VisiCalc® IV

User's Guide

VisiCorp™

Copyright © 1983 VisiCorp
All Rights Reserved.
Printed in U.S.A.

VisiCalc Program Copyright © 1979, 1983
Software Arts Products Corp.
All Rights Reserved.

StretchCalc Program Copyright © 1983
Multisoft Corporation
All Rights Reserved.

VisiCalc® Program by Software Arts Products Corp.

Graphics, Keysaver™ and other StretchCalc™ features by Multisoft Corporation.

IBM® is a registered trademark of International Business Machines Corporation
DIF™ is a trademark of Software Arts Products Corp.
StretchCalc™ is a trademark of Multisoft Corporation
Anadex® is a registered trademark of Anadex, Inc.
Centronics® is a registered trademark of Centronics Data Computer Corp.
Diablo® is a registered trademark of Xerox Corp.
Paper Tiger™ is a trademark of Integral Data Systems Inc.
Spinwriter™ is a trademark of NEC Information Systems, Inc.
Qume® is a registered trademark of Qume Corporation
Okigraph™ is a trademark of Okidata Corporation

Keysaver™ is a trademark of VisiCorp
VisiCalc® is a registered trademark of VisiCorp

VisiCalc® IV

User's Guide
for the IBM® Personal Computer

Program by
Software Arts Products Corp. and Multisoft Corporation

Manual by
VisiCorp

VisiCorp™

CHAPTER 1. INTRODUCTION

CHAPTER 2. THE VISICALC TUTORIAL

LESSON ONE

LESSON TWO

LESSON THREE

LESSON FOUR

LESSON FIVE

CHAPTER 3. THE VISICALC COMMAND REFERENCE

APPENDIX B.

APPENDIX C.

INTRODUCTION

HOW TO USE THIS MANUAL

This manual is organized to help you learn the VisiCalc® IV program easily and quickly. The organization reflects the fact that different people using the program have different levels of experience with computers. Beginners should follow the logical order in sequence, but experienced users can quickly turn to the sections containing explanations they need.

Note: This manual describes the operation of the VisiCalc IV program only. For information on the IBM® Personal Computer or the disk operating system (DOS), see the IBM *Guide to Operations* manual.

A brief description of the chapters and appendixes follows:

- Chapter 1 provides an overview of the VisiCalc IV program. It includes the information you need to begin using the program: equipment requirements, program and storage disk preparation, and complete loading instructions.
- Chapter 2 contains step-by-step lessons that teach you how to use the basic features of the VisiCalc IV program. These lessons show you what to type, keystroke by keystroke, and should be studied while sitting at the computer.
- Chapter 3 describes each basic VisiCalc IV command, arranged in alphabetical order. These descriptions are summarized in the VisiCalc Command Structure Chart at the beginning of the chapter.
- Chapter 4 describes the StretchCalc™ commands, called the *Star* commands*, in alphabetical order. These descriptions are summarized in the Star* Command Structure Chart. The chapter also explains how to run the Star* command demonstration provided on your program disk.
- Appendix A explains how to send special instructions to your printer using the VisiCalc IV program.

- Appendix B describes how the VisiCalc IV program stores files in the DIF™ interchange format. Using DIF format you can manipulate sections of a worksheet, transfer sections from one worksheet to another, or save data in text files that can be read by other DIF-supporting programs. This appendix includes sample programs for the experienced programmer.
- Appendix C lists the printers and adapters that are supported by—and have been tested with—the VisiCalc IV program.

The most effective way to learn how to use the VisiCalc IV program is to study this manual sequentially. Read Chapter 1 first. When you are familiar with the general features of the program, go on to Chapter 2. There you can let the lessons guide you through each example. In a few short sessions you should be able to use the basic VisiCalc program. Refer to Chapter 3 as necessary for a quick refresher on the basic commands. When you have completed the lessons and feel comfortable with the program, turn to Chapter 4 and follow the instructions for running the demonstration of the Star* commands. You should then be ready to apply the VisiCalc IV program to your own needs.

WHAT IS THE VISICALC IV PROGRAM?

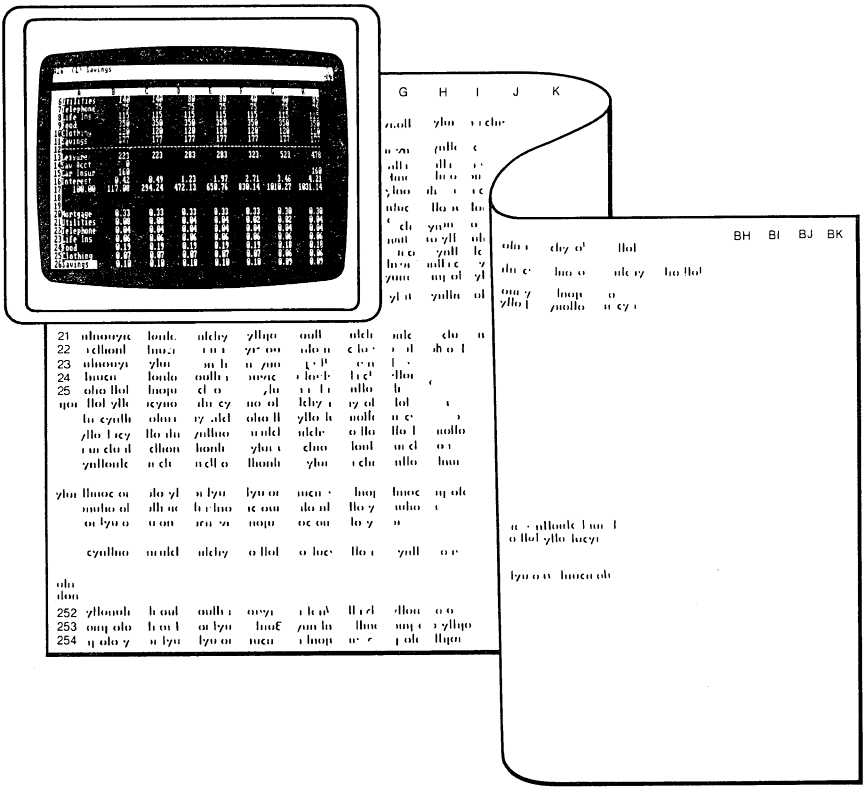
The VisiCalc IV program is an essential tool for the business community. It enhances the features of the original VisiCalc program by adding graphics, sorting, and rearranging capabilities. The new features, called the *Star* commands*, let you reorganize or graph your data quickly and efficiently without leaving the program.

The original VisiCalc program combined the convenience and familiarity of a pocket calculator with the memory and display capabilities of a personal computer. It provided many powerful features, including calculating and recalculating with ease, copying numbers or formulas across the screen quickly, and formatting that could be tailored to your needs. The VisiCalc IV program retains all the original features and adds many more.

With the VisiCalc IV program, you can take the information you have entered and quickly reorganize it into a more appropriate format. Using the same information, you can create several kinds of charts and graphs that summarize the data and help you see trends or significant variations. In addition, you can retrieve information quickly in response to specific queries. And you can do all this without leaving the VisiCalc IV program.

The Electronic Worksheet

With the VisiCalc IV program, the computer's screen becomes a window that shows part of a much larger *electronic worksheet*. You can move, or *scroll*, this window in any direction to look at any part of the worksheet.



005-004
005-020/P

The worksheet is a grid of columns and rows. The intersections of the columns and rows represent entry positions that are identified by their row-column coordinates—for example, position A1 is the intersection of column A and row 1. At each position you can enter either a *value* or a *label*.

One of the strengths of the program is that the computer *remembers* the formulas and calculations you use while working through a problem. If you change any value on the worksheet, all the related values will also change as the program recalculates the entire worksheet.

Recalculation makes the program a powerful planning and forecasting tool. Not only can you easily correct mistakes, but you can also explore alternatives. Playing "what if" is often a matter of changing a single value. For example, you may want to know the impact on your company if a specific product doesn't sell as well as you had anticipated. Finding out takes only seconds.

A value entered at one location can be repeated at other positions. You can change, insert, or delete entire rows or columns. The worksheet is instantly restructured so that all relationships correspond to the changes. Doing the same thing with a calculator, pencil, and paper could take hours of erasing and recalculating.

Once you've set up a worksheet to solve a particular problem, you can save it and use it again for repeated instances or similar problems. All or part of the worksheet can be printed for reports or permanent records.

The Star* Commands

The VisiCalc IV Star* commands expand the capabilities of the the original VisiCalc program to include graphing, sorting and rearranging, and defining your own commands.

Changing the appearance of a worksheet permanently or temporarily is fast and easy. Not only can you rearrange the rows and columns, but also you can treat the worksheet as a visual data base, sorting it or retrieving information according to specific criteria.

Creating graphs from the data on your worksheet is a simple task. You can produce many types of graphs or charts, including area graphs, bar charts, and pie graphs, just to name a few. The Star* commands provide eight different graphs for you to choose from.

Typing the same commands or keystroke sequences over and over can be tedious and cause mistakes. Using the VisiCalc IV program, you can assign a commonly used series of keystrokes to one key. Then you only have to press that key to perform the entire operation, saving you much time and effort.

EQUIPMENT YOU NEED

To use the VisiCalc IV program as it is packaged, you need the following equipment:

- An IBM PC or XT with at least 128K of RAM memory. The computer includes the keyboard unit, a color or monochrome display, and one double-sided disk drive.

Note: If your system has only single-sided disk drives, you need a single-sided program disk. You cannot use the disk packaged with the product. To get your single-sided program disk free of charge, see the *single-sided disk order card* included in the product binder.

- An IBM 80-Character-Per-Second Matrix printer or equivalent, if you want to print the worksheet.
- An IBM Color/Graphics adapter, a color or monochrome graphics monitor, and a graphics printer, if you want to take advantage of the graphics capabilities. See Appendix C for a list of printers and adapters that are supported by the VisiCalc IV program.
- An IBM DOS 1.0, 1.1, or 2.0 disk.
- The VisiCalc IV program disk.
- At least two blank floppy disks. This will be enough to get you started, but you will eventually need more and should plan for it.

Additional floppy or hard disk drives are useful, but not required.

Versions of the Program

The VisiCalc IV program is provided on your program disk in four different versions. They are:

- The original VisiCalc program for a 64K to 512K system with a 40-column monitor.
- The original VisiCalc program for a 64K to 512K system with an 80-column monitor.
- The VisiCalc IV program for a 128K system, which includes graphing, rearranging, and Keysaver™ capabilities.
- The full VisiCalc IV program for a 192K system, which includes all the Star* commands.

These different versions allow you to use the program that corresponds to the amount of memory in your computer. Chapter 4 in this manual describes the Star* commands. The commands that are available in both the 128K and 192K versions of the program are printed normally. The commands available only in the 192K version are shaded with a gray screen.

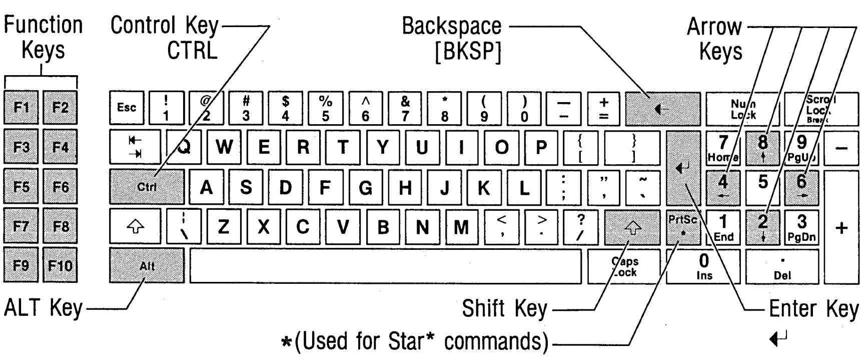
You can use any version that requires *less* memory than you have in your system. This leaves more memory for you to create larger worksheets. If you use one of the smaller versions, however, some or all of the Star* commands will become unavailable to you. If you want to use the full VisiCalc IV program with a larger worksheet, you have to expand the memory that is in your computer.

Memory can be readily expanded by adding IBM-compatible memory cards. The VisiCalc IV program uses this additional memory to increase usable worksheet space. The memory indicator on the screen tells you how much worksheet memory is available. All memory added beyond 128K or 192K, depending on the version of the program you are using, is used as worksheet space.

With larger memory configurations, you might create a worksheet that is too large to store on a disk. See "Storage Command" in Chapter 3 for more information on saving large worksheets.

THE KEYBOARD

The following illustration shows the keyboard. Note the shaded keys in particular; you will be using them often.



036-001

The four keys marked with arrows move the cursor on the screen. The arrow keys are represented in this manual by: ← → ↓ ↑.

The Home key, marked "Home," moves the cursor to the upper-left corner of the worksheet, location A1. Pressing the Home key once is the same as typing >A1. The Home key is represented in this manual by HOME.

The Enter key, marked "↵," is used to end commands and worksheet entries. It is represented in this manual by ↵.

The Backspace key, marked "←," is used to correct typing errors. It is represented in this manual by [BKSP].

The Shift key, marked "⇧," capitalizes letters just as it does on a typewriter. It is represented in this manual by SHIFT.

The Control key, marked "Ctrl," gives special meaning to keys. As with the Shift key, you hold it down and press the other key. This is represented in text by CTRL followed by a hyphen and the other key (CTRL-E, for example, means hold down the CTRL key while you type E).

The Break key, marked "Scroll Lock," is used with the Control key to cancel an entire entry or command. Pressing the two keys also stops the printer or disk drive when a worksheet is being printed or saved. This is represented in this manual by BREAK.

HANDLING FLOPPY DISKS

You can't be too careful with floppy disks. Each disk is a flat, round, magnetically coated plate, sealed in a protective cover. Through the oval cutout in the square cover, you can see the magnetic surface of the actual disk.

Never touch the exposed magnetic surface with your fingers or any implement. Protect the disk from dust by storing it in its paper sleeve. Keep it at least six inches from magnetic fields such as those generated by a TV set. Extremes of temperature (such as in a car trunk on a warm day) can damage a disk, destroying valuable data or your VisiCalc IV program. Don't bend, staple, or write on the square plastic cover with a hard pen or pencil; use only felt-tip pens.

Preparing VisiCalc IV Storage Disks

A blank disk must be *formatted* (prepared for use) before it can store worksheets. The lessons in this manual require at least two formatted disks. Before going further, follow these instructions and format two disks.

1. Make sure the disk operating system (DOS) is ready and A> is displayed.
2. If you have a single-drive system, type **format** and press ↵. If you have a dual-drive system, type **format b:** and press ↵.
3. Insert the storage disk in the correct drive and press ↵.
4. When formatting is complete, press **y** to format another disk or **n** to return to DOS.

IF YOU HAVE ONLY FLOPPY DISK DRIVES

Read this section if you have an IBM Personal Computer with only floppy disk drives. If you have an IBM Personal Computer XT with a hard disk drive, skip to "If You Have a Hard Disk Drive."

The information in this section includes instructions for:

- Putting DOS on the program disk.
- Configuring your program disk.
- Loading the program from drive A.
- Loading the program from drive B.

Putting DOS on the Program Disk

The VisiCalc IV program disk does not include the disk operating system (DOS). Before you load the program, you should copy DOS onto your program disk and then configure your program disk for the version of the program you want to load. Performing these two procedures will make the program disk *self-loading*. This means you won't have to load DOS from a separate disk each time you load the VisiCalc IV program.

Note: If drive A is a single-sided drive, you cannot load the program from that drive. You can, however, copy DOS onto your program disk. If you do not want to copy DOS, skip to "Loading the Program from Drive B."

If you have two disk drives, to copy DOS onto your program disk, you:

1. Load DOS. The system displays A> when ready.
2. Remove the write-protect tab from the VisiCalc IV program disk.
3. Put the VisiCalc IV program disk in your double-sided drive. If drive B is a double-sided drive, put the program disk in drive B. If drive A is your only double-sided drive, put the program disk in drive A, and the DOS disk in drive B.
4. If the program disk is in drive B, type **sys b:** and press ↵. If the program disk is in drive A, type **sys a:** and press ↵. The necessary parts of DOS are copied onto the program disk.
5. When the screen displays A>, choose one of the following:
   (a) If the program disk is in drive B, type **copy a:command.com b:** and press ↵.
   (b) If the program disk is in drive A, type **copy b:command.com a:** and press ↵.
6. When the system completes processing, remove the VisiCalc IV program disk and replace the write-protect tab.

If you only have one disk drive, you will have to change the disk that is in drive A several times while you are copying DOS onto your program disk. To do this, follow steps 1 and 2 in the previous instructions. Then, type **sys b:** ↵ and follow the instructions on the screen. When the screen displays A>, type **copy a:command.com b:**↵ and again follow the instructions on the screen. When the system completes processing, remove the program disk and replace the write-protect tab.

You are now ready to configure your program disk.

Configuring the Program Disk

Follow these instructions if drive A is a double-sided drive. If drive A is a single-sided drive, skip to "Loading the Program from Drive B."

As described earlier, the VisiCalc IV program is provided on your program disk in several different versions. These versions correspond to the memory you have in your system. The following procedure allows you to specify which version of the program you want to load automatically. For information on what each version contains, see "Versions of the Program" earlier in this chapter.

Note: You should choose the version you expect to use most of the time. You can load any of the other versions manually. In addition, you can reconfigure your program disk at any time. You may want to do this, for example, if you increase the memory in your system. To avoid accidentally damaging your program disk, however, you should reconfigure it as little as possible.

To configure the program disk:

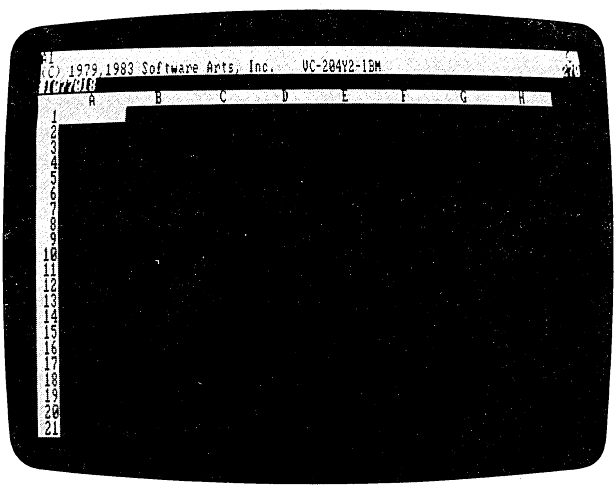
1. With A> displayed, put your program disk in drive A.
2. Type **vconfig** and press ↵.
3. Follow the instructions on the screen by removing the write-protect tab from the program disk and returning the disk to drive A. When you have finished, press ↵.
4. Type **1**, **2**, **3**, or **4**, depending on which version of the program you want to load automatically.
5. The program asks you for the name of a Keysaver command file. You will learn about Keysaver commands later in this manual. For now, press ↵.
6. Remove the program disk and replace the write-protect tab.

You are now ready to load the program.

Loading the Program from Drive A

Follow these instructions if you have made your program disk self-loading. If drive A is a single-sided drive, skip to "Loading the Program from Drive B."

1. Insert the program disk in drive A.
2. Turn on your computer (or, if the computer is on, hold down the CTRL, ALT, and DEL keys simultaneously). The VisiCalc IV program is loaded automatically. Your screen should look like the following photograph:



036-002/P

The second line on the screen contains the copyright notice and the version number. The third line contains the serial number of the disk. Write down the serial number and store it in a safe place.

When the program has loaded, the disk drive has stopped whirring, and the busy light is off, open the drive door and remove the VisiCalc IV program disk. Put it back in its sleeve with the label showing.

Once the program is loaded, you can go to Lesson One in Chapter 2 to learn to use the VisiCalc IV program.

If you want to load any of the versions manually, follow the instructions in "Loading the Program from Drive B." However, since drive A in your system is double-sided, you can skip steps 2 and 3.

Loading the Program from Drive B

If drive B is your only double-sided disk drive, you must load DOS from a separate disk before you can load the VisiCalc IV program. To do this:

1. Make sure that A> is displayed.
2. Put your VisiCalc IV program disk in drive B.
3. Type **b:** and press ↵.
4. With B> displayed, you can load one of four versions of the program. For information on what the different versions contain, see "Versions of the Program" earlier in this chapter. Choose one of the following:

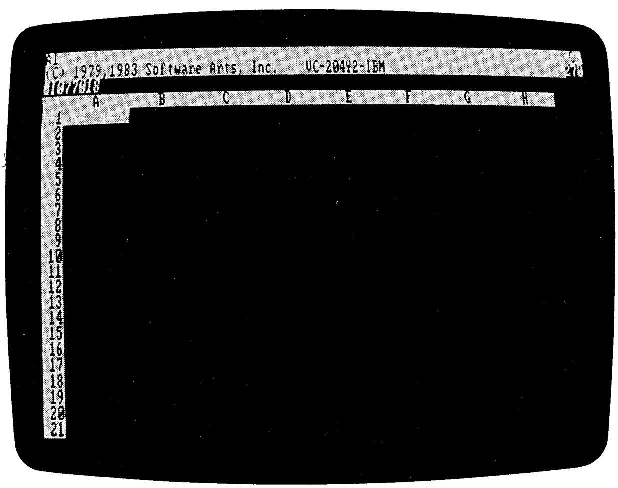
   (a) If you want to load the 128K version of the VisiCalc IV program, type **vc128** and press ↵.

   (b) If you want to load the full version of the VisiCalc IV program, type **vc192** and press ↵.

   (c) If you want to load the 40-column 64K version of the VisiCalc program, type **vc40** and press ↵.

   (d) If you want to load the 80-column 64K version of the VisiCalc program, type **vc80** and press ↵.

Your screen should look like the following photograph:



036-002/P

The second line on the screen contains the copyright notice and the version number. The third line contains the serial number of the disk. Write down the serial number and store it in a safe place.

When the program has loaded, the disk drive has stopped whirring, and the busy light is off, open the drive door and remove the VisiCalc IV program disk. Put it back in its sleeve with the label showing.

Once the program is loaded, you can go to Lesson One in Chapter 2 to learn to use the VisiCalc IV program.

IF YOU HAVE A HARD DISK DRIVE

If you have an IBM Personal Computer XT, we recommend that you install (copy) the VisiCalc IV program onto the hard disk and load the program from the hard disk. The program will be more convenient to use.

You can still load the VisiCalc IV program from the program disk in the floppy disk drive if you choose. Turn back to "Putting DOS on the Program Disk" now if you do not want to install the program on the hard disk.

Installing the Program on the Hard Disk

Before you can install the VisiCalc IV program on your hard disk, you must already have formatted the disk and installed DOS 2.0 on it. If you have *never* done either of these procedures, follow the instructions in the IBM *Disk Operating System* manual. If you have formatted the hard disk and installed DOS, you do not have to do it again.

When the hard disk has been properly prepared, you can install the program. To do this:

1. Load DOS 2.0 from the hard disk. The C> prompt appears.
2. If you want to put the VisiCalc IV program in its own directory on the hard disk, create the directory by typing **mkdir** followed by a directory name and pressing ↵. Then access that directory by typing **chdir** followed by the directory name and pressing ↵. The MKDIR and CHDIR commands are explained in the IBM *Disk Operating System* manual.
3. Type **a:** and press ↵.
4. Put the VisiCalc IV program disk in the floppy disk drive.
5. After the A> prompt, type **install c:** and press ↵.

6. Wait while the VisiCalc IV files are copied onto your hard disk. When the procedure is finished, the message VisiCalc IV installation completed appears. If the message VisiCalc IV installation failed appears, repeat step 5, being careful to type the correct drive letter.
7. Type **c:** and press ↵.

Now that you have installed the program, store the VisiCalc IV program disk in a cool, dry place for safekeeping.

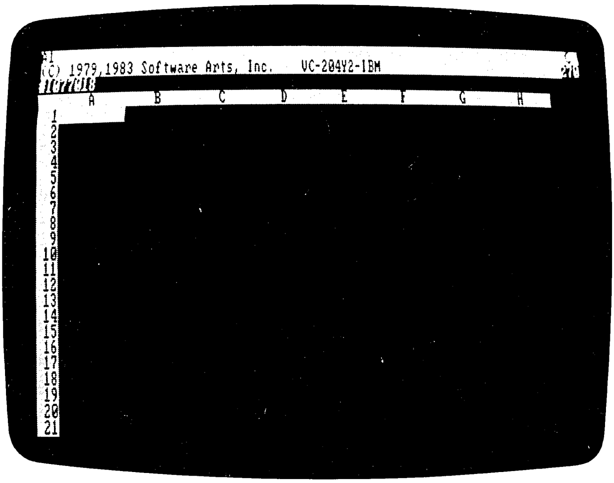
Loading the Program from the Hard Disk

To load the VisiCalc IV program, follow these steps:

1. Start with the DOS C> prompt showing. You can load one of four versions of the program. For information on what the different versions contain, see "Versions of the Program" earlier in this chapter. Choose one of the following:
   (a) If you want to load the 128K version of the VisiCalc IV program, type **vc128** and press ↵.
   (b) If you want to load the full version of the VisiCalc IV program, type **vc192** and press ↵.
   (c) If you want to load the 40-column version of the VisiCalc program, type **vc40** and press ↵.
   (d) If you want to load the 80-column version of the VisiCalc program, type **vc80** and press ↵.

If you installed the program in its own directory (see step 2 in the previous section), each time you load the program you must first use the CHDIR command to access the VisiCalc IV directory; then type the correct loading instruction and press ↵.

2. After a short period, the VisiCalc IV screen appears. Your screen should look like the following photograph.



036-002/P

The second line on the screen contains the copyright notice and the version number. The third line contains the serial number of the disk. Write down the serial number and store it in a safe place.

Special Instructions for Using the VisiCalc IV Program with the Hard Disk

Because you have installed the VisiCalc IV program on the hard disk, you should keep the following points in mind as you read the remainder of this manual:

- To save a worksheet file on the hard disk, type the save command (/**SS**); then type the file name of the worksheet. The worksheet will be saved in the same directory that contains the VisiCalc IV program. You cannot use files on the hard disk that are not in the VisiCalc IV directory.
- To load a file, type the load command (/**SL**); then type the file name. When the instructions in Chapter 2 tell you to load a file you have just saved, the prefix is listed as A: in the text. For you, the prefix will be C:.
- To save or load a file from a floppy disk, type the save or load command (/**SS** or /**SL**), and type the prefix **a:** before the file name. The standard configuration for the IBM Personal Computer XT specifies the one floppy drive as both drive A and drive B. Because floppy drives A and B are really the same drive, you should never need to type the **b:** prefix.

  If you ever do type the **b:** prefix and want to scroll through the list of file names by pressing →, you must press → twice to see the first file name, and once again to see each subsequent file name on the disk. When you switch back to the **a:** prefix, the first time you scroll through the list, you must also press → twice to see the first file name.

Now turn to Lesson One in Chapter 2 to learn how to use the VisiCalc IV program. If you want to use the original VisiCalc program, see the next section, "Using the VisiCalc Program."

USING THE VISICALC PROGRAM

For your convenience, the original VisiCalc program is also provided on your program disk. Using this version, you can create large worksheets or use large worksheets that have been created with an earlier version of the VisiCalc program. Remember, however, that in the original VisiCalc program, you cannot use *any* of the Star* commands.

To load the VisiCalc program:

1. Load DOS. Make sure that the A> prompt is displayed.
2. Put your VisiCalc IV program disk in your double-sided drive. If your program disk is in drive A, skip to the next step. If your program disk is in drive B, type **b:** and press ↵.
3. Choose one of the following:
   (a) If you want to load a 40-column version of the program, type **vc40** and press ↵.
   (b) If you want to load an 80-column version of the program, type **vc80** and press ↵.

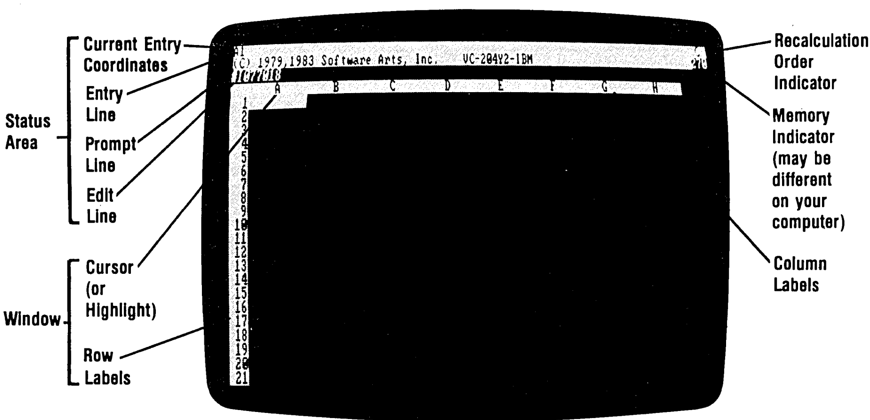
   You can now use the VisiCalc program.

THE VISICALC TUTORIAL

Chapter Two, the VisiCalc Tutorial, consists of five lessons that introduce all the major features of the VisiCalc program. The lessons are written to be used as a keystroke-by-keystroke guide while sitting at the computer.

The lessons require at least one formatted diskette. If you have not yet formatted a diskette, do so now, following the instructions under "Preparing VisiCalc Storage Diskettes" in Chapter 1.

To follow the lessons, load the VisiCalc program following the instructions under "Loading the VisiCalc Program" in Chapter 1. (If the VisiCalc program is already loaded, type /CY to clear the screen.) The screen should look like the following photograph:



036-002/P

The screen has become a *window* into the computer's memory. The VisiCalc program has organized it as an *electronic worksheet* by dividing it into rows and columns. Rows are numbered 1, 2, 3, etc., and columns are lettered A, B, C, etc. Each intersection of a row and a column is an *entry position;* it is referred to by its coordinates (A1, B3, C17, etc.).

Now turn to Lesson One.

LESSON ONE

At each entry position, you can type a message, a number, or a formula. In a moment you'll write at different entry positions on this electronic worksheet.

Look at the white bar and dark line at the top of the screen (just above the column letters). This is the *status area*. The bar is actually two lines; the top line is the *entry line* and the second line is the *prompt line* .

The prompt line is displaying the VisiCalc copyright notice and version number. Should you ever need to call or write to ask questions or to report problems with the VisiCalc program, please refer to your version number and the model of your computer.

The dark line below the prompt line is the *edit line*. It contains another important number: the serial number of your VisiCalc diskette.

The letter in the upper right corner of the screen tells you whether the VisiCalc program calculates new values by rows (R) or by columns (C). The 2-digit number below it tells you how much memory is available.

Press the key marked ↵ . The copyright notice, version number, and serial number disappear. Now type /V and all three appear. Type /V (the Version command) whenever you want to see the VisiCalc version number.

MOVING THE CURSOR

Look at the intersection of column A and row 1; this is location A1. A white rectangle, the *cursor* , covers it. Think of the cursor as the point where your pencil meets the paper. It marks where you can write on the worksheet. To keep you from getting lost on the worksheet, the VisiCalc program displays the cursor's coordinate on the entry line.

The arrow keys move the cursor in the direction indicated by the arrow. Type →. The cursor moves to B1 (column B, row 1). Look at the entry line. Not only are the copyright and version numbers gone, but the new coordinate (B1) is displayed. Type ← and the cursor moves back to A1.

Type ↓. The cursor moves down to entry position A2 (column A, row 2). Now type ↑. The cursor returns to entry position A1.

If, while you are practicing, you make a typing error and see "Value" or "Label" on the prompt line, don't worry. Just type BREAK. The word disappears, along with the number or letter beneath it. The sound when you type BREAK is the VisiCalc program's way of telling you that you canceled a command.

SCROLLING THE WINDOW

When you load the VisiCalc program, the cursor is at A1, the upper-left corner of the worksheet. Type → until the cursor rests at the right edge of the screen. Type → again. The next column to the right comes into view, while column A disappears off the left edge.

The window has *scrolled* to the right. Type → a few more times, watching more columns appear at the right edge of the window while others disappear at the left.

The window also scrolls to the left. (In fact, it scrolls in all four directions.) Type ← until the cursor is at the left edge of the window. Now type ← several more times; the columns that disappeared as you scrolled the window to the right come back into view. Type ← until the cursor is back at A1.

Now type ← once more. The sound is the VisiCalc program's way of telling you that you're bumping into an edge of the worksheet. Type ↑. This time you hear the noise because you're bumping into the top edge of the worksheet. So far you have encountered the left edge and the top edge of the worksheet. There are two other edges.

Type ↓ until the cursor moves down to the bottom of the window. Now type ↓ once more. The next row comes into view, while row 1 disappears off the top of the window. Type ↓ a few more times. As you can see, the worksheet is quite a bit larger than an ordinary sheet of paper.

REPEAT ACTION

You can scroll to the bottom edge of the worksheet more quickly by taking advantage of the repeat action of the keyboard. Hold down ↓; the cursor and the window scroll down rapidly. Continue to hold down ↓ until you bump into the bottom edge of the worksheet. The cursor has reached position A254.

To move the cursor to the right edge of the worksheet, hold down →. The cursor and window go scrolling off to the right. As they scroll, notice how succeeding columns are lettered. (These coordinates also appear in the upper left corner of the screen.) After A, B, C, . . . Z come AA, AB, AC . . . AZ then BA, BB, BC . . . The cursor finally stops at position BK254 as it bumps into the right edge of the worksheet. You are now at the lower right corner of the VisiCalc worksheet.

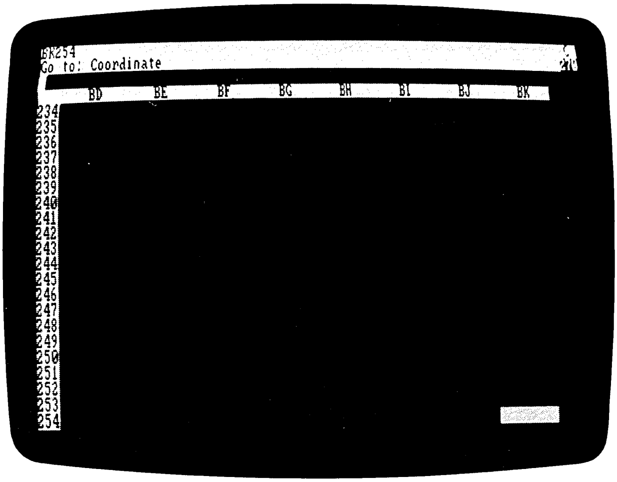
DIRECT CURSOR MOVEMENT

Even with the aid of repeat, it takes a while to scroll all the way to the lower right corner of the VisiCalc worksheet. There's an easy way to move the cursor to another position on the worksheet that takes only a few keystrokes.

Type >. Two things happen:

- "Go To: Coordinate" appears on the prompt line just under "BK254"
- The edit cue (the white box) appears on the edit line, directly below the prompt line.

The screen should look like the following photograph:



005-003/P

In general, each time you press a key the prompt line tells you what you can type next. On the screen in the preceding photograph, the prompt line tells you that the VisiCalc program has recognized the Go To command (>) and wants to know what Coordinate to Go To. It is waiting for you to type in the letter and number of the coordinate to which you want the cursor to move.

Type **A**. The letter "A" appears on the edit line (the dark line just above the column labels), followed by the edit cue. Type 1 to specify position A1.

So far, "A1" is on the edit line followed by the edit cue. The VisiCalc program is still waiting for you to type something. It doesn't know yet whether you want to go to position A1, or A11, or A121, or some other position.

Type ↵. The prompt and edit lines clear, and the cursor and window move back to the upper-left corner of the worksheet at A1.

Try another example. Type >C10 ↵. The cursor now rests in mid-screen, directly below "C" and to the right of "10".

EDITING WITH THE [BKSP] (BACKSPACE) KEY

Earlier, you used BREAK to cancel something you typed. The VisiCalc program has a less drastic way of correcting errors—the backspace key, just to the left of the NUM LOCK key (marked with ←, represented in this manual by [BKSP]).

Type >**A1 1**, but pause for a moment before typing ↵. Suppose you intended to move the cursor to position A1, but accidentally typed 1 twice. The edit line now reads:

A11

followed by the edit cue.

Type [BKSP] once. The edit cue backs up one character and erases the extra 1, leaving "A1". Type ↵. The cursor jumps to entry position A1, and the prompt and edit lines clear.

In general, the VisiCalc program lets you correct typing errors by backing up with the [BKSP] key; each time you type [BKSP], the last character on the edit line is erased. Type >**A1 1** again. Now type **[BKSP]** twice, leaving just "A", then type 2 to get "A2" and ↵ to end the command. The cursor moves to A2.

Besides backing up, you can cancel a command with [BKSP]. Type >**B5**, then pause. Suppose you change your mind and decide you don't want to move the cursor. Type **[BKSP]** and the 5 disappears from the edit line. Type **[BKSP]** again; the "B" disappears from the edit line and "Go To: Coordinate" disappears from the prompt line. You have canceled the Go To command by erasing everything on the edit line; this accomplishes the same thing as typing BREAK.

Before going on, spend a few more minutes moving the cursor around with the arrow keys and >(the Go To command). Try moving the cursor to a nonexistent position such as AB525. What happens? Try moving to an invalid coordinate such as 25A instead of A25. What happens?

WRITING ON THE WORKSHEET

As you have seen, moving the cursor and window around is pretty easy, but so far your worksheet is (or should be) empty. You'll find that writing on the worksheet is even easier.

Type /CY to clear the worksheet. The worksheet disappears, then reappears with the copyright on the prompt line and the VisiCalc serial number on the edit line. The Clear command erases the worksheet and positions the cursor at A1.

To begin, label a row: type **Sales**.

Stop and look at the prompt line. The word "Label" appears on the prompt line; this is the VisiCalc program's name for a worksheet entry that isn't used in making calculations. On the edit line is the word "Sales" followed by the edit cue. The edit cue indicates that you can use [BKSP] to back up and make corrections or BREAK to cancel the command. "Sales" also appears under the cursor at position A1 on the worksheet.

Now type →. The prompt and edit lines clear and the cursor moves to position B1, leaving the label "Sales" at A1. You can use any arrow key instead of ↵ to end an entry and write the label or value on the worksheet.

Type 100. Look at the status area again. The prompt line says "Value", the VisiCalc program's term for a number or formula. The number 100 followed by the edit cue appears on the edit line. Type [BKSP] three times and watch the numbers disappear: first 0, then 0, then 1.

Now type the formula 75+25. If you make a typing error, watch the edit line and type [BKSP] to back up.

The word "Value" is still on the prompt line, and the edit line reads "75+25", followed by the edit cue. Type !. The VisiCalc program calculates 75+25 and displays 100 on the edit line. You can use this feature to perform quick calculations before writing a number on the worksheet.

Nothing appears under the cursor at B1; everything has happened on the edit line. Now type ↵. The prompt and edit lines clear and 100 appears at position B1 on the worksheet. The entry line reads

```
B1 (V) 100
```

The entry line gives a full explanation of the contents of the entry position highlighted by the cursor. Right now the entry line reads "B1" (the coordinate), "(V)" for Value, followed by "100".

Try typing [BKSP]. Nothing happens. Typing ↵ told the VisiCalc program to write 100 on the worksheet. Type ←, and the cursor moves back to position A1. Now the entry line reads:

A1 (L) Sales.

The "(L)" stands for Label.

FORMULAS AND RECALCULATION

Move the cursor down to position A2 by typing ↓. Type **Cost**, then →. The cursor moves to B2, leaving "Cost" at A2.

You're going to write a formula for cost at B2 that says cost is 60% of sales, or .6 times 100. Instead of typing 100, however, you'll use its coordinate (B1). Type .6***B1**.

The * specifies multiplication. The edit line should now read ".6*B1". If it doesn't, use [BKSP] to correct any error. The formula tells the VisiCalc program to multiply whatever is at coordinate B1 by .6. Now type ↵ and watch what happens. The prompt and edit lines clear, and the entry line reads:

B2 (V) .6*B1

Position B2 reads 60, the result of multiplying .6 times 100, the number at B1.

To give you a quick look at the VisiCalc program's power, type ↑ to move the cursor up to position B1. Watch the screen and type 200 ↵. The new number (200) replaces the old number (100) at B1. What else happened?

Type ↓ to move the cursor to B2. The formula you typed earlier, ".6*B1", is on the entry line at the top of the screen. When the number at B1 changed to 200, the VisiCalc program recalculated the formula at B2 as .6*200, or 120. Cost is still 60% of sales. You'll see many more examples of this recalculation feature.

MORE ON LABELS AND VALUES

This topic looks more carefully at labels and values, and explores an even simpler way to write formulas. You'll start by writing a formula that calculates gross profit by subtracting cost from sales. First, position the cursor to write a new label: type >A3 ↵ to move the cursor to A3.

To write the label "Gross", type **G**. The prompt line says "Label." When you write at an entry position, the VisiCalc program looks at the first character you type to determine whether you are typing a label or a value. If you start with one of the letters A through Z, as you did here, the VisiCalc program assumes that you are typing a label.

If you start with one of the digits 0 through 9, a decimal point (.), something that could begin a formula—such as plus (+), minus (–), or left parenthesis (()—or @ or # (explained later), the VisiCalc program assumes that you are typing a value. For now, type **BREAK** to cancel what you have typed.

What happens if you want to write a label such as "–Gross–" or "1st Qtr"? Try typing **-Gross-**. The VisiCalc program interprets the initial dash as a minus sign. It assumes you are typing a formula, and that the letter G starts a coordinate. However, it beeps an error and refuses further input when you type something that can't be a formula. Type **BREAK** to cancel this entry.

To begin a label with a symbol that the VisiCalc program interprets as the beginning of a value, type a quotation mark (") as the first character. The VisiCalc program takes the " to mean that you want to type a label regardless of what the next character is. The quotation mark does not become part of the label itself. As soon as you type ", the prompt line reads "Label" followed by the edit cue.

Type **"-Gross-** (don't type a closing quotation mark unless you want it as part of the label). Now type →; the cursor moves to position B3, leaving "–Gross–" at A3.

> **Note:** To begin a label with a quotation mark, you must type " twice at the beginning; once to identify the entry as a label and again to enter the quotation mark itself.

You're ready to calculate sales minus cost. The formula is sales (B1) minus cost (B2). Type **B1-B2 ↵**. What happens when you type B1–B2? Look at the prompt line. It shows that the VisiCalc program assumed you were entering a label, not a value, as soon as you typed the first B. B1–B2 is a perfectly legitimate label, but it doesn't calculate anything. Only a value (number or formula) can calculate.

Start again by eliminating the label "B1–B2". To erase an entry you have written on the worksheet, use the Blank command, which blanks out the entry position where the cursor rests. Type **/B** ↵. The label B1–B2 under the cursor disappears, and the entry line clears except for the coordinate "B3".

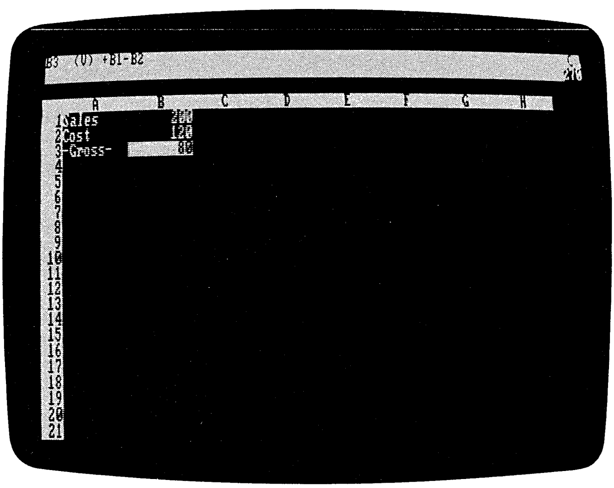
Remember that you used " to type a label when the VisiCalc program would otherwise have treated it as a value? A similar technique makes the VisiCalc program accept the formula B1–B2 as a value instead of a label. The formula +B1–B2 is equivalent to B1–B2, and VisiCalc interprets the + as the first character of a value. Type the following formula for gross profit (be sure to include the +):

+B1-B2 ↵

The entry line reads:

B3 (V) +B1–B2

The cursor rests on 80, the result of calculating +B1–B2, or +200–120.



005-058/P

CURSOR MOVES IN FORMULAS

So far, you know how to move the cursor to an entry position and how to write labels and numbers. You also know how to write a formula (such as +B1–B2 in the previous example) that refers to other positions on the worksheet. You have also seen that if you change the numbers at B1 or B2, the VisiCalc program recalculates the formula +B1–B2 and displays the result at the formula's coordinate, B3.

As you wrote the formula for sales minus cost, you probably had to check the screen to see that the number for sales was at position B1 and that the number for cost was at B3. Imagine what it's like when you write a large number of formulas on the worksheet.

Keeping track of the coordinates of many numbers can be time-consuming and somewhat confusing. Further, when you scroll columns A and B off the left edge of the window to work on other parts of the worksheet, you can't see the numbers next to "Sales" and "Cost". This makes writing a new formula involving sales and cost rather difficult. There's an easy way to solve this problem: you write the formula, but let the VisiCalc program fill in the coordinates.

Type +. The prompt line reads "Value" and the "+" appears on the edit line, followed by the edit cue. At this point you could type B1, but what you really want is the number next to the label "Sales" (currently 200).

Point to the 200 on the screen with your finger. You're about to do the same thing electronically by pointing with the cursor. Watch the edit line and type ↑. The cursor moves up to highlight 120; its coordinate, "B2", appears on the edit line. Now type ↑ again. See what pointing the cursor means? You have taken the cursor from its starting position at B3 and pointed it to B1.

The edit line now reads "+B1" followed by the edit cue. These are the first three characters of the formula you typed earlier. Type - (the hyphen, for a minus sign). The cursor jumps back to B3, the entry position at which you began writing the formula. The edit line now reads "+B1–" followed by the edit cue. As a rule, after pointing the cursor at the position you want to include in the formula, simply continue the formula by typing an arithmetic operator such as –, +, *, or /.

Now enter the second part of the formula by pointing to it. Watch the edit line and type ↑ once more. The cursor moves up to 120, the number next to Cost, and the edit line now reads "+B1–B2"—exactly the same formula you typed before. To write the formula on the worksheet as it appears on the edit line, type ↵. The prompt and edit lines clear.

The result of pointing to coordinates with the cursor is the same as typing those coordinates. The entry line still reads:

```
B3 (V) +B1-B2
```

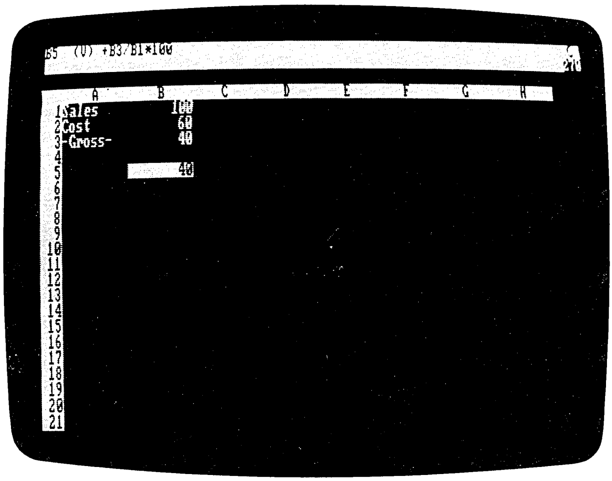
The number on which the cursor rests is 80, the result of calculating B1 minus B2. Nothing has changed except the way you wrote the formula.

To demonstrate the VisiCalc recalculation feature again, type ↑ twice to move the cursor up to B1. Change the number there by typing 100 ↵. Now B2 changes back to 60 (.6 times 100), and B3 changes to 40 (100-60).

In general, whatever you see on the edit line you can type. Likewise, you can point to any position on the worksheet instead of typing its coordinates.

As you gain experience and familiarity with the VisiCalc program, you'll find that the technique of moving the cursor to the positions you want becomes easier. In time, you'll find you can almost forget about coordinates entirely and think only in terms of the visual positions of labels and values on the worksheet.

To test your understanding of the process of moving the cursor as you write formulas, try an example yourself. Move the cursor down to position B5 and write a formula there for gross profit as a percentage of sales. Hint: the formula is gross profit divided by sales, multiplied by 100 (+B3/B1*100). Write this formula on the worksheet just by moving the cursor and typing + and /. The screen should look like the following photograph:



005-004/P

After you obtain the result (40), keep the cursor at B5 and type /B ↵ to blank the coordinate and >**B1** ↵ to move the cursor to **B1**.

MORE ON EDITING

Suppose you wanted to change the cost in the example from 60% to 55%. Move the cursor to B2. The entry line reads:

B2 (V) .6*B1

To change that formula, type /**E** (the Edit command). The prompt line reads:

[Edit]: Value

(because the entry is a value) and the formula .6*B1 is displayed on the edit line; the edit cue (the light block) is over the first character, the decimal point.

To change .6 to .55, type → twice to move the edit cue to the *, just to the right of the character to be changed. Nothing has changed so far except the position of the edit cue. Now type [BKSP]. This works just like it did when you erased with [BKSP] before. The edit cue backs up one space and erases the 6. The edit line now reads ".*B1". Type **55**. The edit line reads ".55*B1"—the formula you want. Type ↵. The prompt and edit lines clear and the entry line reads:

B2 (V) .55*B1

The number at B2 has changed to 55 and B3 is 45.

Typing [BKSP] erases the character just to the left of the edit cue. Typing a character puts that character on the edit line just in front of (to the left of) the edit cue.

To change cost back to 60% of sales, type /**E** →→→ **[BKSP] [BKSP]** 6 ↵. The formula returns to its previous form.

The Edit command can be entered in one of two ways. Typing /E when the edit line is clear lets you edit the contents of the cursor location. Typing CTRL-E while typing a label, value, or command lets you edit what is on the edit line.

You can edit anything you type. Type **>A6** ↵ **January Profit Margin**. Suppose you meant to type "February" instead of "January". Type **CTRL-E** (hold down **CTRL** and type **E**). This time the prompt line reads "[Edit]: Label" because you are typing a label. Type ← until the edit cue is on the blank just to the right of the "Y." Type **[BKSP]** seven times to erase "January", then type **February.** If you were to type ↵ now, you would enter "February Profit Margin" at A6. For now, return to the original sheet by typing **BREAK >B1** ↵.

SAVING THE WORKSHEET ON DISKETTE

If you save the work from this lesson, you can carry it over into Lesson Two. You can save it on one of the diskettes that you initialized earlier. (If you haven't initialized any diskettes yet, follow the instructions under "Preparing VisiCalc Storage Diskettes" in Chapter 1.)

If you don't have any extra diskettes, don't worry. Lesson Two starts by telling you what to type to set up the worksheet with the same labels, numbers, and formulas now written on it. You should, however, at least skim the material in the remainder of this lesson and the beginning of Lesson Two that describes how to save and load the worksheet.

A saved worksheet is called a *file*. The name that identifies a file is called a *file name*. Because a diskette can hold several worksheets, you must give each worksheet on the same diskette a different file name. This way, you can find the worksheet later and load it into the computer.

There are a few simple rules for creating a valid file name:

- It can be up to 8 characters long
- The first character must be a letter
- Only letters and numbers can be used

If the storage diskette isn't in drive A, you must add a *prefix* to the file name that specifies the drive. The prefix consists of the letter that identifies the drive followed by a colon (:). To specify a file named "budget" in drive B, for example, you would type "b:budget".

If you save a file named "b:budget", for example, then later save a file named "forecast", the VisiCalc program tries to save "forecast" on drive B. To save it on a diskette in another drive, you must specify the drive with a prefix.

The VisiCalc program adds the suffix ".VC" to the file name of a worksheet you save with the Save option of the Storage command. This lets you later recognize the file as a VisiCalc worksheet.

To save the worksheet, open the door of the drive. If the VisiCalc program diskette is still in the drive, remove it and put it back in its paper sleeve. Insert a formatted storage diskette, gently push it all the way in, and close the door.

Now type **/S** (the VisiCalc Storage command). The prompt line reads:

Storage: L S D Q #

The characters following "Storage:" are options:

L Load the worksheet contents from a diskette.
S Save the worksheet in a diskette file.
D Delete a file from a diskette.
Q Quit the VisiCalc program.
Save or load a worksheet in the DIF format.

Type **S**. The prompt line reads "Storage: File for Saving" and the edit cue is on the edit line. The VisiCalc program is waiting for you to type the file name. Because this worksheet is an example, type **example** ↵. As usual, you can correct typing errors with [BKSP]. The disk drive begins whirring and the light comes on. After a moment the drive quiets down, the light goes off, and the prompt and edit lines clear. The worksheet is safely filed away on the storage diskette.

PROTECTING YOUR WORK

Have you ever worked out a problem or made some notes to yourself on a sheet of paper, only to find later that you lost the sheet or that someone accidentally threw it away? Or perhaps you lost the results of some work you were doing on a calculator because power was turned off or the battery died. Things can and do go wrong.

As you begin to use the VisiCalc program, you'll find that at times the results you see on the screen are quite important to you. Losing the information could be a real nuisance and, at worst, something of a disaster. How can you protect yourself against such losses?

The VisiCalc program does its best to protect you. If you type the Clear command (/C), the VisiCalc program displays "Clear: Type Y to confirm" on the prompt line. It erases the worksheet only if you type Y at this point. If you type anything else, the Clear command is canceled and the worksheet is left unchanged.

Similarly, if you type /B to blank an entry, the prompt line displays "Blank" but the entry is not erased unless you type ↵ or one of the arrow keys. Any other key cancels the Blank command.

However, things beyond the VisiCalc program's control can go wrong. What if your building has a power failure? What if someone pulls the plug from the socket? What if you are called away from the computer and the janitor turns it off?

SAVING THE WORKSHEET

To protect yourself, you should periodically save the worksheet on diskette. As you work, think how long it has been since you last saved the worksheet. If you have spent more time than you would care to lose if something goes wrong, or if you have results which might be difficult to reconstruct, then it's time to save the worksheet.

To keep track of several versions of the same worksheet on diskette, you can add a sequence number to the file name. Thus, you might save successive versions of a worksheet with the file names "budget1", "budget2", "budget3", etc. (Remember, the VisiCalc program adds ".VC" to the file name you type.)

Note: If you must remove a file from a diskette to make room for your worksheet, use the Delete option of the Storage command (/SD). See "Storage Command" in Chapter 3 for details on saving, loading, and deleting files.

Printing the worksheet is another security measure. You can print not only the worksheet, but also the formulas and formats behind it. See "Print Command" and "Storage Command" in Chapter 3 for descriptions of how to print the content and structure of the worksheet. With printed copies of the worksheet and its underlying formats and formulas, you can reconstruct the work without a disk file, if necessary.

MAKING BACKUP COPIES

Although a diskette is a safe and reliable medium for storing information, saving your work periodically is only the first step in protecting yourself. A diskette can be scratched or pick up grease and dust. It can be damaged by heat, exposed to a magnetic field, accidentally reformatted (which erases it), and eventually it wears out. The average lifetime of a diskette is about 40 hours of use (whenever the disk drive light is on, the diskette is in use). For complete protection, you should make backup copies of your important files on separate diskettes.

Use the Storage command to make a backup copy of a file. Insert the diskette that contains the file you want to copy into drive A (or whatever drive you are currently using). Then use the Load option of the Storage command (/SL) to load the worksheet. Remove the diskette, insert a formatted diskette, and use the Save option of the Storage command (/SS) to save the same worksheet on the second (backup) diskette.

