17
 Deletion of, 48–49
 Formatting of, 17–18
 Insertion of, 49–50, 57
 Moving, 56–57
 Options, 17
Command Letters, 24, 28
Command Lines, 9, 20–22, 24
Commands
 ALPHA, 20–21, 24, 26, 28–30, 40
 BLANK, 53–55
 Cancelling, 19, 27
 CLOSE, 11, 13
 CMD, 11, 13
 COPY, 34–37, 43, 53, 123
 DELETE, 20–21, 48–49, 111

EDIT, 54–57
 eXternal, 105–11, 141
 FORMAT, 68–73, 91–94, 96
 GOTO, 16–18, 26–27
 HELP, 12, 19, 22
 INSERT, 49–52
 LIST, 10, 13
 LOAD, 11–13, 24
 LOCK, 95–96
 MOVE, 56–57
 NAME, 46–48
 OPEN, 11–13
 OPTIONS, 22, 83, 96–99
 PRINT, 11, 13, 23, 32–34, 60, 103
 QUIT, 23
 RUN, 12, 14
 SCRATCH, 10–11
 SORT, 62–63
 Summary of MP, 149
 TRANSFER, 41, 101
 Comparison Operators. *See* Symbols
 Completion Test, 83
 Compounding Interest, 83
 Compressed Print, 62
 Computer. *See* Central Processing Unit (CPU)
 Computing, 31, 83
 Computing Net Worth, 125–29
 Conversion Programs, 12
 Coordinates, 17
 COPY, 34–37, 43, 53, 123
 Copying
 MP Files, 154
 One-Drive Files, 154
 Two-Drive Files, 154
 Copy Utility (MP/COPY), 12
 Correcting. *See* EDIT
 COS, 81, 89, 151
 COUNT, 79, 89
 Cursor, 8, 16–18
 Cell Pointer, 21, 23
 Command, 20, 102
 Keys, 28
 Positioning, 27
D
 Data
 Entry, 1, 101–2
 File, 1
 Not Available, 89
 Default, 92–93
 DELETE, 20–21, 48–49, 111
 Delete, 10–11, 20
 Columns, 48–49
 Files, 111
 F 3 Key, 18
 Rows, 48–49, 57
 DELTA, 97
 Disk Operating System (DOS)
 Command Line, 9, 11, 23
 Drive 8 (one-disk), 6, 101, 154
 Drive 9 (second-disk), 6, 101, 154
 Entering Multiplan, 24
 Files, 10
 Formatting Disks, 11
 Returning to, 23–26
 Diskette
 Accessing, 11
 Blank, 8, 11
 Directory, 10, 11, 101
 Files, 6, 10, 101
 Format, 11, 12
 ID (name and number), 11
 Master, 12
 Multiplan (MP), 14
 Optional, Content of, 155

 Source, 12
 Use of, 7–8, 153–54
 Working, 15
 Display Monitors
 Television Screen, 6, 8, 14
 Video Display Terminal (VDT), 4–5, 8, 14
 Division, 31
 #DIV/0!, 53
 DOLLAR, 83
 Dollar Sign. *See* Symbols

E
 EDIT
 Cell, 55
 Command, 54–57, 149
 Cursor, 54
 Formulas, 55
 Keys, 54
 Rules for, 55
 Entering
 Data, 23
 Multiplan, 24
 Numbers, 23
 Text, 23
 Entry
 Content Line, 83
 Protecting, 95–96
 EPSON, 145
 Equal Sign. *See* Symbols
 Erasing, 53–54, 57
 ERROR, 89
 Error Flag, 73
 Exclamation Point. *See* Symbols
 EXP, 80, 89, 150
 Expense Log, Personal, 113–18
 eXternal, 101, 105–11,

F
 FALSE, 86
 Files. *See* Disk Operating System (DOS)
 FIXED, 83
 FORMAT, 68, 71, 91–92, 99
 Format
 Changing Cell, 96
 Command, 68, 71, 91–92, 99
 Default, 92–93
 Fields
 Alignment, 69, 93, 94, 150
 Format, 69, 93, 99, 150
 Third (# of Decimals), 70, 94, 99, 150
 Global, 91
 Changing, 92
 Formatting, 8, 10, 13, 135
 Formulas, 1, 28, 40, 44, 50, 57
 Formulas, Printing, 83
 FORTRAN, 140
 Free Space, 10
 Function Keys, 54
 Functions
 (arg). *See* Arguments
 Engineering, 80, 89, 150
 Logical, 84–86, 89
 Mathematical, 80, 89, 150
 Net Present Value (NPV), 86–87
 Other, 81, 150
 Single Item
 ABS, 79, 89, 151
 COUNT, 151
 INT, 79–80, 89, 94, 151
 SQRT, 80, 89, 151
 String, 83–84, 90, 150
 Trigonometric, 81, 150
 Using, 44