You can also copy an entire diskette. See the operating manual for instructions. Test your understanding of the procedures on a diskette you make expressly for experimentation. Until you thoroughly understand the procedures, don't risk a diskette with important data on it; you may lose valuable data while learning.

Making backup files is important. It's all too easy to read about these protective measures, use them once or twice, then when you're in a hurry skip the backup steps. Remember Murphy's Law: If anything can possibly go wrong, it will. Protect yourself. The time it takes to make backup files is amply repaid the first time you try to load a file, only to get a message telling you that an error occurred reading the disk.

SUMMARY

This completes Lesson One. You may wish to experiment for a while, moving the cursor around and writing your own labels, numbers, and formulas. Try writing some formulas by pointing the cursor to specify the coordinates. The VisiCalc program has many more features not yet described, and you may stumble upon one of them. If something happens that you don't understand, make a note of it. When you're ready to continue with the next lesson, turn to Lesson Two. Most of your questions will be answered as you go.

You already know enough about the VisiCalc program to use it for some simple applications. Try it!

LESSON TWO

This lesson extends your knowledge of the capabilities of the VisiCalc program learned in Lesson One by applying them in more powerful ways and by learning some new commands. It begins with the example built up in Lesson One, calculating gross profit from sales and cost of goods sold.

If you have just finished Lesson One, everything you need should still be on the worksheet. Check the screen against the following photograph and continue with the heading "Replicating a Formula."

To practice loading the worksheet you saved in Lesson One, type **/CY** and continue with the heading "Loading the Worksheet."

If you're starting Lesson Two in a new session, the first step is to load the VisiCalc program. Follow the instructions under "Loading the VisiCalc Program" in Chapter 1, then continue with the heading "Loading the Worksheet."

If you didn't save the worksheet on diskette, type the following:

HOME
Sales → 100
HOME↓
Cost → .6*B1
HOME↓↓
"- Gross-→ +B1-B2
HOME→

The screen should look like the next photograph. Continue with the heading "Replicating a Formula."

LOADING THE WORKSHEET

Remove the VisiCalc program diskette. Take out the storage diskette on which you saved the worksheet and put it in drive A. Be sure the label on the diskette jacket is up and on the side of the diskette opposite the drive door. Gently push the diskette all the way in and close the drive door.

Type **/S** (the Storage command). The prompt line reads:

Storage: L S D Q #

To review the meanings of the option characters, see "Saving the Worksheet on Diskette" at the end of Lesson One.

Type **L**. The prompt line reads:

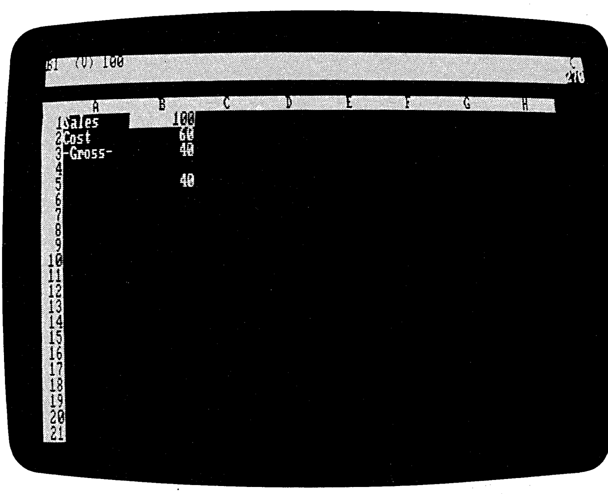
Storage: File to Load

At this point you could type "example.vc ↵", but there's another way: you can display the file names, one by one, from the storage diskette.

Type →. The disk drive whirs for a moment, the light comes on, and "A:EXAMPLE.VC" appears on the edit line. (If a different name appears, type → until you see "A:EXAMPLE.VC". The VisiCalc program is reading the names of the files stored on the diskette.)

If you keep typing →, you eventually display each file name on the diskette. The prompt and edit lines clear, and the VisiCalc program waits for you to type another command.

Assuming that "EXAMPLE.VC" is on the edit line, type ↵. The word "Loading" replaces "File to Load" on the prompt line, followed by a blinking asterisk (*), the VisiCalc program's way of telling you it is loading the file. When the file is loaded, the screen should look like the following photograph:



005-005/P

REPLICATING A FORMULA

The screen should look like the preceding photograph. The cursor should be at B1; if it's not, type **HOME** →. If the number at B1 is not 100, type 100 ↵.

At present, the worksheet contains figures for sales, cost of goods, and gross profit for only one month. Now to project these figures for 12 months.

Begin by assuming that sales will increase by 10% each month. Type → 1.1* ← ↵ to move the cursor to C1 and write the formula. The entry line reads:

C1 (V) 1.1*B1

The number under the cursor at C1 is 110 , which is 1.1*100.

To calculate sales for the remaining 10 months, you would have to move the cursor to D1 and type the formula 1.1*C1, move on to E1 and type 1.1*D1, and so forth. Since this is such a common operation and requires so many keystrokes, the VisiCalc program provides a shortcut.

That shortcut is the VisiCalc Replicate command. It copies—or *replicates*—formulas, labels, numbers, formats, etc., down columns or across rows. This lesson uses the Replicate command in some simple examples. This versatile command is explored more fully in Lessons Three, Four, and Five.

With the cursor at C1, type /**R** (the Replicate command). The prompt line reads:

Replicate: Source range or ENTER

The VisiCalc program is asking what you want to replicate; you can replicate either a single entry or a range of entries. The edit line shows "C1", the coordinate of the formula on which the cursor rests, followed by the edit cue. The source range is only one location, so type ↵ to select C1 as the single location to be replicated. The edit line reads "C1 . . . C1:" followed by the edit cue. You've told the VisiCalc program *what* to replicate (the formula at C1). Now the prompt line reads:

Replicate: Target range

The VisiCalc program needs to know *where* to replicate (the target range). Your intent is to project sales for 12 months. The first month is shown at B1, the second at C1, so the twelfth month is at M1. The formula should be replicated in the target range D1 to M1.

Type → to move the cursor to D1. Now type a period (.) to tell the VisiCalc program that you have specified the beginning of the range. The VisiCalc program displays three periods following "D1". The edit line reads:

C1 . . . C1: D1 . . .

You can define the end of the target range by moving the cursor. Type → until the cursor rests on M1. As you move the cursor to D1, E1, F1, etc., the target range on the edit line expands accordingly:

D1 . . . D1

D1 . . . E1

D1 . . . F1

and so on.

The VisiCalc program is filling in the final coordinate of the target range, just as it did when you moved the cursor while typing a formula. When the cursor is on M1 the edit line reads:

C1 . . . C1: D1 . . . M1

If you move too far with → , you can move back with ←. Now type ↵. The cursor moves back to C1, where the original formula is written. The edit line reads:

C1: D1 . . . M1: 1.1*B1

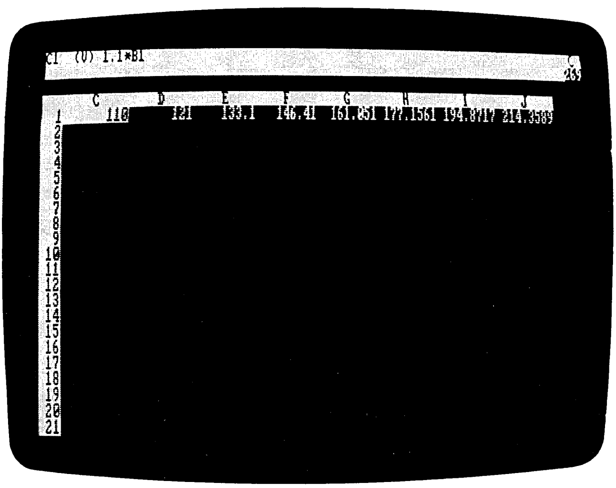
The edit cue highlights "B1". The prompt line reads:

Replicate: N= No Change, R= Relative

The VisiCalc program is asking whether you want the same formula, 1.1*B1, at each of the positions in the target range D1 . . . M1, or whether the coordinate B1 should be interpreted as *relative* to the position of the formula.

Sales are to increase by 10% each month, so the formulas should be 1.1*B1, 1.1*C1, 1.1*D1, etc. In other words, the coordinate B1 should be *relative* to the location of each copy of the formula. That is, each new sales figure should be 1.1 times the previous month's sales.

Watch the screen as you type **R**. The prompt and edit lines clear and numbers appear. The screen should look like the following photograph:



005-006/P

Use → to move the cursor to D1, E1, and F1, and watch the entry line:

1.1*C1

1.1*D1

1.1*E1

and so on.

Type → to scroll from G1 through M1. The VisiCalc program has written the formulas for you and calculated the sales values for all 12 months. M1, the twelfth month's sales, should read 285.3117 (to four decimal places).

Other Replicate options are covered in succeeding lessons. For the moment, however, remember these easy steps for replicating a *single* coordinate:

1. Position the cursor at the location you want to replicate.
2. Type /**R** to start the Replicate command.
3. Type ↵ to define the source range as one coordinate.
4. Point the cursor at the first location where the formula is to be copied.
5. Type a period (.).
6. Point to the last location (you could also type the coordinates instead of pointing to them).
7. Type ↵.
8. In response to the prompts, type either N or R to specify whether the coordinate should be copied with no change or relative to the position of each copy.

REPLICATING A RANGE OF FORMULAS

Type **HOME** → ↓ to move the cursor to B2. To complete the 12-month projection, you'll replicate the formulas for cost of goods sold and gross profit. At the moment, the entry line shows the formula for cost of goods. If you were to move the cursor to B3, you would see the formula for gross profit, "+B1− B2." You can replicate both these formulas simultaneously across the worksheet.

Type /**R**. The prompt line reads:

```
Replicate: Source range or ENTER
```

"B2" is on the edit line, followed by the edit cue. If you were to type ↵ at this point, as you did before, you would replicate just the formula for cost of goods sold at B2. When you finished, you would have to come back to the formula for gross profit at B3 and replicate it into the same range of columns. You can accomplish both these steps at once. Type **.B3** ↵. The edit line now reads:

B2 . . . B3:

The VisiCalc program acknowledges that you want to replicate a source range of two coordinates, B2 through B3, instead of just one. The prompt line reads:

Replicate: Target range

Instead of pointing at the first and last positions as you did before, type the coordinates of the target range: **C2.M2** ↵.

When you specify a source range of B2 . . . B3 and a target range of C2 . . . M2, the VisiCalc program assumes that the formula at B2 (cost of goods) is to be replicated at positions C2 through M2, and the formula at B3 (gross profit) is to be replicated at positions C3 through M3.

The prompt line now reads:

Replicate: N= No Change, R= Relative

The edit line reads:

B2: C2 . . . M2: .6*B1

The edit cue is over the coordinate B1.

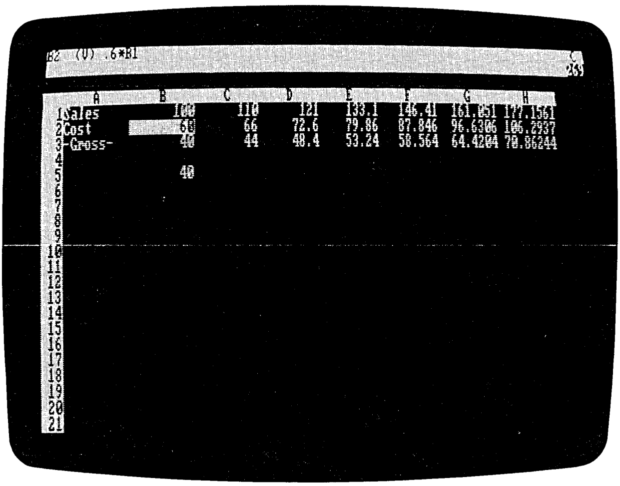
How should the formula at B2 be interpreted? The cost of goods sold in any given month should be 60% of that month's sales. The formula at B2 uses the sales figure above it, or B1. The formula at C2 uses the sales figure above *it*, or C1; and so on. Therefore, B1—the coordinate for "Sales"—should be *relative* to the position of each copy of the formula. Type **R**. Almost instantly, numbers appear in the other columns. The edit line now reads:

B3: C3 . . . M3: +B1

The edit cue highlights "B1." The VisiCalc program is ready to replicate the formula for gross profit at B3 into positions C3 through M3. "+B1" is the beginning portion of the formula +B1-B2. Again, B1 should be relative to the position of each copy of the formula. Type **R** again.

Now the rest of the formula appears on the edit line: "+B1–B2. " This time the edit cue highlights "B2", which contains the formula for cost of goods; it, too, should be relative in the gross profit formula. Type **R** once more.

Numbers appear in the other columns of row 3, and the prompt and edit lines clear. The Replicate command has finished its work. The screen should look like the following photograph:



005-007/P

Scroll across row 3 to examine the formulas and results displayed in columns C, D, E, and so on. Finally, type >**M1** ↵ to display the last month's sales, cost of goods, and gross profit in column M. The VisiCalc program has saved you a good deal of work already. But these numbers are somewhat hard to read because they fill the columns and don't always line up. This can be improved.

FORMATTING THE DISPLAY

You can change the entire worksheet—a *global* change—so that it displays integers. Type /**GFI** (for Global Format Integer). Scroll the window to the left; all the numbers on the worksheet have been rounded to integers and lined up on the right side of each column.

The VisiCalc program, however, does not actually round the numbers; it merely displays the rounded values. Each value is calculated and maintained with 11 (and sometimes 12) significant digits. Each new month's sales, therefore, is based on an accurate version of the previous month's sales, not on the rounded values as displayed. The numbers are rounded only for display, not calculation purposes.

Perhaps you'd prefer to see two decimal places (the Visi Calc program calls this *Dollars-and-cents* format). Type/ (the keystroke that starts all commands). The prompt line reads:

```
Command: BCDEFGIMPRSTVW-
```

Each of the characters following **Command:** represents a different command. So far, you have used /B (Blank), /C (Clear), /R (Replicate), and /S (Storage). Now type **G**. The prompt line reads:

```
Global: C O R F
```

The Global command changes some aspect of the entire screen display. You want to change the global format from Integer to Dollars-and-cents, so type **F** for Format. The prompt line reads:

```
Format: D G I L R $ *
```

The VisiCalc program is ready to change the format (the way labels and values are displayed and printed) over the entire worksheet. A moment ago, you used the letter I to display all numbers in the Integer format. This time, type **$**. The screen displays all values to two decimal places.

The Format option list (D G I L R $ *) offers various ways to format numbers and labels. You can set the format of an individual entry or the entire worksheet. These options are covered more fully in Lessons Three, Four, and Five.

FIXING TITLES IN PLACE

If the cursor is not on M1, type >**M1** ↵. Only numbers are on the screen. The labels "Sales", "Cost", and "– Gross– " have scrolled off to the left.

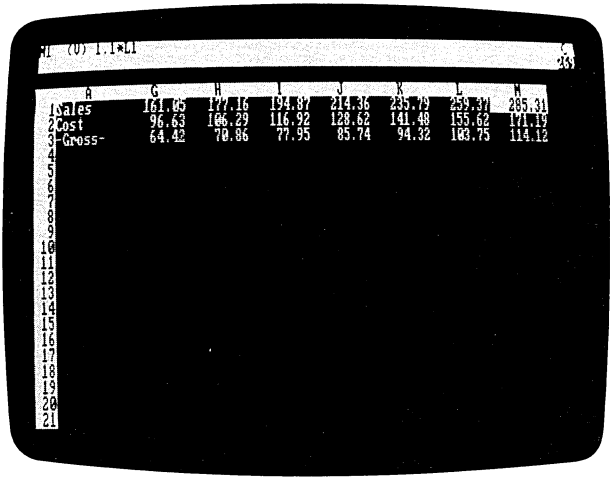
What if you were preparing a more complex income projection, with many rows of numbers for selling and administrative costs, taxes, and other expenses? It would be difficult to remember what each row of figures represented once the labels scrolled off the screen. Here's a way to keep the left edge visible no matter how far you scroll to the right.

First, put the cursor back to the left edge, where the labels are. Type **HOME** to bring "Sales", "Cost", and "– Gross– " into view. Now type /**T** (the Titles command). The prompt line reads "Titles: H V B N". The characters following "Titles:" are options:

H Horizontal titles
V Vertical titles
B Both horizontal and vertical titles
N No titles

Type **V** to tell the VisiCalc program to fix vertical titles (column A, where the cursor lies) as the left column of the screen. Watch the screen and type → until the window scrolls. Column A stays fixed in place while the remaining columns scroll to the left, disappearing when they reach column A.

Now type ← until you scroll back to the titles. If you aren't sure what happened, type ← again. You are bumping into column A, just as you bumped into the left edge of the worksheet earlier. Next type >**M1** ↵. Column A is still visible, making it easy to identify each row of numbers. The screen should look like the following photograph:



005-008/P

A QUICK RECALCULATION

So far, with the aid of the Replicate command, you have written one number (the beginning number for sales at B1) and 35 formulas on the electronic worksheet. How are these formulas related? Type **HOME→** to bring the first column into view with the cursor on the initial sales figure.

The formula for cost at B2 is ".6*B1", which depends on the figures for sales at B1. The formula for gross profit at B3, in turn, depends on both sales and cost ("+B1–B2"). What about succeeding columns? At C1, the formula is "1.1*B1", so this entry also depends on the initial sales figure. Cost of goods at C2 depends on the figure at C1, while C3 depends on both C1 and C2. D1 reads "1.1*C1", and so on.

As you can see, a change to the initial sales figure at B1 affects every other number on the worksheet. Try it. Type a new number, such as 123.45 ↵. How long did it take to recalculate 12 months of sales, cost of goods, and gross profit formulas? Type 100 ↵ and watch the changes ripple through the other figures on the screen. Scroll the screen window to the right to see M1, the last month's sales.

At this point, you might be wondering how to change the percentages themselves—sales growth figure of 10% or the cost of goods percentage of 60% – to recalculate gross profit. For simplicity's sake, the only changeable figure in this example is the initial value for sales. Because the figures 1.1 and .6 are built into each of the 12 formulas for sales and cost of goods, you can't change these percentages without replicating all the formulas again.

A more flexible approach would be to write the factors 1.1 and .6 into separate positions on the worksheet, and make the sales and cost of goods formulas refer to these positions. In this way, changing the sales growth and cost of goods percentages would be as easy as changing the initial sales. Lesson Three uses techniques like this in an example related to personal budgeting.

CHANGING THE COLUMN WIDTH

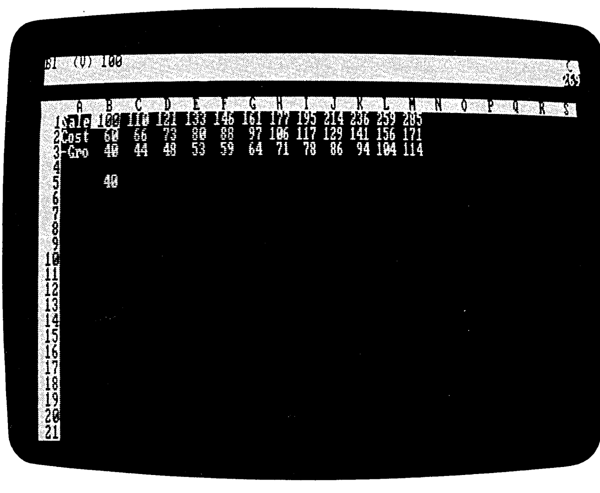
For some time now, you've been scrolling the window back and forth to see the figures for different months. To eliminate some of this scrolling, you can make the columns smaller so that more columns fit on the screen. Type **HOME→** to return the cursor to B1.

Type another Global command to change the column width to 7 characters: **/GC7** ↵. In an instant, the screen displays more columns. Each column has narrowed from nine characters to seven.

In general, you can use the Column Width option of the Global command (/GC) to set the column width to any value from 3 to the maximum number of characters that fit on the screen. Given a column width, the VisiCalc program fits as many columns as it can across the screen.

Right now, the numbers with two decimal places just about fill the available space in these 7-character columns. Type **/GFI** to round the numbers to integers. With this extra space you can narrow the columns further. Type **/GC4** ↵. Even more columns of figures are on the screen; each is four characters wide and displays three digits.

The VisiCalc program leaves the first character of each location blank when a value is displayed to keep columns of numbers from running together. Labels do not have this extra space. The screen should look like the following photograph:



005-009/P

Look at the labels in column A. "Sales" has been shortened to "Sale" and "– Gross– " is now "– Gro". Type **HOME** (if you type ← to get to A1, you bump into column A, which you fixed as a title). Although the label at A1 is displayed as "Sale", the entry line above reads:

A1 (L) Sales

Type ↓ twice to move to A3. The entry line displays the full label:

A3 (L) – Gross–

Just as the VisiCalc program retains the full precision of numbers when it displays rounded values, so does it retain the full length of a label when narrow columns require displaying a shortened version.

Labels aren't limited to nine characters. You can type a label as long as 125 characters, regardless of the current column width; the VisiCalc program accepts the full label. With the cursor still at A3, type **Gross Profit** ↵. Then type ↑ **Cost of Goods Sold** ↵ to move up to A2 and enter a more descriptive label. Finally, type **/GC12** ↵.

The label "Gross Profit" is displayed in full; the number of columns on the screen is reduced. Type **/GC18** ↵. Now the even longer label "Cost of Goods Sold" is displayed. Type **/GC9** ↵ to return to the standard column width.

Note: The VisiCalc program makes all columns in a window the same width. You can only have different column widths on the screen if you split it into two windows (described later in this lesson).

To display a label longer than the current column width, but keep a narrow column width to display a large number of columns, enter the label in two or more columns. For example, to display the full label "Cost of Goods Sold" beginning in column A with a column width of nine, type the following:

Cost of G→oods Sold ↵

The label is displayed in A2 and B2. You can no longer use B2 for numbers or formulas.

The ability to fix titles in place and adjust column widths lets you take maximum advantage of the screen But suppose you wanted to change the initial sales figure at B1 and watch what happens to the final sales and gross profit in column M at the same time.

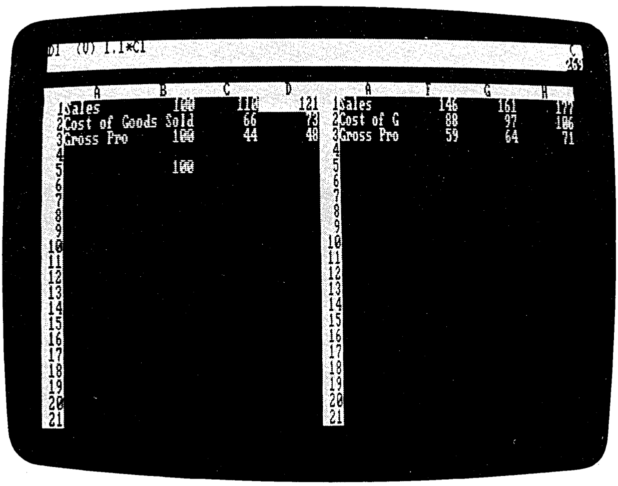
If only there were two screens . . .

SPLITTING THE SCREEN

Type **>E1** ↵ to move the cursor to column E, then **/W** (the Window command). The prompt line reads:

Window: H V 1 S U

Type **V** for Vertical. The screen should look like the following photograph:



005-010/P

You have created two windows, each of which can be scrolled independently to view any portion of the worksheet. The cursor is in the left window. Move the cursor down until the window scrolls down to follow it; the right window remains still. Bring the left window back to the top of the worksheet with HOME.

Type a semicolon (;) to move the cursor into the right window. Scroll the right window across to column M. Now the beginning and ending months' sales, cost of goods, and gross profit figures are visible at the same time.

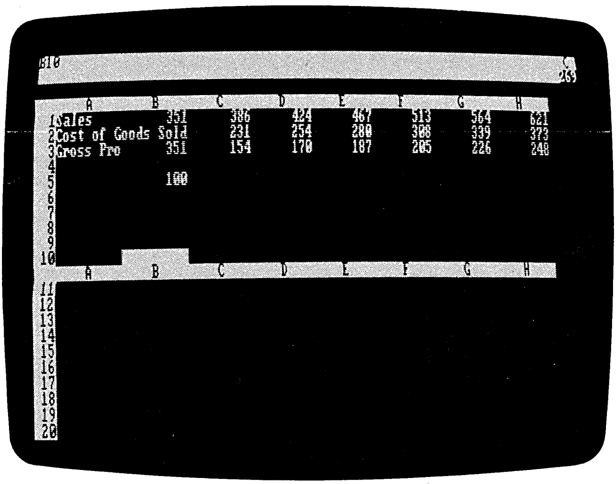
Type ; again. The cursor moves back to the left window. Each time you type ; the cursor moves to the other window. It moves to the same position it was on when it was last in the window.

Now you can change the initial sales figure and see what happens in the final month. With the cursor at B1 type **123** ↵. (While the VisiCalc program is recalculating, an exclamation point appears to the right of the "C" in the upper-right corner of the screen.) "Sales" in column M should be 351. Type a few more numbers with the cursor at B1.

See if you can find, by trial and error, the initial sales figure that yields twelfth month sales of 1000. Hint: You can type a number with a decimal point, even though it is displayed in rounded form.

When you finish experimenting with recalculation, type /**W1** for one window, the normal screen. Much of the screen is empty; perhaps you can use the lower part of the screen to better advantage.

Type >**B11** ↵ to move the cursor down to the middle row of the screen, then /**WH** (for horizontal windows). The screen should look like the following photograph:



005-011/P

This time the screen is split horizontally into a top and bottom window. Type ; to move the cursor into the bottom window. Scroll the bottom window up until the cursor bumps into the top edge of the worksheet. Both windows now display the same portion of the worksheet.

The cursor should be at B1 in the bottom window; change the number there to 100. The recalculation affects both windows. Scroll the bottom window to the right until column M comes into view. Now you can see the figures for the first and last month at the same time.

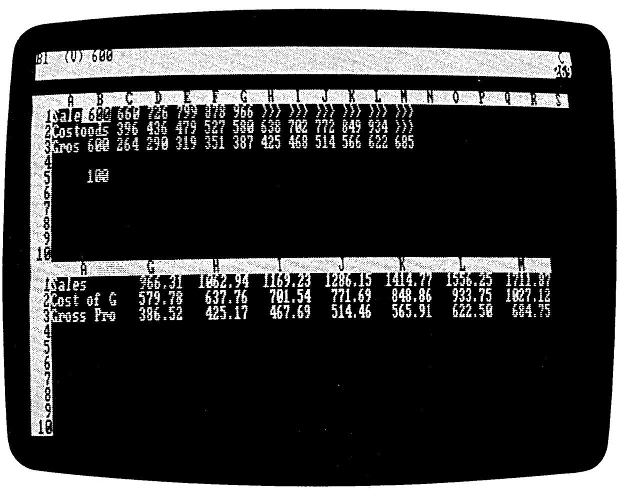
GLOBAL COMMANDS IN SEPARATE WINDOWS

The VisiCalc program allows different column widths in each window. Type ; to move the cursor into the top window, then /**GC**4 ↵ to change the upper window column width to four characters.

Note: The Column Width and Format options of the Global command (/GC and /GF) affect *only* the window in which the cursor rests at the time the command is typed.

You've just done a global column change. Now try a global format change. Type ; to move the cursor to the bottom window, then /**GF$**. Numbers are displayed to two decimal places (Dollars-and-cents format) in the bottom window and as integers in the upper. Type ; to move the cursor into the top window, then **HOME**→ to highlight the original sales figure. At B1, type 300 ↵ and watch the changes ripple through the columns as the VisiCalc program recalculates all the formulas.

Now type 600 ↵. What happens? Columns H through M show >>> in some positions instead of numbers. The calculated results are too large to display in integer form in the narrow columns of the top window. The screen should look like the following photograph:



005-012/P

MEMORY AND THE WORKSHEET

The VisiCalc program expands the size and shape of the worksheet as you use it. It starts as a 1-by-1 worksheet, beginning and ending at position A1. Although you can move the cursor to any position up to BK254, no memory is taken up by the worksheet until you write something on it.

The worksheet grows into a rectangle just large enough to include the rightmost and bottommost positions in which something is written. If you type a label or formula at an entry location, the VisiCalc program uses enough memory for the position, but all other positions on the worksheet remain just large enough to hold the information you have written in them.

The VisiCalc program displays the amount of memory available in the right corner of the prompt line just below the "C." This number is the amount of memory, in units of 1024 characters, available for additional entries on the worksheet. The number varies as you type labels or values or use commands (such as Replicate) to write on the worksheet.

As you write on the worksheet further down and to the right, more and more memory is used. If you finally exhaust all available memory, the VisiCalc program replaces the memory indicator with "0M" and refuses to write anything more on the worksheet.

SHRINKING THE SHEET

The VisiCalc program does not automatically shrink the worksheet. Suppose you have written on various portions of the worksheet, causing it to grow to a 100-by-100 rectangle, then erase or blank out the entry positions near the right and bottom edges. The worksheet remains a 100-by-100 rectangle, with each empty position requiring 2 characters of memory. If you begin using additional memory by writing labels and formulas in other positions, you may run out of memory even though much of the worksheet is empty.

To shrink the worksheet, you must save it on diskette with the Save option of the Storage command (/SS), clear it with the Clear command (/CY), and reload it with the Load option of the Storage command (/SL). As the VisiCalc program reloads the worksheet, it enters only those labels, numbers, and formulas actually saved. The worksheet grows from a 1-by-1 rectangle to just the size needed for the information saved. All unused memory is available for more labels and formulas.

SUMMARY

You have covered a lot of ground in this lesson. Review any difficult points and try your own experiments. The more you work with the VisiCalc program, observing and analyzing the results, the more quickly you will master it for your own work. Just remember the following key points:

- No matter what you type at the keyboard, you cannot hurt either the computer or the VisiCalc program. Moreover, it's fairly difficult to destroy the contents of the worksheet, particularly if you watch the prompt line for keystroke-by-keystroke feedback and save the worksheet periodically on diskette.

- This lesson introduced only four new commands:

  — Replicate (/R), which copies labels, numbers, and formats.
  — Global (/G), which changes the column width (/GC) and the way numbers are formatted (/GF).
  — Titles (/T), which fixes rows or columns of titles in place as part of the top or left border.
  — Window (/W), which splits the screen, either horizontally or vertically, into two independently scrollable, formattable windows.

- Aside from the Replicate command, which saves you time as you write on the worksheet, all the commands described in this lesson affect only the *appearance* of the worksheet (generally in an effort to take greatest advantage of the screen). Nothing you might do with the Global, Titles, or Window commands can affect the labels, numbers, or formulas actually written on the worksheet. When in doubt, you can always type **/W1/TN/GFG/GC9** ↵ to return everything to normal.

Armed with what you have learned up to this point, you should be ready to experiment. Clear the screen and try out these commands with a problem of your own. If you don't understand something, go back through this lesson to see what you might have missed. For more details, check the command descriptions in Chapter 3 or the *VisiCalc Pocket Reference*. Once you understand the fundamentals of this lesson, you'll know enough about the VisiCalc program to use it effectively.

LESSON THREE

In Lessons One and Two, several examples illustrated both the simplicity and the power of the VisiCalc program. Lesson Three expands on the use of previously learned commands, bringing them into more powerful combinations, and introduces several new commands.

The worksheet is set up as a personal budget to present this. Work through the examples, and don't hesitate to experiment. Your skill in using the VisiCalc program grows proportionally with the time you spend practicing with it.

Begin with a clean slate. Load the VisiCalc program as described in "Loading the VisiCalc Program" in Chapter 1 or, if the program is running, clear the worksheet by typing /CY.

To prepare a budget you first project income for the next 12 months. You'll also project necessary expenses such as food, rent or mortgage, telephone, etc., as well as less frequent expenses such as car insurance. Then you'll use the VisiCalc program to find out how much income is left for leisure and savings and what percentage of the income is going to each category of expense. Finally, various enhancements such as calculating the interest on a savings account are presented.

SETTING UP THE BUDGET EXAMPLE

Begin by laying out 12 periods (months) across the worksheet. With the cursor at A1, type **Period** → to label row 1 and move to position B1. You can number the 12 periods either by typing the numbers 1 through 12 at B1 through M1, or typing 1 at B1 and replicating it with a formula that adds 1 to each previous number.

To speed setting up the worksheet, use the second method. As the earlier example showed, if a label at A1 is followed by 12 periods, the twelfth period is at M1. With the cursor at B1, type 1 (the first month), then → The formula, which adds one to each previous number, goes at C1; type 1+**B1** ↵. The entry line reads:

C1 (V) 1+B1

Coordinate C1, highlighted by the cursor, reads 2, the result of that formula.

Replicate this formula at D1 through M1. Type **/R** ↵. The prompt line reads:

Replicate: Target range

The edit line reads:

C1 . . . C1:

Type **D1.M1**; D1 is the beginning of the target range, the period tells the VisiCalc program that we're ready to type the end of the target range, and M1 is the end of the target range. (You could also type D1, a period, → 12 times, and ↵.) The edit line reads:

C1 . . . C1: D1 . . . M1

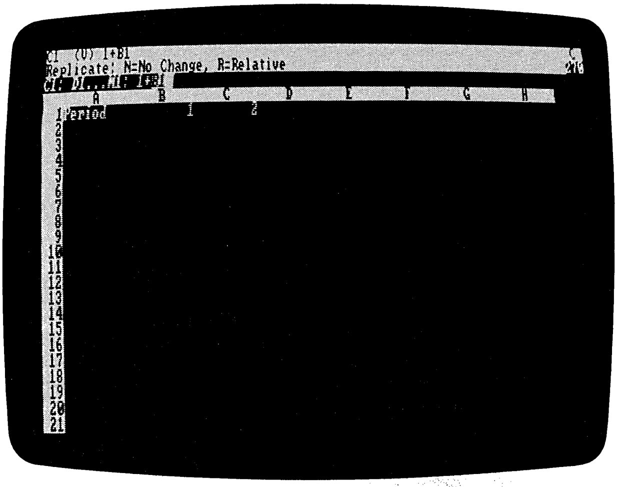
Now type ↵. The prompt line reads:

Replicate: N= No Change, R= Relative

The edit line reads:

C1: D1 . . . M1: 1+B1

The edit cue is on B1. The screen should look like the following photograph:



005-013/P

Type **R** to make the coordinate relative. This produces 1+C1, 1+D1, etc., in the succeeding locations of the target range. (If you typed N–making the coordinate absolute–the formula at each location would be 1+B2.) The prompt and edit lines clear. Move the cursor to column M to check your work. Position M1 should read 12.

REPLICATING NUMBERS AND LABELS

To start filling in the budget worksheet, type **HOME↓ Income** → 1800 ↵.

$1800 is the monthly take-home pay after taxes and other deductions. You could write 1800 in each month. Can you replicate a single number as well as a formula?

Of course. A number is actually the simplest case of a formula. With the cursor at B2 type **/R** ↵. For the target range, type **C2.M2** ↵. The VisiCalc program doesn't ask whether the new formula is relative or not, because 1800 has no coordinates. All 12 columns–positions B2 through M2–read 1800.

Next you'll draw a line across the worksheet. Type **HOME↓↓** to move the cursor, then /–. The prompt line reads:

```
Label: Repeating
```

and the edit cue is on the edit line. Whatever character or characters you type next are repeated to fill entry position A3.

Type – ↵. You should now have a line of nine hyphens at A3. Is this any different from typing the hyphens manually? Yes. Type **/GC12** ↵. As you can see, a repeating label expands to fill a column regardless of its width. Go back to normal column width by typing **/GC9** ↵.

How can you easily extend the line across all 12 columns? The ever-useful Replicate command also replicates labels. Type **/R** ↵. For the target range, type **B3.M3** ↵. It's that simple. You now have a line of hyphens extending from column A to column M. This line is unbroken regardless of the column width.

USING FORMULAS FOR FLEXIBILITY

Before going further, think about what you've done. To save the trouble of typing 1800 for each period, you replicated it. That was fast, but is it the best way to handle income? It would be better if the income figure for all 12 months could be changed just by typing a new figure for the first month and letting the VisiCalc program recalculate the rest. You can set up the worksheet to do this by replicating a formula instead of a number. Type the following:

>C2 ↵
+**B2** ↵

The second month's income is now defined as the income for the first month. Let's replicate this formula. Type **/R** ↵ and give the target range as **D2.M2** ↵. The prompt line reads:

Replicate: N= No Change, R= Relative

Should the same formula (+B2) be in all the remaining locations, or should the formula be relative (+B2, +C2, +D2, etc.)? Either way, the income for all 12 months could be changed simply by typing the new number at B2. But what if the salary increases in the sixth month? If every formula is +B2, the only way to change the value is to change the first month (B2). If the sixth month is changed, only that value changes; the remaining months still have the same value as B2.

On the other hand, if each formula refers to the previous month, when a new number is typed in month 6 the VisiCalc program propagates the change in months 7 through 12. Try it. Type **R** to make coordinate B2 relative. When the Replicate command has finished, use → to move to month 6 (position G2) and type **2000** ↵ to represent a raise at month 6.

Type → a few more times to verify that each succeeding month's income has changed to 2000. Now G2 reads 2000 instead of "+F2". H2 reads "+G2", so its value is the same as G2 (2000). Likewise, I2 reads "+H2", so the value at H2 (2000) is replicated into I2, and so on through M2. If you aren't sure of what you have just done, move the cursor over all 12 income figures and imagine what would happen if all the formulas were +B2.

To emphasize the difference between absolute (No change) and relative coordinates, type the following:

>C2 ↵
+B2 ↵
/R ↵
D2.M2 ↵

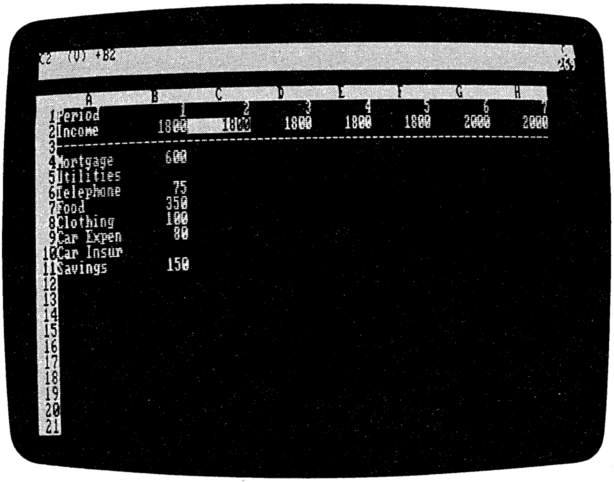
This time type **N** to make B2 absolute instead of relative. Examine C2 through M2 and compare them with the results you got using the relative formula. Repeat the process and make the coordinate relative. Be sure to write 2000 in month 6 (type >G2 ↵ 2000 ↵) to enter the raise.

The next task is to list expense categories and estimate monthly amounts for each category. Some expenses vary from month to month, and other expenses occur perhaps only every six months. Leave them blank for the moment.

Enter all the labels first and then the values for each label. This is a much quicker way to write lists of labels and values than writing each label and its respective value before going on the the next label and value. Type the following:

>A4 ↵
Mortgage ↓
Utilities ↓
Telephone ↓
Food ↓
Clothing ↓
Car Expense ↓
Car Insurance ↓
Savings ↓
>B4 ↵
600 ↓ ↓
75 ↓
350 ↓
100 ↓
80 ↓ ↓
150 ↵
>C2 ↵

The screen should look like the following photograph:



005-014/P

Next you'll replicate the monthly expense figures in column B across the remaining 11 months. Remember the discussion about the merits of replicating a formula, rather than a number, for monthly income? For flexibility, the formulas for monthly expenses should also be relative. C4 should contain the formula +B4; C6 should contain the formula +B6; C7 should contain +B7; and so on. You'll write figures for utilities and car insurance later.

These formulas are so similar to each other and to the income formula, +B2, that it's tempting to look for a shortcut way of writing them. Once again, the Replicate command comes to your aid. This time, you'll replicate a formula down a column instead of across a row.

REPLICATING DOWN A COLUMN

The entry line reads "C2 (V) +B2." Just as you copied B2 to C2, now you want to copy B4 to C4, B5 to C5, and so on. B2, therefore, should be relative. Type **/R** ↵. The prompt line reads:

Replicate: Target range

The edit line reads:

C2 . . . C2:

followed by the edit cue. Type ↓ twice to move the cursor down to the first coordinate in the target range.

Now the edit line reads:

C2 . . . C2: C4

Type a period. The cursor jumps back to C2, and the edit line acknowledges that the target range starts at C4. Next type ↓ nine times to point to C11 (next to the figure for savings). The edit line now reads:

C2 . . . C2: C4 . . . C11

Type ↵. The formula at C2 is to be replicated across the target range of C4 through C11. The cursor jumps back to C2, and the prompt line reads:

Replicate: N= No Change, R= Relative

The edit line reads:

C2: C4 . . . C11: +B2

The edit cue is over B2. This coordinate should be relative, so type **R**. A column of numbers, from 600 to 150, appears in column C.

Move the cursor down, looking at the formulas you replicated. You have what you need: C4 reads "+B4", C6 reads "+B6", and so on. You also have formulas at C5 and C10, but they can be cleared with the Blank command. Type the following:

>C5 ↵
/B ↵
>C10 ↵
/B ↵
>C4 ↵

REPLICATING A COLUMN SEVERAL TIMES

The worksheet now contains a formula for each expense category. The next step is to replicate these formulas across the rows through month 12. Think back to Lesson Two. Remember how you replicated a source range of formulas across the rows for both cost of goods sold and gross profit? You can do the same thing here.

Type **/R**. The prompt line reads:

Replicate: Source range or ENTER

The edit line reads:

C4

followed by the edit cue. This time the source range is an actual range, C4 through C11. Point to each coordinate in that range by typing ↓ seven times to move the cursor down to C11. When it reaches C11, the edit line reads:

C4 . . . C11

Now type ↵. The cursor jumps back to C4 and the prompt line asks for a target range. Type **D4.M4** ↵ to tell the VisiCalc program to copy the formula at C4 into positions D4 through M4, the formula at C5 into positions D5 through M5, the formula at C6 into positions D6 through M6, and so on until the formula at C11 is copied into D11 through M11.

The prompt line reads:

Replicate: N= No Change, R= Relative

The edit line reads:

C4: D4 . . . M4: +B4

The edit cue highlights B4. This is the formula for the first expense, mortgage. Like B2 (income), B4 should be relative, so type **R**. The VisiCalc program replicates the formulas +B4, +C4, +D4, etc., in row 4. It also replicates the blank entry at B5 into C5, D5, E5, etc.

When it replicates a formula, the VisiCalc program asks you to specify N (No change) or R (Relative) for each reference to another location. If you replicate a range of formulas, it asks for each reference in each formula. It starts here by asking you to specify N or R for the formula +B6 on row 6.

Each of these formulas should be relative, too. Type **R** five more times. With relatively few keystrokes and the assistance of the Replicate command, you have written 80 numbers and formulas on the worksheet.

Think about how you replicated the expense formulas. Starting with the original formula +B2 at position C2, you created six copies of it by replicating down a column: +B4 at C4, +B6 at C6, etc. Then you used these formulas as the source range to replicate similar columns of formulas across rows 4 through 11.

Each of the resulting monthly expenses can be changed for all 12 months simply by typing a new number for the first month. For example, type >B8 ↵ 120 ↵. The clothing budget increases to 120 for all 12 months.

To complete the projection of expenses, fill in figures for those expenses that cannot be replicated because they vary from month to month. The utilities bill varies with the season. Car insurance premiums are due every six months, in month 1 and month 7. Type the following:

>B5 ↵
140 → 140 → 80 → 80 → 40 → 40 → 85 →
85 → 50 → 50 → 100 → 140 →
>B10 ↵
160 →
>H10 ↵
160 →

You don't have to type zeros for the other 10 months for car insurance, because the VisiCalc program treats any blank entry as zero. In fact, any entry that doesn't contain a number or formula has a value of zero if it is referenced in a formula.

Now is a good time to save your work on diskette. Remove the VisiCalc program diskette and replace it in the pocket on the inside front cover of this manual. Put in an initialized storage diskette. Be sure the label side is up and that the oval cutout enters the drive first. Close the drive door and type /**SS** (the Save option of the Storage command) then **budg1** ↵ (the file name for the worksheet). When the file is saved, the prompt and edit lines clear.

You should also make a backup copy of this file. Remove the diskette on which you just saved "budg1.vc" and insert another initialized diskette. (You might label it "BACKUP1" and use it only for saving backup files.) Close the drive door and type /**ssbudg1** ↵.

FIXING TITLES IN BOTH DIRECTIONS

Again, the titles ("Income", "Mortgage", etc.) disappear when you scroll horizontally to look at later months. To solve this problem, you can create a border of titles along both the left and top edges of the worksheet. These titles stay in place no matter where the window is scrolled.

Type **HOME↓↓** to move the cursor, then **/T** (the Title command). The prompt line reads:

Titles: H V B N

(The H, V, B, and N options are described under "Fixing Titles in Place" in Lesson Two and "Titles Command" in Chapter 3.)

Type **B** to fix titles in *both* directions.

The position of the cursor has a dual significance to the Titles command. If you fix titles vertically, the VisiCalc program fixes the column in which the cursor rests and all columns to the left of the cursor. If you fix titles horizontally, the VisiCalc program fixes the row in which the cursor rests and all rows above the cursor.

If you fix titles in both directions, the VisiCalc program fixes the column in which the cursor rests and all columns to the left of the cursor, plus the row in which the cursor rests and all rows above the cursor.

The screen now has a border consisting of column A along the left edge, and another border consisting of rows 1, 2, and 3 along the top edge. To check this, move the cursor down until the window scrolls. Rows 1, 2, and 3 remain in place. Now type **>B4** ↵ and scroll horizontally. Column A remains in place.

Only the period and income figures in rows 1 and 2 seem to change, because the other numbers are the same from column to column. Continue scrolling until column O is at the right side of the window. Columns N and O will be used to calculate totals and percentages for income and expenses.

THE @SUM FUNCTION

Type the following:

>N1 ↵
Total ↓

The cursor is at N2. How can you calculate the total income for 12 months? You could type +B2+C2+D2+E2+F2+G2+H2+I2+J2 +K2+L2+M2, but there's a simpler way. Type **@ sum(** . The prompt line reads:

Value

The edit line reads:

@SUM(

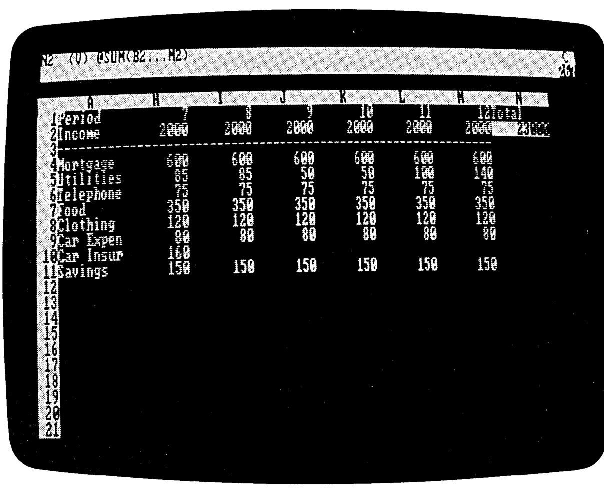
To specify the numbers to sum, scroll to the left border. The cursor is at B2 and the edit line reads "@SUM(B2". Now type . (a period). The cursor moves back to N2 and the edit line reads:

@SUM(B2 . . .

You are specifying a range of entries, just as you did for the Replicate command. To finish the range, type ←) ↵. The entry line now reads:

N2 (V) @SUM(B2 . . . M2)

N2 reads 23000, the total income for the year.



005-015/P

The @ begins the name of a VisiCalc *function*. When you type @ to start an entry, the VisiCalc program immediately knows two things:

- The entry is going to be a Value.
- The next few letters must be the name of a function.

Each function, such as @SUM, performs a calculation on the value or list of values specified with it and produces a numeric result. Functions perform arithmetic calculations more complex than simple addition or multiplication, such as average of a range, minimum or maximum of a range, and trigonometric functions. See "Functions" in Chapter 3 and the *VisiCalc Pocket Reference* for a complete description of all the functions.

The VisiCalc functions accept one of the following types of arguments:

- A coordinate or range of coordinates, such as @SUM(B2 . . . M2).
- A list of specific coordinates, such as @SUM(B2,C3,D8).
- A list of ranges, coordinates, numbers, or formulas, such as @SUM(B2 . . . B7,C3 . . . C6,25,D8,4*C8).

You can use a function anywhere you can use a number. You can even use it as an argument for another function:
@SUM(B2,G5,@SUM(A1 . . . A9),55).

Now to calculate totals for each expense category. With the cursor at N2 type **/R** ↵. To specify the target range, type **N4.N11** ↵. The prompt line reads:

Replicate: N= No Change, R= Relative

The edit line reads:

N2: N4 . . . N11: @SUM(B2

The edit cue highlights B2.

Consider what happens if you make B2 . . . M2 relative. Because you are replicating down a column, the copies of the formula will be in positions N4, N5, N6, etc. The value at N4 should be the sum of B4 . . . M4, the value at N5 should be the sum of B5 . . . M5, and so on. The reference to B2, therefore, should be relative, so type **R** twice (for B2 and M2). The expense totals, from 7200 to 1800, appear in column N.

To calculate the percentage of income represented by each expense total, type the following:

>O1 ↵
Percent ↓↓↓

The formula for calculating the percentage of income represented by the mortgage payment divides total mortgage payments by total income. The VisiCalc program already has those totals–mortgage at N4 and income at N2. To find the percentage, all you need do is divide N4 by N2. Type **+N4/N2** ↵. O4 reads .3130435, or slightly more than 31%.

FORMATTING A SINGLE ENTRY

The VisiCalc program's General format displays numbers to the maximum number of significant digits permitted by the current column width, unless you specify a different display format. For percentages, two decimal places are enough. The Dollars-and-cents format displays values to two decimal places.

Type /**GF$** (specifying the Dollars-and-cents format as global). The mortgage percentage at position O4 now reads 0.31. All other numbers are also displayed to two decimal places. Even the month number at M12 reads 12.00; not quite what you want. Change the global format back to general by typing /**GFG**.

Just the number at O4 should be displayed with two decimal places. With the cursor at O4, type /**F$**. By using the Format command to assign a local format (instead of the Format option of the Global command, which assigns a format to the entire worksheet), just the entry at the cursor location is formatted.

Now that the percentage reads 0.31, look at the label "Total" at the top of column N. It's hard to read because it's too close to the 12 in column M, and it doesn't line up with the numbers below it.

The VisiCalc General format–which currently applies to all entries not individually formatted–starts a label at the left side of its entry position, and lines up the last digit of a number with the right side of its entry position. Type >**N1** ↵ to move the cursor to the label "Total", then /**F** (the Format command). The prompt line reads:

```
Format: D G I L R $ *
```

The characters following "FORMAT:" are Format command options:

- D Default. Reverts to the global format. If a global format is not specified, reverts to the General format.
- G General. Displays a number with several decimal places, even if the global format is integer.
- I Integer. Displays a number as an integer, rounding if necessary.

L Left-aligned. Lines up the first letter or digit of the entry at the left edge of the entry position.

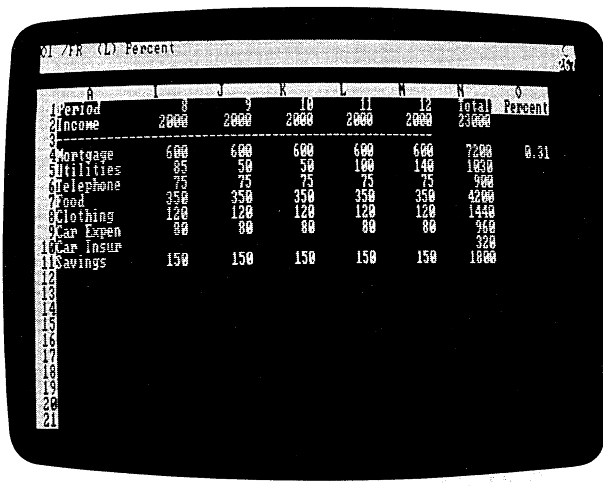
R Right-aligned. Lines up the last letter or digit of the entry at the right end of the position.

$ Dollars-and-cents. Displays two decimal places.

* Graph. Displays asterisks to build bar graphs.

For more details plus examples of the Format command, see "More on Numbers and Formats" and "Transcendental Functions and Graphing" in Lesson Five, "Format Command" in Chapter 3, and the *VisiCalc Pocket Reference*.

To improve this format, type **R**. Now "Total" is lined up with the column of numbers below it. Type → to move the cursor to O1, then **/FR** to line up "Percent" with the right edge of its entry location. The screen should look like the following photograph:



005-016/P

REPLICATING A FORMAT SPECIFICATION

Now that you have a formula for describing an expense as a percentage of income (+N4/N2), you can replicate it down column O for the other expenses. Type **>O4** ↵ to move to the formula for mortgage expense as a percentage of income. Now type **/R** ↵ (the Replicate command), then **O5.O11** ↵ as the target range. The edit line reads:

O4: O5 . . . O11: +N4

The edit cue highlights +N4. The prompt line reads:

Replicate: N= No Change, R= Relative

How should the reference in this formula (+N4/N2) be replicated? At O5, utilities should be divided by income +N5/N2. The first coordinate should change, but the second (N2, or income) should not. The VisiCalc program asks you to specify whether each reference in a replicated formula should be relative or absolute.

Type **R** to make N4 relative. The edit cue moves to N2 on the edit line. Type **N** for No change. It takes the VisiCalc program only a moment to display all the expense percentages, from 0.31 for mortgage to 0.08 for savings. All percentages are displayed to two decimal places.

Type ↓ a few times, watching the entry line. At O5, it reads:

O5 /F$(V) +N5/N2

The format specification (**/F$**) has been replicated along with the formula. In fact, it's possible to replicate a format specification even if the entry being replicated is blank. This technique will be used later.

USING REPLICATE TO COPY A ROW OR COLUMN

To separate different areas of the worksheet, you can draw a line under the list of expenses. You already know one relatively easy way to draw a line of hyphens, by typing /– (the Repeating Label command) at A12 and replicating the hyphens across. Now to use the Replicate command a little differently.

Type >**A12** ↵ /**R**. The prompt line reads:

Replicate: Source range or ENTER

The edit line reads:

A12

followed by the edit cue. Type [BKSP]. "A12" disappears from the edit line, leaving only the edit cue. Now type a new source range: **A3.M3** ↵. As usual, the prompt line reads:

Replicate: Target range

Type **A12.A12**.

The source range (A3 . . . M3) is the line of hyphens on the worksheet. You're telling the VisiCalc program to replicate A3 into A12, B3 into B12, and so on. Type ↵. There's the line. It would be sufficient to type A12 ↵ for the target range; the VisiCalc program takes this to mean A12 . . . A12.

Finally, add an entry for leisure. The money available for leisure is simply income minus the sum of expenses. Type the following:

>A13 ↵
Leisure → +B2–@SUM(B4.B11) ↵

B13 displays the result of the formula for leisure, 125. Replicate this formula across row 13 to get the leisure money for each month. Type /**R** ↵ **C13.M13** ↵, then type **R** three times to make the references relative. Scroll the window to the right and look at the results.

The leisure money starts at 125 the first month and increases fairly steadily thereafter. There is no car insurance premium after the first month, and the utilities bill goes down in months 3, 4, and 5. In month 6 there is a salary increase, which increases the figures for both income and leisure.

Continue scrolling until columns M, N, and O are on the screen, showing "Total" and "Percent", then place the cursor at N13.

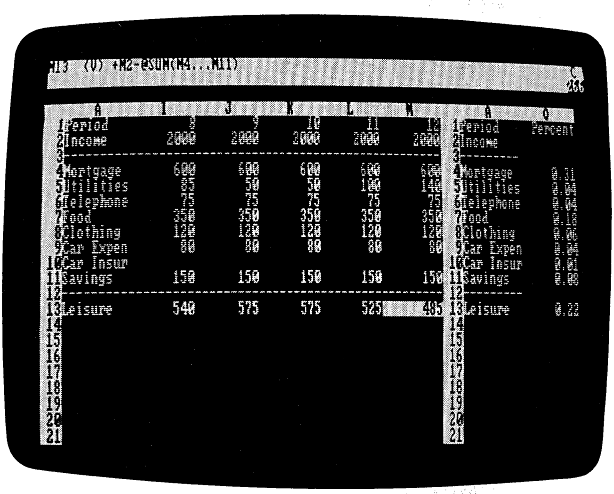
To obtain a total and percentage of income for leisure, use the same method you used earlier to copy the line of hyphens. To allow you to type in formulas in the same form they take on the edit line, the VisiCalc program lets you substitute a colon (:) for ↵. Watch the edit and prompt lines while you type the following:

/R [BKSP] N11.O11:N13 ↵RRRN

The screen shows 5150 for the total and 0.22 for the percentage.

CHANGING WINDOWS AND TITLES

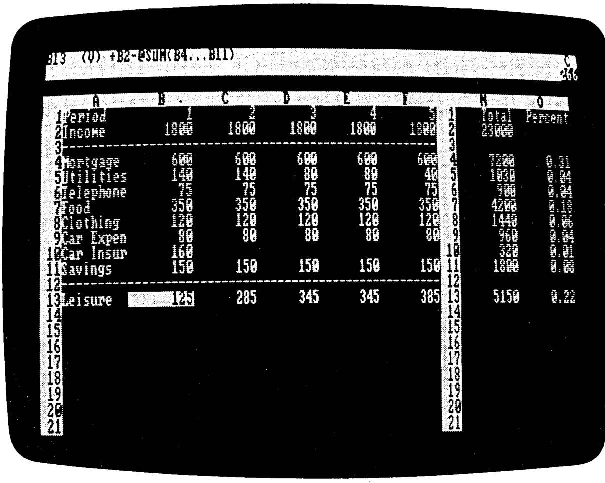
As you found in Lesson Two, the screen cannot display both the starting expense figures in month 1 and the calculated totals and percentages in columns N and O. You can see them both by splitting the screen. With the cursor still at N13, type **/WV**. The titles in column A and rows 1-3 are fixed in place in both windows. The screen should look like the following photograph:



005-017/P

Type ; to move the cursor to the right window, bringing columns A and N into view. Now type **/TN** to eliminate the fixed titles from the window. The title (column A) disappears, exposing column M. Finally, type → to display columns N and O ("Total" and "Percent").

Type ; to move the cursor back to the left window. Scroll this window back to the first month (column B); you bump into column A because the fixed titles are still in effect in this window. The screen should look like the following photograph:



005-018/P

To see how the VisiCalc program recalculates the totals and percentages, change an expense: type >**B9** ↵ **100** ↵. An exclamation point appears to the right of the C in the upper-right corner, telling you that the VisiCalc program is recalculating all values. The available leisure money decreases by 20 each month. The car expense total increases from 960 (4%of income) to 1200 (5%of income), and the leisure total decreases from 5150 (22% of income) to 4910 (21%of income).

Now is a good time to save the worksheet again. Put the diskette you're using for worksheet storage in the disk drive. Type /**SS** and, in response to the prompt "Storage: File for Saving", type →.

The disk drive whirs and a file name appears on the edit line. Continue typing →, if necessary, until "A:BUDG1.VC" is on the edit line. Then type [BKSP] four times until you have erased "1.VC". Now type 2 ↵;the file name reads **BUDG2**, and the VisiCalc program adds .VC to identify the file as a worksheet. You are saving the worksheet with a revised name so you can distinguish it from the first version you saved. As you see, it takes only a few keystrokes to protect yourself from loss of time and data.

SUMMARY

The flexibility and usefulness of the VisiCalc program should now be apparent. This lesson showed how quickly you can set up a worksheet that includes a full year of expenses and formulas to keep all calculations up to date. By eliminating the need to erase or throw away a paper worksheet after some changes, the VisiCalc program not only makes planning and calculating more accurate and efficient, it encourages you to set up planning or budgeting procedures you might otherwise not attempt.

LESSON FOUR

This Lesson describes several ways to change and calculate with the worksheet after you have written on it. Continuing the personal budget example started in Lesson Three, it covers inserting, deleting, and moving rows and columns, controlling recalculations of the worksheet, and printing the worksheet.

DISPLAYING UNKNOWN OR INVALID ENTRIES

Sometimes you must write entries on the worksheet before you know their exact value. Suppose the car insurance premium isn't known. The VisiCalc program has a function to help you deal with just such a problem: @NA (for Not Available). When you write this function at an entry position, that entry takes on the special value NA. Any formula that refers to an entry containing **NA** has a value of NA itself.

Replace 160 at B10 by typing >**B10** ↵ **@NA** ↵. The car insurance premium for month 1 at B10 now reads "NA". Because of this, a number for the leisure balance (income minus the sum of expenses) for month 1 cannot be calculated. Position B13 (for leisure) becomes "NA." Moreover, the total car insurance expense for the year and the corresponding percentage of income also become "NA."

Change B10 back by typing **160** ↵; all the values are restored.

Here's a related issue. Suppose you make a mistake in typing the formula for percentage and try to divide by zero. Type >**O11** ↵ +**N11/0** ↵. Position O11 changes to "ERROR", a message that tells you that the value cannot be calculated. The most common cause is division by zero; other causes include taking the logarithm of a negative number and making an error while typing a formula.

If you save a worksheet with forward references, the locations that contain them display "ERROR" when you reload the worksheet. To restore values to these "ERROR" positions, type ! to force a recalculation.

Like NA, the value ERROR propagates. Any formula that refers to an entry with the value ERROR itself has the value ERROR. You can also deliberately obtain the value ERROR by typing @ERROR. For now, change O11 back by typing **+N11/N2** ↵ so the totals and percentages can be calculated.

Type ; to move the cursor to the right window, then **/W1>A7** ↵ to return to a single window and scroll back to the left edge of the worksheet. The cursor should be on the label "Food".

INSERTING AND DELETING COLUMNS AND ROWS

Suppose you want to add a life insurance policy with monthly premiums of $115. If you were working out this budget on a sheet of paper, you'd have to erase something or write in tiny letters off to the side. But the VisiCalc worksheet is more flexible; it allows you to insert rows and columns.

Type **/I** (the Insert command). The prompt line reads "Insert: R C"; type **R** for Row. The VisiCalc program opens a blank line at row 7 by pushing down the rows that were at or below the cursor. You can enter the life insurance figures in this new row.

Look more closely at what the Insert command has done. Has moving savings from B11 to B12 invalidated the formula for leisure, +B2-@SUM(B4 ... B11)? Type **>B14** ↵ to check it. The entry line reads:

```
B14 (V) +B2-@SUM(B4 ... B12)
```

Whenever you insert, delete, or move a row or column, the VisiCalc program adjusts all formulas on the worksheet.

Now write the life insurance entries on the worksheet. You'll write the basic amount, make a formula based on it, replicate that formula across row 7, then create and replicate formulas for total expense and percent of income figures. Type the following:

>A7 ↵
Life Ins → 115 → + ← ↵
/R ↵
D7.M7:R
>**N6** ↵
/R →:N7:RRRN
>A10 ↵

The leisure money has decreased by the amount of the life insurance premiums each month. Position B14 reads – 10, meaning that spending exceeds income in month 1. To reduce expenditures, eliminate most of the car expense (public transportation and a bicycle can take its place). Type >**B10** ↵ to move to the car expense row, then **/D** (the Delete command). The prompt line reads:

```
Delete: R C
```

With the cursor at B10, you can delete row 10 by typing R or column B by typing C. Type **R** to delete row 10.

Note: Once you have deleted a row or column with the Delete command, the locations and data are gone forever; you cannot recover them. Be sure you really want to delete a row or column before you reply R or C to the command prompt.

The label and values for car expense disappear. The rows below the cursor move up one to fill the gap and are renumbered accordingly. Car insurance is now in row 10 and savings is back in row 11. The leisure money at position B13 has increased to 90. Check the formula at B13; the VisiCalc program has readjusted the range:

```
+B2-@SUM(B4 . . . B11)
```

You can also insert columns. Suppose you wanted to show six-month totals for income and expenses. Type >**H4** ↵. The formula there is "+G4;" because of relative replication the formula at I4 is "+H4". Now type /**IC**. A new, blank column appears; column H and all columns to the right move to the right.

The Insert command inserts a row or column *before* the cursor (closer to row 1 or column A) and moves everything else down or to the right.

Type → to bring column I (formerly column H) into view. The formula at I4 is "+G4", showing that the formulas for income and expense skip over the new blank column. Move back with >**H4** ↵, and type /**D** (the Delete command). The prompt line reads:

Delete: R C

Type **C** to delete column H (if you typed R, row 4 would be deleted).

When the blank column is deleted, all columns to the right move left to fill the empty space. If you've made any mistakes in this section, clear the screen and reload the file named BUDG2.VC to restore the worksheet.

To test your understanding of the Insert and Replicate commands, insert abbreviated names of the months (Jan, Feb, Mar, etc.) just below the month numbers 1 through 12.

Here's one way to do it. Type the following:

HOME↓
/IRMonth → /FR/R:C2.M2 ↵
Jan → Feb → Mar → Apr → **May** → Jun → Jul →
Aug → Sep → Oct → Nov → **Dec**
HOME ←

CALCULATING INTEREST ON A SAVINGS ACCOUNT

According to the present budget, $150 is set aside each month for savings. The interest on this money contributes to gross income. The VisiCalc program can project the interest and the accumulated balance.

Assume that interest on a savings account is paid at the rate of 5% per year, compounded monthly. For flexibility, this interest rate will be written in a separate position on the worksheet; that way, the VisiCalc program recalculates the interest and accumulated balance whenever the interest rate is changed. Type the following:

>A15 ↵
Sav Acct →
.05 ↵

On the first day of each month the account is credited with interest for the previous month's balance plus the monthly deposit of $150. Type the following:

>**A17** ↵
/F$100 ↑
Interest →

The $100 at A17 is the previous balance in the savings account before the budget begins. The interest paid for one month is 1/12 of the yearly rate (.05 divided by 12) times this previous balance. Type:

/F$+B15/12*A17 ↵

The result (at B16) is 0.42. Does this make sense? A year's simple interest at 5% on $100 is $5. One-twelfth of this is 5/12, or .41666 (which rounds to $0.42).

Type ↓ to move to B17. The new savings account balance is the previous month's balance, plus the interest, plus the savings deposit for this month. Type:

/F$+A17+ ↑ **+B12** ↵

If you like, you can point with the cursor to specify all three coordinates in this formula. The result (under the cursor) is 250.42.

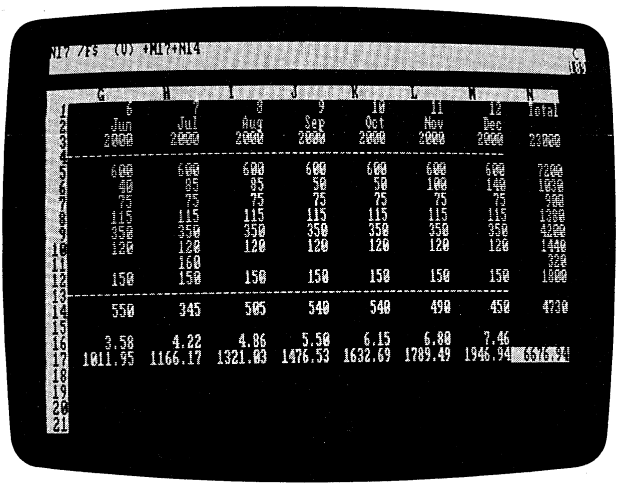
To replicate both the interest and account balance formulas across 12 months, type the following:

>B16 ↵
/R ↓ :C16.M16:NRRRR

Type → to check the results. The interest paid each month increases as monthly deposits and accumulated interest are added to the balance on which the interest is calculated. Continue scrolling until column N comes into view, and then type:

>N17 ↵
/F$+M17+N14 ↵

This is total discretionary income–the sum of savings and leisure money. It should be 6676.94. The screen should look like the following photograph:



005-019/P

MOVING ROWS AND COLUMNS

Type >**A11** ↵ to move the cursor to car insurance. Paying that insurance premium in month 1 is taking a big bite out of the leisure money in month 1 ($90, compared to $250 in month 2). Why not pay the insurance premiums from the savings account?

On a sheet of paper, more erasures and writing in the margins would be necessary. Not so on the VisiCalc worksheet. Simply move the car insurance expense (row 11) out of the range of expenses used to calculate leisure money, putting it down with the savings account items. Type /**M** (the Move command). The prompt line reads:

Move: From . . . To

The edit line reads "A11". Type ↓ The cursor moves down to highlight "Savings", and the edit line reads "A11 . . . A12". Type ↓ four more times; the edit line changes from "A11 . . . A12" to "A11 . . . A16", just as it does when you specify a range for the @SUM function or the Replicate command. The cursor highlights "Interest" at A16. Type ↵. It takes the VisiCalc program a few moments to complete its work:

- The car insurance row moves down from row 11 to row 15.
- The rows for savings, leisure, and savings account move up, filling the gap and creating a space for car insurance in its new position just above interest.
- The cursor returns to where it was when you started the Move command at A11, which is now "Savings".

The figure for leisure in month 1 has increased from 90 to 250. The car insurance premium has been taken out of the sum of expenses used to calculate leisure. The formula at B13 reads:

+B3-@SUM(B5 . . . B11)

Car insurance is now unaccounted for. The formulas in row 17 must be revised to take the car insurance premium out of the savings account balance. Type >**B17** ↵; the entry line reads "+A17+B16+B11". The formula defines each month's savings account balance as the previous account balance, plus a month's interest on that balance, plus the current month's deposit. Now you must add "minus the car insurance premium, if any." Type the following:

+A17+B16+B11- ↑ ↑ ↵
/R:C17.M17:RRRR

This replicates the new account balance formulas +A17+B16+B11-B15, +B17+C16+C11- C15, etc., across the row.

Now type >**N17** ↵ to check total discretionary income, the sum of savings at M17 and leisure money at N13. It has declined from 6676.94, before we took the insurance premiums out of savings, to 6666.10. Savings has declined (1616.10 versus 1946.94) and leisure spending has increased (5050 versus 4730). About $10 interest has been lost from the savings account. Perhaps the monthly deposit should be increased a bit to replenish the funds taken out to pay the insurance premiums.

Type >**A11** ↵ → to bring the titles back on the screen and highlight the initial savings figure at B11. Because the premium is due every six months, increase the monthly deposit by one-sixth of it. Type the following:

150+(B15/6) ↵

The parentheses tell the VisiCalc program to calculate that portion of the formula first. Thanks to the earlier use of formulas, the VisiCalc program propagates the adjusted savings figure across all 12 months, recalculating leisure and account balance figures for each month.

This recalculation makes the screen a bit messy, because the global format for numbers is still General. Clean up the display by typing **/GFI** (setting the global format to Integer). The interest and savings account figures are still displayed with two decimal places because we specified a local Dollars-and-cents format (/F$) for each of these entries. Now type >**N17** ↵ to check total discretionary income again; it has increased to 6673.53.

Because savings has been increased by about $27 each month, the leisure total has declined to 4730. At the same time, most of the lost interest has been regained.

Before continuing, save your work by typing **/SS** and using → as necessary to bring the file name "A:BUDG2.VC" onto the edit line. Type [BKSP] four times, then 3 ↵.

REVIEWING THE REPLICATE COMMAND

This lesson has led you through some fairly sophisticated command combinations. Here's a little project to try on your own–a challenge to test your mastery of the Replicate command. Create monthly percentages for each of your expenses, from mortgage through savings. You can do this by typing just one formula and using the Replicate command three times.

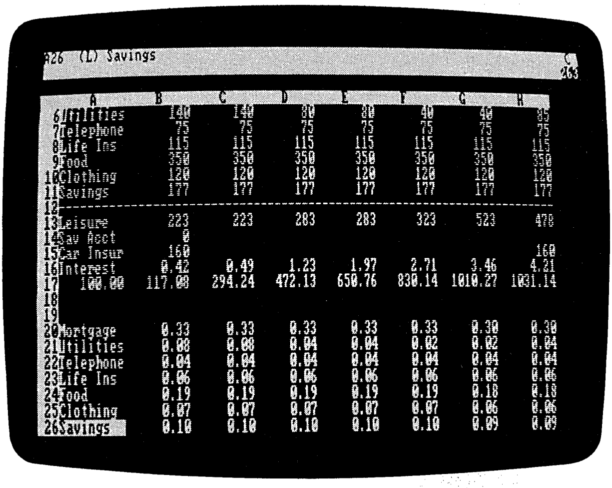
Here are some hints:

- Use the area of the worksheet directly below your list of monthly expenses.
- Remember, you can replicate format specifications.
- You can label each row of percentages with one more use of the Replicate command.

To make sure your worksheet matches the one in this lesson, clear it (/**CY**) and reload the file you just saved with the Load option of the Storage command (/**SL**). Type → until the file name "A:BUDG3.VC" appears on the edit line, then type ↵. Now type the following:

>A20 ↵
/**R [BKSP] A5.A11:A20:** → /F$+B5/B3 ↵
/R:B21.B26:**RN**
/**RB26:** → .M20:RRRRRRRRRRRRRR
(Type **R** 14 times.)
>A26 ↵

The screen should look like the following photograph:



005-020/P

The first replication in this sequence illustrates another use of the Replicate command. You can copy a range of entries into another part of the same column (or to any other area of the worksheet) simply by specifying the starting coordinate of the destination as the target range. Check row 21 (Utilities) to verify that the percentages change from month to month.

SCROLLING SPLIT WINDOWS TOGETHER

The worksheet now extends beyond the boundaries of the screen in both the horizontal and vertical directions. As you scroll down, the month labels, income, and first few expenses disappear. You can split the screen to see both the expense amounts and percentages at the same time.

Move the cursor up to A18 (just above "Mortgage"). Now type the Horizontal option of the Window command (/**WH**). The screen splits horizontally, leaving just enough room for the expense percentages in the bottom window.

Type >**A2** ↵ ↓ ↓. This leaves rows 2 ("Month") through 13 ("Leisure") in the top window; the cursor is at A4. Next type /**TB** to fix both horizontal and vertical borders. (The column and rows forming the borders do not have to start from the edges of the worksheet.) Finally, type ;>**A26** ↵ to bring all the expense per centages on the screen and /**TV** to fix the labels "Mortgage" through "Savings" in place.

Hold down → until the bottom window begins to scroll to the right. You can't easily tell which months these expense percentages represent. The month labels are visible only in the top window, which isn't scrolling.

If the two windows scrolled together horizontally but remained independent vertically, you could view different areas of the worksheet as you are now. Type **/W** again; the prompt line reads:

Window: H V 1 S U

The characters following "Window": are options:

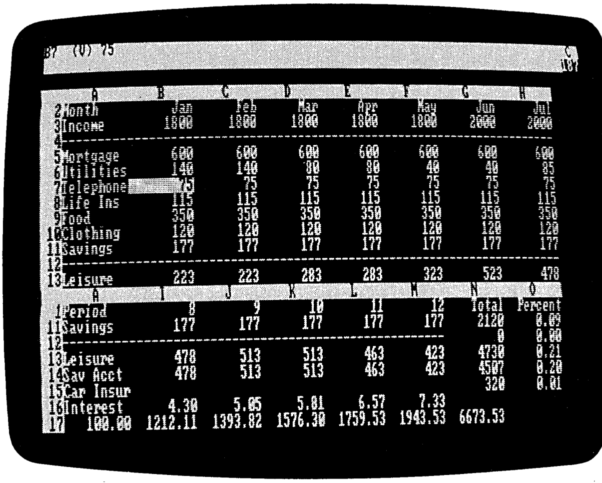
H Splits the screen Horizontally.
V Splits the screen Vertically.
1 Returns to one screen window.
S Synchronizes scrolling.
U Unsynchronizes scrolling.

Type **S**. The top window scrolls over so that portions of the same columns are visible through the top and bottom windows. Now type → a few times; the two windows scroll together. Type ; to move the cursor into the top window, then **>B7** ↵. Change the telephone expense to 100 and watch the line of percentages opposite "Telephone" in the bottom window. Notice that the leisure figures in the top window also change.

Now to use the two windows for a different purpose. Type **/WU** to unsynchronize the windows. Move to the lower window with ; and type **HOME /TB >O17** ↵ to display the leisure total and percentage, final savings account balance, and combined discretionary income.

Type ; to move back to B7 in the top window. Change the telephone expense back to 75 and watch how this affects the budget in other ways.

The screen should look like the following photograph:



005-021/P

If you would like to use this version of the budget worksheet for your own use, save it by typing **/SSbudget** ↵.

ORDER OF RECALCULATION

So far, you've seen that the VisiCalc program recalculates the values of all the formulas on the worksheet, but you haven't been told much about how this is done. Some characteristics of recalculation can affect results on a complex worksheet.

The VisiCalc program recalculates by starting at the upper left corner of the worksheet, working its way down and to the right until it reaches the lower right corner. Each formula is evaluated only once unless you ask for an extra recalculation by typing !.

As a rule, this means that formulas that reference other entries must be located below and to the right of the referenced entries. An entry at position A1 cannot be a formula that references other positions.

The VisiCalc program evaluates the formulas on the worksheet in one of two orders: down the columns or across the rows. Look again at the letter C in the upper right corner of the screen. This tells whether the VisiCalc program recalculates by columns (C) or rows (R).

When you load the VisiCalc program or clear the worksheet, it is set to recalculate by columns. It first evaluates A1, then A2, then A3, etc., followed by B1, B2, B3, etc., then C1, and so on to the lower right of the worksheet.

For many problems, column-first and row-first recalculation produce the same results. But there are times when you must use the right recalculation order to obtain correct results, and it's important to recognize these cases when they arise. Here's an example. Clear the worksheet with **/CY** and type the following:

1 → - A2 **HOME**↓
1+**A1** → 2*B1 ↵
>C5 ↵
+**A1** → 1+C5 ↵
>C6 ↵
- **D5** → 2*C6 ↵
HOME

As you type the formulas, think about how each entry depends on the other entries. The matrix of entries starting at A1 must be recalculated in the order A1, A2, B1, B2 (because B1 depends on A2). The matrix of entries at C5, however, must be recalculated in the order C5, D5, C6, D6 (because C6 depends on D5).

With the cursor at A1, type 2 ↵. A2 becomes 3, B1 becomes – 3, and B2 becomes – 6, as expected. But while C5 becomes 2 and D5 becomes 3, C6 remains – 2 and D6 remains – 4.

The formula at D5 was recalculated, but too late to affect the recalculation of C6 and D6. Now type ! to trigger an extra recalculation. Now C6 reads – 3 and D6 reads – 6.

Now we'll change the order of recalculation from columns to rows. Type /**G**. The prompt line reads "Global: C O R F". Type **O**. Now the prompt line reads "Reeval Order: R C". Type **R**. The recalculation order indicator at the upper-right corner of the screen changes from "C" to "R". Now type 1 ↵.

This time D5 becomes 2, C6 becomes – 2, and D6 becomes – 4. But while A2 becomes 2, B1 reads – 3 and B2 reads – 6. The problem with B1 and B2 is similar to the earlier problem with C6 and D6. Again, you can correct the values by typing !.

The moral of this example is that you should lay out your calculations for recalculation either by column or by row, but not both. If possible, you should arrange the worksheet so that the results are correct regardless of the order in which the VisiCalc program recalculates formulas. Then, if you decide to add a formula or otherwise change the worksheet so that a particular order of recalculation is required, you won't have to redesign other parts of the worksheet or type ! one or more times to get the correct results.

The personal budget example we have used is independent of the recalculation order. To change the life insurance policy (and premiums) to provide a benefit of three times annual income, you could switch to row-order recalculation so that the life insurance premium in month 1 is based on the total income calculated in column N.

If you find yourself with conflicting requirements for recalculation order, the problem may be caused by a *forward reference* or a *circular reference* (to be described in a moment). If a reference problem isn't involved, you must force an extra recalculation by typing ! each time you change a value, or change the worksheet to eliminate the conflict.

FORWARD AND CIRCULAR REFERENCES

Clear the worksheet (/**CY**) and type 1 → – → ↵. The entry line reads:

B1 (V) -C1

and the entry at B1 is 0, as you might expect. Now type → 1+ ← ← ↵. The entry line reads:

C1 (V) 1 + A1

The value at C1 is 2, and the value at B1 has changed to – 2.

Type **HOME**↓ → **- C1** ↵. The same formula, – C1, is at both B1 and B2, and both positions display – 2.

Is there any difference between these two formulas? Indeed there is. One of them recalculates and displays the correct value only if the order of recalculation is by row. The other never displays the correct value after an automatic recalculation.

To see this, type **HOME** 2 ↵. A1 becomes 2 and C1 becomes 3, but both B1 and B2 remain – 2. Type !; both B1 and B2 are updated to – 3.

Type /**GOR**; the recalculation order indicator changes from "C" to "R". Now type 3 ↵. A1 becomes 3, C1 becomes 4, and now B2 becomes – 4 but B1 remains – 3. You'll have to type ! to update B1 to – 4.

When you change A1, B1 displays a value based on the previous contents of C1 and A1. The formula at B1 is an example of a *forward reference*. It contains a reference to an entry that is recalculated *after* B1 is recalculated, regardless of whether you specify row-order or column-order recalculation.

In extremely difficult cases, forward references may refer to other forward references, so that correct results can be obtained only with several recalculations. If the cursor is not on A1, move there. Now type → /**IC**– → ↵.

At the moment, A1 is 3, the new B1 is 4, C1 is – 4, and D1 is 4. Type **HOME** 1 ↵. A1 becomes 1, D1 becomes 2, but B1 and C1 are unchanged. Type !. Now C1 is – 2, but B1 is still 4. Only after you type ! again is B1 updated to 2.

An even more startling effect is caused by a *circular reference*. The value of such a formula cannot be settled with any number of recalculations! Clear the worksheet and type 1+ → ↵. The entry line reads:

A1 (V) 1+B1

and the value under the cursor is 1, as expected.

Now type → 1+ ← ↵ and watch carefully. What happened? The numbers at A1 and B1 actually changed *twice*. When the formula 1+A1 was calculated at B1, it yielded 1+1, or 2. Then, because the value of B1 changed, an automatic recalculation occurred. A1 (1+B1) became 1+2 or 3, and B1 (1+A1) became 1+3, or 4.

Now type !. A1 increases to 5, and B1 becomes 6. These values change every time you type !.

The foregoing examples are somewhat artificial; you probably recognized the forward and circular references as soon as you typed them. If you plan your work carefully, you probably won't write such a formula.

Some forward or circular references, however, are not so obvious. Suppose you set up a worksheet projecting profit, taking into account various revenues and expenses. One of the expenses is employee salaries, which includes profit-sharing. Unless you're careful, you might create a circular reference: salaries depend on profit, but profit depends on salaries. To resolve this circularity, you must calculate a figure for profit before profit-sharing is taken out (a subtotal, in effect), use this figure to calculate profit-sharing, then calculate final profit by subtracting profit-sharing.

Once you understand the issues involved in recalculation, it isn't difficult to avoid recalculation-order conflicts or forward and circular references. These problems usually arise when you must work too quickly to properly plan the worksheet, or hastily modify an existing worksheet. If you have time to approach the problem in an orderly fashion, it is unlikely that you will encounter any problems with recalculation.

THE PRINT COMMAND

Sometimes you need a printed copy of the worksheet. The VisiCalc Print command lets you print any part or all of the worksheet. If you have a printer connected to your computer, you can try it out by printing a copy of the personal budget we've just created.

See "Print Command" in Chapter 3 for a description of how to print the worksheet. If you haven't yet used your printer, see "Appendix A: Controlling the Printer" for instructions.

You can also use the Save option of the Storage command (/SS) and specify the printer, rather than a file name, to print the VisiCalc formulas and formats that produce the worksheet. This doesn't print the values on the worksheet, but it is a permanent record of the structure of the worksheet; you can use it to reconstruct the worksheet if the diskette it is stored on is damaged. It is also a good troubleshooting tool, particularly for finding circular references.

SUMMARY

Once again, this lesson covered a great deal of ground. It reviewed several techniques for using the Replicate command as effectively as possible. It also described a number of new VisiCalc features: the order in which the VisiCalc program calculates and recalculates; functions such as @SUM, @NA, and @ERROR; the Insert, Delete, and Move commands that let you rewrite the worksheet and manipulate entire rows or columns; and synchronized scrolling of the two screen windows.

Most of the important concepts and features of the VisiCalc program have been covered. With what you have learned, this is an excellent time to test your understanding by setting up a worksheet to solve a problem of your own. In this way, you will consolidate your knowledge of the VisiCalc commands and features and develop your own ideas about how they can be used. This will enable you to approach new problems and solve them even more rapidly with the VisiCalc program.

LESSON FIVE

If you have scientific or engineering applications in mind, Lesson Five is particularly relevant to your needs. It concentrates on features that extend the VisiCalc program's usefulness to applications requiring complex or lengthy formulas, numbers with very large or small magnitudes, more sophisticated arithmetic operations, and drawing simple graphs.

MORE ON NUMBERS AND FORMATS

Earlier lessons illustrated some of the ways you can control the display with formatting commands such as /GFI and /F$. This lesson examines the formatting options more closely.

Load the VisiCalc program (as described under "Loading the VisiCalc Program" in Chapter 1) or, if you already have the program running, clear the worksheet with /**CY**. When you clear the worksheet, the global format is set to General, just as if you had typed /GFG. Each entry is controlled by the global format unless you specify a different local format. Type the following:

123.456 ↵
/R: → . →→ :

The same number, 123.456, is displayed at A1, B1, and C1. Because you have not yet set any local formats, all three entries assume the global format–General, which displays numbers with the the greatest precision. As you have seen, however, this is not always the most readable way to display a column of numbers.

Type /**FI** → /**F$** ←. This sets the local format of A1 to Integer and B1 to Dollars-and-cents (two decimal places). The entry line displays the local format setting for A1:

```
A1 /FI (V) 123.456
```

The entries whose local formats you just specified are displayed in rounded form. At A1, 123.456 is rounded down to 123 because .456 is less than .5. At B1, however, 123.456 is rounded up to 123.46 because the last digit (.006) is greater than .005. C1 is controlled by the general format; it still reads 123.456.

Type **/GFI** to change the global format from General to Integer. Entries A1 and B1 are unaffected because they have explicit local formats. C1, however, now reads 123. With the cursor still at A1, change the local format by typing **/FG.** A1 now reads 123.456, and the entry line reads:

A1 /FG (V) 123.456

The local format (General) overrides the global format (Integer).

Type → to move to B1 and erase the local format there by typing **/FD.** This causes the format of B1 to default to the global format, which is currently Integer. B1 now reads 123, and the entry line reads:

B1 (V) 123.456

The local format **/F$** is gone.

Finally, type **/GFG** to set the global format back to General. Now all three entries display 123.456. Position A1 has a local format that overrides the global setting, but the local format is also General. Positions B1 and C1 have no local format, so they are controlled by the General format.

The way numbers are displayed in the General format depends on the column width. Type **/GC7** ↵. Now all three entry positions read 123.46. The VisiCalc program leaves one blank at the left edge of each entry position, then displays as many significant digits as it can.

To compare the flexibility of the global format to a local format, move the cursor to position B1 and type **/F$** to set Dollars-and-cents format, then **/GC6** ↵ to change the column width to six. Positions A1 and C1 now display 123.5, but B1 reads ">>>>>>" (an effect you saw earlier), telling you that a number as large as 123.456 cannot be displayed with two decimal places in a column only six characters wide. Type **12.34** ↵ and the VisiCalc program displays it.

SCIENTIFIC NOTATION

Type /CY to clear the worksheet. At position A1, type 8 nines (99999999) followed by →. This is the largest number that can be displayed in a nine-character column. Type 1+ ← ↵ →.

The calculated result–1+99,999,999, or 100,000,000–is too large to display in ordinary form at B1. To handle this problem, the VisiCalc program switched to *scientific notation*. B1 now reads "1E8", which means 1 times 10 to the 8th power (or 1 followed by 8 zeros).

The E stands for Exponent because the number following it is the exponent of the implied 10. Scientific notation is also used to display very small numbers. At C1 type .000000001 (that's eight zeros) followed by →. The result, 1.E– 9, means 1 times 10 to the – 9th power (or 1 with the decimal point moved left 9 places). Now at D1 type – ← ↵. The result is – 1.E-9 (or –.000000001).

When a number is displayed in the General format, the VisiCalc program shifts between conventional and scientific notation as required to display the calculated value with the greatest precision. Type /**GC**12 ↵. Now all values are displayed in conventional form because the columns are wide enough to show all the required digits. D1, for example, now reads – .000000001.

Next type /**GC**7 ↵ **HOME**. The 99999999 at A1 is displayed as 10.0E7, which means 10.0 times 10 to the 7th power. (The rounded value of 10.0 is only displayed; the accurate version, 99999999, is retained in memory.)

Finally, type /**GC**5 ↵. To display the numbers in these narrow columns, the VisiCalc program has rounded to eliminate all decimal points, displaying A1 as 10E7 and C1 as 1E-9. But D1 shows ">>>>>" because there's not enough room in a five-character column to display – 1E-9 (remember, it leaves one blank at the left of the entry position).

MORE ON VALUE REFERENCES

The VisiCalc program allows you to enter the value–rather than the coordinates–of an entry location in a formula. Clear the worksheet with **/CY** and type the following:

1 → 2 → + **A1/B1** ↵

The entry line reads:

C1 (V) +A1/B1

and C1 reads .5. We know that if the number at either A1 or B1 is changed, the formula at C1 is recalculated. Now type → to move to D1. Watch the edit line as you type **+A1/B1#.** As soon as you type the # after B1, the reference to B1 is replaced by its current value of 2. The edit line now reads:

+A1/2

Type ↵. The entry line reads:

D1 (V) +A1/2

C1 still reads .5. The difference is that the *value* of B1 is entered into the formula at D1, but the *coordinate* B1 is entered at C1. When a different value is entered at B1, the formula at D1 is unchanged but the reference to B1 yields a different result at C1. To verify this, type ← ←4 ↵; C1 changes to .25, but D1 still reads .5.

The effect of # after a value reference on the edit line is similar to the effect of typing ! after a formula, as described in Lesson One. The difference is that ! evaluates the entire formula on the edit line, replacing it with a single number, but # evaluates and enters the value of a single coordinate, so that the rest of the formula can contain changeable elements.

If # is not preceded by a coordinate (such as B1), it is replaced by the current value of the entry where the cursor lies, i.e., the entry you are changing. You can use this feature to take a look at the precise value of a formatted entry on the edit line. For example, set the global format with **/GF$** and type 6 ↵. C1 reads 0.17. Type → to move to C1, then #. The # is immediately replaced on the edit line with the current value of C1 to maximum precision: .166666666666. After checking the number, type **[BREAK]** to cancel the entry.

A word on precision is in order here. The VisiCalc program maintains numbers internally in decimal form. To accommodate large financial figures and high-precision engineering or scientific values, it guarantees precision to 11 digits (and sometimes 12) in base 10.

Certain fractions (such as 1/6) cannot be expressed exactly with any fixed number of significant digits. The twelfth digit (the final 6 in the preceding result) is a guard digit that allows the VisiCalc program to determine which way to round the eleventh digit when a calculation is completed.

MORE ON FORMULAS

In previous lessons you used only simple formulas. As you begin to write more complex formulas involving several arithmetic operations, the way the VisiCalc program evaluates them may not be so obvious.

For example, to evaluate 9+6/3, does it first add 9 to 6 giving 15 and then divide by 3 to obtain 5, or does it first divide 6 by 3 giving 2, and then add 9 to obtain 11? Try it. Clear the screen and type **9+6/3 ↵**. The answer (at A1) is 5. The VisiCalc program, like many hand calculators, evaluates formulas left to right. No arithmetic operator takes precedence over any other.

You can enclose part of a formula in parentheses to force the VisiCalc program to evaluate it first. Move to A2 and type **9+(6/3) ↵**. The answer is 11.

You can put parentheses inside other sets of parentheses up to nine levels. Type the following:

↓ - (↑ ↑ + ((A2-1)/A1)) ↵

The entry line shows the resulting formula: "–(A1+((A2-1)/A1))". The answer (at A3) is – 7. The VisiCalc program first calculates A2– 1 (10), divides it by 5 (2), adds A1 to the result of the earlier calculation (7), then multiplies the whole thing by – 1 (– 7).

More generally, a formula consists of a series of *operands* separated by arithmetic *operators*. An operand can be one of the following:

- A number, with or without a decimal point, minus sign, or exponent:

25	.05
-1267	3.14159
8E6	-.45E-3

- A coordinate or range of coordinates:

A12	B8
B14...M14	H1 . . . H5

- A function:

@SUM(D3 . . . M3,O14)
@MIN(10,B2,B1)
@AVERAGE(C3 . . . M3)

An operator can be one of the following:

+ Addition
- Subtraction
* Multiplication
/ Division
^ Exponentiation

RAISING A NUMBER TO A POWER

The exponentiation operator is the caret (^), above the 6. It raises numbers to a power. Type ↓ 2^3 ↵ to calculate 2 to the 3rd power, or 8.000000. Try another example: type ↓ 2^.5 ↵ to calculate 2 to the 1/2 power, or the square root of 2; the result (at A5) is 1.414214. To find the cube root of 5, type ↓ 5^ (1/3) ↵; the result is 1.709976.

MORE ON FUNCTIONS

Functions were introduced in Lesson Three, where you used @SUM to find yearly totals for income and expenses and to calculate available leisure money as +B2–@SUM(B4 . . . B11), or income minus the sum of expenses. As mentioned earlier, a function can appear in a formula wherever a number or coordinate could appear. Similarly, a formula can be used as an argument in a function.

A function starts with the @ sign followed by the name of the function. Most functions also require a list of arguments; the arguments are separated by commas and enclosed in parentheses. Each argument can be:

- A formula (a series of numbers, values, references to other locations, and functions, separated by arithmetic operators and/or parentheses).
- A range of entries (a series of contiguous locations in a row or column, such as B2 through B7 or D12 through H12). A range is specified by typing or pointing with the cursor to the first location, typing a period (displayed on the edit line as an ellipsis), and typing or pointing to the last location. On the edit line, these ranges would be displayed as B2 . . . B7 and D12 . . . H12.

The number and type of arguments vary from function to function. For example, @PI and @NA require no arguments. Some functions require exactly one or two arguments, while others, such as @SUM, can take any number of arguments.

The @SUM function accepts a variety of arguments. Move to A7 and type:

@SUM(A1.A4,A5*A5,A6^3,11) ↵

The result is 5 + 11 − 7 + 8 + 2 + 5 + 11, or 35.00000.

FINDING MINIMUM AND MAXIMUM VALUES

The @MIN and @MAX functions accept a list of arguments, just like @SUM. The result is the Minimum or Maximum value in the list. If the list of arguments includes negative values, the minimum is the negative value with the greatest absolute value–that is, – 4 is smaller than – 3; the maximum is the negative value with the smallest absolute value.

Type the following:

↓ @MIN(A1.A7)

↓ @MAX(A7,@SUM(A1,A2,A4.A6)) ↵

The result is – 7 at A8 (the minimum) and 35.00000 at A9 (the maximum).

COUNTING NON-BLANK ENTRIES

@COUNT determines the number of nonblank entries in a list of arguments. Any argument that is not a range of coordinates counts as a non-blank entry. This can be puzzling if you include a single coordinate, such as B1, in the argument list; @COUNT counts it as a non-blank entry even if it is blank. To include a single coordinate in the argument list for **@COUNT**, specify it as the beginning and end of a range entry (for example, B1 . . . B1).

CALCULATING THE AVERAGE OF SEVERAL VALUES

@AVERAGE finds the arithmetic mean of the list of arguments; it is equivalent to:

@SUM(arguments)/@COUNT(arguments).

Clear the worksheet and type the following:

1 ↓ ↓ 3 ↓ 4 ↓ ↓ 6 ↓ ↓ 8 ↓ /-- ↓

With the cursor at A10 type **@COUNT(A1.A8)** ↵. A10 contains the count of nonblank entries, or 5. Now type ↓ **@AVERAGE(A1.A8)** ↵. A11 contains the average of 1, 3, 4, 6, and 8, or 4.4.

To check that @AVERAGE is equivalent to @SUM/@COUNT, type ↓ **@SUM(A1.A8)/A10** ↵. This result, too, is 4.4.

Finally, let's change one of the blank entries to a number. Type >**A5** ↵ 5 ↵. The count of nonblank entries at A10 increases to 6; both A11 and A12 increase to 4.5.

DETERMINING THE PRESENT VALUE OF FUTURE CASH FLOW

The @NPV (Net Present Value) function calculates the present value of future cash flow, based on a specified *discount rate*, or cost of money. @NPV requires two arguments: the first is the discount rate (such as 15%, entered as .15); the second is a range of coordinates that contain the cash flow for the periods in question.

The result of the function is the Net Present Value of the cash flows in the range, discounted at the rate specified by the first argument. If we let DR represent the discount rate + 1, the result of the function is:

(ENTRY1/DR) + (ENTRY2/(DR^2)) +
(ENTRY3/(DR^3)) + ... (ENTRYn/(DR^n))

Suppose a project requires an initial cash expenditure of $5000, and is expected to generate cash over a period of five years. Clear the worksheet and make narrower columns with **/CY/GC6** ↵, then type the following to write the cash flows:

→ 1 → 1+ ← ↵
/R:D1.F1:R
HOME ↓
- 5000 → **1000** → **1500** → **2500** → **2000** → **1000**
HOME ↓↓ **.15**↓

Type the @NPV function at position A4:

+A2+@NPV(A3,B2.F2) ↵

The $1000 cash flow in the first year is discounted by 15, the $1500 cash flow in the second year is discounted twice, etc. The result is 288.3.

Now type ↑ and change the discount rate to 10 by typing .1 ↵. The net present value at the lower discount rate (which makes future cash flow worth more today) is 1014.

The internal rate of return of an investment is equal to the discount rate that produces a net present value of 0. You can find this by trial and error. Change the discount rate at A3 until the net present value at A4 is zero (or as close as you can get). When the discount rate is .17, the net present value is 34.81; when the discount rate is .18, the net present value is −85.0.

If you're persistent, you can verify that the net present value is .00000061 (displayed as 6.E− 7) when the discount rate is .1728674256. The internal rate of return of this project is approximately 17.3%.

LOOKING UP VALUES IN A TABLE

The @LOOKUP function selects values from a table. It requires two adjacent lists of values (the table) and two arguments. The lists of values must be in consecutive locations across a row or down a column.

The first list, the *search list*, contains the values that the @LOOKUP function searches. This list must be in ascending order for the function to yield a meaningful result. The second list contains the resulting values that correspond to the search list values. The list of resulting values must be across a row or down a column immediately adjacent to the search list.

The first argument, the *search argument*, is the value to be compared to the search list. The second argument is the range of coordinates that contains the search table.

The @LOOKUP function compares the search argument to each successive entry in the search list. If it finds an entry greater than the search argument, it selects the immediately preceding value from the table of resulting values (the one that corresponds to the last value in the search table that was equal to or less than the search argument).

If no entry in the search list is greater than the search argument, the @LOOKUP function takes the value of the final entry in the list of resulting values. If the first entry in the search table is greater than the search argument, the value of the @LOOKUP function is NA (not available).

To illustrate the use of this function, this example lists the first ten entries in the periodic table of the chemical elements, with their atomic weights and atomic numbers. Clear the worksheet with **/CY/GFL** and type the following:

Element → Weight → Number
HOME↓
H ↓ He ↓ Li ↓ Be ↓ B ↓ C ↓ N ↓ O ↓ F ↓ Ne
HOME↓→
1 ↓ 4 ↓ 7 ↓ 9 ↓ 11 ↓ 12 ↓ 14 ↓ 16 ↓ 19 ↓ 20 ↵
>C2 ↵
1 ↓ 1+ ↑ ↵
/R:C4.C11:R
>**A13** ↵
10.9 → @LOOKUP(←,B2.B11) ↵

The result of the @LOOKUP function at B13 should be 4. Given an experimental atomic weight of 10.9, the @LOOKUP function compared this value against successive values in column B, stopping at the value 11 at B6 which was greater than 10.9. Thus, the atomic weight of 9 at B5 is the matching value, and the corresponding entry, the atomic number in column C, is 4.

If you change the value to be looked up by typing ← 12.1 ↵, the value of the function at B13 changes to 6. If you enter an atomic weight of 0 at A13, the result of the @LOOKUP function is "NA", because the first entry in the range of values to be searched is greater than the value being searched for.

ARITHMETIC FUNCTIONS

The @ABS function finds the absolute value of its argument. For example, @ABS(1) =1, @ABS(- 1) = 1, and @ABS(0) =0.

The @INT function finds the integer portion of its argument without rounding. Think of the @INT function as setting every digit to the right of the decimal point to zero. For example, @INT(1.6) =1 and @INT(- 3.37) =-3.

LOGIC FUNCTIONS

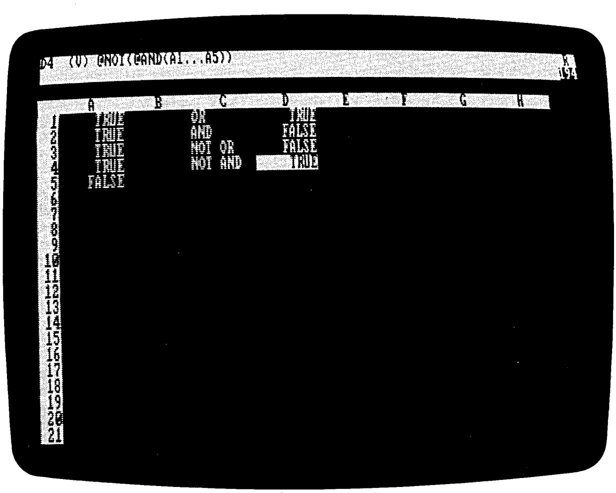
Logic functions deal only with the special values TRUE and FALSE. They include:

- @TRUE and @FALSE, which can be written on the worksheet to produce the logic values TRUE and FALSE. They require no argument.
- @AND and @OR, which test each value specified as an argument. They require a list of one or more arguments.
- @NOT, which negates the logical value of its argument. It requires a single argument.
- @IF, which selects one of two values based on a third, logical value.

The following examples show how the logic functions work together to let you create worksheets that can accommodate different conditions. Type the following:

/CY
/GOR
@TRUE ↓ @TRUE ↓ @TRUE ↓ @TRUE ↓ @FALSE
HOME→ →
OR ↓ AND ↓ NOT OR ↓ NOT AND ↵
>D1 ↵
@OR(A1.A5) ↓ @AND(A1.A5) ↓
@NOT(@OR(A1.A5)) ↓
@NOT(@AND(A1.A5)) ↵

The screen should look like the following photograph:



005-022/P

The logic values (TRUE and FALSE) in column A are referenced by the logic functions in column D. The labels in column C identify which function is in the adjoining location in column D. The @OR function in location D1, whose value is "TRUE" if any of its arguments is TRUE, is TRUE because four of the five referenced locations are TRUE. The @AND function in location D2, whose value is TRUE if all its arguments are TRUE, is "FALSE" because one of the referenced locations (A5) is FALSE.

Location D3, labeled NOT OR, is "FALSE", because the formula at D3 uses the @NOT function to negate (reverse) the value of the @OR function; this value is always to the opposite of the value at D1. Similarly, location D4, labeled NOT AND, is "TRUE" because the formula negates the value at D2. This value is always the opposite of D2.

Change the value at A5 by typing **>A5:@TRUE ↵**. D2 changes to "TRUE" because now all the values at A1 through A5 are TRUE. D4 changes to "FALSE" because the @NOT function there negates the value of the same @AND function used at D2. Change A5 back by typing

@FALSE ↵.

To see how the logic functions can be used to create different numeric values on the worksheet, type **HOME→ /FL25 ↵** to write the value 25 at location B1. Now type the following to write a formula whose value depends on the value at B1:

>A7 ↵
@IF(D1,B1,@SQRT(B1)) ↵

If the first argument of an @IF function (D1 in this example) is TRUE, the value of the function is the value of the second argument (B1 in this example). If the first argument is FALSE, the value of the @IF function is the value of the third argument–@SQRT(B1) in this example.

The formula at A7 tells the VisiCalc program that if any of the values in locations A1 through A5 is TRUE (the test made by the @OR function at D1) the value at A7 is to be the value at B1; if none of the values at A1 through A5 is TRUE, the value at A7 is to be the square root of the value at B1. A7 reads 25, the value at B1.

Change A1 through A4 to FALSE by typing the following:

HOME
@FALSE ↵
/R: ↓ . ↓ ↓ ↓ ↓ ↵

D1 changes to "FALSE", and now A7 reads 5.000000, the square root of A1. Change A1 back to "TRUE" by typing **@TRUE ↵**. D1 changes back to "TRUE" and A7 changes back to 25. You are changing the value at A7 based on whether any of the values in A1 through A5 is "TRUE."

Change the formula at A7 by typing the following:

>A7 ↵
/E → → → → → → **[BKSP] 2** ↵

A7 changes to 5.000000, because now it references the @AND function at D2, whose value is "FALSE" because not all the values at A1 through A5 are TRUE.

A logic function can be used anywhere a formula can be used, so the argument of a logic function can be another logic function. Suppose you wanted to test whether all the values at A1 through A5 are true *and* at least one of the values at D1 through D4 is true; type the following:

@AND(@AND(A1.A5),@OR(D1.D4)) ↵

The outermost @AND function has two arguments: the first is an @AND function that tests the values at A1 through A5, and the second is an @OR function that tests the values at D1 through D4. The value at A7 is "FALSE" because the first argument of the outermost @AND function–@AND(A1 . . . A5)–is not TRUE.

The @AND function used as the first argument is the same as the formula at D2. You can refer to D2 instead of typing the entire function; type the following:

@AND(D2,@OR(D1.D4)) ↵

The value at A7 doesn't change.

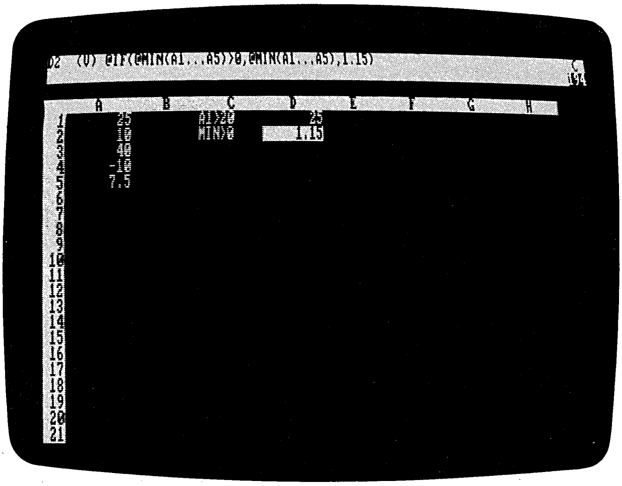
The @TRUE and @FALSE functions aren't the only ways to provide the values tested by logic functions. Comparisons yield either TRUE or FALSE. Type the following:

/CY
25 ↓ **10** ↓ **40** ↓ **-10** ↓**7.5**
HOME → →
A1>20 ↓ **MIN>0** ↵

Column A is a series of numbers whose value is to be tested; Column C labels the functions to be demonstrated. Type the following:

>D1 ↵
@IF(A1>20,A1,20) ↓
@IF(@MIN(A1.A5)>0,@MIN(A1.A5),1.15) ↵

The screen should look like the following photograph:



005-023/P

D1 is 25 (the value at A1) because A1 is greater than 20. D2 is 1.15 because the minimum value at A1 through A5 (– 10, at A4) is not greater than 0. (There is no significance to the number 1.15; it simply demonstrates that a different value can be written.)

The first argument of an @IF function must be a formula that evaluates to either TRUE or FALSE, as the comparisons in the previous example do. The formula can be a reference to another location, if the value at that location is either TRUE or FALSE. Type the following:

HOME→→
+**A**1>20 ↓ @MIN(A1.A5)>0 ↵

Now C1 and C2 are no longer labels, they are logical values. C1 is "TRUE" and C2 is "FALSE". Type the following to use these values to produce the same results as the earlier example:

>D1 ↵
@**IF**(C1,**A**1,20) ↓ @IF(C2,@MIN(A1.A5),1.15) ↵

TRANSCENDENTAL FUNCTIONS AND GRAPHING

The transcendental functions, such as @EXP, @LN, and @SIN, and the Graph format are described in a single example that produces graphs of the transcendental functions. Clear the worksheet and type the following:

1 ↓ 3 ↓ 6 ↓ 20 **HOME**

The Graph format displays the number of asterisks equal to the integer portion of the value of an entry. Type **/F*** at A1. One asterisk replaces the 1. Now type ↓ **/F***. The 3 is replaced by three asterisks. Continue with ↓ **/F*** ↓ **/F***. A3 shows six asterisks and A4 shows eight.

The maximum number of asterisks that can be displayed at an entry position is one less than the column width (remember, the VisiCalc program leaves a blank at the left of each position). Only eight asterisks are displayed at A4, even though its value is 20, because the column width is 9. Type **/GC12** ↵; A4 displays 11 asterisks. Type **/GC30** ↵; now A4 displays 20 asterisks, its full value. Clear the worksheet by typing **/CY**.

To graph a function, you must first supply a series of argument values for the function and calculate the function result for each argument value. For simplicity's sake, begin with a linear function: f(x) =2.5x. Type the following:

>A20 ↵
.1 → .1 **HOME**
+**A20** ↓ + ↑ +B20 ↵
/R: ↓ .A17: RN

A20 is the start value and B20 is the step value for the list of function arguments. A1 to A17 contain the argument values (.1 to 1.7). Next, calculate the corresponding function results. Type the following:

HOME→
2.5***A1** ↵
/R: ↓ .B17:R

The function results are .25, .5, .75, . . . 4.25. To graph these results, you must specify the Graph format as the local format for the locations that contain the function results to Graph. Type the following:

/F*/R: ↓ **.B17:R**

This probably isn't the sort of graph you expected. The problem is that the function results do not fall conveniently in the range 1, 2, 3, etc., which would yield one, two, or three asterisks.