C

Global Formats, 91
 Changing, 92
 Dollar Command, 118
 GOTO, 16-18, 26-27
 Gradebook, 42-56, 95
 Graphs, 73

H

HELP (MP. HLP)
 Applications, 22
 Commands, 22
 Editing, 22
 Formulas, 22
 GENERAL, 19
 Keyboard, 22
 SPECIFIC, 19

I

ID (name and number), 11
 IF, 86
 Income, Projected, 2-4
 Indicator Lights, 7-8
 Individual Retirement Account (IRA), 28, 37-40, 75-77, 105-11
 Initializing, 13
 Input, 1-2, 4
 INSERT
 Column, 49-50, 57
 Row, 49-50, 57
 INT, 79-80, 89, 94, 151
 Interest Computations, 137-41
 IRA. *See* Individual Retirement Account

K

Keyboard, Commodore 64, 18-20
 Keyboard Applications
 Backspace (INST/DEL), 9, 16, 18-20, 23, 27, 55, 95
 Cancel, 22, 27
 Character Left (F4), 54
 Character Right (F6), 54
 CLR/HOME, 9, 16-18
 CTRL, 16-19. *See* Key Combinations
 Edit, 20
 Exclamation Point, 19
 Execute, 20
 Functions (F1-F8), 18, 21, 22, 54
 Action, 18
 Delete (F3), 55
 Edit (F2, F4, F6) 18, 54
 Speed (F5, F7), 19
 INST/DEL (Backspace), 9, 16, 18-20, 23, 27, 55, 95
 Page Left (F7), 17
 Page Up (F5), 17
 PRINT, 9
 Question Mark, 19, 22
 RESTORE, 8
 RESUME, 22
 RETURN, 8-9, 11, 18, 20-23, 26
 RUN STOP, 19, 21, 26-27
 SHIFT, 16, 18-19
 [TAB]
 F1, 17-18, 21-22, 63
 Word Left (F2), 54
 Word Right (F8), 54
 Key Combinations
 CTRL-A, 21
 CTRL-C, 19, 26-27
 CTRL-F, 19, 96
 CTRL-I, 21
 CTRL-R, 19, 22
 CTRL-Y, 55

L

Labels. *See* Titles
 Sorting, 62
 LEN, 84, 151
 Lines, Drawing, 117
 LN, 80, 150
 Loading
 MP (Multiplan) 14, 112
 MPC (Multiplan Color), 14-16
 Optional Diskette, 153
 LOCK, 95-96, 149
 Logarithm, 80, 150
 Logical Functions, 84-86, 89
 LOG 10, 80, 150
 LOOKUP, 87-89, 90

M

Mathematical Functions, 80, 89, 150
 MAX, 75-78
 Message Line, 21, 23
 MID, 84, 89, 151
 MIN, 73-74, 78, 150
 MOVE, 56-57
 Multiplan
 Commands, Summary of, 149-52
 Disk Files
 Directory of, 101-2
 Print, 103, 152
 Saving, 102, 151
 SYmbolic LinK (SYLK), 103-4, 140, 151
 VisiCalc, 104-5, 151
 Entering, 27
 Leaving, 27
 Programs, Creating
 eXternal (X), 105-11

N

NA (), 151
 #N/A, 55
 NAME, 46-48
 #NAME?, 55
 Names, 46-48
 NOT, 86
 #NULL!, 55
 #NUM!, 55, 80
 Numbers
 Rounded, 44
 Sorted, 62

O

On/Off Switch, 8, 13
 OPEN, 11-13
 Option Fields, 17
 OPTIONS, 22, 83, 96-99
 OR, 86
 Output, 1-2
 Output, Graphical, 68
 Overwrite, 25

P

Parentheses. *See* Symbols
 PASCAL, 1
 Percent Sign. *See* Symbols
 PI (), 151
 Portfolio Evaluation, 118-25
 Power Switch, 11, 13-14
 PRINT, 11, 13, 23, 32-34, 60, 103
 Printer, 11, 14