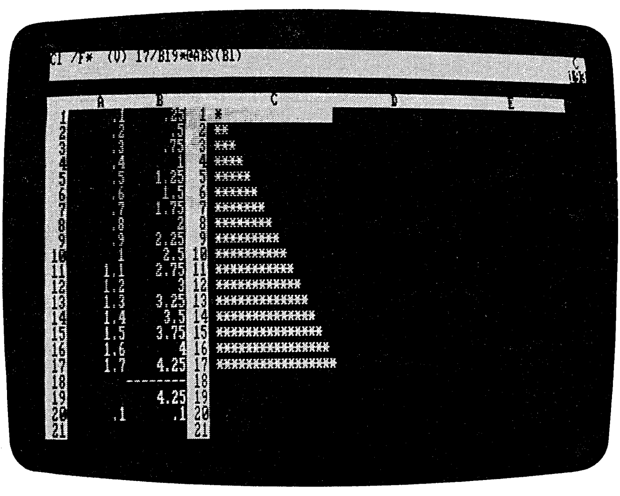
To create a better graph, change the column width to display up to 17 asterisks (to allow a wider range of function results), then scale the function results into the range of 0 to 17. Type **/FD/R:** ↓ **.B17:R** to return to a numeric display, then type the following:

>B18 ↵
/-- ↓ **@MAX(B1.B17)** ↵
/GC8 ↵
HOME → →
/WV;/GC18 ↵
17/B19*@ABS(B1) ↵
/F*/R: ↓ .C17:NR

At B19 the @MAX function finds the upper limit of the function results. C1 through C17 contain the absolute value of the function results, so the lower limit is zero. The screen is split to display the numeric values in narrow columns and the graph in a wider column.

The formula at C1 multiplies each function result by 17 (the maximum number of asterisks) divided by B19 (the maximum function result). The value of this formula lies in the range of 0 to 17/B19*B19, or 17. The graph format is set at C1 and replicated down column C to C17.

This produces a reasonable approximation of a straight line. Because of the way the worksheet is set up, you can graph any set of function results in column B, not just this linear function. The screen should look like the following photograph:

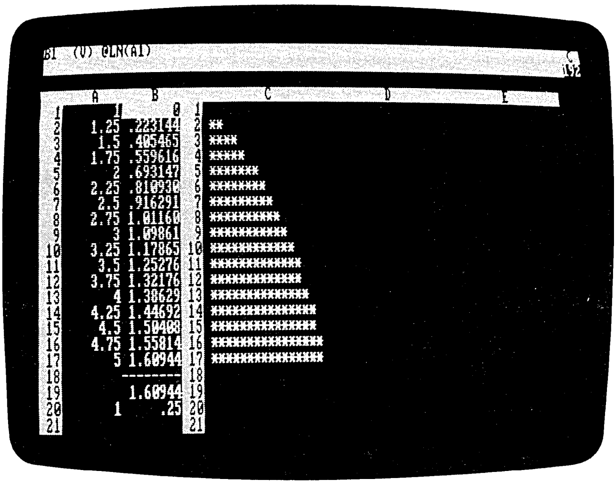


005-024/P

Now type the following:

;>**A20** ↵
1 → .25
HOME→
@LN(A1) ↵
/R:B2.B17:R

The screen should look like the following photograph:



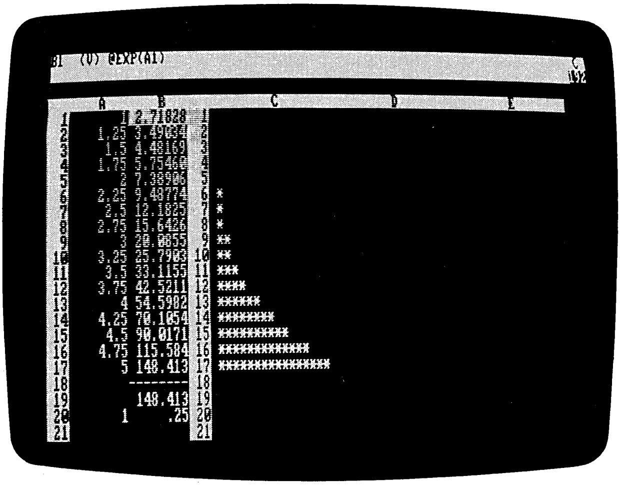
005-025/P

The value of @MAX(B1 . . . B17) at B19 changes to 1.60944. This value affects the formulas in column C so that the results still come out in the range 0 to 17. If the graph of the natural log function doesn't look completely familiar, tilt your head to the right and imagine the X axis on the column and the Y axis on the row.

Now try the exponential function. Type the following:

@EXP(A1) ↵
/R:B2.B17:R

The screen should look like the following photograph:



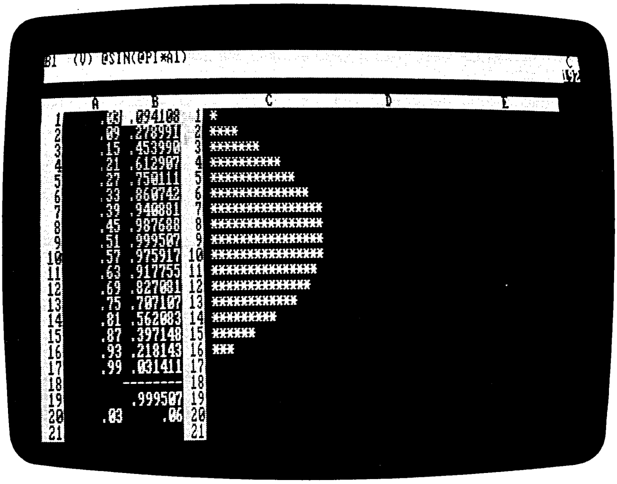
005-026/P

The maximum function result value is now 148.413, and each result is scaled into the range 0 to 17 in column C. Is the graph close to what you expected?

Finally, graph the sine function. Because the VisiCalc program performs its trigonometric calculations in radians, the function is written as @SIN(@PI*A1) and the argument range is changed to obtain a full sine curve in column C. Type the following:

>A20 ↵
.03 → .06
HOME→
@SIN(@PI*A1) ↵
/R:B2.B17:R

The screen should look like the following photograph:



005-027/P

At this point, you may wish to experiment with different argument values and functions. See "Functions" in Chapter 3 for a more detailed description of these and other functions.

CONTROLLING WORKSHEET RECALCULATION

You may have noticed some delay when the VisiCalc program recalculates a worksheet, particularly if there are quite a few transcendental functions. Because function results are calculated to 11- or 12-digit precision, each one takes a fraction of a second to evaluate; a worksheet full of function references can take several seconds to recalculate.

When you load the VisiCalc program or clear the worksheet, the entire worksheet is recalculated each time you change an entry. In many cases this isn't necessary. The Recalculate option of the Global command (/GR) lets you specify whether you want this automatic recalculation or whether you want to specify yourself (by typing !) when to recalculate.

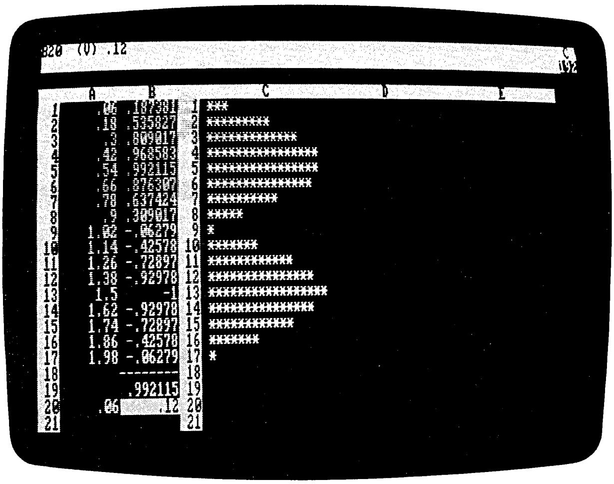
Type **/G**. The prompt line reads "Global: C O R F". Type **R**. Now the prompt line reads "Recalc: A M". The characters following "Recalc:" are options:

A Automatic
M Manual

Type **M**. You have turned off automatic recalculation. The worksheet is recalculated only when you type !.

Now type **>A20** ↵ to move the cursor to the starting value, .03. Type **.06** → **.12** ↵. Now request a recalculation–type !. As you've probably noticed, an exclamation point (!) is displayed next to the C in the upper-right corner of the screen while the recalculation takes place.

The graph changes to display the positive-going portion and the reflection of the negative-going portion of the sine curve. The screen should look like the following photograph:



005-028/P

Now type .06 ← .54 ↵. Specify automatic recalculation by typing /**GRA**. The first thing that happens is a recalculation to update all the figures on the screen.

SUMMARY

This lesson introduced the full range of the VisiCalc program's calculating ability. Besides simple addition, subtraction, multiplication, and division, the VisiCalc program provides exponentiation, transcendental functions, and scientific notation.

You can use functions such as @SUM, @MIN, and @MAX to work with entire rows, columns, or other ranges of numbers. Functions like @COUNT, @AVERAGE, @NPV, and @LOOKUP allow you to quickly handle common problems such as test score averaging, evaluating the terms of a loan, or looking up figures in sales or income tax tables.

You can display the results in a variety of ways, including simple graphs. With practice, you can use the features described in this lesson–in combination with the screen controls and formula replication capabilities–to solve complex problems quickly and easily.

Lesson Five concludes the tutorial chapter of this book. The next chapter is the VisiCalc Command Reference. It contains an in-depth description of each command you used in the lessons.

After you've read this manual and used the VisiCalc program for some of your own applications, we'd appreciate it if you would fill out and send in the Reader's Critique at the back. We hope you find the VisiCalc program an enjoyable and useful tool.

REFERENCE

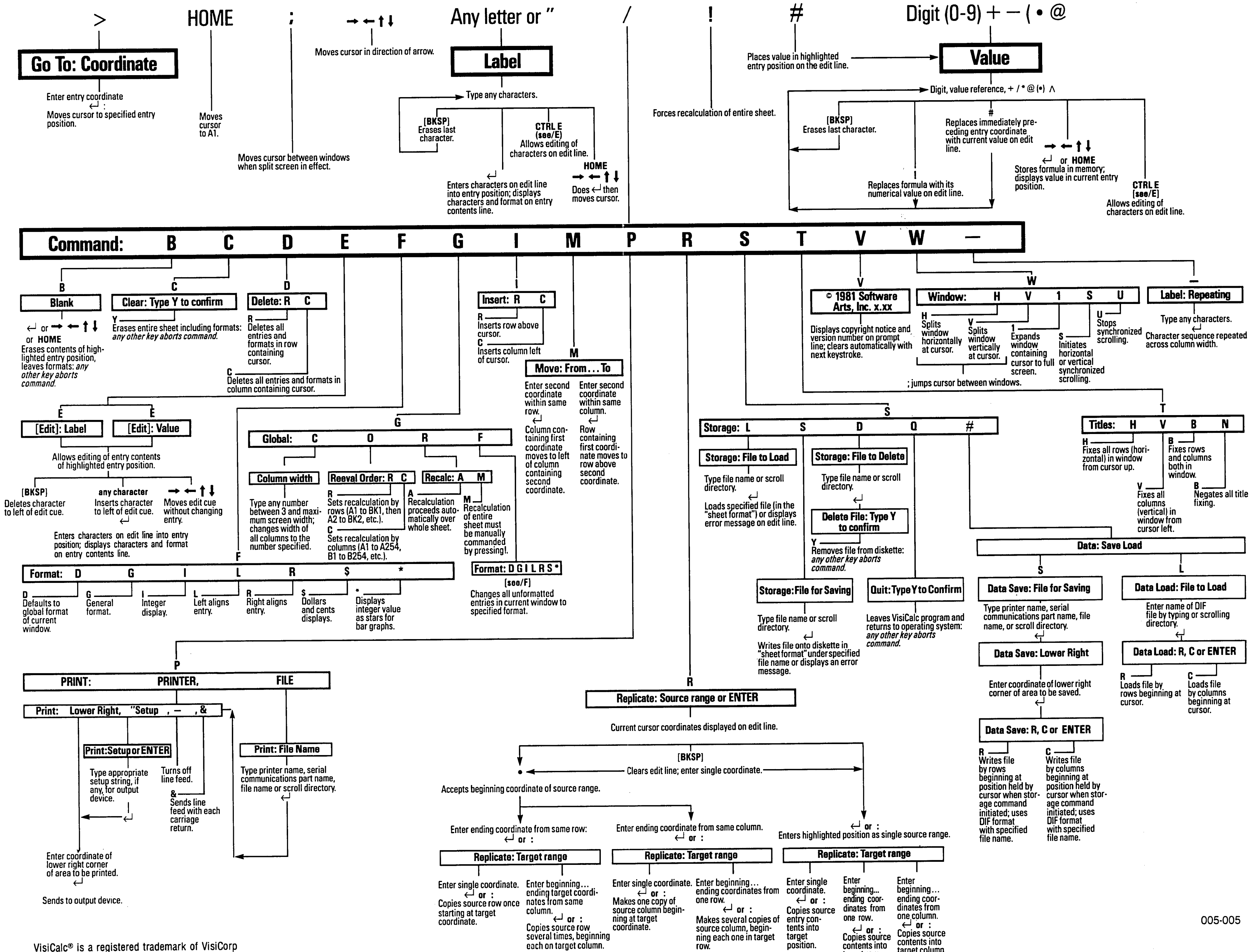
INTRODUCTION

Chapter 3 includes a detailed description of each VisiCalc command, plus topics such as the VisiCalc screen, labels, values, and file names.

The commands and topics are listed alphabetically. Command options are presented in the order they appear on the prompt line of the VisiCalc screen. If you have not read Chapter 2 (the Tutorial), turn first to "The Screen" for general information about the VisiCalc program.

This chapter begins with a chart that shows all VisiCalc commands and their options. Each command description begins with a detailed chart that shows its options.

VISICALC COMMAND STRUCTURE CHART



VisiCalc® is a registered trademark of VisiCorp
DIF™ is a trademark of Software Arts Products Corp.

BLANK COMMAND

B

Blank

↵ or → ← ↑ ↓
or **HOME**
Erases contents of highlighted entry position, leaves formats: *any other key aborts command.*

The Blank command *irretrievably erases* anything written (label or value) at the entry position where the cursor rests. It does not affect the local format, if one is specified.

Type:	Result:
1. Place the cursor on the entry to be erased.	
2. /	Prompt line: Command: BCDEFGIMPRSTVW-
3. **B**	Prompt line: **Blank**
4. ↵, an arrow key, or **HOME**.	The entry position highlighted by the cursor clears.

CLEAR COMMAND

C

Clear: Type Y to confirm

Y

Erases entire sheet including formats:
any other key aborts command.

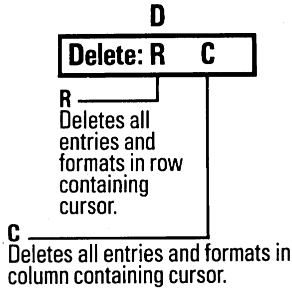
The Clear command *irretrievably erases* all information on the worksheet, returns the cursor to position A1, and resets all worksheet characteristics to the same status as when the program is loaded.

If you want to save the worksheet, be sure to do it (using the Save option of the Storage Command) *before* issuing the Clear command.

Type:	**Result:**
1. /C	**Prompt Line:** Clear: Type Y to confirm
2. Y	The screen darkens for a moment, then reappears with no entries. The status area displays the copyright and version notices.

If you type anything other than Y, the Clear command is canceled.

DELETE COMMAND



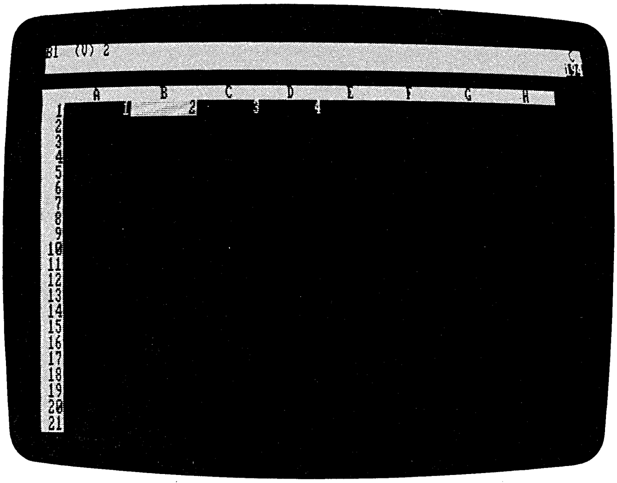
The Delete command *irretrieveably erases* the row or column in which the cursor rests. If a row is deleted, all rows below the deleted row move up to fill the gap; if a column is deleted, all columns to the right move left.

The VisiCalc program adjusts all coordinates on the worksheet to correspond to the new coordinates that result when rows or columns are moved after the Delete command, then recalculates the worksheet. Locations that refer to the deleted row or column display "ERROR".

To use the Delete command, place the cursor in the row or column to be deleted and type /**D**. The prompt line reads "Delete: R C." Type **R** to delete the row in which the cursor is located or **C** to delete the column.

Type:	**Result:**
1. /CY	The worksheet clears and the cursor moves to A1.
2. 1 → 2 → 3 → 1+C1 ↵	

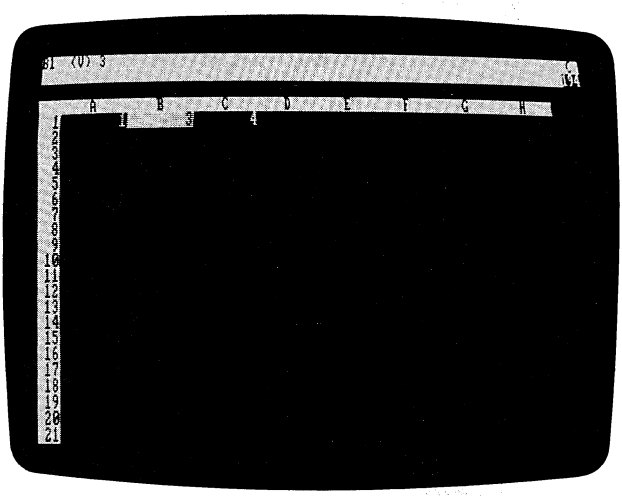
3. **HOME→** The screen should look like the following photograph:



005-029/P

4. **/D** **Prompt Line:** Delete: R C

5. **C** The screen should look like the following photograph:



005-030/P

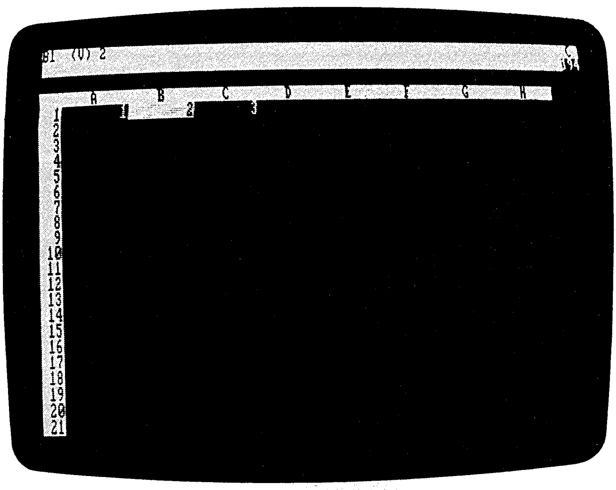
6. → **Entry Line:** C1 (V) 1+B1

When column B was deleted, all columns to the right moved left to fill the gap. This formula has been changed from 1+C1 to 1+B1 so that it still refers to the value you entered at C1 in step 2.

Note: Use the Delete command with care. You cannot recover the information from a row or column once you have deleted it. (The Insert command recreates only the *space* for a row or column, not the actual entries.)

If you delete an entry that is referenced in a formula at another location, all locations that refer to the deleted entry display "ERROR".

Type:	Result:
1. **/CY**	The worksheet clears and the cursor moves to A1.
2. 1 → 2 → 1+B1 ←	The screen should look like the following photograph:

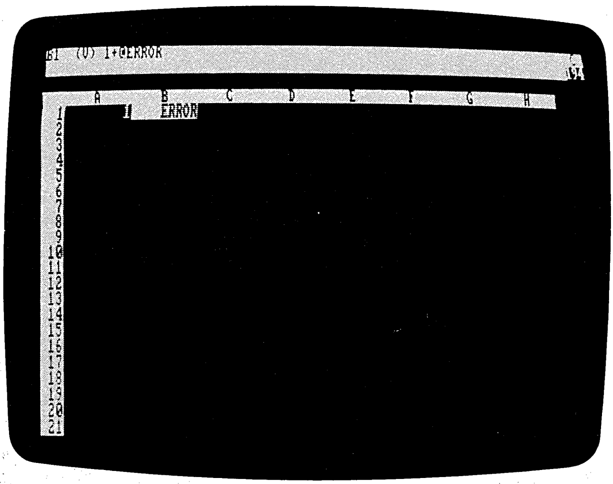


005-031/P

3. **/D**	**Prompt Line:** Delete: R C
4. **C**	**Entry Line:** B1 (V) 1+@ERROR **Position B1:** ERROR

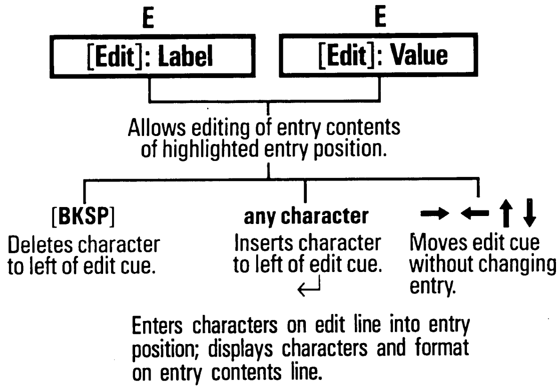
Column B was deleted and the contents of Column C moved over. Position B1 (formerly C1) displays "ERROR" because its formula refers to the row that was deleted. The formula (displayed on the entry line) has changed: the former reference to B1 now reads "@ERROR", the Error function.

The screen should look like the following photograph:



005-032/P

EDIT COMMAND



The Edit Command lets you change an entry without retyping the entire value or label. You can insert or delete characters anywhere in the entry; this is especially helpful when you must change a long or complex formula. The two forms of the Edit Command can be used either to edit something being typed on the edit line (CTRL-E) or to edit an entry already written on the worksheet (/E).

After the Edit command is typed, the entry to be edited is on the edit line. If /E was used, the edit cue is at the beginning of the edit line; if CTRL-E was used, the edit cue is at the end of the edit line. The edit cue can be moved along the edit line without affecting the entry. Typing ← moves it left one character, → moves it right; ↑ moves it to the beginning of the line and ↓ moves it to the end. Typing [BKSP] erases the character to the left of the edit cue; typing any other character inserts that character to the left of the edit cue.

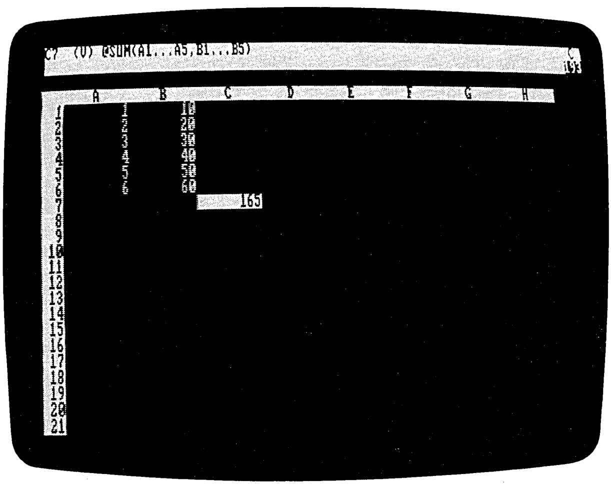
Type ↵ to end the edit command or [BREAK] to cancel it.

To edit the entry at the cursor location, type /E. To edit an entry being typed (not yet written on the worksheet), type CTRL-E.

	Type:	Result:
1.	/CY	The screen clears and the cursor moves to A1.
2.	misteak	Entry Line: Label Edit Line: misteak The edit cue follows the "k".
3.	CTRL-E	Prompt Line: [Edit]: Label Edit Line: misteak The edit cue follows the "k".
4.	← ←	Edit Line: misteak The edit cue highlights the "a".
5.	[BKSP]	Edit Line: mistak The edit cue still highlights the "a".
6.	↓	Edit Line: mistak The edit cue follows the "k".
7.	e ↵	Entry Line: A1 (L) mistake Prompt Line: clear Edit Line: clear A1: mistake

To edit an entry already written on the worksheet, the cursor must be positioned at the location of the entry to be changed.

Type:	Result:
1. **/CY**	The screen clears and the cursor moves to A1.
2. 1 ↓ 2 ↓ 3 ↓ 4 ↓ 5 ↓ 6 **HOME** → 10 ↓ 20 ↓ 30 ↓ 40 ↓ 50 ↓ 60 → ↓ **@SUM(A1.A5,B1.B5)** ↵	The screen should look like the following photograph:



005-033/P

3. **/E**	**Prompt Line:** [Edit]: Value **Edit Line:** @SUM(A1 . . . A5, B1 . . . B5) **C7:** 165 The edit cue highlights the "@ ".

4. → 12 times

Edit Line: @SUM(A1 . . . A5, B1 . . . B5)

The edit cue highlights the comma following "A5".

5. [BKSP]

Edit Line: @SUM(A1 . . . A, B1 . . . B5)

The edit cue remains over the comma.

6. 6

Edit Line: @SUM(A1 . . . A6, B1 . . . B5)

7. ↓ ←

The edit cue highlights the ")".

8. [BKSP]

Edit Line: @SUM(A1 . . . A5, B1 . . . B)

9. 6

Edit Line: @SUM(A1 . . . A6, B1 . . . B6)

10. ↵

Entry Line: C7 (V) @SUM(A1 . . . A6, B1 . . . B6)
Prompt Line: clear
Edit Line: clear
C7: 231

The value at C7 changes to 231 because the @SUM function was changed to add A6 and B6 to the ranges.

FILE NAMES

The worksheet is saved in a *file*. Every file on a diskette must have a different *file name* to distinguish it from the other files. A file name can be up to 8 characters long. The first character must be a letter; the remaining characters can be any letter or number.

The following file names are invalid:

1budget	Begins with a number.
budget%	Includes an illegal character (%).
budgetone	Too long (9 characters).

Disk Drive Prefix

The file name can be preceded by a letter and a colon (:) to specify the disk drive.

File Type Suffix

The VisiCalc program adds a suffix to a file name to identify how a worksheet was saved:

.VC Normal (Saved with /SS).
.PRF Print format (Saved with /PF).
.DIF Data Interchange Format (Saved with /S#S).

See "Storage Command" and "Print Command" in this chapter for a description of these different ways of storing the worksheet. You needn't type the suffix, but the VisiCalc program accepts it if you do.

The suffix distinguishes between file names that otherwise are identical. It's not uncommon to have a file name that differs only in the suffix: "BUDGET.VC", "BUDGET.PRF", and "BUDGET.DIF" are all valid, different file names.

If you save a file with the same name as a file already on the diskette, the file on the diskette is erased and the file you save replaces it. This is fine if you're saving a new version of a worksheet, but could cost you a valuable file if you do it inadvertently. The Storage Command, therefore, asks you to confirm that you want to replace the old version before it stores the new one.

FORMAT COMMAND

F

Format: D	G	I	L	R	$	*
D Defaults to global format of current window.	G General format.	I Integer display.	L Left aligns entry.	R Right aligns entry.	$ Dollars and cents displays.	* Displays integer value as stars for bar graphs.

The Format command controls how an entry is displayed or printed. It applies to one location on the worksheet; the effect is called a *local format*.

Note: The Format option of the Global command controls how the entire worksheet is displayed; the effect is called the *global format.* A local format overrides the global format. When the VisiCalc program is loaded (or cleared with the Clear command) the global format is General, which displays values to the maximum precision possible.

Each location on the worksheet can be assigned a local format. The actual entry (whether a label or value) is not affected by the Format command, only the appearance of the entry. The entry line shows the value or label as it was entered; the full precision of a value is used in calculations.

The entry line shows the local format (if any) assigned to a location. The local format is assigned to the entry, not the location; if the entry is moved to another location, the local format moves with it. Erasing the contents of an entry position with the Blank command, however, does not remove a local format. Clearing the worksheet with the Clear command removes all local formats.

To assign a format to a row or column before writing on it, format the first entry and replicate the format with the Replicate command across the row or down the column.

The width (number of spaces) of an individual location or column cannot be changed. The Column Width option of the Global command changes the width of every column in the display; if the screen is split into two windows, each window can have its own column width.

The following example illustrates all the Format command options. Type this example and save it; it is used to illustrate all the options:

Type:	**Result:**
1. **/CY**	The worksheet clears, the cursor moves to A1, and the global format is set to General.
2. **Label entry** → 1.23456789 → 99.999 ↵	One label and two values are displayed.
3. **/SSformex** ↵	The example is saved in a file named "formex.VC".

Default Format —/FD

The Default format is the format last set with the Global command. If no global format is in effect, the Default is the General format, described later in this topic.

This example first uses the Global command to set the global format so that all numbers are displayed with two decimal places (Dollars-and-cents format, specified with $), uses the Format command to set the local format at C1 to Integer, then uses the Format command again to default to the global Dollars-and-cents format.

Type:	**Result:**
1. Load or type the sample worksheet.	
2. **/GF$**	Entry Line: C1 (V) 99.999 B1: 1.23 C1: 100.00 The worksheet is displayed in Dollars-and-cents format.
3. **/FI**	Entry Line: C1 /FI (V) 99.999 B1: 1.23 C1: 100 C1 is now displayed in integer format; B1 is unchanged.

4. /FD

Entry Line: C1 (V) 99.999
B1: 1.23
C1: 100.00

The Default format is Dollars-and-cents because that was the last global format set.

Even though the entry at C1 was displayed in three different formats, the actual value–as shown on the entry line–remained the same.

General Format —/FG

The General format is in effect when you load the VisiCalc program or clear the worksheet:

- Numeric values are displayed to the maximum precision possible in the column width, aligned with the right edge of the column. The leftmost position of each column is blank so that values won't run together across the worksheet.
- Labels are displayed aligned with the left edge of the column. The label is displayed only to the right edge of the column, but the full label is kept in memory (and displayed on the entry line).

This example uses the Global command to change the global format to Integer, then uses the Format command to set the local format at C1 to General.

Type:	Result:
1. Load or type the sample worksheet.	
2. /GFI	Entry Line: C1 (V) 99.999 B1: 1 C1: 100 The Integer format controls the entire worksheet and has rounded both entries.

3. **/FG**

Entry Line: C1 /FG (V) 99.999
B1: 1
C1: 99.999

The local format at C1 is General, so the value is displayed just as it was when the global format was General. B1, however, is still controlled by the global Integer format.

Integer Format —/FI

Integer format displays all values rounded to the nearest whole number.

Type:	Result:
1. Load or type the sample worksheet.	
2. **/FI**	Entry Line: C1 /FI (V) 99.999 B1: 1.234568 C1: 100 C1 is rounded to the nearest integer. B1 is still controlled by the global format (General).

Left-Aligned Format —/FL

The Left-aligned format displays a label aligned with the left edge of the entry position, and a value aligned with the second space of the entry position. The first character of each entry position is left blank when a value is displayed to keep columns from running together across the worksheet.

This format causes values to be displayed as in the General format (maximum precision); decimal values cannot be displayed both as rounded integers and left-aligned.

Type:	Result:
1. Load or type the sample worksheet.	
2. /FL	Entry Line: C1 /FL (V) 99.999 B1: 1.234568 C1: 99.999 The value at C1 moves to the left side of the entry position (the first position remains blank).
3. /FI	Entry Line: C1 /FI (V) 99.999 B1: 1.234568 C1: 100 The Integer format rounds the value at C1 to 100 and displays it aligned with the right edge of the column.
4. /FL	Entry Line: C1 /FL (V) 99.999 B1: 1.234568 C1: 99.999 The value at C1 is displayed just as it is in the General format (maximum precision) and moved to the left of position C1 (the first position remains blank).

Right-Aligned Format —/FR

The Right-aligned format displays a label or value aligned with the right edge of the entry position.

Type:	Result:
1. Load or type the sample worksheet.	
2. **HOME↓**	Cursor moves to A2.
3. **Label ↵**	**Entry Line:** A2 (L) Label
4. **/FR**	**Entry Line:** A2 /FR (L) Label "Label" shifts to the right of entry position A2.
5. **HOME**	**Entry Line:** A1 (L) Label entry **A1:** Label ent
6. **/FR**	**Entry Line:** A1 /FR (L) Label entry **A1:** Label ent Because the label fills all of position A1, it can't move to the right. As the entry line shows, however, the Right-aligned format is assigned to the entry location.
7. **/GC14 ↵**	Now the label is moved in the entry position to be aligned with the right edge because the label is shorter than the column width and can be moved within the entry position.

Dollars-and-Cents Format —/F$

The Dollars-and-cents format displays all values rounded to two decimal places; it adds two decimal places (.00) to integers and an extra 0 to values with just one decimal place. A $ is not displayed. The command has no effect on labels.

Type:	Result:
1. Load or type the sample worksheet.	
2. /F$	Entry Line: C1 /F$ (V) 99.999 C1: 100.00

Graph Format —/F*

The Graph format displays the number of asterisks equal to the truncated (not rounded) integer value of the entry location. An entry position can display one fewer asterisk than the column width (the first position is blank).

Type:	Result:
1. /CY	The worksheet clears and the cursor moves to A1.
2. 1 ↓ 2 ↓ 3 ↓ 4 ↓ 5 HOME→	Puts a column of numbers in column A.
3. /F*	Sets the Graph format at B1.
4. /R ↵ ↓ .B5 ↵	

5. + ← ↵ — The formula +A1 sets B1 equal to A1.

6. /**R** ↵ ↓ .B5 ↵ **R** — Replicating the formula with a relative reference to A1 sets B2-B5 equal to A2-A5. Now the values are displayed as numbers in column A and a bar graph in column B.

Any number typed at A1-A5 is displayed as a number in column A and a bar of asterisks in column B. (A number larger than the column width displays the maximum number of asterisks, one less than the column width.) See "Transcendental Functions and Graphing" in Lesson Five for more detailed examples of the graph function.

FUNCTIONS

Functions perform more complex calculations than simple addition, subtraction, multiplication, or division. Some functions save the effort of typing frequently-used formulas (such as adding the values of a range of locations); some perform calculations that are not otherwise possible (such as trigonometric functions); and some choose alternative values for calculations (such as looking up tax rates from a table).

A function can be used anywhere a formula can be used. It consists of the name of the function (which always begins with @) followed by an *argument* (or list of arguments) in parentheses. An argument consists of the values (formulas and references to other locations) the function uses to calculate its own value.

The @SUM function, for example, could be written:

@SUM(B1,S2,A4*.23)

This adds the value of locations B1 and S2, and .23 times the value of location A4.

An example of the @SQRT function:

@SQRT(625)

Its value is 25, the square root of 625.

The @CHOOSE function selects one of several alternative values based on the value of the first argument:

@CHOOSE(A4,17,8,23,44)

The value of this function depends on the value of its first argument, A4. If A4 is 1, the value is 17; if A4 is 2, the value is 8; and so forth.

The @ starts a value entry (no preceding + is necessary).

Functions That Use A Single Argument

The arithmetic and trigonometric functions require a single argument.

Arithmetic Functions

The functions in the following table perform the listed arithmetic calculation on a single argument (specified by *v* in the table):

Function	Result
@ABS(*v*)	Absolute value of the argument.
@EXP(*v*)	e (2.71828 . . .) to the power specified by the argument.
@INT(*v*)	Integer portion of the argument.
@LN(*v*)	Natural log (base e) of the argument.
@LOG10(*v*)	Logarithm (base 10) of the argument.
@SQRT(*v*)	Square root of the argument.

Trigonometric Functions

The functions in the following table perform the listed trigonometric calculation on a single argument. All angles are specified in radians (2 pi radians =360 degrees):

Function	Result
@SIN(*v*)	Sine of the argument.
@COS(*v*)	Cosine of the argument.
@TAN(*v*)	Tangent of the argument.
@ASIN(*v*)	Arc sine of the argument.
@ACOS(*v*)	Arc cosine of the argument.
@ATAN(*v*)	Arc tangent of the argument.

Functions That Use a List of Arguments

The functions in the following table perform a calculation with a list of arguments (represented by *list* in the table).The arguments are separated with commas:

Function	Result
@AVERAGE(*list*)	Arithmetic mean of the values in the list. The result is equivalent to @SUM(*list*) divided by @COUNT(*list*).
@COUNT(*list*)	Number of non-blank entries in the list.
@MAX(*list*)	Largest value in the list.
@MIN(*list*)	Smallest value in the list.
@SUM(*list*)	Sum of each value in the list.

Net Present Value —@NPV

The @NPV function calculates the net present value of future cash flows. It takes two arguments: the discount rate, or cost of money, used to discount the future cash flows, and a range of locations that include the cash flows themselves.

Functions Without Arguments

Several functions do not require an argument.

@PI: @PI is the ratio of the circumference of a circle to its diameter, 3.1415926536.

@NA and @ERROR: @NA (Not Available) is used when a worksheet is set up before the data is written. Because a blank location evaluates to 0, "ERROR" is displayed at each location where zero appears as a denominator and incorrect or misleading values can be produced at other locations.

Writing @NA at the blank locations causes the VisiCalc program to evaluate all entries that refer to those positions as "NA". All formulas on the worksheet are legal.

The result of an illegal calculation is displayed on the worksheet as "ERROR". This can be caused by a too-deep nesting of "(" or "+", an error in writing a formula, or deleting a row or column that is referenced in a formula at another location. "ERROR" is displayed at the entry position that contains the error and all locations that refer to it.

The @ERROR function causes "ERROR" to be displayed at the location where it is entered and all locations that refer to it.

@TRUE and @FALSE: @TRUE and @FALSE are used with the logic functions described later in this section. They cause "TRUE" or "FALSE" to be displayed at the locations where they are entered. The values "TRUE" and "FALSE" are also displayed when the comparison operators (<, >, =, < = , > = , and < >) are used.

Logic Functions

A logical value is one whose value is either TRUE or FALSE. A logic function is one that manipulates logical values. Logical calculations are similar to mathematical operations, but operate only on this special set of values.

Comparison Operators: The comparison operators work on two numeric values and evaluate to a logical value. The equal (=) and not equal (< >) operators can be used to compare logical values. For example, the formula 4>1–four is greater than one–evaluates to the logical value TRUE. The formula 5= 3– five is equal to three–evaluates to the logical value FALSE.

These comparisons can be used either as entries on the worksheet or arguments in a logic function. If used as entries on the worksheet, they should not be written at a location referenced by a function that does not use logic functions. If this happens, the value of such functions is ERROR.

The following table lists the comparison operators (it assumes the operator is preceded by *value1* and followed by *value2*):

Operator	Value
<	TRUE if *value1* is less than *value2*, FALSE if it is not.
>	TRUE if *value1* is greater than *value2*, FALSE if it is not.
=	TRUE if *value1* is equal to *value2*, FALSE if it is not.
< =	TRUE if *value1* is less than or equal to *value2*, FALSE if it is not.
> =	TRUE if *value1* is greater than or equal to *value2*, FALSE if it is not.
< >	TRUE if *value1* is not equal to *value2*, FALSE if it is.

@NOT: @NOT takes a single logical value as its argument (i.e., one whose value is TRUE or FALSE); the function's value is the opposite logical value. The value of @NOT(A1) is FALSE if A1 is TRUE and TRUE if A1 is FALSE. If the value of A1 is NA, the value of @NOT(A1) is also NA. If the value of the argument is anything other than TRUE, FALSE, or NA, the value of the @NOT function is ERROR.

@AND: @AND takes any number of arguments, each of which must be a logical value or a range of logical values. Its value is TRUE if all the arguments are TRUE, FALSE if any of the arguments is FALSE. If any of the arguments is not logical or evaluates to ERROR, the value of @AND is ERROR. If any of the arguments evaluates to NA and all other arguments are TRUE or FALSE, the value of @AND is NA.

@OR: @OR takes any number of arguments, each of which must be a logical value or a range of logical values. Its value is TRUE if any of the values is TRUE and FALSE if all the values are FALSE. If any of the arguments is not logical or evaluates to ERROR, the value of @OR is ERROR. If any of the arguments evaluates to NA and all other values are TRUE or FALSE, the value of @OR is NA.

@IF: @IF takes three arguments. The first must be a logical value; the second and third can be any value. The function evaluates to the value of the second or third argument, depending on the value of the first:

Value of First Argument	Value of @IF
@TRUE	Value of second argument
@FALSE	Value of third argument
@NA	@NA
Not logical or @ERROR	@ERROR

For example, the value of @IF(D5,2,3) is 2 if D5 is TRUE and 3 if D5 is FALSE. The value of @IF(D5,E1,E2) is the value of E1 if D5 is TRUE and the value of E2 if D5 is FALSE.

@ISNA and @ISERROR: @ISNA takes one argument. Its value is TRUE if the value of the argument is @NA and FALSE if the value of the argument is anything else.

@ISERROR takes one argument. Its value is TRUE if the value of the argument is ERROR, and FALSE if the value of the argument is anything else.

@ISNA and @ISERROR are used to manipulate entries written as @NA or @ERROR and return a value that is not automatically designated NA or ERROR.

Functions That Select Alternative Values

Two functions can be used to select alternative values for calculations. These allow the worksheet to handle different situations or projections.

@CHOOSE: The @CHOOSE function takes one of the values in its list of arguments. The first element in the list is the index to the following arguments.

For example, in @CHOOSE(A4,17,6,33,39), A4 is evaluated first. If A4 is 1, the result is 17; if A4 is 2, the result is 6; and so on.

@CHOOSE evaluates to NA if the first argument is zero or less, or if the value of the first argument is greater than the number of remaining arguments.

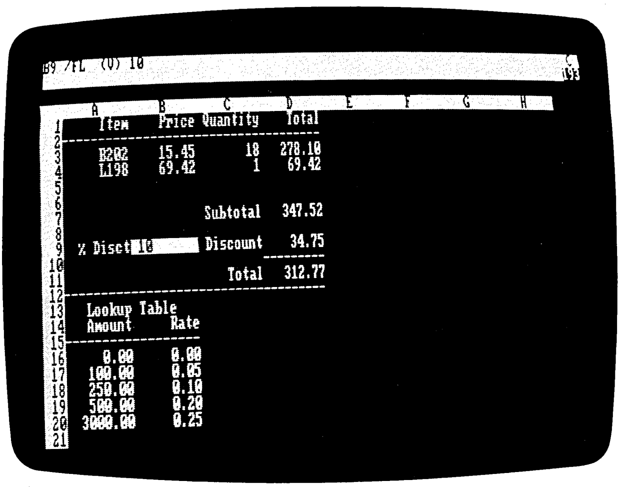
@LOOKUP: The @LOOKUP function looks up a value in a table, a form of calculation frequently used in financial calculations. Tax calculations, for example, require looking up gross pay in a tax table and using the corresponding tax rate.

@LOOKUP searches for a value in a range of locations and evaluates to a corresponding value from an adjacent range. The range to be searched can be in either a row or a column; the adjacent range must be in the column to the right of the range searched or the row below the range to be searched.

The value being looked up is compared to successive values in the range to be searched a value is found that is larger than the value being looked up (or until the end of the table is reached). The entry in the adjacent range that precedes the match position is the value that the @LOOKUP function assumes.

Two arguments are required: the first is the value to be searched for, and the second specifies the range that contains the table of values to be searched.

The @LOOKUP function can be used to calculate an invoice. The total amount of goods purchased is looked up in a table, and the corresponding value from the adjacent range is used as the discount percentage to calculate total amount of the invoice is found. Those calculations are shown in the following photograph:



005-034/P

The lookup table is in positions A16-A20. The cost of goods purchased is at D7. The @LOOKUP function is written at B9:

@LOOKUP(D7,A16...A20)*100

Multiplying the value by 100 makes it a percentage that is later divided by 100 (at D9). Although the lookup range is technically a forward reference, it makes no difference in this case because the values in the table are constants.

The formula at D9 is +D7*B9/100. The value at A20 is 3000.00, corresponding to a discount of 0.25 at B20. If the invoiced items total 347.52, the discount is 10% (the value that corresponds to 250.00). If the invoiced items total 3000.00, the discount is 25%.

The **R** in the upper-right corner of the screen indicates that the worksheet is recalculated by row because the price-times-quantity calculations are made across rows. The left-aligned format (indicated by "/FL" in the entry line) is used to improve the readability of the display; for a description of display formats, see "Format Command" in this chapter.

Making the VisiCalc Program Less Precise

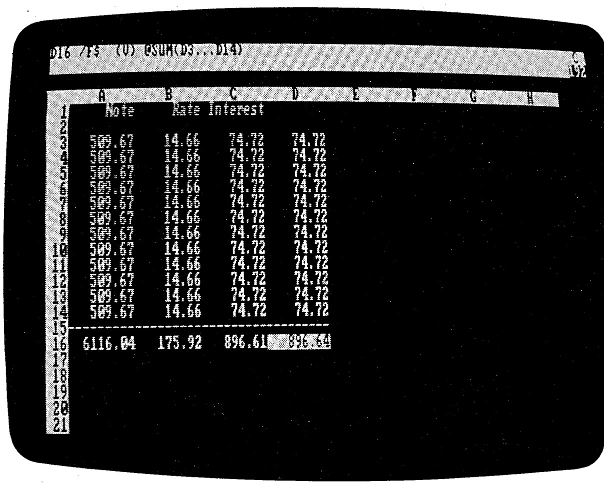
Because the VisiCalc program calculates and rounds to 11 (and sometimes 12) decimal places, differences of pennies and even dollars can occur between VisiCalc results and those produced by ordinary 2-place precision calculations. This difference can cause problems when attempting to balance books and during audits.

To solve this problem, you can use the following *rounding formula* to reduce the calculating precision to just two decimal places:

(@INT((*coord*)*100+.5))/100

coord is the location of the original formula.

For example, assume there are twelve notes worth $509.67 each. The interest paid on those notes is 14.66%. What will the value of the twelve notes be after one year? The following photograph illustrates the effect of the rounding formula:



005-035/P

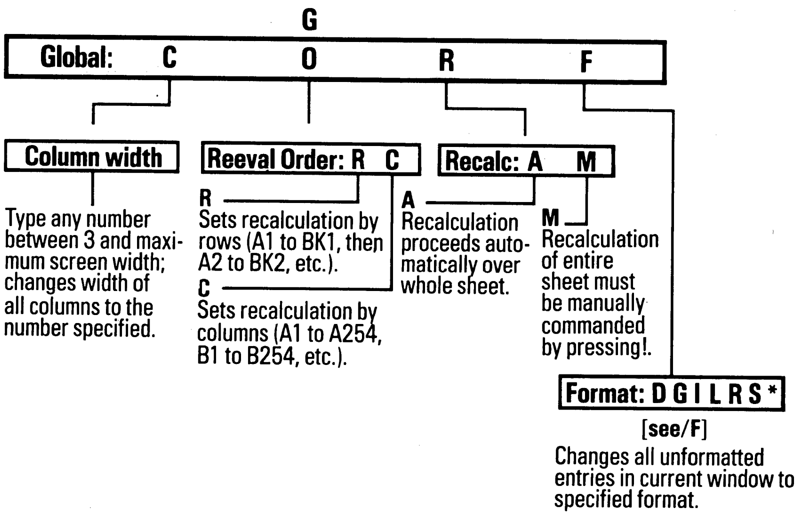
The value 74.717622 is rounded to 74.72 *for display only.* Column C shows the result of multiplying the full precision by twelve: 896.61. Column D shows the result of multiplying the rounded amount by twelve (the result produced by a calculator or paper and pencil): 896.64. If the difference of 3 cents isn't acceptable, a rounding formula can be used.

The formula at C3, which produces the more accurate result, is +A3*B3/100. The rounding formula, at D3, is:

(@INT((C3)*100+.5))/100

The rounding formula is replicated from D4 to D14; the @SUM function is used at D16 to add D3 through D14. This sum matches the less-precise calculation because the rounding formula has held precision to two decimal places.

GLOBAL COMMAND



The Global command affects the entire display, unless the screen is split into two windows, in which case it affects the window that contains the cursor. The command has four options:

C Column width
O Order of recalculation
R Recalculation priority (Automatic or Manual)
F Format

Column Width —/GC

The Column Width option changes the width of all columns in a window. The minimum width is 3; the maximum is the width of the screen less 3 (for the row numbers down the left side). Individual column widths cannot be set.

As many columns as possible are displayed. Setting the column width to half the maximum or more removes vertical titles.

Changing the column width changes the way some entries are displayed. Labels longer than the column width are truncated. Values can be affected in one of three ways:

- Numbers are rounded when necessary.
- Numbers are displayed in scientific notation if this allows greater precision.
- The entry location is filled with > signs if the number is too large to be represented in any form.

These changes affect only the way entries are displayed. Values and labels are unchanged in memory. The entry line displays the full value or label at the cursor location, regardless of how the entry is displayed.

To enter the value appearing on the entry line onto the edit line, type #. The prompt line must be clear to do this.

Type:	Result:
1. **/CY**	The worksheet clears, the cursor moves to A1, and the column width is set to 9.
2. **This line is too long.** ↵	**Entry Line:** A1 (L) This line is too long. **A1:** This line
3. **/G**	**Prompt Line:** Global: C O R F
4. **C**	**Prompt Line:** Column width
5. **18** ↵	**Entry Line:** A1 (L) This line is too long. **A1:** This line is too l
6. **/GC22** ↵	The entire label is displayed.

Numbers are rounded as necessary to fit in the entry location. If a number is too large to display in the number of available spaces (the column width less 1), the entry location is filled with > signs.

Type:	Result:
1. /CY	The worksheet clears and the cursor moves to A1.
2. 123456789 ↵	Entry Line: A1 (V) 123456789 A1: 1.2346E8 The first column position is blank. The number is displayed in scientific notation, with the final digit rounded up.
3. → 1.23456789 ↵	Entry Line: B1 (V) 1.23456789 B1: 1.234568
4. /GC3 ↵	Entry Line: B1 (V) 1.23456789 A1: >> B1: 1. A1 cannot represent the number in any form. The number at B1 is rounded.

Order of Recalculation Option —/GO

The Order of Recalculation option tells the VisiCalc program whether it should recalculate the worksheet by rows or by columns. The order of recalculation is noted by the letter in the upper right corner of the screen–C for Column-order recalculation, R for Row-order. When the VisiCalc program is loaded, it is set for column-order recalculation.

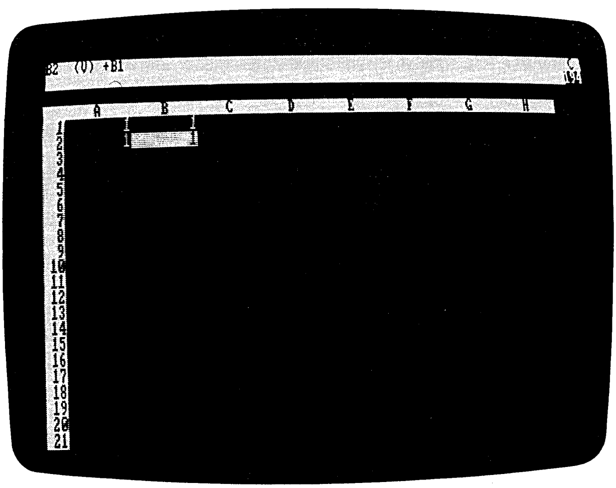
When the VisiCalc program calculates in *column* order, it begins at A1 and works down column A until it comes to the final entry in that column. Then it moves to B1 and works its way down to the last entry in that column. This continues until all values are calculated.

When the VisiCalc program calculates in *row* order, it begins at A1 and works across row 1 until it comes to the final entry in that row. Then it moves to A2 and works its way across to the last entry in that row. This continues until all values are calculated.

If the VisiCalc program appears to evaluate formulas incorrectly, formulas may be written at locations where they are calculated before locations they reference are calculated (a condition called a *forward reference*). Order of recalculation is described in more detail under "Values" in this chapter and "Order of Recalculation" in Lesson Four. "Forward and Circular References" in Lesson Four describes these conditions and how to circumvent them.

The following example requires column-order recalculation to produce correct results, and shows what happens when the worksheet is recalculated by rows.

Type:	Result:
1. **/CY**	The worksheet clears, the cursor moves to A1, and column-order recalculation is set.
2. **1 ↓ +A1 HOME → +A2 ↓ +B1 ↲**	The screen should look like the following photograph:

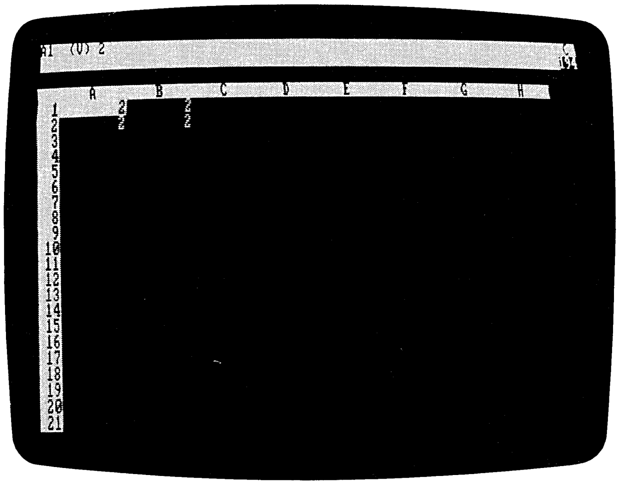


005-036/P

3. **HOME** 2 ↵

A1 changes to 2 and the other values are recalculated.

The screen should look like the following photograph:



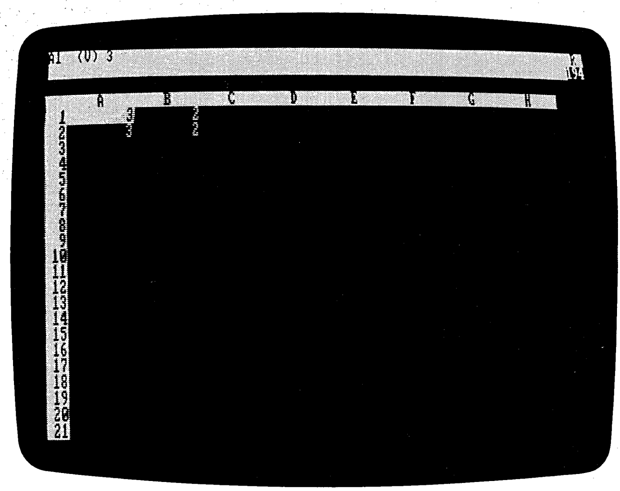
005-037/P

4. **/GOR**

The order of recalculation indicator changes to "R"

5. 3 ↵

The screen should look like the following photograph:



005-038/P

This is incorrect. B1 should be the same as A2, but it displays a different value because B1 was recalculated *before* A2 was recalculated. To get a correct result in a case like this, you must force a recalculation by typing ! when the prompt line is blank.

The ERROR value isn't necessarily displayed when you create a forward or circular reference. If you save the worksheet and reload it, however, "ERROR" is displayed at each location that contains a forward or circular reference. In this case, you must also force a recalculation by typing !.

Note: Because forward and circular references can make it difficult to use a worksheet, you should avoid writing them, and correct them when you find them. If you load a worksheet and the ERROR value is displayed at entry positions, check the formulas at those locations for forward or circular references unless you know the ERROR value is caused by division by zero.