Q

QUIT, 23
 Quotation Mark. *See* Symbols

R

#REF!, 55

Reference

- Absolute, 38–39
- Relative, 41, 52
- REPT, 84, 90, 151

Reverse Video, 10

ROUND, 80, 89, 150

Rows

- Deleting, 48–49, 57
- Inserting, 49–50, 57
- Moving, 56

S

Sales Projections, 131–37

Screen Color Programs (MPC), 12, 14

Scrolling, 18, 58

Scrolling, Synchronized, 67

SIGN, 80, 89, 150

SIN, 81, 89, 151

Slash. *See* Symbols

SORT, 62–63

SQRT, 80, 89, 151

Status Line, 21

Subcommands

- Alternative, 20
- Row-Col, 17

SUM, 44–45

Symbols (Special Characters), 156

T

Tables

- Creating, 1–5

TAN, 81, 89, 151

Template, 141, 143, 145–46

Text. *See* Alpha

Title

- Entering. *See* ALPHA
- Freezing, 64, 78
- Long, 72
- Value as, 55

TRANSFER Subcommands, 151–52

Triangles, 82

Trigonometric Functions, 81, 150

TRUE, 86

V

VALUE, 28, 40

#VALUE!, 55

Value, 28, 40, 53

Video Display Terminals (VDTs), 23–26

W

WINDOW, 64–65

Window, 5, 15–19, 20, 58

- Command, 64–65

- Split, 66–67

Worksheet

- Correcting, 27
- Identifying, 31
- Linking, 105–11
- Margins, 33
- Printing, 32, 59–62
- Retrieving, 25, 27
- Saving, 25, 27, 102
- Scrolling, 58

ZZero, Dividing by. *See* AVERAGE

TO OUR READERS. . . .

Would you like to know more about microcomputer books published by Wm. C. Brown? If so, please provide the following information:

Name _____

Company (if applicable) _____

Street _____

City _____ State _____ Zip _____

Most titles are available through better book and computer stores.

READER'S CRITIQUE. . . .

Wm. C. Brown wants to publish microcomputer books that meet your needs. Your feedback therefore is vital for us to improve current books and publish new and even better books in the future.

The feedback we need from you is outlined in the following questions. This page, when folded along the dotted lines in thirds, can be mailed back to us at no cost to you. Thank you in advance for the valuable information you are providing.

1. In which Wm.C. Brown microcomputer book did you find this form?

Author: _____ Title: _____

2. Please rate this book in the following areas:

	Excellent	Good	Fair	Poor
a. Clarity of information	___	___	___	___
b. Completeness of explanations	___	___	___	___
c. Quantity of Examples	___	___	___	___
d. Usefulness of Examples	___	___	___	___
e. Degree to which Book Met Your Expectations	___	___	___	___
f. Overall Rating of Book	___	___	___	___

3. From what source did you purchase this book?

() Bookstore () Convention Order () Directly from Wm. C. Brown
() Computer Store () Other: _____

4. What brand(s) and model(s) of microcomputer(s) do you own/use?

5. What computer language(s) do you use? _____

6. For how many years have you been using a computer?

() Less than 1 year () 1-2 years () 3-5 years () More than 5 years

7. Which, if any, computer magazines/newspapers do you read regularly?

8. On what topics would you be most interested in purchasing microcomputer books? _____

9. Please suggest ways in which WCB can improve future microcomputer books we publish.

QUESTIONS #10-14 ARE MERELY FOR CLASSIFICATION AND WILL BE USED ONLY TO DEVELOP A STATISTICAL PROFILE OF OUR RESPONDENTS.

10. Your occupation/job title? _____

11. Do you use a computer in your work? () Yes () No

12. Your age?

() Under 18 () 25-29 () 40-49 () 65 and over
() 18-24 () 30-39 () 50-64

13. Are you male or female? () Male () Female

14. What is the highest level of education you have attained?

() 8th grade or less () High school graduate () College graduate
() Attended/attending high school () Attended/attending college () Post graduate study

THANK YOU

wcb



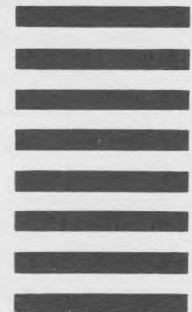
NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED
STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 551 DUBUQUE, IOWA 52001

POSTAGE WILL BE PAID BY ADDRESSEE

WM. C. BROWN PUBLISHERS
Marketing Research Department
2460 Kerper Boulevard
P.O. Box 539
Dubuque, Iowa 52001



ABOUT THE BOOK

Multiplan is the popular software package from Microsoft™ that allows you to make business projections at the touch of a button. Through step-by-step instructions you can master features of Multiplan for your Commodore 64.

ABOUT THE AUTHORS

Michael V. Laric received his M.A. in marketing from the Hebrew University in Jerusalem and his Ph.D. in marketing from the City University of New York. Presently he is Professor of Marketing at the University of Baltimore and a member of the American Marketing Association and the American Academy of Advertising. In addition to a consulting editorship for the Journal of Advertising, Dr. Laric is the author of many publications in the fields of management, consumerism, and productivity.

M. Ronald Stiff received his M.A. in social psychology from Northwestern University, his M.B.A. in marketing from the University of Chicago, and his Ph.D. in management from Northwestern University. His professional career includes academic work for the government and private enterprise. Dr. Stiff is a member of the American Marketing Association, the American Council on Consumer Interests, and the American Psychological Association. His publications include articles and research in the areas of management, marketing, and computer science.

wcb

*Wm. C. Brown Publishers
Dubuque, Iowa*

\$16.95

ISBN 0-697-00424-4