The next example requires row-order recalculation to produce correct results, and shows what happens when the worksheet is recalculated by columns.

	Type:	**Result:**
1.	**/CY**	The worksheet clears, the cursor moves to A1, and recalculation is set to column-order.
2.	**/GOR**	Changes to row-order recalculation.
3.	**1 → +A1 HOME↓ +B1 → +A2 HOME 2 ↵ 3 ↵**	The results are correct.
4.	**/GOC**	Changes to column-order recalculation.
5.	**4 ↵**	These results, as in the previous example, are incorrect.
6.	**!**	The extra recalculation produces correct results.

Recalculation Priority Option —/GR

The Recalculation Priority option specifies whether the VisiCalc program recalculates the worksheet each time a value is typed, or whether the worksheet is recalculated only when ! is typed. Manual recalculation speeds entry of columns or rows or figures by eliminating the automatic recalculation after each entry.

In response to /GR, the prompt line reads:

```
Recalc: A M
```

The characters following "Recalc:" are options:

A Automatic
M Manual

Under manual recalculation, only the formula at the cursor location is recalculated. To update the worksheet, you must force a recalculation by typing !. This does not restore automatic recalculation; the Global command must be used again (/GRA).

Format Option—/GF

The Format option assigns a format to each entry position on the worksheet that does not have a local format. The formats available are the same as those for local formats; see "Format Command" in this chapter for their description.

If the screen is split into two windows, each can have a different global format.

GO TO COMMAND

>

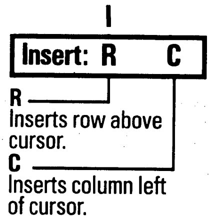
Go To: Coordinate

Enter entry coordinate
↵ :
Moves cursor to specified entry position.

The Go To command moves the cursor directly to an entry position. It is invoked by typing >; the prompt line asks for the coordinate of the destination.

	Type:	**Result:**
1.	>	**Prompt Line:** Go to: Coordinate
2.	B20 ↵	The cursor moves to location B20 (column B, row 20).

INSERT COMMAND



The Insert command places space for an additional row or column on the worksheet.

In response to /I, the prompt line reads "Insert: R C". The "R" and "C" are options:

R Insert a Row.
C Insert a Column.

An individual entry location cannot be inserted, only an entire row or column.

When a row or column is inserted into the range of a function, the range is expanded to include the insertion. (A row or column cannot be inserted before the first location or after the last location of the range; the point of insertion must lie between the first and last locations.)

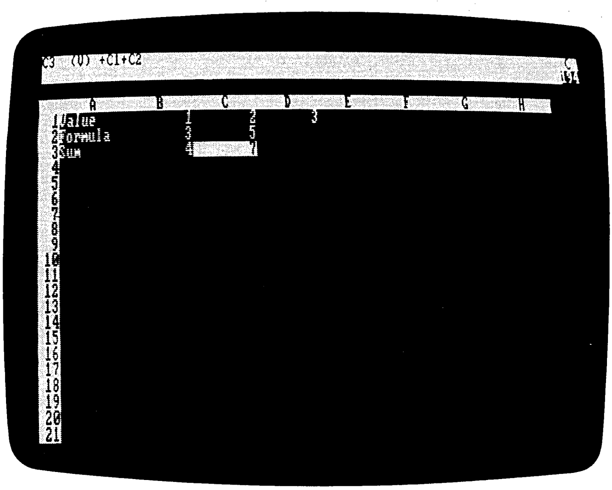
Because rows move down and columns move to the right to make space for an insertion, a row cannot be inserted if row 254 contains an entry; a column cannot be inserted if column BK contains an entry. To make an insertion, the entries in these locations must be erased (with the Blank command) or moved (with the Move command) and the worksheet saved and loaded.

Row Option —/IR

The Row option inserts a blank row immediately above the row that contains the cursor. The row that contains the cursor and all rows below move down one to make room for the new row.

All location references in formulas on the worksheet are changed to correspond to the new coordinates of the rows that moved. For example, if a formula contains the coordinate C2 and a row is inserted above row 2, the coordinate in the formula is changed to C3.

	Type:	Result:
1.	**/CY**	The worksheet clears and the cursor moves to A1.
2.	**Value** → 1 → 2 → 3 **HOME**↓ **Formula** → +B1+C1 → +C1+D1 **HOME**↓↓ **Sum** → +B1+B2 → +C1+C2 ↵	Entry Line: C3 (V) +C1+C2 The screen should look like the following photograph:



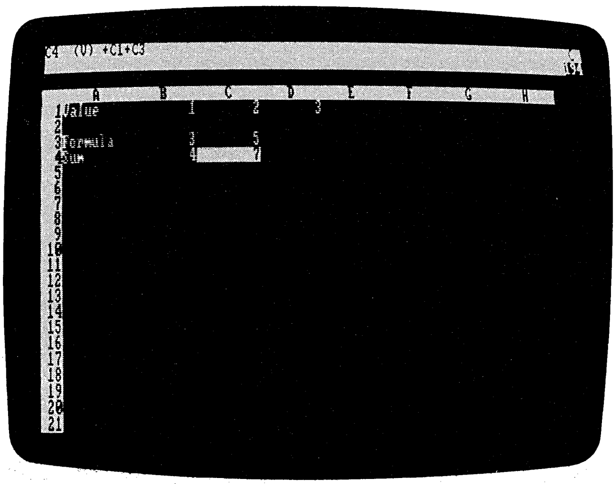
005-039/P

3. ↑ — Cursor moves up to C2.

4. /IR — Row 2 moves down (becoming row 3), creating a new blank row 2.

5. ↓ ↓ — Entry Line: C4 (V) +C1+C3

The location reference in the formula has changed because of the inserted row– from +C1+C2 to +C1+C3. The screen should look like the following photograph:



005-040/P

6. ↑ ↑ /DR ↓ — Entry Line: C3 (V) +C1+C2

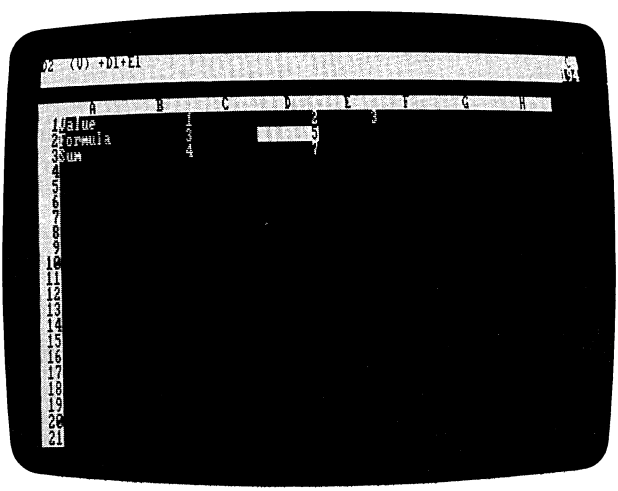
The new row is deleted and the formula returns to its original value.

Column Option —/IC

The Column option inserts a blank column immediately to the left of the column that contains the cursor. The column that contains the cursor and all columns to the right move right one to make room for the new column.

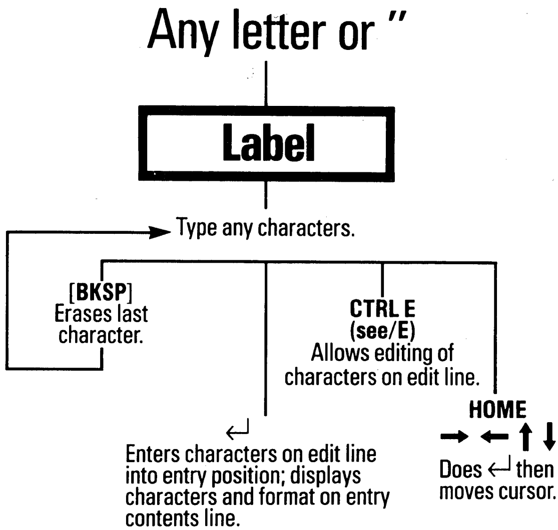
All location references in formulas on the worksheet are changed to correspond to the new coordinates of the columns that moved. For example, if a formula contains the coordinate C2 and a column is inserted to the left of column C, the coordinate is changed to D2.

Type:	Result:
1. If the Row option example isn't on the screen, type steps 1 and 2 from the preceding example.	
2. ↑	Entry Line: C2 (V) +C1+D1
3. /IC	Column C moves right (becoming column D), creating a new, blank column C.
4. →	Entry Line: D2 (V) +D1+E1
	The location reference in the formula has changed to reflect the inserted column–from +C1+D1 to +D1+E1. The screen should look like the following photograph:



005-041/P

LABELS



A label is an entry not intended to be used in calculations; it usually gives some information about the entries on the worksheet. All keyboard characters are allowed in a label.

When the prompt and edit lines are clear, the VisiCalc program looks at the first character typed to determine whether the entry is a label or a value. If the character is not a number or arithmetic operator, the entry is assumed to be a label. The prompt line reads "Label" and the edit line contains the character that was typed followed by the edit cue. As more characters are typed, they appear on the edit line.

The label is displayed at the cursor location as it is typed; it is not permanently written until ↵, an arrow key, or HOME is typed. If something else is written at that location, the label replaces it. If the label is longer than the column width, it is truncated on the display but the full length is kept in memory. When the cursor is at a location that contains a label, the entry line contains the coordinate, followed by (L) and the full label.

To display a label longer than the column width, write it in consecutive locations across a row. When the first position is filled, type → and the next part of the label; continue this until the entire label is written.

If you make a typing error while typing a label, type [BKSP] to erase the last character typed and continue typing. The Edit command can also be used to correct a label after it is written on the worksheet.

If a formula references a location that contains a label, the reference has a value of 0.

Type:	Result:
1. **HOME**	The cursor moves to A1.
2. **P**	Entry Line: A1 Prompt Line: Label Edit Line: P A1: P
3. **eriod**	Prompt Line: Label Edit Line: Period A1: Period
4. ↵	Entry Line: A1 (L) Period Prompt Line: clear Edit Line: clear A1: Period If an arrow key is typed to end the label, the cursor moves to the next location in the direction specified by the arrow.

To start a label with a character that would normally begin a value–a number, arithmetic operator, (, or @– type a quotation mark (") before you type the first character of the label. The " merely tells the VisiCalc program that the entry is a label; it isn't the first character of the label.

Type:	Result:
1. HOME→	The cursor moves to B1.
2. "	Prompt Line: Label Edit Line: clear
3. .575*B2	Prompt Line: Label Edit Line: .575*B2 B1: .575*B2
4. ↵	Entry Line: B1 (L) .575*B2 Prompt Line: clear Edit Line: clear B1: .575*B2 The "(L)" on the entry line identifies the entry as a label. If the entry were a value, the entry line would contain "(V)" and B1 would contain the calculated value of the formula.

Move Command

M

Move: From...To

Enter second coordinate within same row.	Enter second coordinate within same column.
↵	↵
Column containing first coordinate moves to left of column containing second coordinate.	Row containing first coordinate moves to row above second coordinate.

The Move command moves the entire row or column in which the cursor is located to another position on the worksheet. All location references in formulas are changed to correspond to the new coordinates that result when the rows or columns are moved.

To carry out the Move command, the VisiCalc program asks for the coordinates of the origin (From) and destination (To) of the move. Both the origin and destination locations of the move are specified with coordinates (either by typing or pointing with the cursor). The coordinates are separated with three periods.

In response to /M, the prompt line reads "From . . . To" and the current cursor location appears on the edit line. The current location is assumed to be the origin of the move. Moving the cursor or typing a period (.) confirms the origin coordinate and causes three periods (. . .) to appear; the destination coordinate can now be specified either by typing it or moving the cursor.

The VisiCalc program determines whether to move a row or column by the difference between the origin and destination coordinates: if the cursor is moved *horizontally* (or the *letter* is changed in the typed destination coordinate), a column is moved; if the cursor is moved *vertically* (or the *number* is changed in the typed destination coordinate) a row is moved. If *both* the row and column are changed between the origin and destination coordinate (or the cursor is moved to a different row *and* a different column), the VisiCalc program can't tell whether to move a row or column; the command is canceled.

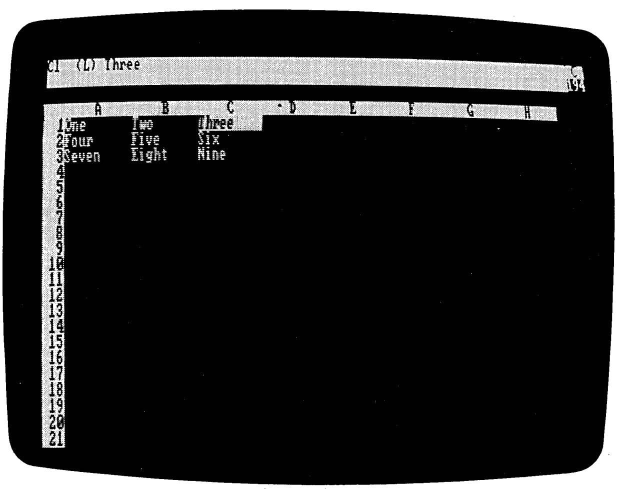
Moving a row or column can create a forward reference or circular reference, requiring extra recalculations or even preventing the VisiCalc program from yielding accurate results. See "Values" in this chapter and "Order of Recalculation" and "Forward and Circular References" in Lesson Four for a description of these conditions and how to circumvent them.

Moving a Row

The Move command deletes the row from the origin, moves all rows below the origin up one to fill the gap, then inserts the row by moving the destination and all rows below it down one to create a gap. Because of this, when moving a row down the destination row is specified as the row *below* the intended location; when moving a row up the destination is specified exactly.

After the row is moved, the cursor returns to its original location.

Set up the screen to look like the following photograph:



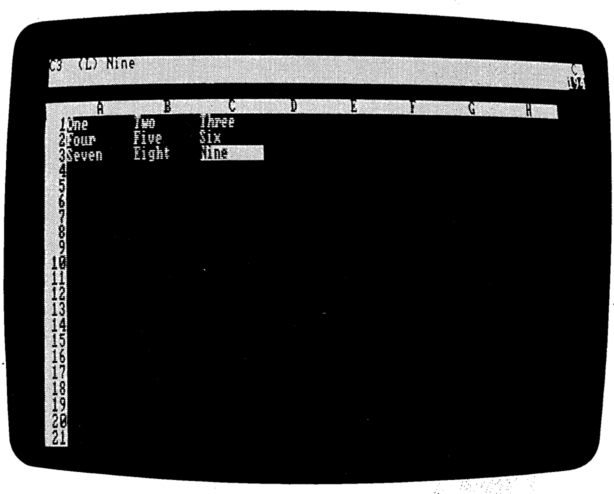
005-042/P

Type:	**Result:**
1. /M	**Entry Line:** C1 (L) Three **Prompt Line:** Move: From . . . To **Edit Line:** C1
2. ↓	**Edit Line:** C1 . . . C2 The VisiCalc program adds the " . . . " to separate the origin from the destination. The cursor has moved to C2.
3. ↓ ↓	**Edit Line:** C1 . . . C4 Because a row is being moved down, the destination is specified as the row *below* the intended destination. Row 4 is specified; row 1 will move to row 3, because all rows below row 1 move up after it is deleted from its present position.
4. ↵	**Entry Line:** C1 (L) Six Row 1 moves to row 3. The screen should look like the following photograph:

005-043/P

The next example moves row 3 back to its original position. Because it moves a row up, the actual destination row is specified.

Type:	Result:
1. ↓ ↓	Entry Line: C3 (L) Three The cursor moves to C3.
2. /M	Prompt Line: Move: From . . . To Edit Line: C3
3. .	Edit Line: C3 . . . The period confirms the origin location.
4. C1	Edit Line: C3 . . . C1 The destination row is 1.
5. ↵	Row 3 moves to row 1. The screen should look like the following photograph:



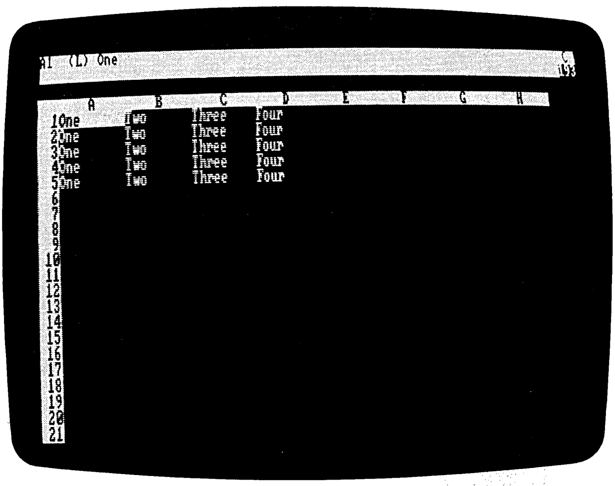
005-044/P

Moving a Column

The Move command deletes the column from the origin, moves all columns to the right of the origin left one to fill the gap, then inserts the column at the destination. Because of this, when moving a column to the right the destination column is specified as the column *to the right of* the intended location; when moving a column to the left the destination is specified exactly.

After the column is moved, the cursor returns to its original location.

Type:	Result:
1. **/CY** **One**→ Two→ Three→ Four **HOME /R.D1** ↵ **A2.A5** ↵	The screen should look like the following photograph:



005-045/P

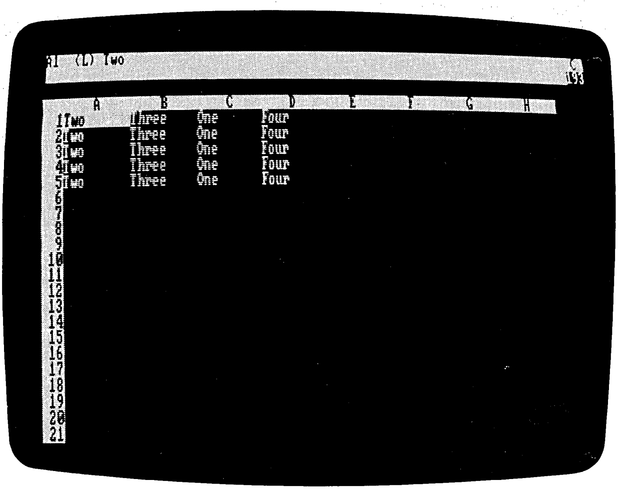
2. **/M**	**Entry Line:** A1 (L) One **Prompt Line:** Move: From . . . To **Edit Line:** A1
3. →	**Edit Line:** A1 . . . B1 The VisiCalc program adds the " . . . " to separate the origin from the destination. The cursor has moved to B1.

4. → → Edit Line: A1 . . . D1

Because a column is being moved to the right, the destination is specified as the column *to the right of* the intended destination. Column D is specified; column A will move to column C, because all columns to the right of column A move to the left after it is deleted from its present position.

5. ↵ Entry Line: A1 (L) Two

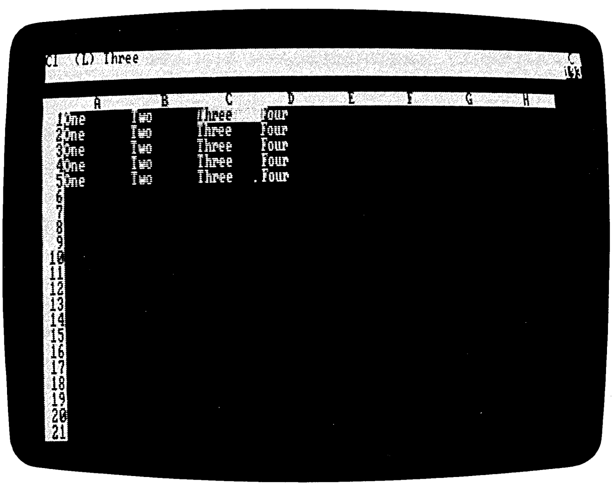
Column A moves to column C, and the cursor returns to A1. The screen should look like the following photograph:



005-046/P

The next example moves column A back to its original position. Because it moves a column to the left, the exact coordinate of the destination row is specified.

Type:	**Result:**
1. **HOME→→**	**Entry Line:** C1 (L) One The cursor moves to C1.
2. **/M**	**Prompt Line:** Move: From . . . To **Edit Line:** C1
3. **.**	**Edit Line:** C1 . . . The origin column is confirmed.
4. **A1**	**Edit Line:** C1 . . . A1
5. ↵	Column C moves back to column A. The screen should look like the following photograph:



005-047/P

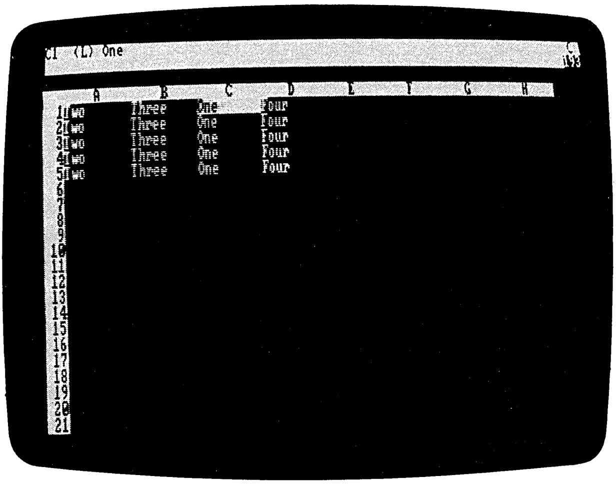
Even though the VisiCalc program assumes the cursor location when the Move command is typed is the coordinate of the row or column to be moved, you can type that coordinate, too. After typing the Move command (/M), use [BKSP] to erase the coordinate on the edit line and either type the origin coordinate or move the cursor to it.

Type:	Result:
1. /M	Entry Line: C1 (L) Three Prompt Line: Move: From . . . To Edit Line: C1
2. [BKSP]	Prompt Line: Move: From . . . To Edit Line: clear The origin coordinate is erased but the Move command is still in effect.
3. A1	Edit Line: A1 Now A1 is the origin.
4. .	Edit Line: A1 . . . The origin is confirmed.

5. **D1** ↵

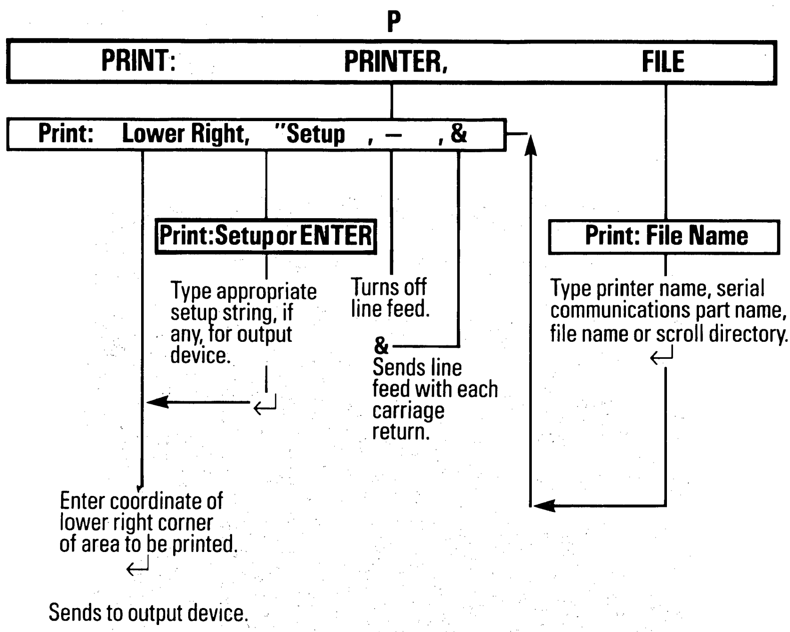
Edit Line: clear

Column A is now column C. The columns that were B and C have moved to A and B. The screen should look like the following photograph:



005-048/P

PRINT COMMAND



The Print command produces an image of the worksheet just as it appears on the screen (not the formulas and formats that produce the worksheet, which are saved by the Storage command). This image can be sent to one of several destinations:

- A printer. (See Appendix D.)
- A serial communications port.
- A diskette file.

The Save option of the Storage command is used to produce a record of the formulas and formats behind the worksheet. An example of its use is included under "Printing Formulas and Formats" in this section.

Printing on the Printer —/PP

The Printer option of the Print command (/PP) sends the image of the worksheet to the last printer specified with the File option of the Print command (/PF); if no printer has been specified, it sends the worksheet to printer LPT1:. The upper-left corner of the rectangle to be printed is the location of the cursor when the command is entered; the program prompts for the lower-right corner.

Note: For a description of how to control the options of the printer, see "Appendix A: Controlling the Printer."

If the screen is split when the command is entered, the windows are ignored; only the rectangle specified by the upper left and lower right coordinates is printed. If titles are fixed and the upper left corner of the rectangle to be printed is in a location fixed as a horizontal or vertical title, the Go To command (>) must be used to move the cursor to that location.

Typing [BREAK] stops the printer.

The following example prints the area of the worksheet bounded by A1 on the upper left and J14 on the lower right:

Type:	Result:
1. **HOME**	The cursor moves to A1. This defines the upper left corner of the rectangle to be printed.
2. **/P**	Prompt Line: Print: File, Printer
3. **P**	Prompt Line: Print: Lower right, "Setup, - , &
	Edit Line: clear
4. **J14 ↵**	The last printer specified in a command starts printing the worksheet. (If no printer has been specified, it is printed on LPT1:.)
5. **[BREAK]**	The printer stops and the status area clears.

Printing the Worksheet in Sections

If the rectangle to be printed is wider than the number of characters the printer can print on one line, the part of the line beyond the printer limit is printed on a separate line. For example, if your printer prints 80 characters per line, the 81st character starts the next row down.

To reproduce a worksheet wider than the printer can print, print it in sections. Calculate how many columns of the worksheet can be printed at a time, then divide the worksheet into the number of rectangles required to print the full width. The separate printed copies can be taped together to reproduce the worksheet.

If the printer uses single sheets rather than continuous- form paper, the worksheet may also have to be printed in sections to accommodate the depth.

Creating a Print File —/PF

The File option of the Print command (/PF) command sends the image of the worksheet to a diskette file, printer, or serial communications port. A print file is a standard text file that contains exactly what is required to print the worksheet, including carriage returns and line feeds. It can be read by other programs, such as a word processor.

The VisiCalc program adds the suffix .PRF to the file name you specify to identify the file as a print file, rather than a worksheet file (which is given the suffix .VC), or a DIF file (which is given the suffix .DIF). Because the file contains only the titles and numbers displayed on the worksheet (not the formulas, formats, etc., used to set up the worksheet), this file cannot be loaded with the Load option of the Storage command.

If you specify the name of a file that exists on the diskette, the VisiCalc program prompts you to make sure you want to replace the file: "File exists: Type Y to replace". If you type Y, the print file of the worksheet replaces the file with the same name on the diskette; if you type anything else, the Print command is canceled.

Printers

Up to three printers can be attached to the computer. They are named LPT1:, LPT2:, and LPT3: (the colon is part of the name). Once a printer is specified in a command, the VisiCalc program assumes it is to be used in all subsequent Print commands unless another one is specified.

Serial Communications Ports

The serial communications ports are named COM1: and COM2: (the colon is part of the name). A baud rate (speed with which the port sends or receives data) can be specified as part of the port name by typing a B followed by a valid rate after the colon. The valid baud rates are 110, 300, 600, 1200, 2400, 4800, and 9600. To specify serial port COM1: with a baud rate of 1200, for example, specify the name as COM1:B1200. If no baud rate is specified, 300 is assumed.

Diskette Files

Specifying a diskette file causes the image of the worksheet to be saved in a diskette file. See "File Names" in this chapter for a description of file names and how to specify them.

The following example shows how to save the area of the worksheet bounded by B10 at the upper left and F20 at the lower right in a print file named BUDGET.PRF (the VisiCalc program adds the suffix .PRF) on drive A:

Type:	Result:
1. **>B10 ↵**	The cursor moves to B10.
2. **/P**	**Prompt Line:** Print: File, Printer The upper left corner of the rectangle to be saved is defined as B10.
3. **F**	**Prompt Line:** Print: Filename
4. **budget ↵**	**Prompt Line:** Print: Lower right, "Setup, -, & The VisiCalc program adds the suffix .PRF to the file name.
5. **F20 ↵**	**Edit Line:** F20 The worksheet is saved as a print file.

The next example prints the same area of the worksheet on a serial printer attached to serial communications port COM1: at a baud rate of 1200:

Type:	Result:
1. >B10 ↵	The cursor moves to B10.
2. /P	**Prompt Line:** Print: File, Printer The upper-left corner of the rectangle to be saved is defined as B10.
3. F	**Prompt Line:** Print: Filename
4. com1:B1200 ↵	**Prompt Line:** Print: Lower right, "Setup, – , & **Edit Line:** com1:1200
5. F20 ↵	**Prompt Line:** Print: Lower right, "Setup, – , & **Edit Line:** F20 The worksheet is printed on the serial printer.

Printing Formulas and Formats

The Printer option of the Print command (/PP) prints an image of the worksheet as it is displayed on the screen. But what about the formulas and formats behind the data?

The Save option of the Storage command (/SS) saves a record of the formulas and formats as they were written on the worksheet. Like the other options of the Storage command, this record can be sent to a diskette file, a printer, or a serial port. The information is so complete the worksheet can be reproduced from it.

The printer, serial communications port, or diskette file is specified just as in the File option of the Print command (/PF).

The following example prints the formulas and formats of each entry position on the worksheet on printer LPT1: (the cursor can be positioned anywhere on the worksheet):

Type:	Result:
1. **/S**	**Prompt Line:** Storage: L S D Q #
2. **S**	**Prompt Line:** Storage: File for Saving **Edit Line:** clear
3. **LPT1:** ↵	The worksheet is printed entry-position-by-entry- position, from the lower-right entry to the upper-left entry, followed by format information. Each entry position takes one line, so quite a bit of paper is required to print a large worksheet this way.

REPEATING LABEL COMMAND

—

Label: Repeating

Type any characters.

↵

Character sequence repeated across column width.

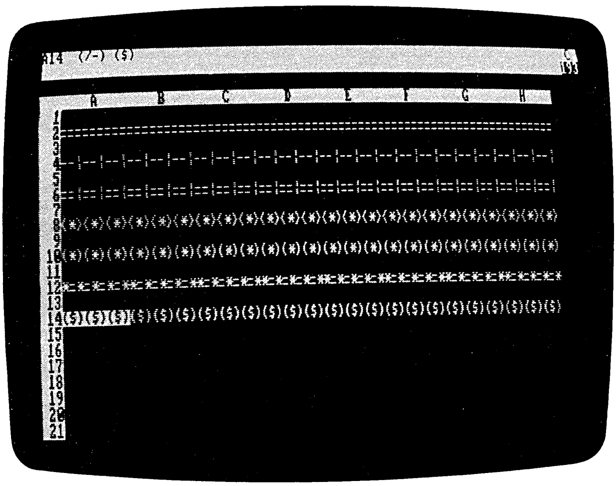
The Repeating Label command fills an entry position with characters. It is commonly used to draw a line across the worksheet to separate titles from columns of numbers, or otherwise distinguish between areas on the worksheet. The entry location is filled with the character regardless of the column width.

A column filled with characters can be copied across a row (with the Replicate command) to draw a line of any length across the worksheet.

Type:	Result:
1. /CY	The screen clears.
2. Jan → Feb → Mar **HOME**↓↓ 1 → 2 → 3 ↵	The numbers are separated from the titles by a blank line.
3. **HOME**↓ /—	Entry Line: A2 Prompt Line: Label: Repeating The program is waiting for you to enter the character that is to fill the entry location.
4. — ↵	Entry Line: A2 (/-) - A2 fills with hyphens.
5. /GC20 ↵	A2 is still filled with hyphens, even though the column is wider.
6. /GC9 ↵	The columns return to the normal width.
7. /R ↵ → . → → ↵	The line of hyphens extends across the first three columns.

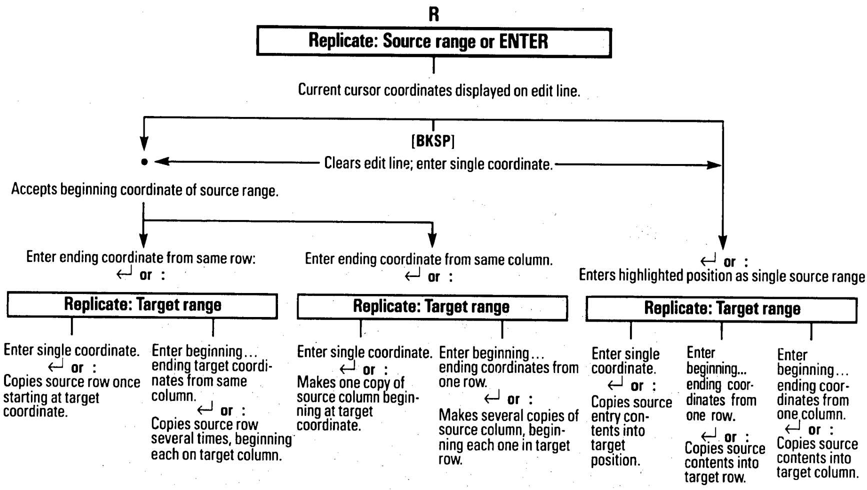
Repeating labels can be drawn with any character or combination of characters. The hyphen can be used to indicate a subtotal, for example, and the equal sign to indicate a total. Decorative borders can be drawn with combinations like <*> and *= .

The following photograph shows different repeating labels across the screen.



005-049/P

REPLICATE COMMAND



The Replicate command copies an entry position to another location on the worksheet. It can copy a label, value, a blank entry, or just the local format. It can create one copy of a single entry, multiple copies of a single entry, one copy of a range of entries, or multiple copies of a range of entries.

The Replicate command cannot copy a row into a column, a column into a row, or a combination of rows and columns at the same time. Complex copying operations can be performed, however, with several uses of the command.

The Replicate command asks for the following information:

- The source range (location or locations to be copied).
- The target range (location or locations where the source range is to be copied).
- If the source range includes formulas that reference other locations, it asks whether each such reference should be copied with no change at each new location or relative to each new location.

A range can be one location (for example, B5) or a series of contiguous locations across a row or down a column (for example, F9-F12 or B2-M2). A range is defined by the starting coordinate, a period, and the ending coordinate (for example, B2 . . . M2). If the range is a single location, the starting and ending coordinates are the same (for example, B2 . . . B2). The coordinates can be specified by typing them or pointing to them with the cursor.

If the beginning and ending source range coordinates are identical (for example, B5 . . . B5), only the specified location is copied to the target range. If the letters are the same but the numbers are different (for example, B5 . . . B15), a portion of a column is copied. If the numbers are the same but the letters are different (for example, B9 . . . M9), a portion of a row is copied.

To make just one copy of a source range, the target range should consist of a single coordinate (for example, F9). To make more than one copy, the target range should consist of a range of locations (for example, F9 . . . F15).

Copying a Single Entry Position

To copy one entry position, both the source and target coordinates specify a single location.

	Type:	**Result:**
1.	**/CY**	The worksheet clears and the cursor moves to A1.
2.	**100 ↵**	**Entry Line:** A1 (V) 100
3.	**/R**	**Prompt Line:** Replicate: Source range or ENTER **Edit Line:** A1 A1 is assumed to be the source range because the cursor was at that location when the Replicate command was typed.

4. ↵	**Prompt Line:** Replicate: Target range **Edit Line:** A1 . . . A1 The source range is the single location A1.
5. **D1**	**Prompt Line:** Replicate: Target range **Edit Line:** A1 . . . A1: D1 D1 is the start of the target range.
6. ↵	The value at A1 is copied to D1.

Creating a Column of Entries from One Entry

To create a column by making several copies of one entry, the source range is specified as the entry to be copied and the target range is specified as the starting and ending coordinate of the portion of the column to receive the copy.

Type:	**Result:**
1. Type steps 1-5 from the preceding example.	
2. **.D10** ↵	The value at A1 is copied to entry positions D1 through D10.

Formats can also be replicated (see "Format Command" in this chapter for a description of formats). A format must be replicated *before* anything else is written at the target locations, because the Replicate command copies the entry as well as the format; if you replicate a format into a target location that contains a label or value, the label or value is replaced by the label or value of the source range.

To display a column of sales figures rounded to two decimal places, place the cursor on a blank entry position and type **/F** to specify the Dollars-and-cents format. Then replicate that entry position using the procedure in the preceding example. If there are no values in the target range, there is no effect on the display. When you enter a value into any location in the target range, however, it is displayed with two decimal places.

Making One Copy of a Column

To make one copy of a column, the source range is specified as the top and bottom locations (for example, A1 . . . A32). The target range is specified as the coordinate of the top location of the new column.

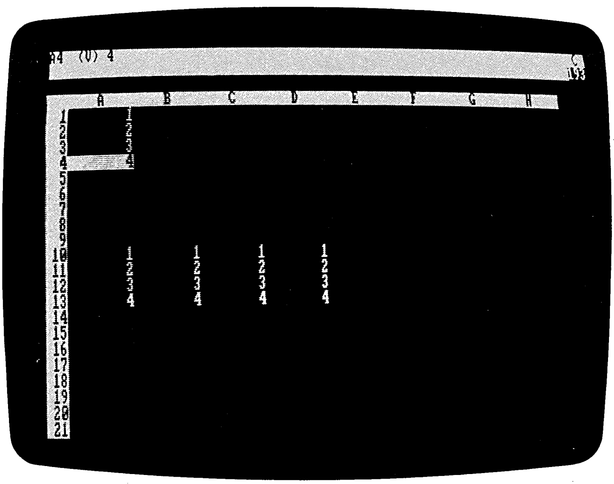
The cursor need not be located at the beginning of the source range. You can erase the starting coordinate from the edit line with [BKSP] and type both the source and target ranges.

	Type:	Result:
1.	/CY	The worksheet clears and the cursor moves to A1.
2.	1 ↓ 2 ↓ 3 ↓ 4 ↵	A column of numbers in column A.
3.	/R	**Prompt Line:** Replicate: Source range or ENTER **Edit Line:** A4
4.	[BKSP]	Erases the first source coordinate.
5.	A1	**Edit Line:** A1
6.	.	**Edit Line:** A1 . . .
7.	A4 ↵	**Prompt Line:** Replicate: Target range **Edit Line:** A1 . . . A4:
8.	C4	**Edit Line:** A1 . . . A4: C4
9.	↵	Column A (A1-A4) is copied to column C (C4-C7).

Making Several Copies of a Column

To make several copies of a column, the source range is specified as the top and bottom coordinates of the source range and the target range is specified as a range of locations across a row. The column is copied starting at each location in the target range.

Type:	Result:
1. Steps 1-7 from the preceding example.	
2. **A10.D10** ↵	Column A (A1-A4) is copied to columns A through D (A10-A13 through D10-D13). The screen should look like the following photograph:

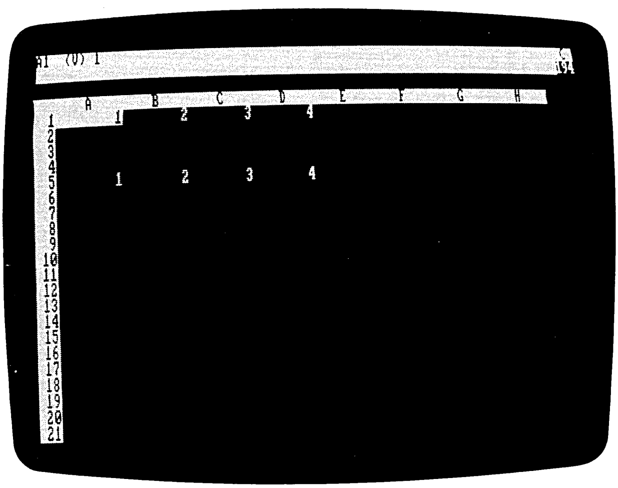


005-050/P

Making One Copy of a Row

To make one copy of a row, the source range is specified as the beginning and ending coordinates of the row to be copied (for example, A1 . . . C1). The target range is specified as the beginning coordinate in the row to receive the copy.

	Type:	Result:
1.	/CY	The worksheet clears and the cursor moves to A1.
2.	1 → 2 → 3 → 4 **HOME**	A row of numbers in row 1.
3.	/R.D1 ↵	**Prompt Line:** Replicate: Target range **Edit Line:** A1 . . . D1:
4.	A5 ↵	Row 1 (A1-D1) is copied to row 5 (A5-D5). The screen should look like the following photograph:

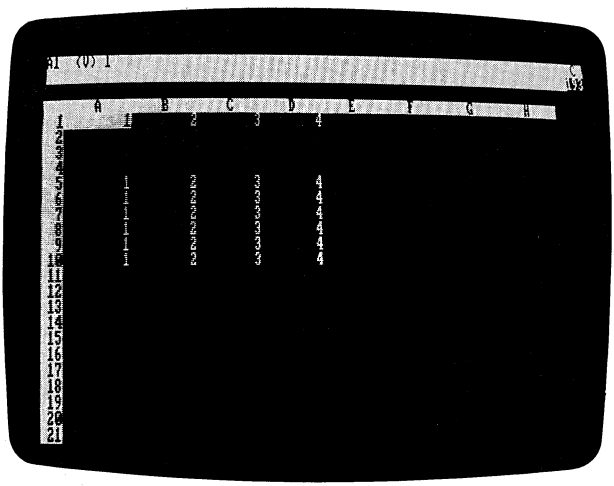


005-051/P

Making Several Copies of a Row

To make several copies of a row, the source range is specified as the left and right coordinates of the source range and the target range is specified as a range of locations down a column. The row is copied starting at each location in the target range.

Type:	Result:
1. Steps 1-3 of the preceding example.	
2. **A5.A10** ↵	Row 1 (A1-D1) is copied to rows 5-10 (A5-A10 through D5-D10). The screen should look like the following photograph:



005-052/P

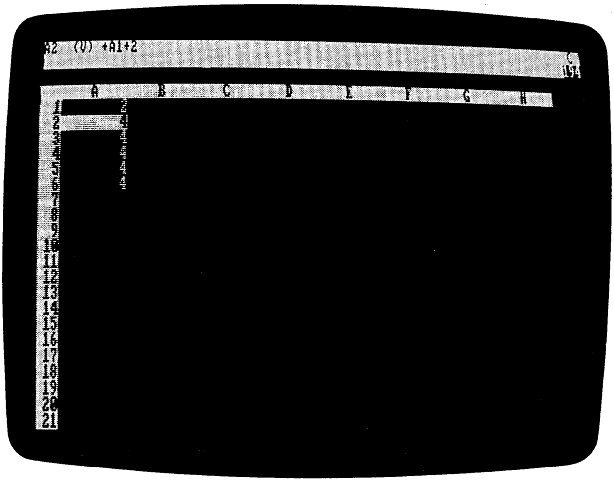
Replicating Formulas

When the Replicate command copies a formula that contains one or more references to another location (for example, .6*B5-A21), it either copies the references with No change or Relative to the new location. It prompts for an N or R for each reference in the formula being replicated.

Type:	Result:
1. **/CY**	The worksheet clears and the cursor moves to A1.
2. **2 ↓ +A1+2 ↵ /R ↵**	**Entry Line:** A2 (V) +A1+2 **Prompt Line:** Replicate: Target range **Edit Line:** A2 . . . A2: The ↵ after "/R" tells the VisiCalc program that the source range consists of the single location A2.
3. **A3.A6 ↵**	**Prompt Line:** Replicate: N= No change, R= Relative **Edit Line:** A2: A3 . . . A6: +A1 The edit cue over +A1 indicates that the VisiCalc program is asking whether to copy the reference to location +A1 unchanged (the same in each position in the target range) or whether it is to be copied *relative* to each new position.

4. **N**

The reference to location A1 is copied with no change. The screen should look like the following photograph:



005-053/P

Move the cursor down the column. Each copy of the formula contains the reference to A2, just as the source location does. This is the result of choosing No Change instead of Relative.

The next example sets up the same structure for comparison in column B.

Type:	**Result:**
1. **HOME→**	The cursor moves to B1.
2. **2 ↓ +B1+2 ↵ /R ↵**	**Entry Line:** B2 (V) +B1+2 **Edit Line:** B2 . . . B2 **Prompt Line:** Replicate: Target range The ↵ after "/R" tells the VisiCalc program that the source range consists of the single location A2.

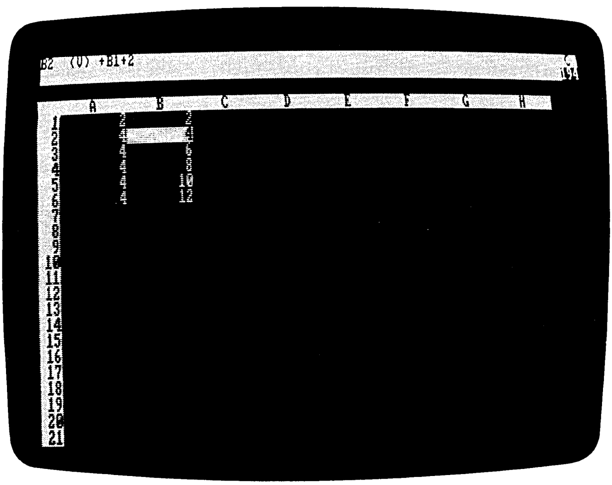
3. **B3.B6** ↵

Prompt Line: Replicate: N= No change, R= Relative
Edit Line: B2: B3 . . . B6: +B1

The edit cue highlights +B1 just as it did +A1 in the previous example. Again, it wants to know whether to copy the reference to location B1 unchanged (N) or relative (R) to the target locations.

4. **R**

The reference to location B1 is copied relative to each target location; that is, it becomes B2 in the formula at B3, B3 in the formula at B4, etc. The screen should look like the following photograph:



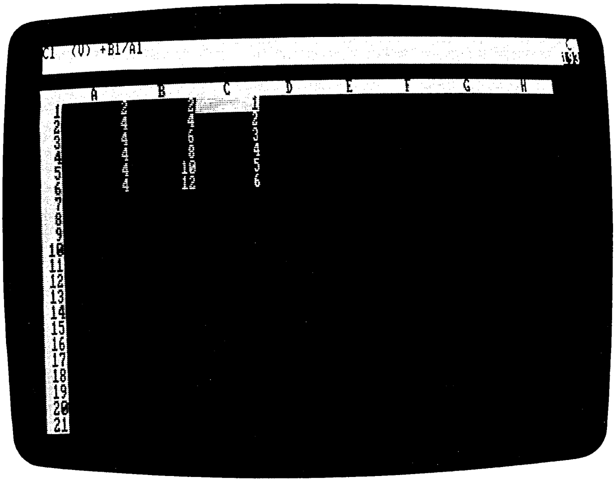
005-054/P

The formulas behind columns A and B are quite different. Move the cursor down column B to see how the formula changes on each line. Now move down column A; each formula copied from A2 is the same.

If a formula contains more than one reference to another location, the VisiCalc program asks how to copy each reference.

Type:	Result:
1. **HOME→→**	The cursor moves to C1.
2. **+B1/A1 ↵**	**Entry Line:** C1 (V) +B1/A1 The formula references two locations (B1 and A1).
3. **/R ↵**	**Edit Line:** C1 . . . C1: **Prompt Line:** Replicate: Target range
4. **C2.C6 ↵**	**Edit Line:** C1: C2 . . . C6 +B1 **Prompt Line:** Replicate: N= No change, R= Relative The edit cue highlights the reference to B1, asking how to copy that reference.
5. **R**	**Edit Line:** C1: C2 . . . C6: +B1/A1 **Prompt Line:** Replicate: No change, R= Relative The edit cue highlights the reference to A1, asking how to copy that reference.

6. **N** The screen should look like the following photograph:



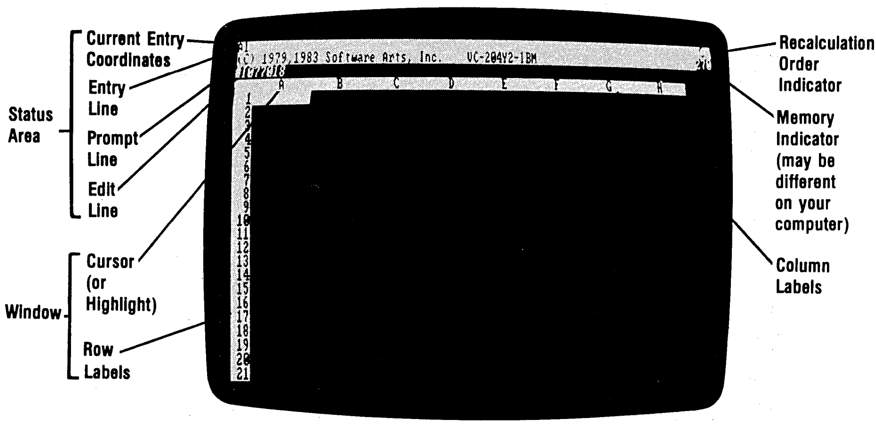
005-059/P

Move the cursor down column C to see how the dividend changes at each location but the divisor remains the same. To experiment further, replicate this formula with both dividend and divisor relative and with both unchanged. See "Replicating a Formula" and the headings that follow in Lesson Two for more examples of the Replicate command.

It is possible to replicate a formula into a position so that valid relative value references cannot be assigned. If you inserted a new column at A and replicated the formula at C1 to A1, the relative position of the value references A1 and B1 would be located off the worksheet to the left. The VisiCalc program assigns the value ERROR to locations that contain such a reference.

Also be careful not to create forward or circular references or incorrect calculation order when you replicate formulas. See "Order of Recalculation" and "Forward and Circular References" in Lesson Four and "Values" in this chapter for a description of these conditions and ways to circumvent them.

THE SCREEN



036-002/P

The screen includes the *status area* (the top three lines), the column and row borders, and the entry positions that make up the window through which you view the electronic worksheet.

Status Area

The three lines of the status area are (from the top down) the *entry line*, the *prompt line*, and the *edit line*. The entry line and the prompt line share the wide white bar at the top; the edit line is the dark line just above the column border.

Entry Line

The *entry line* is the top line of the status area. It can display as many as five items of information about an entry position:

1. The coordinate of the cursor location. This is always displayed.
2. The local format, if one is assigned to the location.
3. The type of entry ("L" for Label or "V" for Value), if the location contains an entry.
4. The entry just as it was written, if the location contains an entry.
5. A character that indicates the order of recalculation ("C" for column-order, "R" for row-order). An exclamation point is displayed next to it when the worksheet is being recalculated.

Prompt Line

The *prompt line* is the middle line of the status area. It displays a prompt at the left side that describes the options at any point in a command sequence, and a two-digit number at the right side that tells how much memory is available (in units of 1024 characters).

Edit Line

The *edit line* is the bottom line of the status area. It displays each character as it is typed, or the coordinate as the cursor is moved to write coordinates in a formula. The edit cue (a small white block) appears after the last character entered.

The characters on the edit line can be edited by erasing them with the [BKSP] key and retyping to correct any errors. The VisiCalc program also, on occasion, uses this line to display information it wants you to confirm or clarify before it carries out a command.

The Window and Worksheet

The VisiCalc *window* is that portion of the screen beneath the column border and to the right of the row border. The window can be scrolled with the arrow keys to show any portion of the worksheet.

The column border extends across the top of the worksheet and labels 63 columns, A through BK. The row border extends down the left side and labels rows 1 through 254.

Entry Positions

The intersection of each column and row defines an *entry position*. Each is identified by a column letter followed by a row number (A5, for example). This identifier is the *coordinate* of the entry position.

The Cursor

Each time the VisiCalc program is loaded or cleared with the Clear command, a column-wide rectangle covers entry position A1. This rectangle is the *cursor*.

The cursor marks the position on the worksheet where entries can be written or commands are to be carried out. The coordinate of the cursor location is displayed at the left side of the entry line. During the course of some commands, the cursor can be moved with the arrow keys.

Moving the Cursor

The arrow keys (← → ↑ ↓) move the cursor in the indicated direction to any position on the worksheet. Typing an arrow key moves the cursor one entry position at a time.

When the cursor is at the edge of the screen, further cursor movement in the same direction causes the window to move over the worksheet, or *scroll*, until it reaches the edge of the worksheet. The Go To command moves the cursor directly to a location without scrolling.

The HOME Key

The HOME key moves the cursor to location A1. Typing HOME is the same as typing >A1 ↵.

Repeating Keys

To cause any key to repeat, hold it down for about half a second; it repeats until you release it. Use of the repeating capability of the arrow keys moves the cursor more quickly without extra keystrokes. The window scrolls with the cursor.

Pointing with the Cursor

If a command requires the coordinate of an entry position, the coordinate can be typed or entered in the formula by pointing to it with the cursor. As the cursor moves, the coordinate on the edit line changes. If a command does not require the coordinate of an entry position, typing one of the arrow keys may end the command and move the cursor to the next entry position.

Typeahead

If you type faster than the VisiCalc program can accept the characters, it stores the characters and catches up as soon as it can. If you type long enough and fast enough, it cannot save any more characters and beeps each time you try to type another character until it has room to save more.

Correcting Mistakes

[BREAK] cancels a command or entry. The Blank command erases an entry from the worksheet.

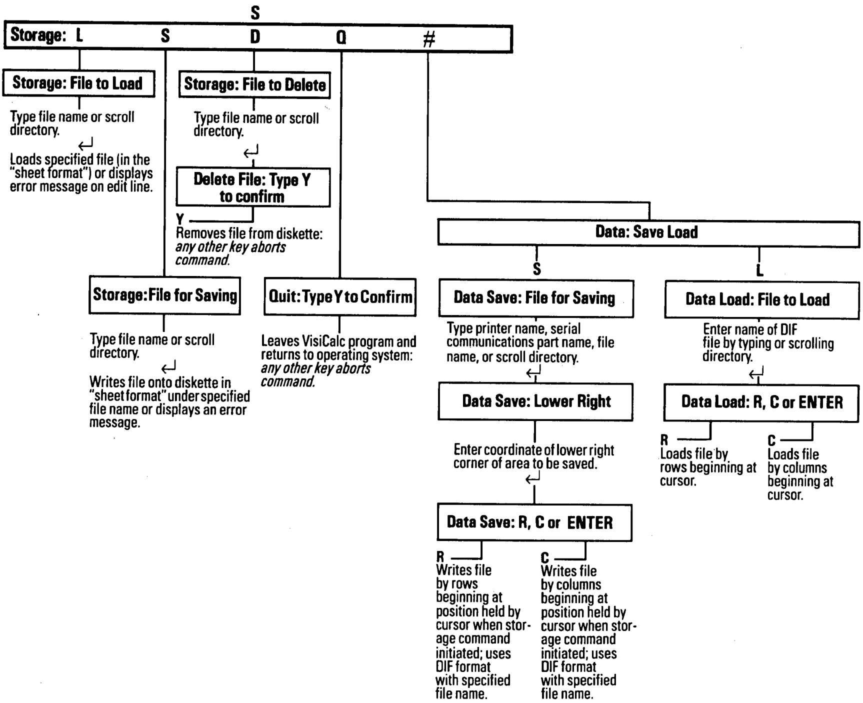
While an entry is being typed, characters on the edit line can be erased by typing [BKSP]. Each time [BKSP] is typed, the edit cue moves left and erases the last character typed.

An existing entry can be replaced by moving the cursor to the entry position and typing a new entry. The new entry replaces the old one when it is ended with ↵ or one of the arrow keys. If the entry is canceled with [BREAK], the old entry is unchanged. The edit line shows the new entry as it is typed; the entry line shows the old entry until the new one is written.

The Edit command changes the contents of the edit line; it can be used to correct an entry while it is being typed or change an entry already written on the worksheet. The edit cue can be moved back and forth over the entry: → and ← move it one character right or left; ↑ moves it to the beginning of the edit line and ↓ moves it to the end. To erase the character to the left of the edit cue, type [BKSP]; to insert a character to the left of the edit cue, simply type the letter. To end the Edit command and write the entry on the worksheet, type ↵.

To use the Edit command to change an entry being typed (before it is written on the worksheet), type CTRL-E; the edit cue remains at the end of the edit line. To change an entry already written on the worksheet, move the cursor to the location and type /E. The entry is copied to the edit line and the edit cue is positioned at the beginning.

STORAGE COMMAND



The Storage command includes several options:

L Loads a previously saved worksheet.
S Save the VisiCalc worksheet on diskette.
D Deletes a file from the storage diskette.
Q Quits the VisiCalc program.
Loads or saves a file in DIF format.

When saving the worksheet, the Storage command produces a record of the formulas, formats, and other instructions that produce the worksheet (not the image of the worksheet as it appears on the screen, which is saved by the Print command). This record can be sent to one of several locations:

- A printer.
- A serial port.
- A diskette file.

Load Option —/SL

The Load option loads a worksheet that was saved with the Save option. When the file is loaded, the worksheet is displayed exactly as it was when it was saved unless it contains forward or circular references; those locations display "ERROR". (If "ERROR" appears while the file is loading, ignore it.)

If one worksheet is on the screen when another is loaded, the first is not cleared; the second worksheet is loaded over the first. If a location is filled on both worksheets, the entry on the second worksheet replaces the entry on the first. If a location on the second worksheet is blank, it does not erase the entry of the corresponding location on the first. This makes it possible for one worksheet to be combined with another. If the worksheets are not to be combined, the screen should be cleared (with the Clear command) before the second worksheet is loaded.

The following example shows how to load a worksheet named "budget" from drive B:

	Type:	Result:
1.	**/S**	**Prompt Line:** Storage: L S D Q #
2.	**L**	**Prompt Line:** Storage: File to load
3.	**b:budget ↵**	During the loading process, the prompt line reads "Storage: Loading" followed by a flashing asterisk. When the worksheet is loaded, it appears on the screen just as it did when it was saved (including any split windows, fixed titles, and global format).

Instead of typing in the file name, you can display the name of each file on the diskette and select the one you want by typing ↵. Type → instead of "budget" at step 3 in the preceding example. The name of the first file on the diskette is displayed on the edit line. Each time you type →, the next file name is displayed until the name of each file on the diskette has been displayed. (If you type → again, the Storage command is canceled.) When the file name you want is on the edit line, type ↵ and the VisiCalc program loads the worksheet just as if you had typed the file name.

These file names can be edited, just as if you had typed them. To edit the file name on the edit line, type CTRL-E and edit the line as described in "Edit Command" in this chapter. When the name is correct, type ↵. This feature is useful when, for example, you have forgotten a file name. As you display the file names, one might remind you of the name you forgot. The name "projfeb", for example, might remind you that the file you want is called "projaug". To change the file name from "projfeb" to "projaug", type **[BKSP]** to back up the edit cue until you've deleted "feb", then type **aug** ↵. The worksheet saved in the file named "projaug" is loaded.

If the VisiCalc program cannot find the file you tell it to load, the prompt line reads "Error: File not found". This could mean you misspelled the name, the wrong diskette is in the drive, or the wrong drive is being searched. Either correct the name, put in the proper diskette, or add the proper drive letter (followed by a colon) to the beginning of the file name.

Save Option —/SS

The Save option of the Storage command (/SS) saves the record of the worksheet. The record can be sent to a file, a printer, or a serial port.

See "Print Command" in this chapter for a description of how to specify a printer or serial port. See "File Names" for a description of how to specify a file name.

Note: A diskette must be formatted before anything can be written on it. See the computer operations manual for instructions on how to format a diskette.

If you specify the name of a file that is already on the diskette, the VisiCalc program prompts you to make sure you want to replace the existing file: "Storage: File exists. Y to replace". If you type Y, the saved file replaces the file with the same name; if you type anything else, the Storage command is canceled.

The ability to display the names of the files on a diskette makes it easy to name files as new versions of existing files. For example, you could modify a file called "sales5" and, instead of replacing it with the new version, display the file name, type [BKSP] to erase the 5, then type **6**. The result: a new version named "sales6"; "sales5" is still on the diskette.

Although a diskette has enough room to hold many worksheets, it still has a finite amount of space. If the disk fills while a worksheet is being saved, "Error: Disk full" appears on the edit line; the VisiCalc program saved only part of the worksheet. If this happens, delete the incomplete file from the disk (see "Delete Option" later in this topic) and save the worksheet on a diskette with enough room.

With larger memory configurations you might create a worksheet that is too large to store on diskette. There is no way to accurately predict when a worksheet has become too large to save. Two worksheets with the same memory indicators can take different amounts of diskette storage space depending on the number of formulas used. Worksheets with many formulas will take less diskette space than worksheets with many values and labels.

You can determine the size of each worksheet saved on the diskette by quitting the VisiCalc program and using the DOS DIR Command. See the *IBM Technical Reference* or your dealer for information on the storage capacity of the media you are using.

Save very large worksheets frequently to check that they are not becoming too large to save. If a worksheet is too large, a "Disk full" message appears when you try to save. If you are overwriting a previously saved worksheet, that worksheet will have been deleted. It is a good practice to save to alternate diskettes to prevent losing your work. If a "Disk full" message appears, delete a small part of the worksheet and save again. Continue deleting until the worksheet can be saved.

The following example saves the worksheet in a file named "budget":

	Type:	Result:
1.	**/S**	**Prompt Line:** Storage: L S D Q #
2.	**S**	**Prompt Line:** Storage: File for saving
3.	**budget ↵**	The status area clears and the drive activates, saving the worksheet with the file name typed.

Delete Option —/SD

The Delete option of the Storage command (/SD) *irretrievably erases* a file from the diskette.

The following example deletes a file named "budget" from the diskette in drive A:

	Type:	Result:
1.	**/SD**	**Prompt Line:** Storage: File to Delete

2. **a:budget** or **a:** →	**Edit Line:** a:budget
	Edit Line: The name of the first file on the diskette in drive A.
	Each subsequent → causes the next file name to be displayed. Continue until the name of the file you want is displayed (if you type → after the last file name is displayed, the Storage command is canceled).
3. ↵	**Prompt Line:** Delete file: Type Y to confirm.
4. **Y**	The file is deleted from the diskette. If you type anything other than "Y", the Storage command is canceled.

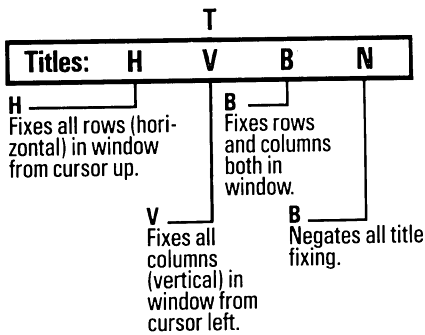
Quit Option —/SQ

The Quit option of the Storage command (/SQ) returns control to the Disk Operating System. It first prompts "Quit: Type Y to confirm". In response, type "Y". If you don't want to start another program, simply remove the diskettes and turn the computer off.

DIF™ Options

The two remaining options–S#S and S#L–are used to save and load worksheets stored in the DIF format, which makes it possible for the VisiCalc program to share files with other programs. For a complete description of the DIF format and these options, see Appendix B.

TITLES COMMAND



The Titles command fixes rows and columns in place so that they remain in view as the window scrolls over the worksheet. The position of the cursor at the time the Titles command is entered determines which column(s) and/or row(s) are fixed.

In response to /T (the Titles command), the prompt line displays "Titles: H V B N" The characters following "Titles:" are options:

- H Horizontal titles.
- V Vertical titles.
- B Both horizontal and vertical titles.
- N No titles (cancels any titles that are fixed).

The Titles command causes no apparent change in the worksheet. The effect of title fixing becomes apparent, however, when the window scrolls; the locations that are not fixed move across the screen,but the titles remain unmoving.

The arrow keys won't move the cursor to a location that is in a row or column fixed as a title. The cursor beeps an error when it bumps into a fixed title just like it does when it bumps into one of the borders of the worksheet. The Go To command (>), however, can be used to move to a location in a row or column fixed as a title.

If vertical titles are fixed and the column width is increased so that only one column can be displayed, the VisiCalc program cancels the vertical title settings. If the column width is later reduced, vertical titles must be fixed again.

Horizontal Option —/TH

The Horizontal option of the Titles command fixes all rows at and above the row that contains the cursor. To fix rows 1 and 2 as titles, place the cursor anywhere in line 2. To fix only row 1, place the cursor in row 1.

Vertical Option —/TV

The Vertical option of the Titles command fixes all columns at and to the left of the column that contains the cursor. To fix columns A and B, place the cursor anywhere in column B. To fix only Column A, place the cursor in column A.

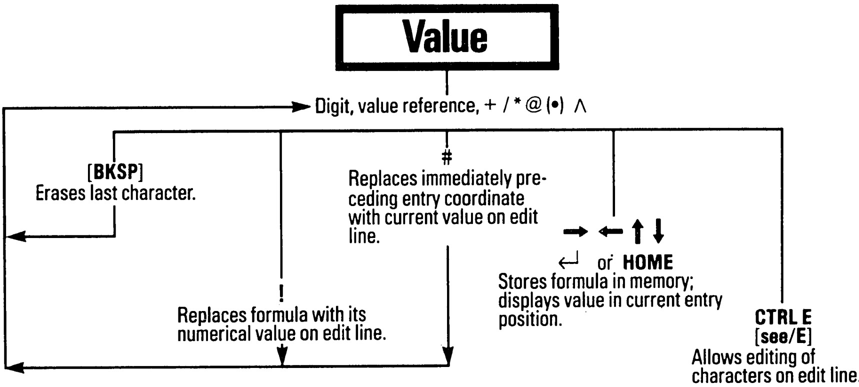
Both Option —/TB

The Both option of the Titles command fixes Both rows and columns at the same time. The rows are fixed at and above the row that contains the cursor and the columns are fixed at and to the left of the column that contains the cursor. To fix row 1 and column A, place the cursor at location A1. To fix rows 1 and 2 and columns A and B, place the cursor at B2.

None Option —/TN

The None option of the Titles command cancels any rows or columns fixed by the Titles command. The cursor can be anywhere when the command is entered.

VALUES



A value is an entry on the worksheet used in calculations. The VisiCalc program looks at the first character of what is typed to determine whether the entry is a value or a label; if the entry begins with any of the following characters, is it assumed to be a value:

0-9	The digits zero through nine.
+	Plus sign.
—	Minus sign.
@	The beginning of a function name.
(	Left parentheses (start an inner portion of a formula).
#	Value at cursor location.

A value is a *formula* that consists of one or more of the following:

- A number: 1, – 11, 25.5, 3.4E4.
- A reference to the coordinate of another location: A5, J23.
- A range of such references: B2 . . . B12, F5 . . . F10.
- An arithmetic operator: +, – , = , *, /, >.
- A function: @SUM(M2 . . . M12), @PI.

As soon as the entry is recognized as a value, the word VALUE appears on the prompt line and the typed character appears on the edit line. Each subsequent character appears on the edit line as it is typed until the value is written on the worksheet (by typing ↵, an arrow key, or HOME) or canceled (by typing [BREAK]).

If you make a typing error while typing a value, type [BKSP] to erase the last character typed and continue typing. The Edit command can also be used to correct a value after it is written on the worksheet.

When the formula is written on the worksheet, its calculated value is displayed. The formula itself appears on the entry line of the status area.

References to Another Location

The value of another worksheet location can be used in a formula by including the coordinates of the other location. The value changes whenever the entry at the other location is changed.

When the first element of a formula is a reference to another location, the formula must begin with a character that starts a value such as + or 0+. If it begins with a letter, the VisiCalc program assumes the entry is a label (see "Labels" in this chapter).

If the last character typed is an arithmetic operator, typing an arrow key does not end the value and write it on the worksheet; it changes the coordinate on the edit line so that a reference to another location can be included in a formula by pointing the cursor to the location to be referenced. To write the formula on the worksheet, ↵ must be typed.

For example, if 1+ is on the edit line and the cursor is moved to A5, the formula on the edit line is 1+A5. Typing a cursor-moving key at this point does not write the formula on the worksheet, it changes A5 to the coordinate of the new cursor position. ↵ must be typed to end the formula.

If the formula is to be 1+A5–B5, typing – after moving the cursor to A5 returns the cursor to the formula entry position. Now the cursor can be moved to B5, which makes the formula on the edit line 1+A5–B5; again, ↵ ends the formula.

The VisiCalc program does not allow an illegal formula to be entered (such as one that ends with an arithmetic operator). It beeps an error and waits for the formula to be completed. To continue after an illegal entry of this type, the formula must be edited or canceled. A valid formula that causes an illegal calculation (such as dividing by zero) causes the special value ERROR to be displayed where the illegal calculation occurs and at all other locations that reference the calculation.

The complexity of a formula–the number of references to other locations, arithmetic operators, parentheses, and functions and their arguments–and the amount of memory available together determine the maximum length of a formula. If a formula becomes too complex, the VisiCalc program beeps an error and stops displaying additional characters on the edit line. Only the formula up to that point can be written on the worksheet.

Precision of VisiCalc Values

The VisiCalc program stores all values in base 10 with either 11 or 12 significant digits. The largest number the VisiCalc program can accurately calculate is .999999999999E62. The smallest is 9.999999999E−66. When a number is displayed in the General format (see "Format Command" in this chapter), the VisiCalc program shifts between conventional and scientific notation as required to display the value with the greatest precision.

In scientific notation, the number 123456789123 is displayed as 1.235E11 with a column width of 9. The E11 means "times 10 to the 11th power." Except in Dollars-and-cents format, which displays all numbers with two decimal places, non-significant zeros are dropped.

If the columns are too narrow to display a number even in scientific notation, the VisiCalc program fills its entry position with > signs. The display format (Integer, Dollars-and-cents, etc.) and column width do not change the number stored in memory, only the way it is displayed.

Order of Precedence of Calculation

The VisiCalc program performs calculations in the order it encounters each operator from left to right. No operator takes precedence over another. Portions of a formula in parentheses are calculated first. If there are parentheses within parentheses, the VisiCalc program calculates the innermost first. For example, 5+6/2*4 evaluates to 22, but 5+((6/2)*4) evaluates to 17.

Calculating Values on the Edit Line

Typing an exclamation point (!) at the end of a formula causes the calculated value to replace the formula on the edit line. Typing a cros-shatch (#) immediately after a reference to another location causes the value of that location to replace the reference to it on the edit line.

The following example shows both these features:

Type:	Result:
1. /CY	The worksheet clears and the cursor moves to A1.
2. 1 → 2 → 3 HOME	Three values in row 1.
3. ↓ + ↑ + ↑ → + ↑ → →	Entry Line: A2 Edit Line: +A1+B1+C1
4. #	Edit Line: +A1+B1+3 The reference to C1 is replaced by the value of C1. If ↵ if typed now, the formula on the edit line is written at A2.
5. !	Edit Line: 6 The entire formula on the edit line is replaced by the calculated value of the formula. If ↵ is typed now, the value 6 is written at A2.
6. [BREAK]	Entry Line: clear Edit Line: clear A2: clear

The entry is canceled.

This feature makes it possible to use the VisiCalc program as a calculator. The edit line becomes the calculator display; any valid formula can be typed and evaluated using the !, then cleared with [BREAK].

The # can be used to copy the value at a location (not the formula, just the calculated value) to another location. With the cursor at the destination location, type +, move the cursor to the location whose value is to be copied (or type its coordinate) and type # ↵. The value of the origin location is calculated and written at the destination. If the value of the origin location is later changed, the value at the destination location is unchanged because the reference to the origin location was not copied, just the value.

Recalculation Order

A formula can contain as many references to other locations as the complexity of the formula allows. Unless manual recalculation is specified, the worksheet is recalculated each time an entry is written on the worksheet. Recalculation always starts at location A1.

When first loaded, the VisiCalc program is set to calculate column-by-column. It calculates the value at A1, then A2, then A3 to the end of column A. Then it recalculates B1, B2, B3 to the end of column B; then C1, C2, C3 and so on. The C in the upper-right corner of the screen indicates that the order of recalculation is by Column.

The Global command can be used to change the order of calculation from column-by-column to row-by-row. When recalculation by row is in effect, an R is displayed in the upper-right corner of the screen.

Forward and Circular References

The worksheet should be arranged so that all formulas that contain references to other locations are located *after* the referenced locations are calculated. When the VisiCalc program is set to recalculate by columns, all references to other locations should appear either in a column to the right of the location that is referenced, or below the referenced location in the same column. When it is set to recalculate by rows, all references should appear in a row below the referenced location or to the right of the referenced location in the same row.

If the worksheet is not arranged in this way, the formula containing a reference to another location is recalculated *before* the new value is calculated for the referenced location. After it is recalculated, the worksheet displays the value of the formula as calculated using the *old* value of the referenced location. The referenced location itself, however, displays its *new* value.

This problem, called a *forward reference*, is often difficult to diagnose and might cause you to suspect that the VisiCalc program has made an error. If you suspect the worksheet contains a forward reference, type !. This forces another recalculation.

Watch the location whose value is incorrect. If its value changes, look for forward references; the worksheet probably must be redesigned to eliminate them. (It is possible to leave the forward reference and type ! to force an extra recalculation, but this can be inconvenient, especially if there are forward references to other forward references, a situation that can require several forced recalculations to produce a correct result.)

Forward references sometimes occur in some areas of accounting when column totals are placed at the *top* of the page.

Note: If a worksheet that includes forward references is saved and later loaded from diskette, the locations that contain forward references display "ERROR". A recalculation must be forced (by typing !) to eliminate the ERROR values and display correct results. The forward references should be eliminated by reorganizing the worksheet.

A *circular reference* is a formula that cites itself, such as 1+A1 in location A1. Each time the worksheet is recalculated, the value of this formula changes even if no other change is made to the worksheet. A circular reference prevents accurate results and can be particularly difficult to diagnose.

VERSION COMMAND

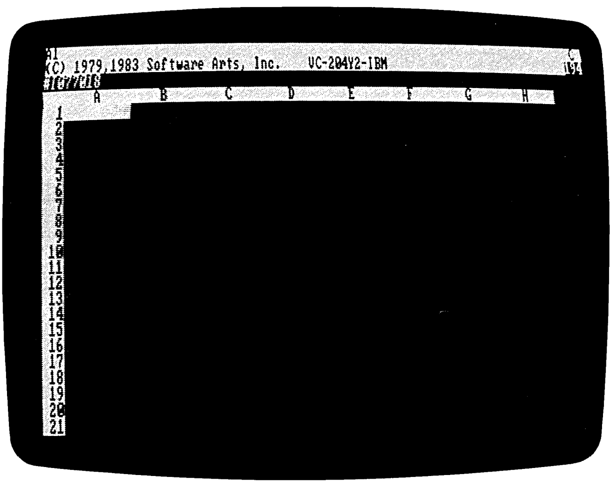
V

© 1981 Software Arts, Inc. x.xx

Displays copyright notice and version number on prompt line; clears automatically with next keystroke.

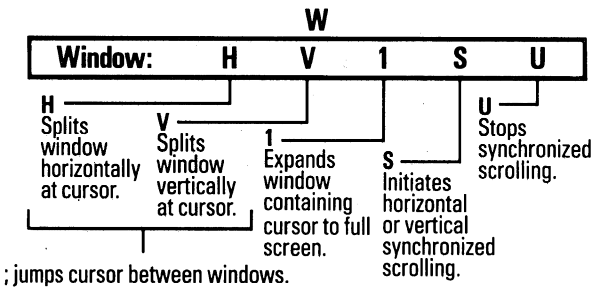
The Version command displays the copyright notice and version number of the VisiCalc program in the status area. It does not affect the worksheet. The Version command is entered /V.

Should you need to call or write with questions about the VisiCalc program, be sure to include the version number displayed by the Version command. The following photograph shows how the notice appears on the screen:



005-002/P

WINDOW COMMAND



The Window command (/W) splits the screen vertically or horizontally so that the worksheet can be viewed through two windows simultaneously. The windows can be positioned to show rows or columns too far apart to be viewed through the single window.

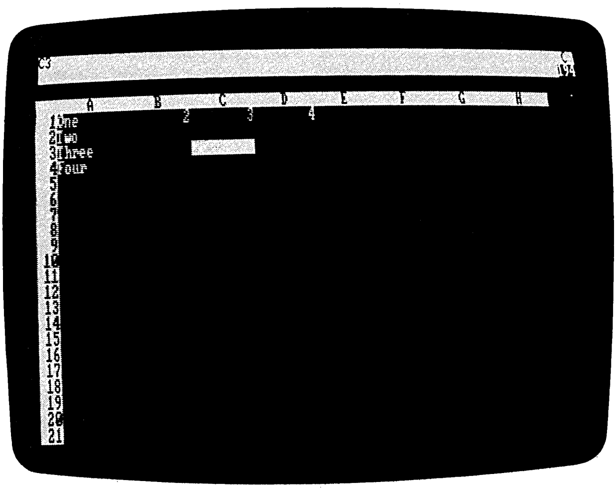
When the screen is split, the windows can be scrolled independently or together. Worksheet locations can be displayed in different formats in each window; each window can have a different column width.

Horizontal Option —/WH

The Horizontal option of the Window command (/WH) splits the window by placing a second column border (A, B, C, etc) just above the row that contains the cursor. When the command is executed, the cursor moves up one row into the upper window. For example, typing /WH when the cursor is at A4 places the horizontal border between A4 and A3 and moves the cursor to A3.

Because the column bar is placed between two rows and because it moves the cursor up, /WH cannot work if the cursor is in row 1–there's no "between" and nowhere to put the cursor. The following example shows both the horizontal and vertical window options:

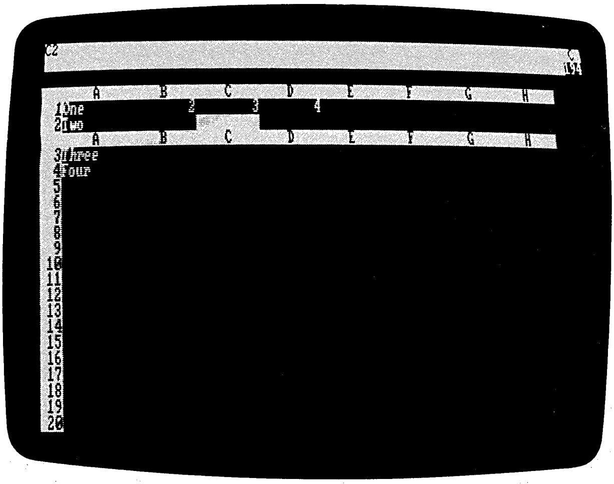
Type:	Result:
1. /CY	The screen clears and the cursor moves to A1.
2. → 2 → 3 → 4	A row of numbers in row 1.
3. HOME	
4. One ↓ Two ↓ Three ↓ Four ↓ ↵	A column of words in column A.
5. >C3 ↵	The screen should look like the following photograph:



005-055/P

6. **/W** Prompt Line: Window: H V 1 S U

7. **H** The screen should look like the following photograph (note the position of the cursor and the horizontal bar):



005-056/P

Each window now shows the worksheet independently. The cursor can move around the worksheet in the top window. Move the cursor to A1 to highlight "One" in the top window. Type ; to move the cursor to entry position C3 in the bottom window.

Typing ; moves the cursor to its last position in the other window. In this case, that position was C3. Type ; again. The cursor returns to A1, the last entry position it rested on before it moved to the lower window.

Any VisiCalc command can be entered in either window; with two exceptions, the effect of a command on the worksheet is shown in both windows. The two exceptions are the Column and Format options of the Global command, which are set in one window at a time (see "Format Command" and "Global Command" in this chapter for a more detailed description of formats and how to set them).

Move to the bottom window with ; and remove the horizontal window by typing /**W1**. There is now one window; it has the format options of the window that contained the cursor when the Window command was entered.

The screen can be split into two windows only when one window is displayed (you can't change directly from horizontal windows to vertical windows or vice versa). The size of each window is determined by the position of the cursor at the time the Window command is entered.

If you intend to follow the vertical window example, type **HOME** to bring the example back into view.

Vertical Option —/WV

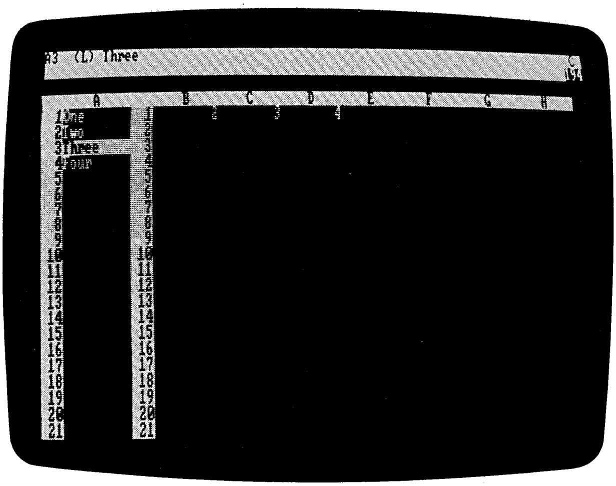
The Vertical option of the Window command (/WV) splits the window by adding a second row border (1, 2, 3, etc.) just to the left of the column that contains the cursor. For example, typing /WV when the cursor is at B3 places the vertical border between A3 and B3 and moves the cursor left to A3.

When the screen is split vertically, the columns in the right window may be slightly narrower than those in the left window, to make room for the additional row border. As a rule, the VisiCalc program narrows the right window by one space if the narrowing preserves a column on the screen.

The behavior of vertical windows–cursor movement, effect of commands, etc.–is like horizontal windows.

If the previous example is not on the screen, type steps 1 through 4 from the horizontal window example before continuing.

Type:	Result:
1. >B3 ↵	The cursor moves to B3.
2. /W	Prompt Line: Window: H V 1 S U
3. V	The screen should look like the following photograph (note the position of the border and cursor):



005-057/P

One Window Option —/W1

The One Window option of the Window command (/W1) displays a single window. If the command is entered while the screen is split, the global format and column width settings of the window that contains the cursor are applied to the single window.

Synchronized Scrolling Option —/WS

The Synchronized Scrolling option of the Window command (/WS) synchronizes horizontal scrolling in horizontal windows or vertical scrolling in vertical windows. If the window that contains the cursor is scrolled, the other window also scrolls.

Unsynchronized Scrolling Option —/WU

The Unsynchronized Scrolling option of the Window command (/WU) turns off synchronized scrolling.

Note: The last three options of the Window command (/W1, /WS, and /WU) can only be used when the screen is split.

THE STAR* COMMAND REFERENCE

INTRODUCTION

Chapter 4 describes each of the VisiCalc IV Star* commands in detail. Each Star* command appears in alphabetical order. Command options are presented in the order that appears on the prompt line.

This package includes a chart showing all of the Star* commands and their options. Each command description begins with a separate chart of each Star* command.

Typing /* opens the door to the Star* commands. When you type /*, the prompt line offers you the following choices. You choose the command you want by pressing that letter.

G The Star* Graph command (/*G) creates graphs from the data on your worksheet. The Star* Graph command options quickly convert your data to pie charts, bar charts, line graphs, area graphs, dot graphs, component graphs, scatter diagrams, or high/low/close graphs.

Star* Graph Options (/*GO) can display these graphs in color (on a color monitor) or shaded (on a monochrome monitor), or print them on any of the graphics printers listed in Appendix C.

M The Star* Move command (/*M) allows you to reorganize the data in your worksheet either temporarily or permanently. Using the command options you can rearrange the rows and columns, collect all rows meeting your criteria, and sort the rows of your worksheet.

K The Star* Keysaver™ Maintenance command (/*K) allows you to assign sequences of keystrokes (called *Keysaver commands* because they save your keystrokes) to each function key or letter on your keyboard. Later, when you press a function key, the program types the Keysaver command for you.

The VisiCalc IV program lets you have as many as 67 Keysaver commands defined at any one time. And you can save additional Keysaver commands on a disk and load them whenever you need to change the keystroke sequence for any Keysaver command.

In addition to letting you create Keysaver commands, the Star* Keysaver Maintenance options let you edit, clear, print, store, and retrieve your Keysaver commands from disk.

A The Star* Adapters command (/*A) allows you to tell the program about the graphics hardware in your system. If you have an IBM Color/Graphics adapter, and your printer is attached to LPT1:, you don't need to use this command.

D The Star* Devices command (/*D) allows you to tell the program what kind of printer you are using to print your graphs. If you have an IBM graphics printer or equivalent, you don't need to use this command.

O The Star* Options command (/*O) allows you to choose between two modes of entering commands. One mode, called Standard mode, requires more keystrokes but is easier to learn. The other mode, Menu mode, requires fewer keystrokes and is more efficient after you have become familiar with the Star* commands.

When you use a Star* command, an arrow appears in front of one option. Unless otherwise stated in this manual, an arrow in front of an option indicates which option you chose the last time you used the command. For details, see "Star* Options Command" at the end of this chapter.

For more information about any of these commands, see the appropriate section in this chapter.

USING THE STAR* COMMAND DEMONSTRATION

The demonstration program, designed to acquaint you with the Star* commands, uses a series of Keysaver™ commands. To run the demonstration:

1. Load DOS. Choose one of the following options:
   (a) If drive A is a double-sided drive, remove your DOS disk and replace it with your program disk.
   (b) If drive A is a single-sided drive, put your program disk in drive B, type **b:**, and press ↵.
2. Choose one of the following options:
   (a) If you have a 128K system, type **vdemo128** and press ↵.
   (b) If you have a 192K system, type **vdemo192** and press ↵.

   Several prompts flash across the prompt line as the demonstration loads. When the demonstration is loaded, the prompt and edit lines clear and an explanation appears on the screen.
3. When you have read the explanation, press **[F1]** to display the main menu for the demonstration.
4. Choose as many options of the demonstration as you like by pressing the function keys that are listed on the screen.
5. When you have finished the demonstration, you can return to the VisiCalc IV program by pressing **ALT-[F8]** or to DOS by pressing **ALT-[F9]**.

Note: If you return to the program from the demonstration, the Keysaver commands that make up the demonstration are still in memory. For more information on Keysaver commands, see "Star* Keysaver Maintenance Commands" later in this chapter.

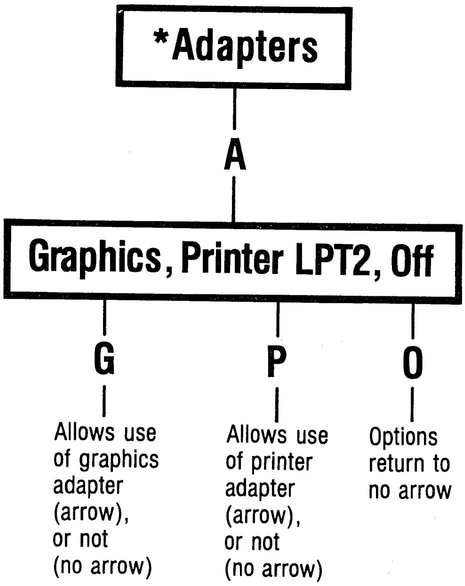
Helpful Hints for the Demonstration

The following hints will help you with running the demonstration:

- If your system has a graphics monitor only, press ↵ or **[F9]** to return to the demonstration worksheet from a displayed graph.
- If you have an IBM or Epson printer with a Graftrax™ or Graftrax™-Plus ROM, typing **P** when a graph is displayed on the screen will print that graph on the printer.

Now that you are acquainted with the various Star* commands, study the following sections, arranged alphabetically, for more detailed information about each command.

STAR* ADAPTERS COMMAND



The Star* Adapters command (/*A) lets you tell the VisiCalc IV program what kind of hardware is attached to your system, so that the program can make the best use of the hardware.

Using this command, for example, you can inform the program that your printer is attached to LPT2: instead of LPT1:. Once aware of this, the program automatically prints on LPT2:.

Since most people who use the VisiCalc IV program do not need the Star* Adapters command, let's try to save you some time. If the following statements are true, you don't need to use this command:

- You do not intend to create any graphs.
- Your printer is attached to LPT1:.

In contrast, you do need the Star* Adapters command if your computer has more than one connector for your graphics printer, and the graphics printer is connected to a port other than LPT1:.

If you're still not sure whether to use this command, it's time to experiment. Try graphing and printing a graph without using the command. If you succeed, you don't need the command. If you fail, you probably do.

How to Use the Star* Adapters Command

To use the Star* Adapters command, just type **/*A** and then press **G**, **P**, or **O** repeatedly.

Pressing **G** repeatedly turns on and off an → in front of Graphics on the prompt line. The presence of this arrow tells the program that you have an IBM Color/Graphics adapter. If the arrow is absent, the program thinks you do not have a graphics adapter and prevents you from using the Star* Graph command.

Warning: If you do not have a graphics adapter but the program thinks that you do, you could damage your worksheet when you use the Star* Graph command.

Pressing **P** repeatedly turns on and off an → in front of Printer LPT2 on the prompt line. If the arrow is present, the program prints your graphs on LPT2:. If the arrow is absent, the program prints your graphs on LPT1:.

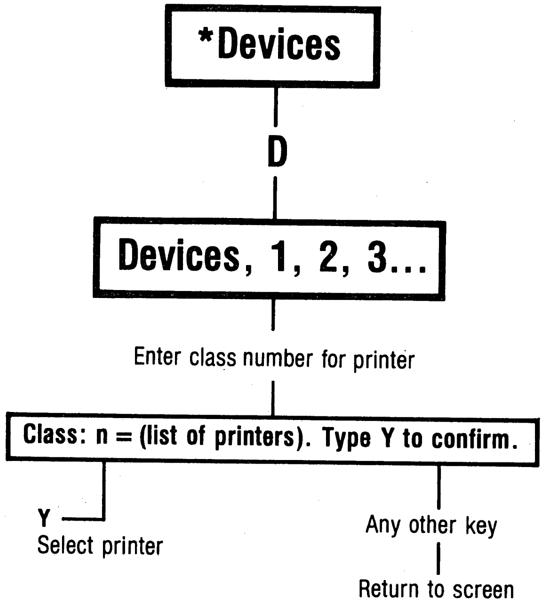
Pressing **O** turns off any arrow that is present. The ability to set the arrows is most useful when you define Keysaver commands, which cannot sense the state of the arrows.

When you have the arrows where you want them, press ↵. This tells the program that you are finished with the command.

When to Use the Star* Adapters Command

Use the Star* Adapters command before you start graphing. Once you've used it, you need not use it again (although you can if you wish) until the next time you load the VisiCalc IV program.

STAR* DEVICES COMMAND



The Star* Graph command (/*G) displays graphs on your graphics monitor and lets you print your graphs if you have a graphics printer. Because there are many different types of printers, however, the VisiCalc IV program must know what type you have before it can print your graph. The Star* Devices command (/*D) lets you tell the VisiCalc IV program what type of graphics printer you have connected to your system.

Classes of Graphics Printers

The VisiCalc IV program works with many graphics printers. Appendix C lists the printers that are supported by and have been tested with the VisiCalc IV program.

Look at Appendix C and note that the printers are divided into *classes*. The printers within each class are similar enough that the program need not distinguish between them.

When you use the Star* Devices command, you must enter the *number* of the class that contains your graphics printer.

Who Must Use the Star* Devices Command

You must use the Star* Devices command only if *both* of the following statements are true:

- You intend to print your graphs in addition to displaying them on the screen.
- Your printer is not in Class 1 (see Appendix C for printer classification).

If your printer is in Class 1, you don't need to use the command because the program automatically assumes Class 1.

When to Use the Star* Devices Command

Each time you load the VisiCalc IV program, use the Star* Devices command *before* you print a graph. As long as the program remains in your computer's memory, and as long as you do not connect a printer of a different class, you need not use the command again.

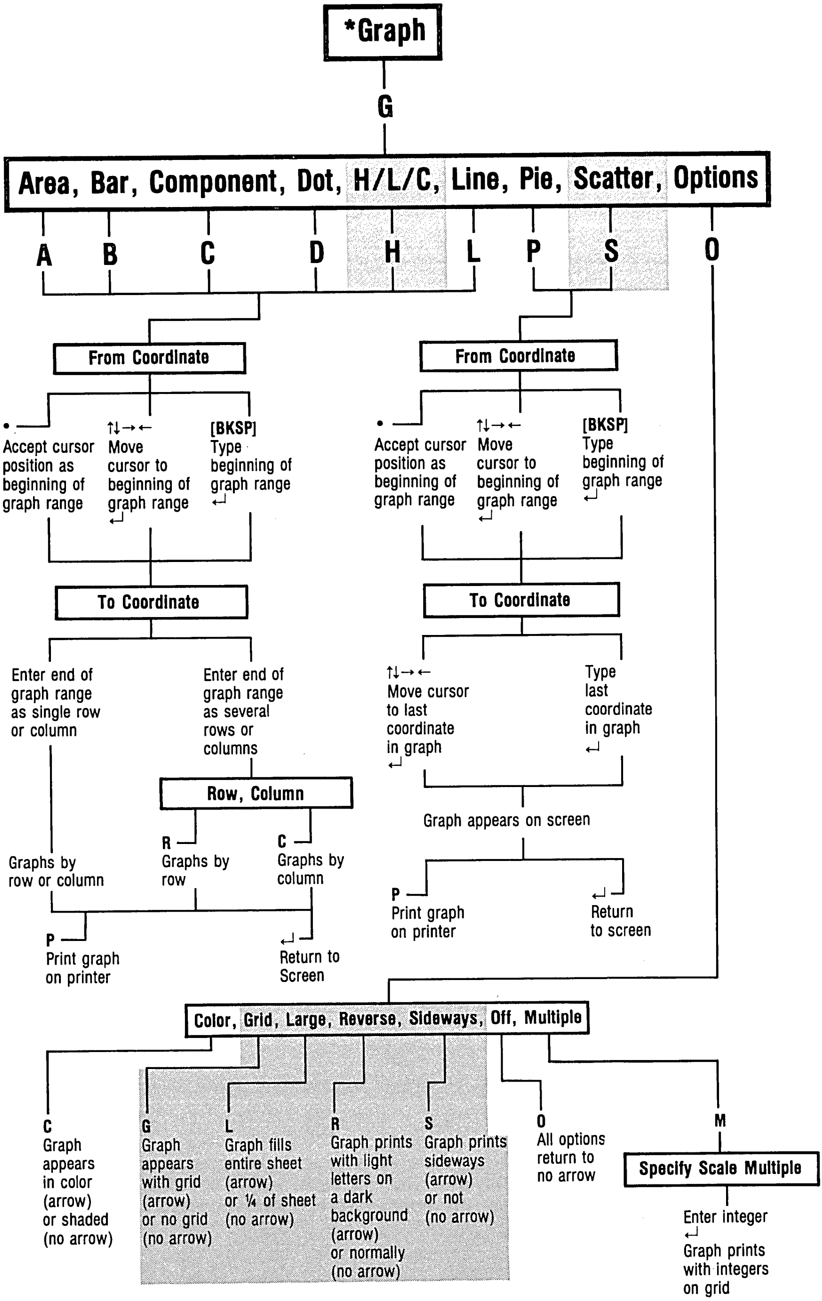
Example Using the Star* Devices Command

Suppose that you have just loaded the VisiCalc IV program, and that you have a C. Itoh Prowriter printer. Since this printer is listed in Class 2, the following keystrokes are needed to tell the VisiCalc IV program what kind of graphics printer you have.

Type:	Result:
1. **/*D**	**Prompt Line:** *Graphics Devices: 1, 2, 3 The prompt line shows you which classes of printers the program currently recognizes. An arrow points to the class that is currently selected.
2. **2**	**Prompt Line:** Device Class: 2= [C. Itoh Prowriter]. Type Y to confirm. The prompt line lists the printers in Class 2, allowing you to verify that your printer is in this class.
3. **Y**	This tells the program that you have verified that you want Class 2. The program is now able to print graphics on your printer. Pressing any other key cancels the command without changing the class.

You do not need to use the Star* Devices command again until you reload the VisiCalc IV program or attach a graphics printer of a different class.

STAR* GRAPH COMMAND



The Star* Graph command (/*G) lets you create graphs on your graphics monitor or on any of the printers listed in Appendix C. By using this command, you can turn the mass of numbers on your worksheet into an easy-to-understand graph.

The Star* Graph command (/*G) gives you the ability to create any of the following graphs:

- An area graph (/*GA)
- A simple or clustered bar graph (/*GB)
- A component (stacked) bar graph (/*GC)
- A dot graph (/*GD)
- A high/low/close graph (/*GH)
- A line graph (/*GL)
- A pie graph (/*GP)
- A scatter graph (/*GS)

In addition, using Star* Graph Options (/*GO) lets you control how your graph looks both on your monitor and on your printer. Specifically, you can tell the VisiCalc IV program:

- Whether displayed graphs appear shaded (on a monochrome monitor) or in color (on a color monitor).
- Whether you want a grid of reference dots.
- Whether you want your printed graphs to be large or small.
- Whether you want graphs to print in normal type (dark on light background) or in reverse type (light on dark background).
- Whether you want graphs to print horizontally or vertically.
- What numbers appear on your graph's axes.

Equipment Required

If you want to use the Star* Graph command, your system *must* have some special equipment:

- To display graphs...

  You must have an IBM Color/Graphics adapter and a graphics monitor (either color or monochrome).
- To print the graphs...

  You must have an IBM Color/Graphics adapter and a graphics printer attached to either the IBM Monochrome adapter or to a separate parallel-printer adapter.

Note: If your system does not have an IBM Color/Graphics adapter, you cannot use the Star* Graph command (/*G).

If your printer is attached to LPT2:, you must use the Star* Adapters command (/*A) before using the Star* Graph command. For more information, see the "Star* Adapters Command" section earlier in this chapter.

Using the Star* Graph Command

When you use any of the Star* Graph command options, except Star* Graph Options (/*GO), the program displays a graph on your graphics monitor. Before the program can create the graph, it must know what values you want to use.

Using the Star* Graph command is similar to using the Replicate command (/R). To tell the program which values to graph, you use the same technique used to specify a source range in the Replicate command. You can specify the coordinate by pointing to it with the cursor or by explicitly typing it.

Each of the Star* Graph options can graph the values contained within a *rectangular area* of your worksheet. In general, you identify the rectangle by entering the coordinates of the upper-left and the lower-right corners of the rectangle. This rectangle is called the *range* of the graph.

When the prompt line asks you to specify the range, enter the upper-left and lower-right coordinates.

Some of the command options can graph only ranges having particular dimensions. Specifically:

- Pie Graph (/*GP)

  The Pie Graph option accepts adjacent values from any single row or single column within your worksheet. The upper-left and lower-right coordinates must be in the same row or the same column.

- Scatter Graph (/*GS)

  The Scatter Graph option graphs two collections of data against each other. This kind of graph is useful for exploring the correlation between the two data sets. Because of the focus on the relationship between the two data sets, this option expects values from two adjoining rows or two adjoining columns.

- High/low/close Graph (/*GH)

  The High/low/close Graph option graphs three collections of values against a numerical scale. This is particularly useful for examining stock values over intervals of time. This option requires the range to lie within three rows or three columns.

You can reorder your worksheet temporarily to create the adjacent rows or columns necessary for the graph you want. When you produce the graph, you can return the worksheet to its original order. See the "Star* Move Command" section later in this chapter for more information.

The other Star* Graph options can graph information from ranges of any size. If you use more than one row or column, the option graphs the successive rows or columns on the same set of axes.

CHANGING COLORS OR PATTERNS

The VisiCalc IV program uses colors or (if your monitor is monochrome) patterns to distinguish between areas within a graph. The program selects these colors and patterns for you, and you cannot change them.

If, however, you do not like the arrangement of the colors or patterns, you can modify the arrangement. Use the Insert command (/I) to place a blank row or column between the labels and the values in your worksheet. This additional row or column forces the program to redistribute the colors or patterns within the graph.

LABELING A GRAPH

The VisiCalc IV program uses the labels from row 1 or column A of your worksheet to label your graph. If the program finds values or blanks in the first row or column, it leaves the graph unlabeled. Consequently, if you want a labeled graph, make sure your first row or first column contains labels.

As you know, you can enter labels that exceed the width of your worksheet's columns. Although the worksheet displays on the screen only the characters that fit in a column, the program keeps track of any additional characters. The labels for your graphs will include the full labels, not just the characters displayed on the screen.

The Star* Graph options can use any additional characters for the labels on your graphs. This is especially useful for position A1, which provides the title for the graph.

Each Star* Graph option uses different guidelines for selecting labels. These guidelines are explained in the description of each option.

NUMBERS ON AXES

Every graph, except a pie graph, has a horizontal and a vertical axis. Although the markings on a horizontal axis differ from graph to graph, all the vertical axes are marked with numbers.

The VisiCalc IV program computes the numbers for the vertical axes using this information:

- The largest number being graphed.
- The scale multiple used in the graph.

For specific information on how the program computes the numbers, see the "Setting the Scale Multiple" section later in this chapter.

If the program computes a number that has too many digits to fit next to the axis, the number is replaced with a series of greater-than signs (>>>>>>>). This situation is called *overflow*.

If you see the overflow symbol, you can correct the problem by changing the units of measurement for information within your worksheet.

For example, suppose that your vertical axis represents distance in inches. If the program computes a number that overflows, you can fix the problem by changing the units of distance and regraphing. Depending on the severity of the overflow, you could switch to feet or yards (or even miles).

Suppose that you are creating a graph showing last year's volume of sales in dollars. If your graph overflows, you might consider measuring sales in thousands or millions of dollars rather than just dollar units.

By reducing the number of digits in your numbers, you do more than just fix the overflow. You also make your data easier to understand. For example, 62.4 billion is easier to understand than 62387415032.

OVERCROWDING GRAPHS

Each graph must fit on a single screen, so putting too much information into one graph pushes the elements of the graph closer together. If you pack too much data in a single graph, the result may be unreadable.

If you create a graph that is too crowded, you can correct the problem by using a different type of graph, one that requires less room per element. For example, suppose that you create a component bar graph and find it too crowded. Try a line graph of the same information. This could eliminate the problem because lines require less space than bars. If the line graph is still too crowded, try a dot graph. If none of this helps, you may have to divide the information and create two graphs.

QUICK REGRAPHING

The VisiCalc IV program provides a shortcut method for regraphing the same information.

When you type /*G, followed by a letter identifying the kind of graph you want, press PGDN to accept the range, and press ↵, the program creates the graph specified by the letter. It answers the prompts for you by using the answers you gave when you last created a graph. This technique is most helpful when you want to regraph the same data, but produce a different kind of graph.

You can also change several values within the range of the graph, and then quickly regraph. For example, suppose that you create a line graph and then realize that you have an error in your data. If the line graph is displayed on your monitor, press ↵ to return to the worksheet, and change the data. Then type **/*G** and press ↵. The program creates the line graph using the new data.

When you type /*G and press ↵, the program performs the Star* Graph option that you most recently entered. It remembers how you answered the prompts, and it repeats your answers for you. This technique is most helpful when you want to change data within the graph's range and repeat the same kind of graph.

When you change from one kind of graph to another, remember that some graphs have range restrictions. For example, a pie graph requires that the range lie within a single row or a single column. If you try changing format from a dot graph (which accepts a range of any size) to a pie graph, your range might be invalid.

You do not have to let the program automatically answer all the prompts as you did when you last created a graph. When the worksheet is displayed, type **/*G**. Then, for each prompt, you have three choices:

1. You can answer the prompt, and the program will display the next prompt.
2. You can press **PGDN**, and the program will answer the prompt for you, using the answer that you provided the last time you created a graph. Then it will display the next prompt.
3. You can press ↵, and the program will answer any remaining prompts for you using the answers you supplied when you last created a graph. However, if you press ↵ in response to Specify range, the program accepts only the displayed range.

If you create many graphs, you will find that quick regraphing significantly reduces the number of keystrokes you must type.

GRAPHING NEGATIVE VALUES

All of the Star* Graph options, except the Component Bar Graph option and the Pie Graph option, accept negative values in a range. The Component Bar and Pie Graph options display an error message if the range contains a negative value.

Although the Scatter Graph option accepts negative values, the resulting graph can be surprising. If you create a scatter graph with negative values, the negative points being graphed appear outside the axes of the graph.

Printing Your Graph

When the VisiCalc IV program creates a graph, it displays it on your screen. It does not automatically print the graph. If you want to print your graph, you must tell the program to do so. The program prints the graph currently displayed on your screen.

Note: If you have not printed any graphs using the VisiCalc IV program, see the "Star* Devices Command" and "Star* Adapters Command" sections before attempting to print.

To print a graph:

1. Even if you have a color graphics monitor, press **/*GO** and select monochrome graphing (see the "Star* Graph Options (/*GO)" section later in this chapter for instructions). Shaded graphs print more clearly than colored graphs.
2. Ensure that the printer's power is on and the graph is displayed on your screen.
3. Optionally, you can:
   - Press **D** to cause a five-second delay before the program uses the next keystroke. This delay is useful to Keysaver commands—it allows them to control how long a graph remains on the screen.
   - Press **F** to advance the paper to the next form or page. This allows you to put each graph on a separate sheet of paper.
   - Press **L** to advance the paper one line. This allows you to control the spacing between graphs.
   - Press **R** to reset the printer.
4. When you have prepared your printer, press **P** to print your graph.

You can repeat steps 3 and 4 as many times as necessary.

When you have finished, press any key other than D, F, L, P, or R. This tells the program that you are ready to enter another command.

You can control the appearance of printed graphs using Star* Graph Options (/*GO). For more information, see "Star* Graph Options (/*GO)" later in this chapter.

USING A SYSTEM THAT HAS TWO MONITORS

If your system has both a graphics monitor and a monochrome monitor, it behaves differently than a single-monitor system. When you create a graph in a two-monitor system, the graph appears on the graphics monitor while the worksheet remains on the monochrome monitor. In a single-monitor system, the graph replaces the worksheet.

When you print a graph using a two-monitor system, the prompt Delay, Formfeed, Linefeed, Print, Reset is displayed on the monochrome monitor. In a one-monitor system, however, this prompt does not appear. You have to remember to press D, F, L, P, or R, as described in the previous section.

Area Graph

An *area graph* plots successive rows or columns of values from your worksheet. It plots each row or column as a series of line segments with the area below the line shaded (on monochrome monitor) or colored (on a color monitor). You use the Area Graph option (/*GA) to create an area graph.

Area graphs are most useful when the values of one row or column are greater than or equal to the values of the corresponding coordinates in the next row or column.

RANGE OF AREA GRAPHS

When you create an area graph, you can graph coordinates from any number of *adjacent* rows or columns. Regardless of the number of rows or columns you graph, they must be adjacent. The rectangle of the adjacent coordinates is the *range* of the graph.

For an area graph to be meaningful, the elements (*areas*) must occupy a moderate percentage of the space on the screen. If there are too many rows or columns in a single graph, the areas will become too small to be useful.

GRAPHING ROW BY ROW OR COLUMN BY COLUMN

If you want to graph several rows or columns rather than a single row or column, the program displays the following prompt:

```
Area Graph: Rows, Columns
```

If you press **R**, the command graphs information row by row. If you press **C**, it graphs column by column.

If you choose to graph row by row, the program graphs the values from the row that is nearest to the bottom of the worksheet first. Next, it graphs the row above the last row graphed, and repeats this process for each row within the range.

Similarly, if you choose to graph column by column, the program begins graphing the values from the rightmost column in the range, graphing columns successively from right to left.

An area graph looks best when the lines separating the areas do not cross. If you are creating a row-by-row graph and the values near the top of the range are greater than or equal to the values near the bottom of the range, the lines will not cross. In the same manner, if you are creating a column-by-column graph and the values at the left end of the range are greater than or equal to the values at the right end of the range, the lines will not cross. As long as the lines separating the areas do not cross, your area graph will look correct.

EXAMPLE: CREATING AN AREA GRAPH

This example shows you how to create an area graph from a worksheet. If you recreate the worksheet shown in the following illustration and follow these steps, you will generate an area graph.

Note: When you are creating this worksheet, use the following information: (1) the entry in position A1 is "Profit per Year"; and (2) the years (1982, 1981, etc.) should be entered as labels and formatted with /FR.

	A	B	C	D	E	F
1	Profit pe	1982	1981	1980	1979	1978
2	Revenue	1023	905	800	480	400
3	Pre-Tax	180	160	140	95	65
4	Net	95	83	72	50	35

036-005

Type:	Result:
1. **/*GOM100**↵	Uses the Multiple feature of Star* Graph Options to ensure that the number at the top of the vertical axis is a multiple of 100.
2. **>B3**↵	Positions the cursor on the upper-left coordinate in the range.
3. **/*GA**	Tells the program to create an area graph. **Prompt Line:** Area Graph: Specify range **Edit Line:** B3 The edit line displays the current cursor location.

4. **.** — Tells the program to accept B3 as the upper-left coordinate of the range.

   Prompt Line: Area Graph: Specify range

   Edit Line: B3...

5. **F4** — Enters the lower-right coordinate of the range.

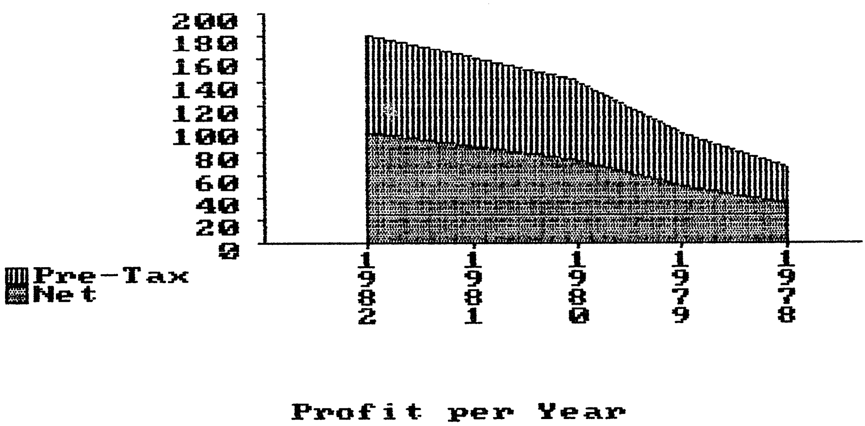
   Edit Line: B3...F4

6. ↵ — Verifies that F4 is the lower-right coordinate of the range.

   Prompt Line: Area Graph: Rows, Columns

   Because the range contains more than one row and one column, the program is asking whether you want row-by-row or column-by-column graphing.

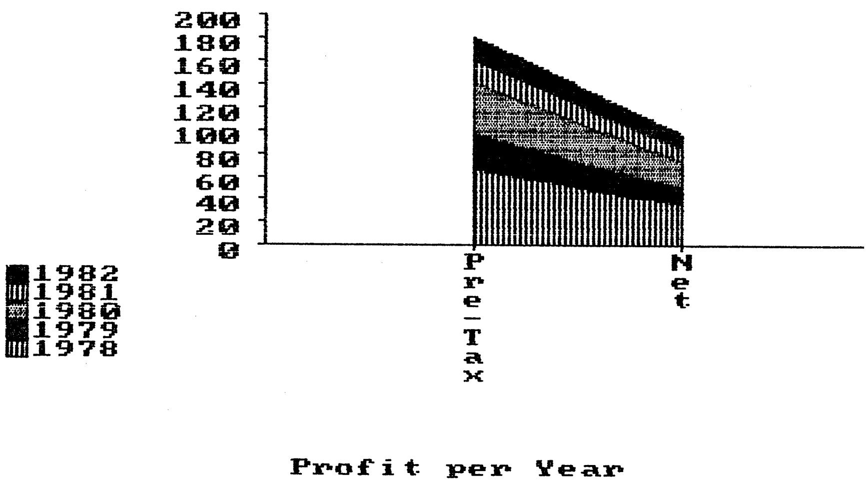
7. **R** — Selects row-by-row graphing. Your screen should display the following graph:



036-006

8. ↵ — Returns to the worksheet.

If you want to create a column-by-column graph of the same information, follow the instructions, with the exception of step 7. At step 7, press **C** instead of R. Your screen should display the following graph:



036-007

Bar Graph (Simple or Clustered)

A *bar graph* creates a vertical bar for each value that you want graphed. If you graph only a single row or column in your worksheet, all the bars are the same color or pattern. If you graph several rows or several columns, the bars within the cluster have different colors or shading, making the bars more distinct. The bars are clustered for a given x-value. You use the Bar Graph option (/*GB) to produce simple or clustered bar graphs.

RANGE OF BAR GRAPHS

When you create a bar graph, you can graph the coordinates from any number of *adjacent* rows or columns. Regardless of the number of rows or columns you graph, they must be adjacent. The rectangle of adjacent coordinates is the *range* of the graph.

For a bar graph to be meaningful, the elements (*bars*) must occupy a moderate percentage of the space on the screen. If there are too many rows or columns in a single graph, the bars will become very narrow and difficult to read.

GRAPHING ROW BY ROW OR COLUMN BY COLUMN

If you want to graph several rows or columns rather than a single row or column, the program displays the following prompt:

Bar Graph: Rows, Columns

If you press **R**, the command graphs information row by row. If you press **C**, it graphs column by column. The differences between the resulting graphs are significant.

Row-by-row graphing works best for comparing corresponding information from adjoining rows. If you choose to graph row by row, the program creates several clusters of bars. Each cluster corresponds to one column in the range. Within each cluster, there is one bar for each row in the column.

Similarly, column-by-column graphing works best for comparing corresponding information from adjoining columns. If you choose to graph column by column, the program creates several clusters of bars. Each cluster corresponds to one row in the range. Within each cluster, there is one bar for each column in the row.

EXAMPLE: CREATING A BAR GRAPH

This example shows you how to create a bar graph from a worksheet. If you recreate the worksheet shown in the following illustration, and follow these steps, you will generate a bar graph.

Note: When you are creating this worksheet, use the following information: (1) the entry in position A1 is "Electric Costs"; (2) the years (1980, 1981, etc.) should be entered as labels, and (3) the month names should be formatted with /FR. This is the same worksheet used in the area graph example.

	A	B	C	D	E	F	G	H
1	Electric	JAN	FEB	MAR	APR	MAY	JUN	Total
2								
3	1980	223.00	172.00	150.00	92.00	70.00	61.00	768.00
4	1981	202.00	182.00	155.00	95.00	75.00	65.00	774.00
5	1982	220.00	195.00	170.00	105.00	84.00	73.00	847.00
6	----------	----------	----------	----------	----------	----------	----------	----------
7	Monthly	645.00	549.00	475.00	292.00	229.00	199.00	2389.00

036-008

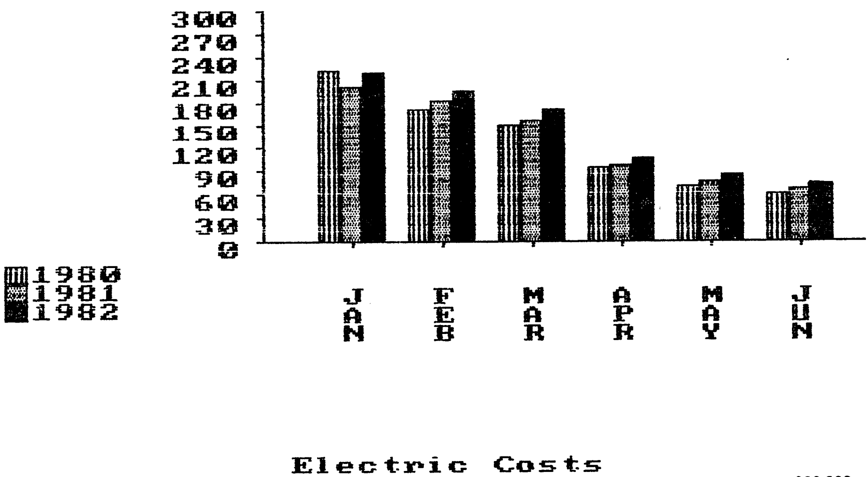
Type:	**Result:**
1. **/*GOM100↵**	Uses the Multiple feature of Star* Graph Options to ensure that the number at the top of the vertical axis is a multiple of 100.
2. **/*GB**	Tells the program that you want to create a bar graph. **Prompt Line:** Bar Graph: Specify range **Edit Line:** (displays the coordinate of the cursor's current location)
3. **[BKSP]**	Tells the program that you want to explicitly type the upper-left coordinate of the range. **Prompt Line:** Bar Graph: Specify range **Edit Line:** (clear)
4. **B3**	Shows the upper-left coordinate of the range. **Prompt Line:** Bar Graph: Specify range **Edit Line:** B3
5. **.**	Tells the program to accept B3 as the upper-left coordinate. **Prompt Line:** Bar Graph: Specify range **Edit Line:** B3...

6. **G5**↵ — Tells the program that the range is from B3 to G5. Because the range consists of more than one row or column, the program issues another prompt.

   Prompt Line: Bar Graph: Rows, Columns

   Your response to this prompt tells the program whether to graph row by row or column by column.

7. **R** — Tells the program to graph row by row. The screen should display the following graph:

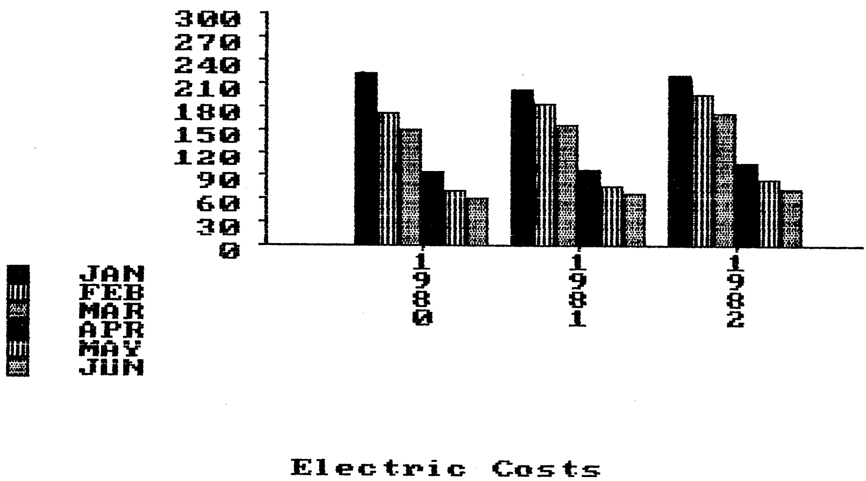


036-009

8. **P** — Prints the graph on your graphics printer.

9. ↵ — Returns to the worksheet.

If you want to create a column-by-column graph of the same information, follow the instructions, with the exception of step 7. At step 7, press **C** instead of R. Your screen should display the following graph:



036-010

Component Bar Graph

A *component bar graph* is similar to a bar graph. However, a component bar graph stacks the bars on top of each other, whereas a clustered bar graph puts the bars for each item next to each other. The height of the stacked bar equals the sum of the values represented in the bar.

A component bar graph is useful for comparing the sums of several items. You use the Component Bar Graph option (/*GC) to create a component bar graph.

RANGE OF COMPONENT BAR GRAPHS

When you create a component bar graph, you can graph the coordinates from any number of *adjacent* rows or columns. Regardless of the number of rows or columns you graph, they must be adjacent. The rectangle of adjacent coordinates is the *range* of the graph.

For a component bar graph to be meaningful, the elements (*bars*) must occupy a moderate percentage of the space on the screen. If there are too many rows or columns in a single graph, the bars will become very narrow and difficult to read.

GRAPHING ROW BY ROW OR COLUMN BY COLUMN

Since you are graphing several rows or columns, the program displays the following prompt:

Component Bar Graph: Rows, Columns

If you press **R**, the command graphs information row by row. If you press **C**, it graphs column by column. The differences between the resulting graphs are significant.

Row-by-row graphing works best for comparing corresponding information from adjoining rows. If you choose to graph row by row, the program creates a bar with several stacks. Each bar corresponds to one column in the range. Within each bar, there is one section for each row in the column.

Similarly, column-by-column graphing works best for comparing corresponding information from adjoining columns. If you choose to graph column by column, the program creates a bar with several stacks. Each bar corresponds to one row in the range. Within each bar, there is one section for each column in the row.

EXAMPLE: CREATING A COMPONENT BAR GRAPH

This example shows you how to create a component bar graph from a worksheet. If you recreate the worksheet shown in the following illustration, and follow these steps, you will generate a component bar graph.

Note: When you are creating this worksheet, use the following information: (1) the entry in position A1 is "Electric Costs"; (2) the years (1980, 1981, etc.) should be entered as labels, and (3) the month names should be formatted with /FR. This is the same worksheet used in the area graph example.

```
    !  A          B        C        D        E        F        G        H
-------------------------------------------------------------------------
 1  !Electric    JAN      FEB      MAR      APR      MAY      JUN    Total
 2  !
 3  !1980      223.00   172.00   150.00    92.00    70.00    61.00   768.00
 4  !1981      202.00   182.00   155.00    95.00    75.00    65.00   774.00
 5  !1982      220.00   195.00   170.00   105.00    84.00    73.00   847.00
 6  !--------------------------------------------------------------------
 7  !Monthly   645.00   549.00   475.00   292.00   229.00   199.00  2389.00
```

036-008

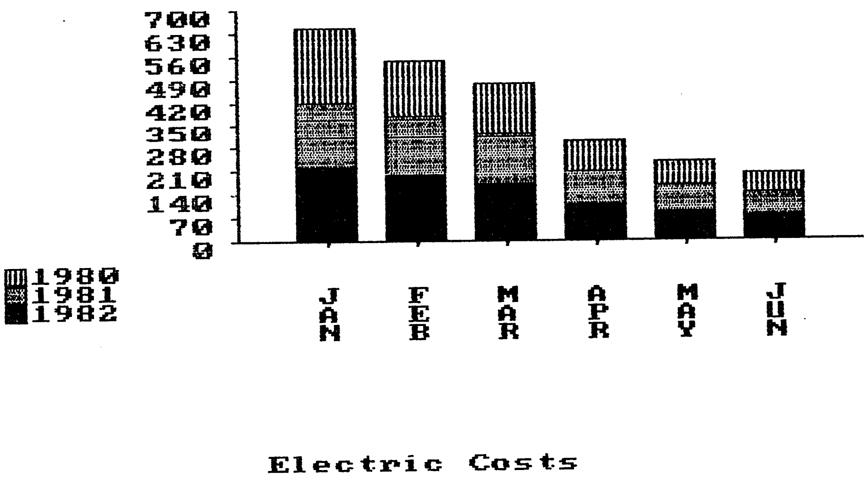
Type:	Result:
1. /*GOM100↵	Uses the Multiple feature of Star* Graph Options to ensure that the number at the top of the vertical axis is a multiple of 100.
2. /*GC	Tells the program that you want to create a component bar graph. **Prompt Line:** Component Bar Graph: Specify range **Edit Line:** (displays the coordinate of the cursor's current location)
3. [BKSP]	Tells the program that you want to explicitly type the upper-left coordinate of the range. **Prompt Line:** Component Bar Graph: Specify range **Edit Line:** (clear)
4. B3	Shows the upper-left coordinate of the range. **Prompt Line:** Component Bar Graph: Specify range **Edit Line:** B3
5. .	Tells the program to accept B3 as the upper-left coordinate. **Prompt Line:** Component Bar Graph: Specify range **Edit Line:** B3...

6. **G5↵** — Tells the program that the range is from B3 to G5. Because the range consists of more than one row or column, the command issues another prompt.

   Prompt Line: Component Bar Graph: Rows, Columns

   Your response to this prompt tells the program whether to graph row by row or column by column.

7. **R** — Tells the program to graph row by row. The screen should display the following graph:

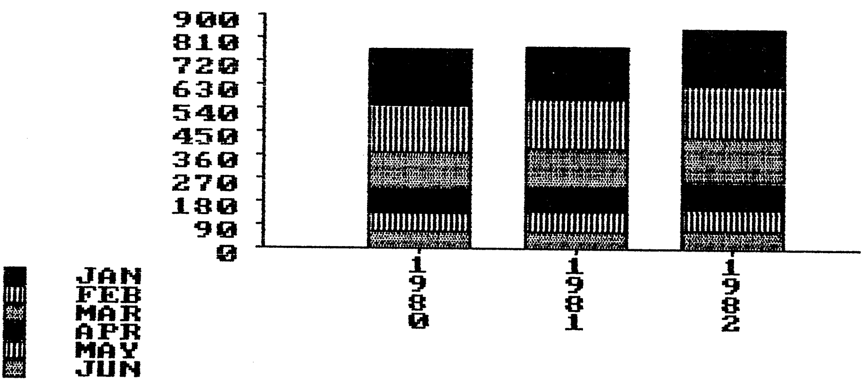


036-011

8. **P** — Prints the graph on your printer.

9. **↵** — Returns to the worksheet.

If you want to create a column-by-column graph of the same information, follow the instructions, with the exception of step 7. At step 7, press **C** instead of R. Your screen should display the following graph:



Electric Costs

036-012

GRAPHING NEGATIVE VALUES

You cannot include negative values in the range of a component bar graph. If your range includes any negative values, the program displays the following error message:

ERROR: Negative value(s)

After the message is displayed, press ↵. The program terminates the option, allowing you to enter a new command.

If you are more interested in the *size* of the number rather than the *sign* of the number, you can convert the negative value to positive by using the @ABS function. Once you've converted the number, you can create the component bar graph.

Dot Graph

A *dot graph* plots points against a vertical axis. Each point is marked by a small symbol, such as a plus sign (+). The points are not connected. The graph's vertical axis consists of numbers, and the horizontal axis consists of labels taken from the worksheet. You use the Dot Graph option (/*GD) to create a dot graph.

RANGE OF DOT GRAPHS

You can create dot graphs consisting of one or more rows or columns. The program assigns each row or column a color and a symbol. Then, for each row or column, the program graphs the symbols against a numerical vertical axis.

GRAPHING ROW BY ROW OR COLUMN BY COLUMN

If you want to graph several rows or columns, rather than a single row or column, the program displays the following prompt:

Dot Graph: Rows, Columns

If you press R, the program graphs row by row. It also assigns a color and a symbol to each row and plots each dot one row at a time.

If you press C, the program graphs column by column. Again, it assigns a color and a symbol to each column and plots each dot one column at a time.

EXAMPLE: CREATING A DOT GRAPH

This example shows you how to create a dot graph from a worksheet. If you recreate the worksheet shown in the following illustration, and follow these steps, you will generate a dot graph.

Note: When you are creating this worksheet, use the following information: (1) the entry in position A1 is "Electric Costs"; (2) the years (1980, 1981, etc.) should be entered as labels, and (3) the month names should be formatted with /FR. This is the same worksheet used in the area graph example.

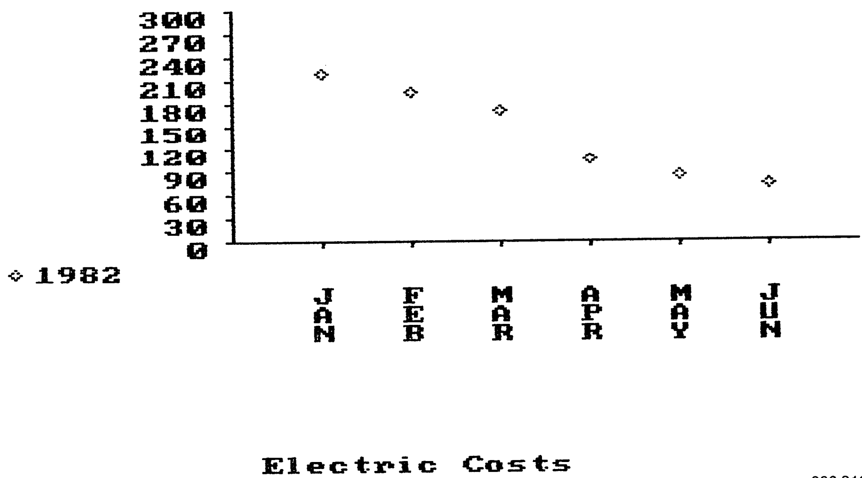
	A	B	C	D	E	F	G	H
1	Electric	JAN	FEB	MAR	APR	MAY	JUN	Total
2								
3	1980	223.00	172.00	150.00	92.00	70.00	61.00	768.00
4	1981	202.00	182.00	155.00	95.00	75.00	65.00	774.00
5	1982	220.00	195.00	170.00	105.00	84.00	73.00	847.00
6	----------	----------	----------	----------	----------	----------	----------	----------
7	Monthly	645.00	549.00	475.00	292.00	229.00	199.00	2389.00

036-008

Type:	Result:
1. /*GOM100↵	Use the Multiple feature of Star* Graph Options to ensure that the number at the top of the vertical axis is a multiple of 100.
2. /*GD	Tells the program to create a dot graph. **Prompt Line:** Dot Graph: Specify range **Edit Line:** (displays the coordinate of the cursor's current location)
3. [BKSP]	Tells the program that you want to explicitly type the upper-left coordinate of the range. **Prompt Line:** Dot Graph: Specify range **Edit Line:** (clear)
4. B5	Shows the upper-left coordinate in the range. **Prompt Line:** Dot Graph: Specify range **Edit Line:** B5
5. .	Tells the program that you have verified your upper-left coordinate. **Prompt Line:** Dot Graph: Specify range **Edit Line:** B5...

6. **G5↵** Tells the command that the range is B5...G5.

Your screen should display the following dot graph:



036-013

7. ↵ Returns to the worksheet.

High/Low/Close Graph

A *high/low/close graph* is useful for reporting stock prices. It graphs values from three columns or three rows (one for high values, one for low values, and one for closing values) against a numerical vertical axis. You use the High/low/close Graph option (/*GH) to create this graph.

Each triplet of values is graphed as a vertical bar with a cross mark. The vertical bar connects the high and low values, and the cross mark represents the closing value.

RANGE OF HIGH/LOW/CLOSE GRAPHS

When you create a high/low/close graph, you must specify a range lying within three rows or three columns. The uppermost row or the leftmost column must contain the high values. The next row or column must contain the low values, and the third row or column must contain the closing values.

If your range does not contain three rows or three columns, the program displays the following error message:

ERROR: Invalid number of rows or columns

and it terminates the option.

If your range is three rows by three columns, the program displays the following prompt:

H/L/C Graph: Rows, Columns

If the closing values lie in a row, press R; if they lie in a column, press C.

LABELING A HIGH/LOW/CLOSE GRAPH

The title for a high/low/close graph is always taken from position A1 and displayed at the bottom of the graph.

The coordinates in your worksheet provide the labels for the horizontal axis and for the legend. The orientation of your information governs which coordinates provide the labels. For example:

- If your closing values lie in a row, then row 1 provides the labels for the horizontal axis. The labels come from the coordinates of row 1 that are above the closing values on the worksheet.

  The labels for the legend come from the coordinates in column A lying to the left of the three rows.
- If your closing values lie in a column, column A provides the labels for the horizontal axis. The labels come from the coordinates in column A to the left of the element's triplet of values.

  The labels for the legend come from the coordinates in row 1 lying above the three columns.

If any of these coordinates are blank, empty, or contain values, the corresponding labels will be blank.

EXAMPLE: CREATING A HIGH/LOW/CLOSE GRAPH

This example shows you how to create a high/low/close graph from a worksheet. If you recreate the worksheet shown in the following illustration and follow these steps, you will generate a high/low/close graph.

Note: When you are creating this worksheet, use the following information: (1) the entry in position A1 is "Glacial Memories Stock"; and (2) the day names should be formatted with /FR.

	A	B	C	D	E	F
1	Glacial M	MON	TUE	WED	THU	FRI
2						
3	High	11	11.5	12	14	15
4	Low	9.25	9.75	11.25	12	13
5	Close	9.5	11.5	11.75	13.25	14.25

036-014

Type:	Result:
1. **/*GOM15**↵	Uses Star* Graph Options (/*GO) to set the value at the top of the vertical axis to a multiple of 15.
2. **/*GH**	Tells the program that you want to create a high/low/close graph. Prompt Line: H/L/C Graph: Specify range Edit Line: (displays the current location of the cursor)
3. **[BKSP]**	Tells the program that you want to explicitly type the upper-left coordinate. Prompt Line: H/L/C Graph: Specify range Edit Line: (clear)

4. **B3** Shows the upper-left coordinate of the range.

Prompt Line: H/L/C Graph: Specify range

Edit Line: B3

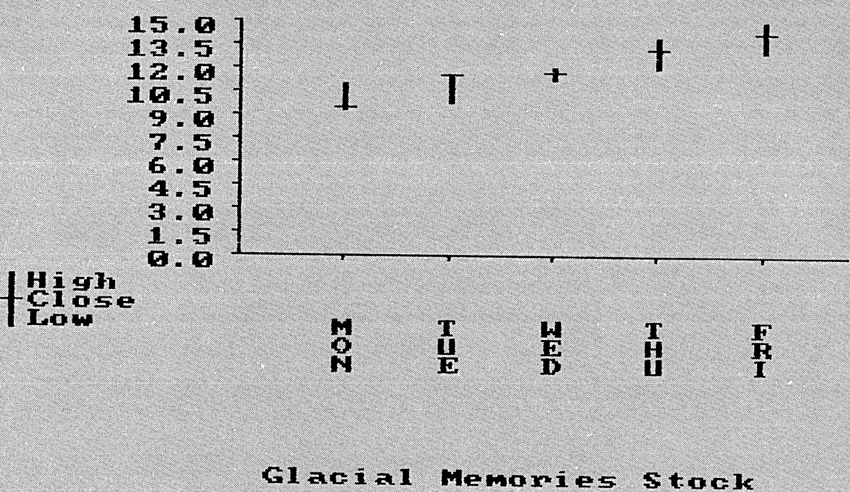
5. **.** Tells the program that B3 is correct.

Prompt Line: H/L/C Graph: Specify range

Edit Line: B3...

6. **F5**↵ Tells the program that F5 is the lower-right coordinate in the range.

Your screen should display the following graph:



036-015

7. ↵ Returns to the worksheet.

Line Graph

A *line graph* is very similar to a dot graph. A line graph, however, connects the dots of each row or column with lines. You use the Line Graph option (/*GL) to create a line graph.

EXAMPLE: CREATING A LINE GRAPH

This example shows you how to create a line graph from a worksheet. If you recreate the worksheet shown in the following illustration, and follow these steps, you will generate a line graph.

Note: When you are creating this worksheet, use the following information: (1) the entry in position A1 is "Electric Costs"; (2) the years (1980, 1981, etc.) should be entered as labels, and (3) the month names should be formatted with /FR. This is the same worksheet used in the area graph example.

	A	B	C	D	E	F	G	H
1	Electric	JAN	FEB	MAR	APR	MAY	JUN	Total
2								
3	1980	223.00	172.00	150.00	92.00	70.00	61.00	768.00
4	1981	202.00	182.00	155.00	95.00	75.00	65.00	774.00
5	1982	220.00	195.00	170.00	105.00	84.00	73.00	847.00
6	------------	------------	------------	------------	------------	------------	------------	------------
7	Monthly	645.00	549.00	475.00	292.00	229.00	199.00	2389.00

036-008

Type:	**Result:**
1. **/*GOM100↵**	Uses the Multiple feature of Star* Graph Options to ensure that the number at the top of the vertical axis is a multiple of 100.
2. **/*GL**	Tells the program to create a line graph. **Prompt Line:** Line Graph: Specify range **Edit Line:** (displays the coordinate of the cursor's current location)

3. **[BKSP]** — Tells the program that you want to explicitly type the upper-left coordinate of the range.

   Prompt Line: Line Graph: Specify range

   Edit Line: (clear)

4. **B5** — Shows the upper-left coordinate in the range.

   Prompt Line: Line Graph: Specify range

   Edit Line: B5

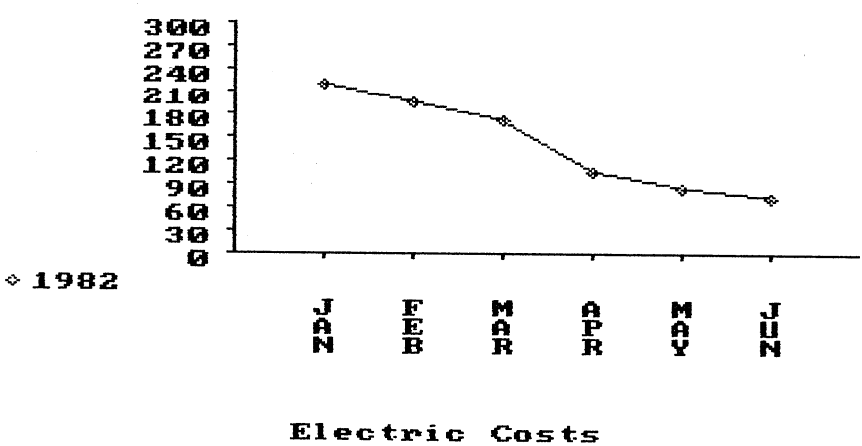
5. **.** — Tells the program that you have verified your upper-left coordinate.

   Prompt Line: Line Graph: Specify range

   Edit Line: B5...

6. **G5↵** — Tells the command that the range is B5...G5.

   Your screen should display the following line graph:



036-016

7. **↵** — Returns to the worksheet.

Pie Graph

A *pie graph* graphs the values of a single row or column as sectors, or slices, within a circle. The size of each slice is directly proportional to the sum of the values in the pie. Consequently, each slice is a percentage of the whole pie. You use the Pie Graph option (/*GP) to create a pie graph.

Pie graphs are most useful for comparing the values of a row or column to each other and to the sum of the values. However, the comparison is very coarse. Pie graphs show relative sizes very well, but they show absolute sizes poorly.

RANGE OF PIE GRAPHS

The Pie Graph option accepts ranges only if they are contained within one row or one column. If you graph a range exceeding one row or column, the program displays the following message:

```
ERROR: Invalid number of rows or columns
```

and it terminates the option.

LABELING PIE GRAPHS

Pie graphs use four kinds of labels:

- The main title for the graph (displayed above the pie) is taken from position A1. If A1 contains a value or is empty or blank, the pie graph has no title.
- The subtitle is displayed in the lower-left corner of the screen. The program takes the subtitle from the intersection of column A and the row being graphed (if you are graphing a row), or the intersection of row 1 and the column being graphed (if you are graphing a column).

  If the coordinate contains a value or is empty or blank, then the subtitle is left blank.
- The label for each slice consists of two parts. The first part is the slice's name. The name is taken from the coordinate in row 1 directly above the value represented by the slice (if you are graphing a row), or the coordinate in column A directly to the left of the value represented by the slice (if you are graphing a column).

  If the coordinate contains a value or is empty or blank, the name for the slice is left blank.

The second part of the label for each slice is the value of the slice. The program displays the value in parentheses below or next to the name of the slice.

- Finally, the program adds the values of all the slices and displays the sum in the lower-right corner of the screen.

EMPHASIZING ONE SLICE

The Pie Graph option allows you to emphasize a particular slice of your pie graph. If you want to emphasize a slice, position the cursor at that coordinate before typing /*GP. Then type **/*GP** and respond to the prompts. When the graph is displayed or printed, the emphasized slice is offset from the rest of the pie.

If you do not want to emphasize any slices, you must make sure, when positioning the cursor before typing /*GP, that the cursor is not on any of the coordinates being graphed.

SMALL SLICES NOT FILLED IN

If your pie graph includes any extremely small slices (less than approximately 2 percent of the sum of all the values), the Pie Graph option draws the sections, but does not fill them with a pattern or a color.

PRINTING PIE GRAPHS

Some graphics printers show pie graphs as ovals rather than circles. If your printer does this, try using the Sideways option of Star* Graph Options (/*GO). The resulting graph may look more like a circle.

EXAMPLE: CREATING A PIE GRAPH

This example shows you how to create a pie graph from a worksheet. If you recreate the worksheet shown in the following illustration, and follow these steps, you will generate a pie graph.

Note: When you are creating this worksheet, use the following information: (1) the entry in position A1 is "Electric Costs"; (2) the years (1980, 1981, etc.) should be entered as labels; and (3) the month names should be formatted with /FR. This is the same worksheet used in the area graph example.

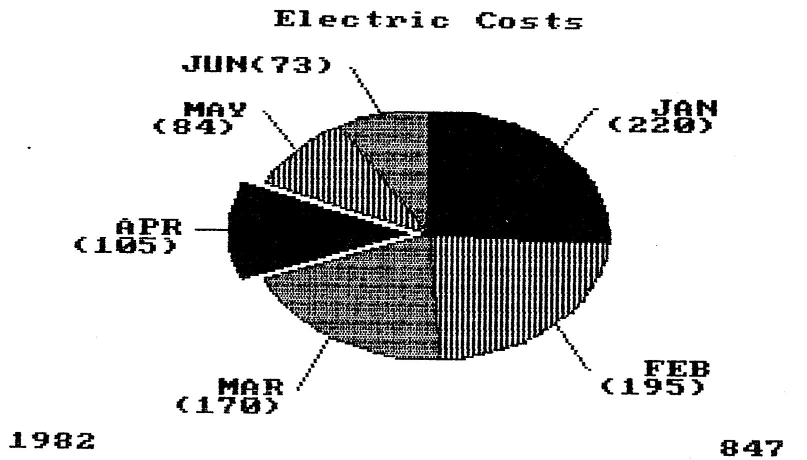
	A	B	C	D	E	F	G	H
1	Electric	JAN	FEB	MAR	APR	MAY	JUN	Total
2								
3	1980	223.00	172.00	150.00	92.00	70.00	61.00	768.00
4	1981	202.00	182.00	155.00	95.00	75.00	65.00	774.00
5	1982	220.00	195.00	170.00	105.00	84.00	73.00	847.00
6	------------	-------	-------	-------	-------	-------	-------	-------
7	Monthly	645.00	549.00	475.00	292.00	229.00	199.00	2389.00

036-008

Type:	Result:
1. >**E5**↵	Positions the cursor at the value to be emphasized.
2. **/*GP**	Tells the program that you want to create a pie graph. **Prompt Line:** Pie Graph: Specify range **Edit Line:** E5
3. **[BKSP]** ← ← ←	Repositions the cursor to cell B5. **Prompt Line:** Pie Graph: Specify range **Edit Line:** B5
4. **.**	Tells the program that you have verified that B5 is the upper-left coordinate. **Prompt Line:** Pie Graph: Specify range **Edit Line:** B5...

5. **G5**↵ — Tells the program that G5 is the lower-right coordinate of the range.

Your screen should display the following graph:



036-017

6. ↵ — Returns to the worksheet.

GRAPHING NEGATIVE VALUES

If the range of your graph includes negative values, the program terminates the option and displays the following message:

ERROR: Negative value(s)

If you are more interested in the relative sizes of the values than in their signs, use the @ABS function to make the values positive, and then graph again.

Scatter Graph

A *scatter graph* uses only two rows or columns. It computes the horizontal axis from the first row or column and the vertical axis from the second. It plots each pair of values using the value from the first row or column as the horizontal coordinate and the other value as the vertical. You use the Scatter Graph option (/*GS) to produce a scatter graph.

Scatter graphs are useful for demonstrating a correlation between ordered collections of values.

RANGE OF SCATTER GRAPHS

When you create a scatter graph, you must specify a range that lies within two rows or two columns. If your range does not meet this restriction, the program displays this message:

ERROR: Invalid number of rows or columns

and terminates the option.

If your range is precisely two coordinates by two coordinates, the program uses an additional prompt to find out whether you want to graph row-by-row or column-by-column. If your sets of data are in rows, type R; if they are in columns, type C.

If your range includes negative values, you could produce surprising results. Your graph will display negative values outside the axes of your graphs.

LABELING A SCATTER GRAPH

A scatter graph uses three labels — a main title, a title for the vertical axis, and one for the horizontal axis. The program takes these labels from the worksheet. The main title is always taken from position A1.

The source of the labels of the axes depends on whether your graph's range is rows or columns. If it is in rows, the labels are taken from column A. If the range is in columns, the labels are taken from row 1.

The format of the numbers displayed on the horizontal axis differs from that of numbers on the vertical axis. The values on the vertical are accurate to the nearest one-hundredth; the values on the horizontal are accurate only to the nearest unit.

EXAMPLE: CREATING A SCATTER GRAPH

This example shows how to create a scatter graph from the following worksheet. If you recreate the worksheet shown in the illustration and follow these steps, you will genterate a scatter graph.

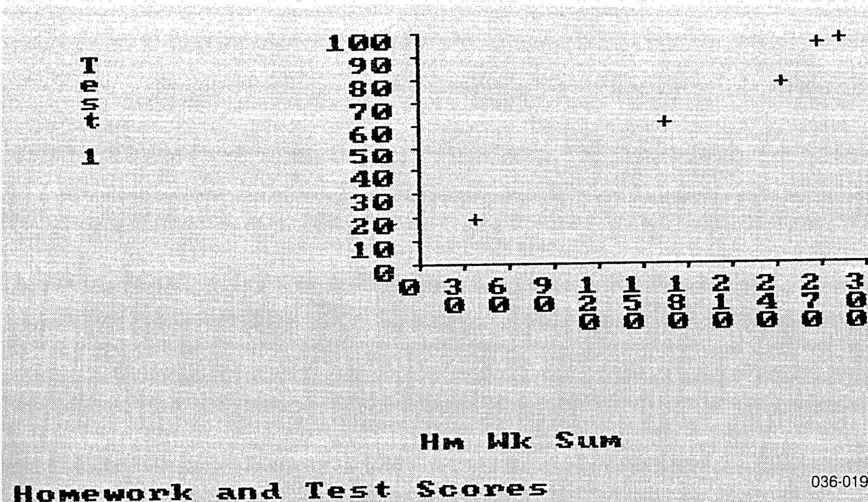
Note: When you are creating this worksheet, use the following information: (1) the entry in position A1 is "Homework and Test Scores"; and (2) the student's names should be formatted with /**FR**.

	A	B	C	D	E	F
1	Homework	Adams	Baker	Crawford	Dunning	Edwards
2	Hm Wk 1	91	82	91	21	53
3	Hm Wk 2	85	76	97	0	60
4	Hm Wk 3	93	85	96	17	52
5	Hm Wk Sum	269	243	284	38	165
6	Test 1	95	78	97	19	61

036-018

Type:	Result:
1. **/*GOM100**↵	Uses the Multiple feature of Star* Graph Options to ensure that the number at the top of the vertical axis is a multiple of 100.
2. **/*GS**	Tells the program that you want to create a scatter graph. **Prompt Line:** Scatter Graph: Specify range **Edit Line:** (displays the cursor's current location)
3. **[BKSP]**	Tells the program that you will explicitly type the upper-left coordinate of the range. **Prompt Line:** Scatter Graph: Specify range **Edit Line:** (clear)
4. **B5.**	Shows is the upper-left coordinate. **Prompt Line:** Scatter Graph: Specify range **Edit Line:** B5...

5. **F6**↵	Tells the program that the range is from B5 to F6. Your screen should display the following graph:



036-019

6. ↵	Returns to the worksheet.

Star* Graph Options (/*GO)

The Star* Graph Options (/*GO) control the appearance of graphs on your monitor and printer. After you type **/*GO**, you have the following options:

- Pressing **C** turns on and off an → to the left of Color on the prompt line.

  If you have a monochrome graphics monitor, this arrow must be off. With the arrow off, the program will send shaded rather than colored graphs to your monitor. If the arrow is on and your monitor is monochrome, the elements of your graphs will be indistinguishable.

  If you have a color monitor, and you prefer color graphs to shaded graphs, turn the arrow on. However, when you decide to print a graph, you should turn the arrow off and regenerate the graph before printing, because shaded graphs print more clearly than color graphs.

  When you load the program, graphs will appear shaded, unless you change the option.

- Pressing **G** turns on and off an → to the left of the word Grid on the prompt line. If the arrow is present, all displayed or printed graphs will have a dotted reference grid. This does not apply to pie graphs.

  When you load the program, graphs will appear without reference grids.

- Pressing **L** turns on and off an → to the left of the word Large on the prompt line.

  This controls the size of your printed graphs. If the arrow is absent, your printed graphs occupy about 1/4 of a sheet of 8 1/2-by-11-inch paper. If the arrow is present, each graph occupies the entire sheet. The size of the graph on your monitor is not affected.

  Large graphs do not fit on paper that is 8 1/2 inches wide unless you also use the Sideways option.

  When you load the program, graphs are printed at the normal size.

- Pressing **R** turns on and off an → to the left of the word Reverse on the prompt line. This controls the appearance of your printed graphs, but does not affect graphs displayed on your monitor.

  When the arrow is absent, the graph is printed with dark characters on a light background. In contrast, when the arrow is present, the graph is printed with light characters on a dark background. This is often called *reverse type*.

  By using this option, you can emphasize important graphs—but you will use a lot of ink.

  When you load the program, graphs print normally.

- Pressing **S** turns on and off an → to the left of the word Sideways on the prompt line. This controls the orientation of the graph on your printer. It does not affect graphs displayed on your monitor.

  The purpose of this option is to let you print large graphs even if your paper is only 8 1/2 inches wide. Without this option, large graphs cannot fit across the standard paper width.

  When you load the program, graphs print normally.

- Pressing **O** removes all the arrows on the prompt line. This option is particularly useful within Keysaver commands. It allows your Keysaver commands to cancel and then reset graphics options to a known state. This does not change the value of the scale multiple.

- Pressing **M** allows you to set a scale multiple for your graph. You must set the scale multiple option last.

After you have selected the options that you want, press ↵. This tells the program that you are finished.

SETTING THE SCALE MULTIPLE

When you type /*GOM, the program displays the following prompt:

Multiple: Specify scale multiple

Your response to this prompt governs the number at the top of the vertical axis for any graphs other than pie graphs. It also applies to the rightmost number on the horizontal axis for scatter graphs.

Note: The rest of this discussion refers only to vertical axes. However, the same discussion applies to the horizontal axis of scatter graphs.

Before placing numbers on an axis, the program must calculate the numbers to put there. It takes the largest value to be graphed, rounds it to the next highest whole number, and places it at the top of the axis. Then it labels the bottom of the axis as zero and spaces the intermediate values *evenly* between zero and the highest value.

If you are using graphing to answer "what if ..." questions, any numbers on the axis are probably fine. But suppose that you are preparing a report, and you want to make sure that all the numbers on the axis are whole numbers.

You can use the Multiple feature to control these numbers. For example, to make sure that each number on the axis is a whole number, set the scale multiple to 10. Or, to make sure that each number on the axis is a multiple of 10, set the scale multiple to 100.

In general, use these two steps to decide what your scale multiple should be:

1. Decide what you want the numbers on the axis to look like. Do you want them to be multiples of 1? Of 10? Of 100?
2. Multiply your answer by 10, and enter it as your scale multiple.

Suppose you want all the numbers on the axis to be multiples of 100. What should the scale multiple be? To find the answer, multiply 100 by 10. Your scale multiple should be 1000. To set the scale multiple to 1000, type:

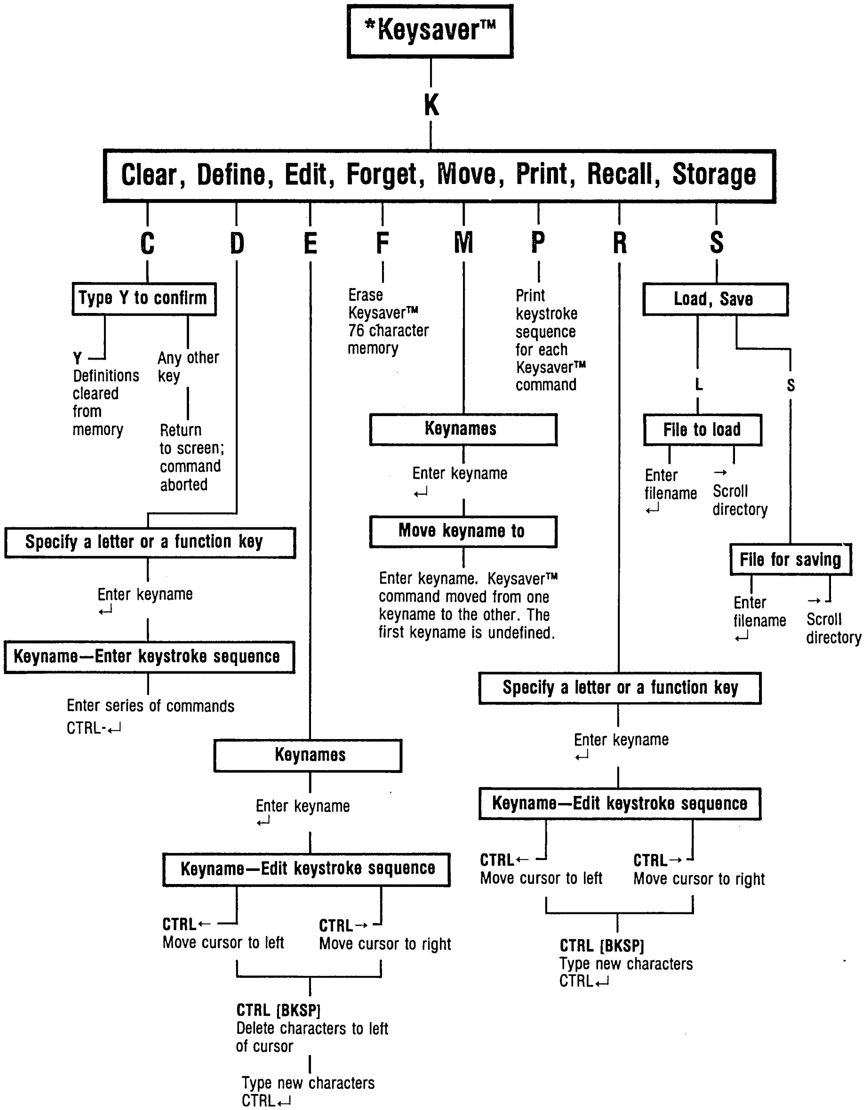
/*GOM1000↵

When you press ↵ after typing the scale multiple, the program accepts it and returns to the worksheet.

If the range of your graph contains negative values, the Multiple feature still controls the numbers displayed on the axis. The number at the top of the axis is greater than or equal to the largest positive value, and the number at the bottom of the axis is less than or equal to the smallest negative number. The numbers at both ends of the axis are multiples of the scale multiple.

The scale multiple is originally set to 1. If you change the scale multiple, it remains as you set it until you reset it or reload the program.

STAR* KEYSAVER MAINTENANCE COMMAND



The Star* Keysaver Maintenance command lets you create, edit, save, load, and print Keysaver commands. This command has the following options:

C The Clear option erases any Keysaver command definitions from memory.

D The Define option lets you create Keysaver commands.

E The Edit option lets you change the keystroke sequence for a Keysaver command.

F The Forget option erases the 75 most recent keystrokes from memory. It erases only the *memory* of the most recent keystrokes; it does not undo their effect on your worksheet.

M The Move option transfers the definition of a Keysaver command from one keyname to another, leaving the original keyname undefined.

P The Print option prints the definitions of all currently defined Keysaver commands on your printer.

R The Recall option displays the 75 most recent keystrokes on the edit line; it lets you edit them and create a Keysaver command from them.

S The Storage option saves and loads Keysaver commands.

The following sections describe each option in detail.

Keysaver Commands

A *Keysaver command* is a sequence of keystrokes that the VisiCalc IV program remembers for you. When you want to run one of these sequences, you tell the program which one to run and it types the sequence for you. The program allows you to have as many as 67 Keysaver commands defined or loaded in memory at any one time.

HOW ARE KEYSAVER COMMANDS NAMED?

Before you create a Keysaver command, you should know how to name it. This is important because there are two ways of invoking Keysaver commands, and the name you assign determines how you invoke it.

- Function-Key Keysaver Commands

  You can assign a Keysaver command to a function key—for example, [F10]. [F10] becomes the *keyname* of the keystroke sequence.

- Single-Letter Keysaver Commands

  You can also assign a Keysaver command to one of the 26 alphabetic keys—for example, A. A becomes the keyname of the keystroke sequence.

The function-key Keysaver commands are more numerous than the single-letter Keysaver commands. Although there are only 10 function keys on the keyboard, you can combine the function keys with the ALT, SHIFT, and CTRL keys for a total of 40 function-key Keysaver commands. On the other hand, you can have only 26 single-letter Keysaver commands.

Note: The 67th Keysaver command is a special command assigned to the key 0. For information on Keysaver command 0, see the "Automatically Loading a File of Keysaver Commands" section later in this chapter.

The following table shows the function keys that are available to be used as keynames and the notation used in this manual.

Function Keys	Notation Shown on Edit Line	Notation Used in Manual
F1 ... F10	F1 ... F10	**[F1]** to **[F10]**
ALT-F1 ... ALT-F10	A1 ... A10	**[A1]** to **[A10]**
CTRL-F1 ... CTRL-F10	C1 ... C10	**[C1]** to **[C10]**
SHIFT-F1 ... SHIFT-F10	S1 ... S10	**[S1]** to **[S10]**

The major difference between the two kinds of Keysaver commands is not their keynames; it is the way they are invoked.

HOW ARE KEYSAVER COMMANDS INVOKED?

The VisiCalc IV program provides two techniques for invoking Keysaver commands. One is for function-key Keysaver commands, and the other is for single-letter Keysaver commands.

To invoke a function-key Keysaver command, press the function key that names the command. For instance, to run the [F3] Keysaver command, press **[F3]**. To run the [S6] Keysaver command, press **[S6]**.

To invoke a single-letter Keysaver command, hold down the **ALT** key and press the key that names the Keysaver command. For instance, to run J, press **ALT-J**. To run M, press **ALT-M**.

You can use either kind of Keysaver command whenever you wish.

ACCURACY IN KEYSAVER COMMANDS

If you make a mistake while typing a Keysaver command, you could ruin some valuable data on your worksheet. Suppose, for example, that your Keysaver command includes an invalid command, such as /?. The program cannot detect this kind of mistake, so it continues processing the Keysaver command until all the keystrokes are finished. You must ensure that the keystroke sequence for your Keysaver command is correct.

The best way to avoid errors in your Keysaver commands is to manually test the sequence of keystrokes *before* it becomes a Keysaver command. When you are entering each keystroke of the sequence, you can detect errors as they happen. This way you can correct any errors and prevent a problem. Once you have verified that it does what you want it to, go ahead and create the Keysaver command.

The consequence of an error within a Keysaver command is not predictable. You should remember to be very careful when you define your Keysaver commands. Before you experiment with any Keysaver commands, you should save your worksheet. This way, if a Keysaver command has an undesired effect, you can reload the original worksheet and no damage will be done.

LENGTH OF KEYSAVER COMMANDS

You can have as many keystrokes in a Keysaver command as will fit on the edit line—a maximum of 76. However, because of the notation used for some keystrokes (for example, [F1] means function key 1), most Keysaver commands will contain fewer than 76 characters.

Sample Keysaver Commands

The VisiCalc IV program provides 18 sample Keysaver commands that you can use. These samples are supplied on the program disk in the file, KEYSAVER.SC.

USING THE SAMPLE COMMANDS

If you want to use the sample Keysaver commands, put your program disk in a drive. If your program disk is in drive A, type **/*KSLKEYSAVER.SC** and press ↵. If your program disk is in drive B, type **/*KSLB:KEYSAVER.SC** and press ↵. This loads the Keysaver commands into memory, allowing you to use them. For more detailed explanations of loading files of Keysaver commands into memory, see "Storage Option (/*KS)" later in this chapter.

The sample Keysaver commands are:

[F1]*coord*	GoTo command; moves cursor to *coord*.
[F2]	Defines entry as label.
[F3]	Displays value in integer format.
[F4]	Displays value in dollars-and-cents format.
[F5]	Displays entry left-aligned.
[F6]	Displays entry right-aligned.
[F7] to [F10]	Moves cursor up, down, left, or right, respectively. Frees the numeric keypad for entering numbers.
ALT-[F7]	Moves cursor up ten cells.
ALT-[F8]	Moves cursor down ten cells.
ALT-[F9]	Moves cursor left four cells.
ALT-[F10]	Moves cursor right four cells.
ALT-P	Prints normally.
ALT-C	Prints compressed.
ALT-D	Prints double strike.
ALT-B	Prints both double strike and compressed.

The sample printing commands apply to the IBM and equivalent printers only.

Clear Option (/*KC)

The Clear option deletes the definitions of all currently defined Keysaver commands from memory. It does not delete any commands you have stored on a disk. This is very useful if you frequently load Keysaver commands from files on a disk.

Without this option, newly loaded Keysaver commands would always be merged with existing Keysaver commands. By using the Clear option, you can erase existing definitions before loading new ones.

After you type **/*KC**, the program asks you to verify that you want to clear the memory. If you press **Y**, it deletes the commands and displays the message:

```
Keysaver Clear complete
```

If you type anything other than Y, the program leaves the Keysaver commands unchanged.

The Clear option can be very useful. Suppose that you have a worksheet that prompts users to press one of several function keys. When a user presses a function key, the program executes a Keysaver command. You should ensure that only the function keys you want them to use are defined. This prevents users from accidentally running the wrong Keysaver command. The easiest way to ensure this is to type **/*KCY** and press ↵ before loading the correct Keysaver commands.

The Clear option can be called from within a Keysaver command. It automatically detects which Keysaver commands are running, and it leaves their definitions intact.

If you invoke one Keysaver command from another, both can be running at the same time. For example, if [F1] invokes A, and A uses the Clear option, neither Keysaver command will be deleted. The Clear option detects that both [F1] and A are running, so it leaves their definitions intact.

In contrast, if [F1] uses the Clear option first and then invokes A, the computer beeps indicating that A is undefined. In this example, the Clear option erased the definition of A because A was not running when the Clear option was invoked.

Define Option (/*KD)

The Define option allows you to specify a sequence of keystrokes and assign the keystrokes to a keyname. This sequence of keystrokes is called a *Keysaver command*.

Later, when you want to use the sequence of keystrokes, you type only the keyname. The program then enters the keystroke sequence for you as if you had typed it from the keyboard.

Note: After you become more experienced with the program, read about the Recall option (/*KR). It also creates Keysaver commands, but it requires fewer keystrokes.

EXAMPLE OF A KEYSAVER COMMAND

Suppose that you frequently use a worksheet that has too many columns to fit on your printer. If your printer can print in *compressed mode*, you can fit more information on a sheet of paper. The Print command (/P) lets you enter a *control code* that tells the printer to print in compressed mode. (For detailed information on printing with control codes, see Appendix A.)

Using the Print command, however, you must type the printer control code each time you print. This is fine if you use compressed mode infrequently. The Define option allows you to create a Keysaver command that includes the printer control codes that you could use every time you print.

The characters *∧H0F* make up the control code for printing in compressed mode on an IBM printer. The following example assigns these keystrokes to [F1]. If you make any typing errors, see the next section, "Editing a Keystroke Sequence."

Type:	Result:
1. **/*KD**	**Prompt Line:** Define: Specify a letter or a function key
2. **[F1]**	**Prompt Line:** Define: F1 – Enter keystroke sequence The program is asking you to enter the keystrokes to be run whenever you press [F1].
3. **/PP"∧H0F↵**	This is the entire Print command except for the lower-right coordinate. The ↵ appears as a ◄ on the screen.
4. **CTRL-↵**	By simultaneously pressing the CTRL and ↵ keys, you inform the program that you have finished typing the sequence of keystrokes. Prompt Line: (clear) You can now enter the next command.

Whenever you want to print a worksheet in compressed mode, press **[F1]**, type the lower-right coordinate of the area being printed, and press ↵. The program will print your worksheet in condensed mode.

This example illustrates two important points about the Define option:

- The program does not run the Keysaver command when you type it in. It is run only when you invoke it.
- You cannot use ↵ alone to mark the end of Keysaver command. Instead, you must use a combination of the CTRL and ↵ keys. If you use the ↵ key alone, the program assumes that ↵ is part of the Keysaver command, and waits for more keystrokes.

EDITING A KEYSTROKE SEQUENCE

If you make a mistake while entering a Keysaver command, you can fix it without retyping the entire command. To do this, move the edit cue to the character after the mistake. The following keys move the edit cue within the Define option:

CTRL-→	Moves the edit cue to the right.
CTRL-←	Moves the edit cue to the left.
CTRL-HOME	Moves the edit cue to the beginning of the edit line.
CTRL-END	Moves the edit cue to the end of the edit line.

When the edit cue is correctly positioned, press **CTRL-[BKSP]** to delete the character(s) to the left of the edit cue. Then you can type the correct characters. They will also appear to the left of the edit cue.

SAVING KEYSAVER COMMANDS ON DISK

Once you've defined a Keysaver command, you can save it on a disk. You can load the Keysaver command from the disk at any time, instead of typing it again. For more information about saving and loading Keysaver commands, see the "Storage Option (/*KS)" section later in this chapter.

REDEFINING A KEYSAVER COMMAND

You can redefine existing Keysaver commands using the Define option. When you type **/*KD**, the program asks you to verify that you are not *accidentally* redefining the Keysaver command. The edit line displays the current definition of the Keysaver command, and the prompt line displays the following message:

```
Keysaver Command "n" exists. Type Y to redefine.
```

If you press **Y**, the program deletes the old definition and prompts you for the new keystroke sequence. If you press anything other than Y, the program leaves the current definition unchanged.

You can use the same keystrokes to delete the definition of a Keysaver command. First, type **/*KD** followed by the keyname. Then press **Y** in response to the prompt. Finally, press **CTRL-↵**. This erases the Keysaver command, leaving it undefined.

Note: This does not affect any Keysaver commands you have saved on a disk.

KEYSAVER COMMANDS WITHIN KEYSAVER COMMANDS

You can include a previously defined Keysaver command within the keystroke sequence of another Keysaver command. This provides you with three benefits:

- You save time entering the command.
- You reduce the likelihood of typing mistakes.
- You can create several small Keysaver commands and combine them into one large Keysaver command that would be too long otherwise.

Although you can include Keysaver commands within definitions of other Keysaver commands, you cannot include a Keysaver command within its own definition.

Warning: You should not use any of the following options within a Keysaver command:

Keysaver Define	/*KD
Keysaver Edit	/*KE
Keysaver Move	/*KM
Keysaver Recall	/*KR

Doing so could damage your worksheet.

Edit Option (/*KE)

The Edit option allows you to change the keystroke sequence for an existing Keysaver command. For example, suppose [F1] generates the following keystrokes: /PP"∧H0F↵, and you want to change that to another set of similar keystrokes. The following example shows how to use the Edit option to change the keystroke sequence for [F1] to /PP"∧EE↵.

Type:	Result:
1. **/*KE**	Tells the program that you want to edit a Keysaver command. **Prompt Line:** Edit: F1 If any other Keysaver commands are defined, their keynames will appear on the prompt line.
2. **[F1]**	Tells the program that you want to edit the [F1] Keysaver command. **Prompt Line:** Edit: F1 — Edit keystroke sequence **Edit Line:** /PP"∧H0F ◄ The edit line shows the current keystroke sequence of the [F1] Keysaver command with the edit cue at beginning of the sequence.
3. **CTRL -→** **CTRL -→** **CTRL -→** **CTRL -→** **CTRL -→** **CTRL -→** **CTRL -→** **CTRL -→**	Moves the edit cue eight positions right, to the ◄.
4. **CTRL—[BKSP]** **CTRL—[BKSP]** **CTRL—[BKSP]**	Erases three characters to the left of the edit cue. **Edit Line:** /PP"∧◄ The cursor is still on the ◄.

5. **EE** — Type EE just to the left of the cursor.

Edit Line: /PP"∧EE ◄

6. **CTRL-↵** — Tells the program that you are finished editing the Keysaver command.

Edit Line: (clear)

After you type **/*KE**, the prompt line displays the names of the currently defined Keysaver commands. If there are too many Keysaver commands to fit on the prompt line, the program displays an ellipsis (...) at the end of the line. This means that more commands are defined than are shown.

If this happens, and you want to know the names of the commands that are not displayed, you can print your Keysaver commands. See the "Print Option (/*KP)" section later in this chapter for details.

The following keys can be used in the Edit option:

- CTRL-[BKSP] deletes the character immediately left of the edit cue.
- CTRL-← moves the edit cue one position to the left.
- CTRL-→ moves the edit cue one position to the right.
- CTRL-HOME moves the edit cue to the beginning of the edit line.
- CTRL-END moves the edit cue to the end of the edit line.

Forget Option (/*KF)

The Forget option is provided by the VisiCalc IV program as a convenience to you. It simplifies the process of using the Recall option (/*KR), and it is useful only if you intend to use the Recall option within the next 75 keystrokes. If you never use the Recall option, you do not need the Forget option.

The remainder of this explanation assumes that you are familiar with the Recall option. If you are not, see "Recall Option" later in this chapter.

The Forget option erases the last 75 keystrokes that you typed from memory. This *forgetfulness* on demand simplifies using the Recall option.

For example, suppose that you want to create a Keysaver command. Before you actually create the command, you should enter the keystrokes manually to verify that the sequence is correct. The program remembers the sequence for you in its 75-keystroke memory, allowing you to use the Recall option to create your Keysaver command.

However, if your keystroke sequence contains fewer than 75 keystrokes, the memory may contain unwanted keystrokes that preceded the desired sequence. When you use the Recall option, you can edit these surplus keystrokes—but there is an easier way to eliminate them.

Before you start to verify your keystroke sequence, use the Forget option to erase the 75-keystroke memory. Later, when you use the Recall option to create your Keysaver command, it will require less effort.

To use the Forget option, type

/*KF

This erases the 75-keystroke memory. The program does not issue any messages; it waits for your next instruction.

Move Option (/*KM)

The Move option transfers a keystroke sequence from one keyname to another. For example, you can move the keystrokes from [F1] to A. The following example shows how to do this:

Type:	Result:
1. **/*KM**	**Prompt Line:** Keysaver Move: F1 The prompt line shows the names of the Keysaver commands that are currently defined.
2. **[F1]**	**Prompt Line:** Keysaver Move: Move F1 To The program is asking you where you want the sequence moved.

3. **A**

Prompt Line: Keysaver Move complete

The program moves the keystroke sequence from [F1] to A. [F1] is now undefined.

You may now enter another command.

After you type **/*KM**, the prompt line displays the names of the currently defined Keysaver commands. If there are too many Keysaver commands to fit on the prompt line, the program displays an ellipsis (...) at the end of the line. This means that more commands are defined than are shown.

If this happens, and you want to know the names of the commands that are not displayed, you can print your Keysaver commands. See the "Print Option (/*KP)" section later in this chapter for instructions.

The Move option checks for four possible problems:

- Are any Keysaver commands defined? If not, the program displays the following message:

  No Keysaver commands defined

  and aborts the command.
- Is a Keysaver command associated with the first keyname? If not, the computer beeps displays the following message:

  Keysaver command not defined

  and aborts the option.
- Does the second keyname already have an assigned Keysaver command? If so, the program displays the following message:

  Keysaver command "keyname" exists. Type Y to redefine.

  If you still want to move the Keysaver command to the keyname, press **Y**. If you press anything else, the program aborts the option.
- Are both the first and second keynames valid for Keysaver commands? If not, the computer beeps and aborts the option without displaying a message.

Print Option (/*KP)

When you type **/*KP**, the program prints the keystroke sequence for each currently defined Keysaver command. If no Keysaver commands are defined, the program displays the following message:

No Keysaver commands defined

and aborts the option.

Note: Make sure your printer is properly connected *before* using this option. If your printer is turned off or improperly connected, the behavior of the program is unpredictable.

The following illustration is an example of a printout created using the Print option.

```
Keysaver Command Definitions

B        /pp"^hOf^eG[Enter]
C        /pp"^hOf[Enter]
D        /pp"^EG[Enter]
P        /pp
[F1]     >
[F2]     "
[F3]     /fi
[F4]     /f$
[F5]     /fl
[F6]     /fr
[F7]     [Up]
[F8]     [Down]
[F9]     [Left]
[F10]    [Right]
[A7]     [Up][Up][Up][Up][Up][Up][Up][Up][Up][Up]
[A8]     [Down][Down][Down][Down][Down][Down][Down][Down][Down][Down]
[A9]     [Left][Left][Left][Left]
[A10]    [Right][Right][Right][Right]
```

Recall Option (/*KR)

The Recall option allows you to create a Keysaver command from keystrokes that you have recently typed. This process can save you time and keystrokes. If you are not already familiar with the Define option (/*KD) or the Edit option (/*KE), see the "Define Option (/*KD)" and the "Edit Option (/*KE)" sections earlier in this chapter.

To use the Recall option, type:

/*KR

The program asks you for a keyname. If the keyname already has a Keysaver command assigned to it, the program asks you to verify that you want to redefine it.

The program then displays your 75 most recent keystrokes on the edit line, and it lets you use the editing features of the Edit option (/*KE) to change the sequence of keystrokes.

After you have finished editing the sequence, hold down the **CTRL** key and press ↵. This tells the program that you have finished editing. The Recall option then creates your Keysaver command.

The program can remember as many as 75 of the most recently typed keystrokes. Sometimes, however, the program forgets keystrokes. This occurs:

- When you use the Forget option (/*KF).
- After you use the Define (/*KD), Edit (/*KE), or Recall (/*KR) options.

The 75-keystroke memory does not contain keystrokes entered on behalf of a Keysaver command. For example, if you press [F1], the memory remembers only [F1], not the entire Keysaver command associated with [F1].

Storage Option (/*KS)

Once you have defined your Keysaver commands, the Storage option (/*KS) lets you write the commands into a file on a disk. Later, you can read your Keysaver commands from the disk into memory.

The Storage option provides two commands. The Load command (/*KSL) reads definitions of Keysaver commands from a disk into the computer. The Save command (/*KSS) records on a disk any Keysaver commands that are currently defined in the computer's memory.

LOAD COMMAND (/*KSL)

When you use the Load command, the program asks you for the name of a file and then reads all of the Keysaver commands from the file into the computer's memory. *This file is not the file that stores your worksheet.* It contains only Keysaver commands.

The program asks for the name of the file by displaying the following prompt:

Keysaver Storage: File to load

If you have defined any Keysaver commands and not saved them, the program prompts you to do so. If you want to save your Keysaver commands, press a key other than Y. If you want to continue with loading, press **Y**.

If the commands in memory have names different from the commands being loaded, the loaded commands are added to the commands in memory. None of the existing commands are redefined.

If some of the commands in memory have the same names as some of the commands being loaded, the existing commands will be redefined. However, the existing commands with unique names remain as they were before the new commands were loaded.

You can respond to the prompt File to load in two ways:

- You can enter the name of the file containing the Keysaver commands. For example:

  B:BUDGET↵ or **FORECAST↵**

  You can load your commands from any disk drive attached to your system. When your file name has no drive indicator, the program automatically uses the default drive. Generally, the default is drive A.

  Although the two sample file names are B:BUDGET and FORECAST, the program loads the file named BUDGET.SC or FORECAST.SC. Whenever you enter a file name with no file name extension, the program automatically adds the extension .SC.

- You can scroll through file names using →. When you scroll, the program does not display the names of all files on the disk. It displays only the names of files with a .SC extension. Press ↵ when the correct file name appears.

Before executing this command, the program checks for the following things:

- Is the file on the disk? If the program cannot find the file, it displays the following message:

  ERROR: File not found

  This tells you the Load command has been aborted, and that you can enter another command.
- Does the file contain Keysaver commands? If the file contains any information other than Keysaver commands, the program displays the following message:

  ERROR: Bad Keysaver command file

  This tells you the Load command has been aborted, but you can enter a new command.
- Are you trying to load Keysaver commands from a drive that is not attached to the system? If so, the program aborts the option and displays the following prompt:

  ERROR: File not found

  When you see this message, you can enter another command.

The Load command allows you to work around the limit of having only 67 Keysaver commands at any one time. You can use one of the 67 Keysaver commands to load another file of Keysaver commands. If you use this technique, however, you cannot load a new definition for any Keysaver command that is running when the Load command is invoked.

For example, suppose that Keysaver command [F1] loads a file of Keysaver commands including a new definition of [F1]. The old definition of [F1] will remain unchanged because the program prohibits redefining Keysaver commands that are running.

SAVE COMMAND (/*KSS)

When you use the Save command, the program asks for the name of a file and then saves all of the Keysaver commands currently in memory in that file. *This is not the same file that contains your worksheet.* It is a separate file that contains only Keysaver commands.

The program asks for the name of the file by displaying the following prompt:

Keysaver Storage: File for saving

If there are no Keysaver commands currently defined, the program displays the following message:

ERROR: No commands to save

and aborts the Save command. Press ↵. You can now enter another command.

You can respond to the File for saving prompt in either of two ways:

- You can enter the name of the file in which you want to store the Keysaver commands. For example:

  B:BUDGET↵ or **FORECAST↵**

  You can save your file on a disk in any drive attached to your system. When your file name has no drive indicator, the program automatically uses the default drive.

  Although the two sample file names are B:BUDGET and FORECAST, the program saves the file named BUDGET.SC or FORECAST.SC. Whenever you enter a file name with no extension, the program automatically adds the extension .SC.
- You can scroll through existing file names using →. When you scroll, the program does not display the names of all files on the disk. It displays only the names of files with a .SC extension. Press ↵ when the correct file name appears.

Before executing this command, the program checks for the following things:

- Does the drive contain a formatted disk having sufficient room to store all the currently defined Keysaver commands? If not, the program displays the following message:

  ERROR: Save aborted

  and aborts the Save command. You can now enter another command.
- Does the file already exist? If the disk already contains a file of the same name, the program warns you that saving the commands in that file will erase any information already in the file. The program displays the following prompt:

  File exists. Type Y to replace

  If you don't care about the current contents of the file, press **Y** and the program will save it. If you want to preserve the current contents of the file, press any key other than Y and the program aborts the Save command.

- Are you trying to save the Keysaver commands on a disk drive that is not attached to your system? If so, the program aborts the command and displays this message:

  ERROR: Save aborted

  When you see this message, press ↵. You can enter another command.

After the program saves your Keysaver commands on disk, you can use the Load command (/*KSL) to reload them whenever you wish.

Note: Saving Keysaver commands does not remove them from the computer's memory. You can continue to use the commands even after you have saved them.

DELETING A FILE

If you want to delete a file containing Keysaver commands, use the Delete option (/SD) of the Storage command. See "Storage Command" in Chapter 3 for details.

When you use the Storage command to delete a file containing Keysaver commands, be sure to explicitly type the extension of the file that you want to delete. If you do not type the .SC extension, the program will incorrectly assume that you want to delete the file of the same name that has a .VC extension. To reduce the likelihood of deleting the wrong file, adopt the practice of typing the extension whenever you delete a file.

EXAMPLES OF LOADING AND SAVING

The following two examples show how to load and save Keysaver commands.

Loading Example

Suppose that three Keysaver commands (A, B, and C) are already defined, and that you need two other Keysaver commands (D and a different version of C) that are contained in a file named PROJECT.SC. To load the two new commands from a disk in drive B, use the following steps:

Type:	Result:
1. /*KS	**Prompt Line:** Keysaver Storage: Load, Save
2. L	**Prompt Line:** Warning: Some commands already defined. Type Y to confirm. The program noticed that some commands are already defined. It is asking if you want to load the commands from disk anyway.
3. Y	Tells the program to load the commands from disk regardless of existing commands with duplicate names. **Prompt Line:** Keysaver Storage: File to load
4. B:PROJECT↵	**Edit Line:** B:PROJECT **Prompt Line:** Loading commands The program loads the file PROJECT.SC. **Prompt Line:** Commands loaded

The A, B, C, and D Keysaver commands are now in memory. The old definition of the C command has been replaced with the new definition. You can now use any of these Keysaver commands.

Saving Example

Now suppose that you have just defined two new Keysaver commands (X and [F3]), and that you want to save them in a file named PROFIT.SC on a disk in drive A. Perform the following steps:

Type:	Result:
1. **/*KS**	**Prompt Line:** Keysaver Storage: Load, Save
2. **S**	**Prompt Line:** Keysaver Storage: File for saving
3. **A:PROFIT**↵	**Prompt Line:** Storing commands **Edit Line:** A:PROFIT The program saves the commands in file A:PROFIT.SC. **Prompt Line:** Commands stored

The two new Keysaver commands (X and [F3]) have been saved in the PROFIT.SC file along with any other Keysaver commands that are currently defined in the memory of the computer.

AUTOMATICALLY LOADING A FILE OF KEYSAVER COMMANDS

Note: You must have two disk drives to load a Keysaver command file automatically.

You may find some Keysaver commands so useful that you want these commands in memory whenever the VisiCalc IV program is running. If you put your favorite commands in one file, you can make the program automatically load the file for you each time you load the program. In addition, you can have the program execute one of the Keysaver commands in the file as soon as loading is complete.

Before you can load the Keysaver command file, however, you have to save it on a disk. To do this:

1. Define or load the Keysaver commands that you want to load automatically.
2. Save the Keysaver commands on a data disk. If your system has a hard disk, save the file on the hard disk in the same directory as the VisiCalc IV program. *Make sure you remember the name of the file.*

How you automatically load the Keysaver command file depends on what equipment you have in your system. The sections that follow describe the necessary procedures in detail.

Loading the File from Drive A

Follow these instructions if you have two floppy drives and drive A is a double-sided drive. If drive A is a single-sided drive, skip to "Loading the File from Drive B." If your system has a hard disk, skip to "Loading the File from the Hard Disk."

To make the Keysaver command file load automatically, you have to reconfigure your program just as you did when you first loaded the program. To do this:

1. With A> displayed, put your program disk in drive A.
2. Type **vconfig** and press ↵.
3. Follow the instructions to remove the write-protect tab and replace the program disk in drive A. When you are finished, press ↵.
4. Type **3** or **4** depending on which version of the program you use.
5. Type the name of the file that contains the Keysaver commands you want loaded. *Remember to include the drive, B:, in the file name.* Press ↵.
6. Remove the program disk from drive A and replace the write-protect tab.

You can now load your program and the Keysaver command file automatically. To do this:

1. With A> displayed, put your program disk in drive A and the data disk with the Keysaver command file in drive B.
2. Press **CTRL**, **ALT**, and **DEL** all at the same time.

The program and file load automatically.

Loading the File from Drive B

Follow these instructions if you have two floppy drives and drive A is a single-sided drive. To load the program and the file:

1. With A> displayed, put the data disk in drive A and the program disk in drive B.
2. Type **b:** and press ↵.

3. Type **vc128** or **vc192** depending on which program you use, and press the space bar.
4. Type the name of the Keysaver command file. *Remember to include the drive prefix, A:, in the file name*. Press ↵.

The program and file load automatically.

Loading the File from the Hard Disk

Follow these instructions if you have a hard disk. To load the program and the file:

1. With C> displayed, type **vc128** or **vc192** depending on which program you use, and press the space bar.
2. Type the name of the Keysaver command file and press ↵.

If you installed the program in its own directory, each time you load the program you must first use the CHDIR command to access the VisiCalc IV directory; then type the correct loading instruction and press ↵.

The program and file load automatically.

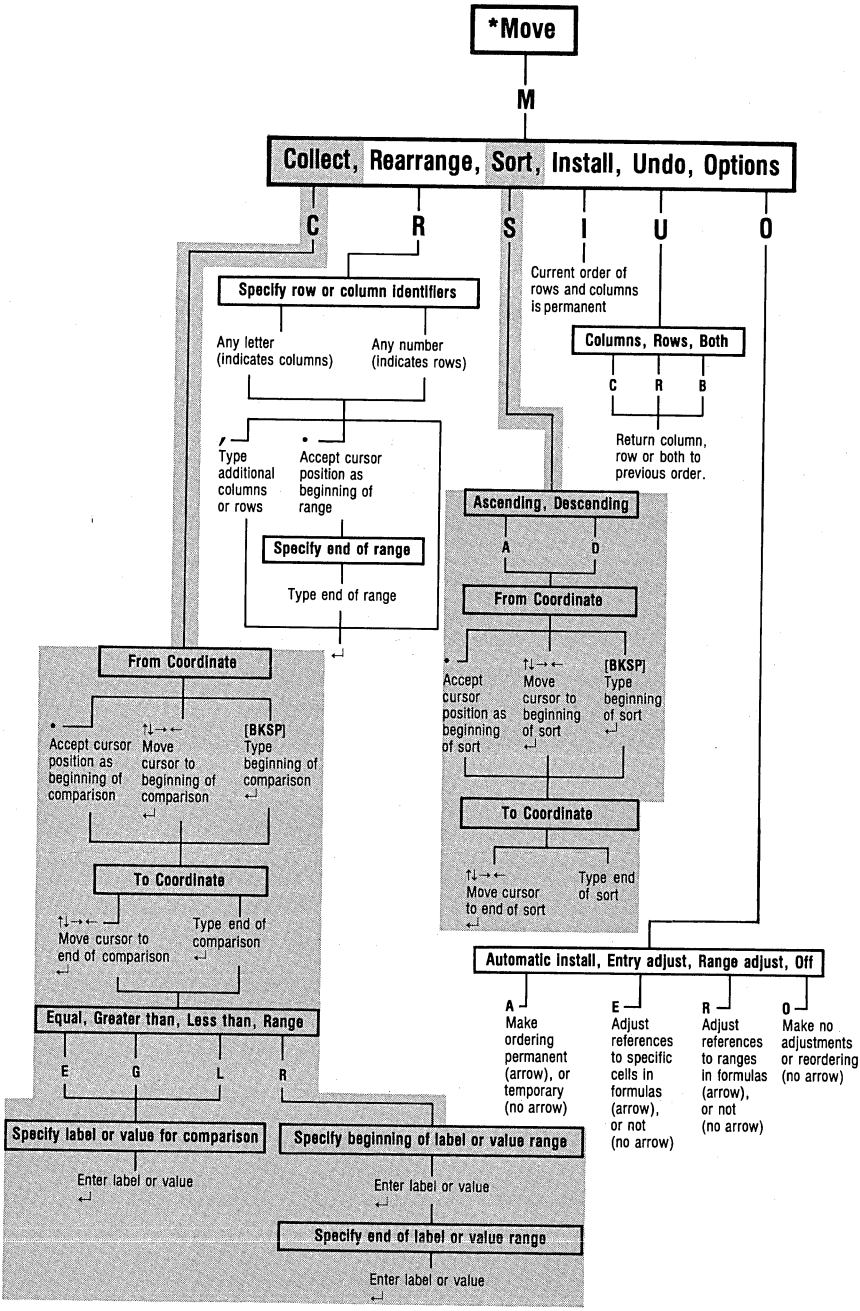
EXECUTING A KEYSAVER COMMAND AUTOMATICALLY

In addition to having a Keysaver command file that loads automatically, you can assign one Keysaver command that is executed every time the program is loaded. All you have to do is define the Keysaver command and assign it to the keyname 0 (zero) just as you would create any Keysaver command. Then include it in the Keysaver command file that loads automatically. When you load the program, Keysaver command 0 will be executed automatically.

Keysaver command 0 is the only Keysaver command that can be invoked automatically, but you can run other Keysaver commands from it. This means that Keysaver command 0 can invoke any currently defined Keysaver command.

You can invoke Keysaver command 0 manually using the same technique as is used for single-letter Keysaver commands. To do this, press **ALT-0**.

STAR* MOVE COMMAND



The Star* Move command allows you to quickly retrieve rows, sort rows, rearrange columns, or return your worksheet to its original order.

It has the following options:

C The Collect option examines the entries in one column of the worksheet, comparing each to criteria you supply. The option then collects all rows meeting your criteria.

R The Rearrange option moves one or more rows to the top of the worksheet, or moves one or more columns to the left side of the worksheet.

S The Sort option reorders the rows of the worksheet according to the values in a column of your choice. You can sort labels or values in ascending or descending order.

I The Install option changes the worksheet displayed on the screen from temporary to permanent.

U The Undo option returns the rows, columns, or both to the order they were in before you used the Collect, Rearrange, or Sort options, if you are using a temporary worksheet.

O Star* Move Options (/*MO) allow you to specify all movement as temporary or permanent and to specify how formulas are adjusted.

Temporary vs Permanent Movement

The Star* Move command is very powerful. It allows you to quickly change the appearance of your worksheet by sorting rows, collecting rows that meet your criteria, or explicitly rearranging rows or columns. Because these commands are so powerful, you must decide whether you want their consequences to be *temporary* or *permanent*. These words have special meanings to the Star* Move command.

To remind you of whether your changes are temporary or permanent, the program displays a message on the right side of the prompt line. If the program is set for permanent changes, the right side of the prompt line displays Changes Permanent. If the program is set for temporary changes, the prompt line displays Changes Temporary.

TEMPORARY MOVEMENT

When you load the program, it will make permanent changes to your worksheet. You can, however, tell the program to reorder the rows and columns of the worksheet temporarily. To do this, use Star* Move Options (/*MO) and turn off automatic installation. This means that the Star* Move command produces a temporary worksheet containing the reordered information and leaves your original worksheet intact.

You can quickly return a temporary worksheet to its original order. For example, if you rearrange a worksheet, you can look at the results and, using the Undo option (/*MU), return the worksheet to its previous order.

While your worksheet is temporary, you can use the GoTo command, the cursor control keys, the Print command or any Star* command. For example, you can temporarily rearrange a worksheet and print the rearranged worksheet on your printer.

If you try to do anything other than move the cursor, print on printer, or use a Star* command, the temporary worksheet returns to its original order. This includes attempting to change the contents of any coordinate.

If you like the results, you can change the worksheet from temporary to permanent. To do this, type **/*MI** (the Install option) and the worksheet becomes permanent. Once you've made the worksheet permanent, however, you cannot go back to the original unless you can reload it.

PERMANENT MOVEMENT

When you load the VisiCalc IV program, it automatically makes the results of the Star* Move command permanent.

Note: If you don't want to make permanent changes, use Star* Move Options to make the changes temporary.

Making changes permanent means:

- You cannot use the Undo option (/*MU) to return to the original worksheet. If you want to return to the original order, you must reload the worksheet or recreate it.
- You can use any command or option on a permanent worksheet, except the Install option or the Undo option.
- The program adjusts any affected formulas according to Star* Move Options settings. If they are set for entry or range adjustment, the program modifies all formulas after any Star* Move command option is used.

Note: If you decide to use permanent movement rather than temporary, be sure to save your worksheet before using the Star* Move command. This can save you hours of rework if you make a mistake.

Repeating Commands Quickly

The VisiCalc IV program provides a shortcut method for repeating the same Star* Move command option. To do this, type **/*M**. When the program displays a prompt, you have three choices:

- You can answer the prompt as you normally would, and the program will then display the next prompt.
- You can press PGDN and the program will answer the prompt for you, using the answer that you provided the last time you used the Star* Move command. The → on the prompt line indicates what your answer was.
- You can press ↵ and the program will answer remaining prompts for you by using the answers you supplied when you last used the Star* Move command. If you press ↵ in response to Specify range, the program accepts only the displayed range.

This quick-repetition feature is most useful for sorting and collecting. You can collect rows repeatedly using different criteria or a different column as the basis for collecting. This allows you to collect all the rows that meet several criteria.

Similarly, you can sort rows repeatedly, changing only the column on which you base the sorting. For example, you could use this technique to sort first by name and then by ZIP code.

Collect Option (/*MC)

The Collect option examines adjoining rows within a range, tests each to see if it meets your criteria, and moves the rows that satisfy the test to the top of the collection range. In effect, this option converts your worksheet to a visual *data base*. When you consider a worksheet as a data base, the rows become *records*, and the entries in the rows become *fields*.

You can use the Collect option whenever you need to find rows that meet specific criteria. For example, suppose that you are a sales manager, and you have a worksheet containing one row for each salesperson. If one column in each row contains the volume (in dollars) of merchandise they sold, you can use the Collect option to find all the salespeople whose sales exceed one million dollars.

USING THE COLLECT OPTION

When you use the Collect option, you must provide the program with a *search range* and the *criteria* for testing the search range.

The search range is the column of data that the program examines. In the previous example, the search range is the column containing the volume of merchandise for each salesperson. The range can be from the first row to the last row in the column, or it can begin and end with the same row. However, the program examines only entries that lie within one column.

The criteria you choose to test against can be from one of four categories:

- Equal to a specific label or value.
- Greater than a specific label or value.
- Less than a specific label or value.
- Between two specific labels or values.

To use the Collect option, type **/*MC**. The program displays the following prompt:

Collect: Specify range

Now you must enter the search range. Remember that the coordinates you enter must be in the same column, and that the first coordinate must be closer to the top of the worksheet than the last coordinate. To do this, type the first coordinate, a period (.), and the last coordinate. For example, K14.K93 is a valid search range, but J14.K93 is not. When you have entered the search range, press ↵.

After prompting you for the search range, the program displays the next prompt:

Collect Criteria: Equal, Greater than, Less than, Range

Your response determines the kind of test that the program uses. You have four choices:

- If you press **E**, the program asks you for a label or value. It tests each entry in the search range to see if any of them are equal to the label or value you entered. The program moves any rows that pass the test to the top of the search range.

- If you press **G**, the program asks for a label or value. It tests each entry in the search range to see if any of them are greater than the label or value you entered. The program moves any rows that pass the test to the top of the search range.
- If you press **L**, the program asks for a label or value. It tests each entry in the search range to see if any of them are less than the label or value you entered. The program moves any rows that pass the test to the top of the search range.
- If you press **R**, the program asks for *two* labels or *two* values. It tests each entry in the search range to see if any of them lie between the two labels or values. Specifically, it checks to see if the contents of each entry position are greater than or equal to your first response and less than or equal to your second response. The program moves any rows that pass the test to the top of the search range.

DIFFERENCES BETWEEN LABELS AND VALUES

When you enter a label or value as your criterion, be sure that the program can tell that you have entered a label or a value.

If you accidentally use values when you mean to use labels, your results can be very disappointing. For example, if your test criterion is the label 456, be sure to use "456 when you enter the label. If you forget the quote, the program interprets your entry as a value and will not find the correct rows.

If you are collecting using a criterion that is a range rather than a single entry, the program tries to save you a keystroke. If your response to the prompt is a label, the program assumes that the end of the range is a label also. Do not use the quote (") in your second response.

If you are collecting labels, the program sorts the labels in the following order:

(1) The space character
(2) The special characters ! " # $ % & ( ) * + , - . /
(3) The digits 0 through 9
(4) The special characters : ; < = > ? @
(5) The uppercase alphabet
(6) The special characters [\ ] ^ _ `
(7) The lowercase alphabet
(8) The special characters { | } ~

If you are collecting rows using values as the test criteria, you can enter numbers only. You cannot enter a formula that refers to a value, and you cannot enter an expression that is the product, sum, difference, or quotient of any two numbers.

For example, you can enter any of the following values:

```
  100
 +100
-100.003
```

But you cannot enter any of the following formulas or expressions:

```
 +A1
100+D3
 72*6
```

Regardless of whether an entry is a label or a value, collecting is based on the full value or label. This can lead to some surprising results. For instance, suppose you have these entries:

Entry	Format	Value	Display
1	Dollars-and-cents	3.25	3.25
2	Integer	3.41	3

If you use the Collect option with the criterion being "greater than 3.3," the row containing entry 2 passes the test and is moved upward in the worksheet. In contrast, the row containing entry 1 does not pass the test and is not moved. This happens because collection is based on the coordinate's true value rather than the displayed value.

EXAMPLE OF COLLECTING ROWS

The following example shows what you have to type to collect the rows for the salespeople whose sales exceed one million dollars. In this example, the entries for the salespeople begin at row 4 and end at row 200. The sales totals for each person are in column AK.

Type:	Result:
1. **/*MC**	Tells the program that you want to use the Collect option. **Prompt Line:** Collect: Specify range
2. **[BKSP]**	Edit Line: (clear)

3. **AK4.**	Row 4 is the first row of the search range, and column AK governs. **Prompt Line:** Collect: Specify range **Edit Line:** AK4...
4. **AK200**↵	Tells the program that the last row of the search range is row 200. **Prompt Line:** Collect Criteria: Equal, Greater than, Less than, Range **Edit Line:** (clear)
5. **G**	Selects the greater than criterion. **Prompt Line:** Collect: Specify a single label or value for comparison
6. **1000000**	**Prompt Line:** Value Collect Because you entered a value, the Collect option recognizes that you want to collect values.
7. ↵	The program now collects all the rows between 4 and 200 in column AK that contain a value in excess of 1000000. The command moves these rows to row 4, the top of the search range. **Prompt Line:** Collected 8 row(s) into row(s) 4 ... 11 This message tells you how many rows passed the test (8). Also, just in case you forgot the number of the row at the top of the searching range, this message tells you the current location of the collected rows (4 ... 11).

The example showed how to use the Collect option using the greater than criterion. The less than and equal criteria work the same way. The range criteria requires that you enter a second label or value in step 6.

Install Option (/*MI)

When you type **/*MI**, the program immediately makes the current order of your worksheet's rows and columns permanent.

This option is most useful when you have been temporarily changing the order of rows and columns, and have finally decided on a permanent arrangement.

Note: Using the Install option makes the current worksheet permanent. If you have been temporarily changing a worksheet, and you now want to make those changes permanent, you only have to type **/*MI**. You do not have to redo any changes. You can then save the worksheet on a disk using the Storage command (/S). However, any additional changes that you make will be temporary.

For an explanation of the difference between temporary and permanent arrangements of rows and columns, read the "Temporary vs Permanent Movement" section earlier in this chapter.

The difference between this option and Star* Move Options (/*MO) is important. The Install option applies only to the displayed worksheet.

In contrast, Star* Move Options affects the behavior of the program until you explicitly change it making *all* movement permanent or temporary. For more details, see the "Star* Move Options" section later in this chapter.

Rearrange Option (/*MR)

The Rearrange option quickly reorders the rows or columns of your worksheet. The Rearrange option can work with either rows or columns, but it cannot rearrange both simultaneously. If you want to rearrange both rows and columns, you must use the command twice.

The Rearrange option has the following features:

- You can move many rows or many columns at one time.
- Your columns are automatically moved to the *left side* of the worksheet or your rows are moved to the *top* of the worksheet where the information is easier to examine or print.
- You can quickly return the worksheet to its original state using the Undo option (/*MU), if your rearrangement is temporary.

USING THE REARRANGE OPTION

When you rearrange your worksheet, you must specify what rows or columns you want to move. To use the Rearrange option, type **/*MR**. The program displays the following prompt:

Rearrange: Specify row or column identifier(s)

You must now enter what you want moved. If you enter a letter, the program assumes you want to move columns to the left. If you enter a number, the program assumes you want to move rows up.

When you have entered the first row or column, you have three choices. You can:

- Press **,** (comma) if there are more rows or columns to be rearranged.
- Press **.** (period) if you are rearranging a range of rows or columns.
- Press ↵ if you are finished specifying rows and columns.

You can rearrange several groups of rows or columns in the same command. For example, D,BB.BG,K ↵ is a valid response to the Specify identifier(s) prompt.

EXAMPLE USING THE REARRANGE OPTION

The following example shows how you rearrange several groups of columns.

Type:	Result:
1. **/*MR**	Prompt Line: Rearrange: Specify row or column identifier(s) The Rearrange command is asking you to state which rows or columns you want moved.

2. **D,**

Prompt Line: Rearrange: Specify row or column identifier(s)

Edit Line: D,

Column D is to be leftmost on your worksheet. The comma tells the command that you want more columns moved. The program knows that you want to move columns leftward because you entered a letter.

3. **BB.**

Prompt Line: Rearrange: Specify end of range

Edit Line: D,BB...

Because you entered a period rather than a comma, the command knows that you want a range rather than a single column. It expands the . into an ellipsis (...), and prompts for the end of the range.

4. **BG,**

Prompt Line: Rearrange: Specify rows or column identifier(s)

Edit Line: D,BB...BG,

Because of the comma, the command expects more columns.

5. **K↵**

The command knows this is the last column that you want moved because you pressed ↵. It moves the columns, in the order you typed them, to the left side of the worksheet.

This example shows that you can select one column or several columns. If you want to rearrange rows rather than columns, just enter row numbers instead of column letters. The program notices the difference and recognizes that you want to move rows upward instead of columns leftward.

DON'T REACH BEYOND THE END OF YOUR WORKSHEET

Most worksheets use far fewer rows and columns than the program allows. If your worksheet does not use the 254 rows and 63 columns, do not reach beyond the ends of your worksheet when rearranging it. The Rearrange option ignores any references to rows or columns that are beyond the ends of your active worksheet.

Sort Option (/*MS)

The Sort option orders adjoining rows of your worksheet according to the contents of a column. For example, suppose that your home budget for one year occupies a worksheet consisting of 40 rows and 13 columns. If column M (the rightmost column) contains the annual total for each category of expense, you can rearrange the rows from most expensive to least expensive by using the Sort option.

USING THE SORT OPTION

When you use the Sort option, you must provide the program with a *sort range* and the *order* to sort in.

The sort range is the column of data that the program examines. In the previous example, the sort range is the column containing the annual expense totals. The range can be from the first row to the last row in the column, or it can begin and end with the same row. However, the program examines only entries that lie within one column.

The range must be either labels and blanks or values and blanks. You cannot sort labels and values in the same range or a range that contains any repeating labels.

Note: The program considers the values @ERROR, @ISNA, @TRUE, and @FALSE equivalent to zero in a sort.

The order can be ascending or descending. The precise meaning of ascending and descending depends on whether you are sorting values or labels.

- If you are sorting values, ascending means from the smallest value to the largest; descending means the opposite.
- If you are sorting labels, ascending means from **A** to Z; descending means the opposite. However, the entire uppercase alphabet precedes the lowercase alphabet in ascending sorts, and digits precede the uppercase alphabet.

To use the Sort option, type **/*MS**. The program displays the following prompt:

Sort: Ascending, Descending

If you press **A**, the order is ascending; if you press **D**, the order is descending. When you have selected the sort order, the program prompts:

Specify range

Now you must enter the sort range. Remember that the coordinates you enter must be in the same column, and that the first coordinate must be closer to the top of the worksheet than the last coordinate. To do this, type the first coordinate, a period (.), and the last coordinate. For example, K14.K93 is a valid sort range. When you have entered the sort range, press ↵ and sorting begins.

While the program is sorting, the prompt line displays one of two messages:

Performing label sort or Performing value sort

When sorting is completed, the program displays:

Sort complete

indicating that you can enter another command.

ERROR MESSAGES

If the program detects any errors, it displays one of the following messages:

- ERROR: Invalid range

  This message can be caused by two mistakes: (1) the two range coordinates are in different columns (for example, K10.J33), or (2) the first coordinate of the range is numerically greater than the second (for example, K93.K14).

- ERROR: Invalid sort data

  This message means that the range contains information that can't be meaningfully sorted. Troublesome ranges contain: (1) both values and labels, rather than values and blanks or labels and blanks, or (2) repeating labels.

Once you have found and corrected the cause of the error, you can use the Sort option to successfully sort your rows.

UNEXPECTED RESULTS

Sorting is based on the full value or label of each entry in the range. Even if the width of the entry position is too small to show the last few characters, the Sort option compares the contents using even the undisplayed characters.

This can lead to surprising results. For instance, suppose that you are performing an ascending sort in a range containing these two entries:

Entry	Format	Value	Display
1	Dollars-and-cents	3.25	3.25
2	Integer	3.41	3

Although the first entry appears to be larger in the display (3.25 vs 3), the full value is smaller (3.25 vs 3.41). After an ascending sort, the first entry would precede the second because sorting is based on full value.

EXAMPLE OF SORTING

The following example shows what to type to sort rows 12 through 15, in ascending order, based on the contents of column M.

Type:	Result:
1. **/*MS**	**Prompt Line:** Sort: Ascending, Descending
2. **A**	Tells the Sort option to sort from smallest to largest. **Prompt Line:** Sort Ascending: Specify range **Edit Line:** (displays the current coordinates of the cursor)

3. **[BKSP]**	Tells the program that you want to explicitly type the coordinate. Edit Line: (clear)
4. **M12.**	Prompt Line: Sort Ascending: Specify range Edit Line: M12...
5. **M15**↵	The screen flickers, and the worksheet reappears with rows 12 through 15 sorted. Prompt Line: Sort complete

RANGE ADJUSTMENT AND SORTING

If you are sorting permanently, you can significantly alter the formulas of your worksheet. This is especially true if the Range Adjust option of Star* Move Options (/*MO) is turned on.

If you are not careful when you design your worksheet, you can create formulas that do not adjust properly after the worksheet is sorted. But with a little care, you can create a worksheet that can be sorted and still have valid formulas.

The following figure shows the formulas used to sum the values in a worksheet. Pay special attention to the formulas shown in row 17. Note that range of the sums includes the repeating labels in rows 11 and 16. You can include the repeating labels in the sum because the program treats them as though they have a value of zero.

	A	B	...	L	M
1		JAN	...	DEC	TOTALS
2					
.		.		.	.
.		.		.	.
.		.		.	.
11	----------	----------	----------	----------	----------
12	Ins Premium	xxx	...	xxx	@SUM (B12...L12)
13	Loan Paymt	xxx	...	xxx	@SUM (B13...L13)
14	Fuel Costs	xxx	...	xxx	@SUM (B14...L14)
15	Maintenance	xxx	...	xxx	@SUM (B15...L15)
16	----------	----------	----------	----------	----------
17	Auto Expense	@SUM (B11...B16)	...	@SUM (L11...L16)	@SUM (M11...M16)

The formulas in row 17 demonstrate how you design worksheets that are easier to sort. When you create a worksheet that uses the @SUM function to add ranges of coordinates lying in columns, bracket the range with repeating labels and include the repeating labels in the sum.

Including the repeating labels in the sum is an important technique if you plan to permanently sort your worksheet. It ensures that the ranges in your formulas remain unchanged when you sort the rows between the repeating labels.

To understand why this is important, consider what could happen if the formulas did not refer to the repeating labels. In other words, suppose that the formulas in row 17 referred to the range 12 through 15.

Although row 12 is the top row of the sorting range before sorting, it could move to the middle or bottom during sorting. Similarly, row 15 could move from the bottom to the middle or top of the range.

If either row 12 or 15 moves, and Range Adjustment is turned on, the program will change the formulas in the worksheet to reflect the new positions of rows 12 and 15. If row 15 moves to the middle of the sorting range, the formulas in row 17 would be adjusted to ignore any of the rows following the new position of row 15. This would cause row 17 to contain an incorrect sum.

By having the formulas refer to rows 11 and 16, the rows with the repeating labels, you eliminate this problem. Your sorting is focused on the rows between the repeating labels, so the repeating labels don't move. Consequently, the Range Adjustment option does not modify formulas that refer to the rows containing the repeating labels. This leaves the formulas in row 17 correct even after sorting.

Undo Option (/*MU)

The Undo option allows you to return a temporarily reorganized worksheet to its original state before any changes were made by the Star* Move command. The Undo option allows you to:

- Return only the rows to their original order by pressing **R**.
- Return only the columns to their original order by pressing **C**.
- Return both rows and columns to their original order by pressing **B**.

If you try to use this option to undo a permanently sorted or rearranged worksheet, the program displays the following message:

ERROR: Undo not possible

After receiving this message, you can return to the main prompt by pressing /.

Star* Move Options (/*MO)

The Star* Move Options (/*MO) give you control over how and when your worksheets are made permanent.

After you type **/*MO**, you can press any of four letters:

- When you press **A**, an → appears and disappears to the left of the words Automatic install on the prompt line.

  When an arrow is present, any use of the Star* Move command creates a permanent worksheet. When the arrow is absent, the program leaves the results of the Star* Move command temporary.

  When you load the program, all changes to worksheets are permanent.
- When you press **E**, an → appears and disappears to the left of the words Entry adjust on the prompt line.

  If an arrow is present, the program adjusts all the formulas in your worksheet when the worksheet is made permanent. However, it only adjusts references to specific cells. It does not adjust any ranges within your formulas.

Note: When the program adjusts formulas in the worksheet, it adjusts all the formulas, not just those in the part of the worksheet that has been moved.

- When you press **R**, an → appears and disappears to the left of the words Range adjust on the prompt line.

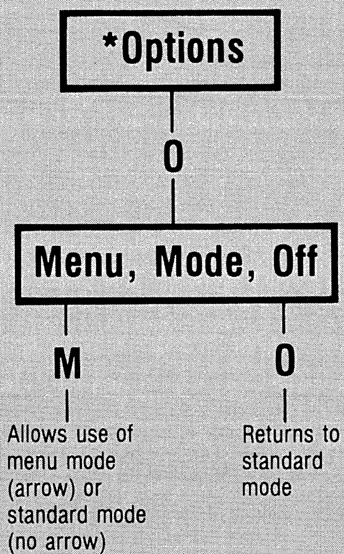
  If an arrow is present, the program adjusts all the formulas in your worksheet when the worksheet is made permanent. However, it only adjusts ranges. It does not adjust any specific entries within your formulas.

Note: When the program adjusts formulas in the worksheet, it adjusts all the formulas, not just those in the part of the worksheet that has been moved.

- Pressing **O** removes all the arrows on the prompt line. It is most useful in Keysaver commands because it allows you to reset the options to a known state.

After you have selected all the options you want, press ↵ to tell the program you have completed your selection. For more information on range adjust and entry adjust, see the "Range Adjustment and Sorting" section earlier in this chapter.

STAR* OPTIONS COMMAND (/*O)



The Star* Options command allows you to choose between two *modes* of entering Star* commands: Menu mode and Standard mode.

To set the mode, type **/*O**. Then, you have two options:

- Pressing **M** makes an → appear and disappear in front of Menu mode on the prompt line. If the → is present, the program accepts commands in Menu mode. This mode is useful for people who are very familiar with the Star* commands and who often use several similar commands consecutively.

  If the → is absent, the program accepts commands in Standard mode.
- Pressing **O** removes any arrows that are on the prompt line. This allows Keysaver commands to put the options into a known state before selecting Menu mode.

Standard Mode

Standard mode is the usual way of entering a command. If the program is in Standard mode and is not processing a command, it displays the main command prompt whenever you press /.

To actually select the command you want to run, you must press the symbol of the command. For example, to select the Star* commands, you must press *.

You can then select any of the Star* commands by pressing the correct letter.

You must continue typing letters until the program obtains the information it needs to run the command. And each time you want to run another command, you must start again from the main prompt.

Menu Mode

Menu mode is a shortcut for entering the Star* commands. In Menu mode, you are not required to start at the main prompt every time you enter a command. The program remembers the Star* command and the data you enter, and reminds you of your previous selections. This can reduce your keystrokes when you use the same procedure repeatedly.

If the program is in Menu mode, it keeps track of which Star* command you selected. Later, when you press /, the program displays the prompt for the Star* command that you last used, with an → in front of the previously selected option.

You can use the PGUP and PGDN keys to move quickly through the sequences of the prompts. Press **PGDN** to select the alternative marked by →, or press **PGUP** to return to the previous prompt.

For example, suppose you are in Menu mode and you last used the Sort option (/*MS). When you press /, the program displays the prompt for the Star* Move command, with an → in front of Sort. If you want to use the same option, press **PGDN**. The program interprets this as selecting the command marked by the →. If you want to use a different Star* Move option, simply press the first letter of its name. If you don't want to use any option of the Star* Move command, press **PGUP**. The screen displays the Star* command prompt.

Menu mode also provides a way to repeat an entire command by typing only two keystrokes. To do this, press /. Because the program is in Menu mode, it displays the prompt for the Star* command you last used. Then press ↵. This tells the program to repeat the command using the same parameters as last time.

Repeating a command with two keystrokes, even after changing values on the worksheet, is useful for sorting and graphing. For instance, suppose you are graphing some hypothetical data. If the graph reveals a problem in the data, you can correct the worksheet and then use this quick-repeat feature to display the graph again.

To return to Standard mode from Menu mode, press **PGUP** until the main command prompt appears. Then type ***OO**↵ to turn off Menu mode.

APPENDIX A

CONTROLLING THE PRINTER

The standard printer is the IBM 80-Character-Per-Second Matrix Printer. The *Option Installation Instructions*, packed with the printer, describe how to unpack, install, and check out the printer; it also describes in some detail how to control the output by sending control codes to the printer.

This appendix describes how to use the VisiCalc program to send those control codes to the IBM printer using the Setup option of the Print command. Control codes govern such printer functions as line spacing, print intensity, character width, etc. See "Guide to Operations" in the *Option Installation Instructions* for a complete description of the setup codes that can be sent to the printer and their effect.

If you are using a different printer, follow the installation and checkout procedures in its manual. The procedure for sending control codes works for all printers; the meaning of the codes, however, varies. Check your printer manual for the codes the printer accepts and their effect.

Note: If you haven't printed anything yet, go through the checkout procedures in the *Option Installation Instructions* to confirm the printer's operation before attempting to follow the procedures in this appendix.

When you specify a printer to either the Print or Storage commands, the VisiCalc program prompts you for details about the print operation:

```
Print: Lower right, "Setup, - , &
```

As described in "Print Command" in Chapter 3, "Lower right" refers to the lower-right corner of the rectangle to be printed. This appendix describes the Setup option, which can be used to send a control code (or series of control codes) to the printer, and the last two options (– and &), which control whether the VisiCalc program sends a Carriage Return to the printer at the end of each line.

All the examples in this appendix are for the IBM 80-Character-Per-Second Matrix Printer and assume that the lower-right corner of the worksheet to be printed is location C5. If you have saved a worksheet, load it; otherwise, enter some labels and values in the portion of the worksheet bounded by A1 on the upper left and C5 on the lower right.

Sending Setup Characters to the Printer

The Setup option sends one or more characters to the printer before the worksheet is printed. These can be printable characters, control codes, or both. Several character combinations have a special meaning to make it possible to send the control codes accepted by the printer (the ^ is the caret character from the keyboard):

- ^C Marks the next character as a control character (the ^C has the same effect as holding down the CTRL key). Only the letters A through Z are valid following ^C. Typing ^CE, for example, sends CTRL-E; ^CC sends CTRL-C. The CTRL key cannot be used.
- ^E Sends an ESC. The ESC key cannot be used.
- ^R Sends a Carriage Return ↵.
- ^L Sends a Line Feed.
- ^H Treats the next two characters as hexadecimal digits and sends the single ASCII character whose code is the corresponding hexadecimal number. ^H1B, for example, also sends the ESC code; ^H0D sends a Carriage Return.
- ^^ Sends a caret character (^).

To send a Carriage Return, for example, type ^R ↵. The ↵ only ends the response to the prompt, it doesn't send a Carriage Return to the printer; the ^R sends a Carriage Return to the printer. To send two Carriage Returns type ^R^R ↵. To send ESC E, type ^EE ↵; to send ESC A0, type ^EA0 ↵. To send the hexadecimal value 0F (decimal 15), type ^H0F (the 0 is required).

The setup characters are sent to the printer and discarded. You can send several sets of characters with the Setup option before printing the worksheet, but once the worksheet has started printing you cannot send any more.

The VisiCalc program doesn't determine whether the control codes are valid. If you send an incorrect code or series of codes, the only indication is the (possibly) unexpected behavior of the printer. In extreme cases, it may be necessary to reload the VisiCalc program to restore proper printer operation.

Changing the Typeface

The following example prints the worksheet in emphasized type:

Type:	Result:
1. **/P**	**Prompt Line:** File, Printer
2. **P**	**Prompt Line:** Print: Lower right, "Setup, - , &
3. **"**	**Prompt Line:** Setup or ENTER:
4. **^EE ↵**	**Prompt Line:** Print: Lower right, "Setup, - , &
5. **C5 ↵**	The worksheet is printed in emphasized type.

The next example prints the worksheet in condensed type:

Type:	Result:
1. **/P**	**Prompt Line:** File, Printer
2. **P**	**Prompt Line:** Print: Lower right, "Setup, - , &
3. **"**	**Prompt Line:** Setup or ENTER:
4. **^H0F ↵**	**Prompt Line:** Print: Lower right, "Setup, - , &
5. **C5 ↵**	The worksheet is printed in condensed type. This prints up to 132 characters across a sheet of paper only 8-1/2" wide.

Printing a Title on the Worksheet

The following example prints "Budget" in enlarged printing as a title, followed by two blank lines, at the top of the worksheet:

Type:	Result:
1. **/P**	**Prompt Line:** File, Printer
2. **P**	**Prompt Line:** Print: Lower right, "Setup, - , &
3. "	**Prompt Line:** Setup or ENTER:
4. **^H0EBudget^R^R** ↵	**Prompt Line:** Print: Lower right, "Setup, - , & "Budget" is printed in enlarged type, followed by two blank lines. The printer stops enlarged printing at the end of the line.
5. **C5** ↵	The worksheet is printed in normal type.

The next example prints a multi-line title that consists of "Personal Budget" in enlarged emphasized printing, a blank line, "1981 Estimated" in condensed printing, and two blank lines.

Type:	Result:
1. **/P**	**Prompt Line:** File, Printer
2. **P**	**Prompt Line:** Print: Lower right, "Setup, - , &

3. "	**Prompt Line:** Setup or ENTER:
4. **^H0E^EE Personal Budget^R^R ↵**	**Prompt Line:** Print: Lower right,"Setup, – , &
	"Personal Budget" is printed in enlarged characters (^H0E), emphasized printing (^EE). The extended printing is turned off by the printer at the end of the line.
5. "	**Prompt Line:** Setup or ENTER:
6. **^EF^H0F1981 Estimated^R^R^R ↵**	**Prompt Line:** Print: Lower right, "Setup, – , &
	"1981 Estimated" is printed in condensed characters (^H0F). The printer doesn't turn off condensed printing at the end of a line, so if you don't send the code to turn off condensed printing (^H12), the worksheet will be printed condensed.
7. "	**Prompt Line:** Setup or ENTER:
8. **^H12 ↵**	**Prompt Line:** Lower right, "Setup, – , &
	Turns off condensed printing.
9. **C5 ↵**	The worksheet is printed in normal type.

Controlling Printer Spacing

The – and & at the end of the Print command prompt (Print: Lower right, "Setup, –, &) control whether the VisiCalc program sends a Carriage Return and Line Feed to the printer at the end of each line or just a Carriage Return. When the program is first loaded, a Carriage Return and Line Feed are automatically sent to the printer at the end of each line. To suppress the Line Feed, type –; to turn Line Feed back on, type &. Typing & when the Carriage Return and Line Feed are being sent has no effect; similarly, typing – when the Line Feed is not being sent has no effect.

If a worksheet is printed double-spaced and you want it single-spaced, use the following procedure to suppress the extra Line Feed:

	Type:	Result:
1.	**/P**	**Prompt Line:** File, Printer
2.	**P**	**Prompt Line:** Print: Lower right, "Setup, – , &
3.	**-**	**Prompt Line:** Print: Lower right, "Setup, – , &
4.	**C5** ↵	A Line Feed is suppressed at the end of each line of the worksheet.

APPENDIX B

EXCHANGING FILES

The DIF™ format is a standard file format that allows unrelated programs to share data. A file saved in the DIF format is a text file that can be read by other DIF-supporting programs. Thus, a file created by the VisiCalc program can be saved and read by other programs that support DIF, and files created by these other programs can be loaded by the VisiCalc program. These programs can be written (in BASIC, for example) so that the VisiCalc program can be integrated into a broader set of personal computing tools.

Another useful feature of the DIF format is that it allows for the transfer of data from one VisiCalc sheet to another, enabling users to consolidate data for purposes such as corporate or annual calculations.

This appendix describes the commands that save and load DIF files, the file format itself, and three sample programs in BASIC that read and write DIF files.

Additional information about DIF™ is available through the DIF Clearinghouse. The DIF Clearinghouse is set up to:

Further information is available through the DIF Clearinghouse, which has been set up to coordinate and distribute information about the DIF format. The Clearinghouse maintains and distributes the DIF Technical Specification, which is a detailed technical description of the DIF format, and information about the commercially available programs that support the format. To obtain this information, please send your name, address, $6, and a note requesting this information to: DIF Clearinghouse, P.O. Box 526, Cambridge, MA 02139.

DIF™ is a trademark of Software Arts, Inc.

The Save DIF™ Option —/S#S

The Save DIF option of the Storage command (/S#S) saves the worksheet in the DIF format. The VisiCalc program adds .DIF to the file name so it can be recognized later as a worksheet saved as a DIF file.

Values are saved to their full precision. This precision may cause the saved worksheet to differ from the displayed version. To save the worksheet with all values calculated to the same precision as the version on the screen, you must either use a rounding formula (see "Making the VisiCalc Program Less Precise" in "Functions" in Chapter Three) or save the file with the File option of the Print command (/PF) and then convert it to the DIF format with a separate program.

The VisiCalc program saves and loads a worksheet in one of two ways–by row or by column. The VisiCalc program prompts you to specify the orientation by typing "R" or ENTER or "C". Specify "R" or press ENTER when saving a worksheet to be loaded later by the VisiCalc program.

The following example saves the area of the worksheet whose upper-left boundary is A1 and lower right boundary is J14 in a file named BUDGET.DIF on the disk in drive B:

	Type:	Result:
1.	**HOME**	The cursor moves to A1.
2.	**/S#**	**Prompt Line:** Data: Save Load
3.	**S**	**Prompt Line:** Data save: File for Saving
4.	**b:budget ↵**	**Prompt Line:** Data save: Lower right **Edit Line:** b:budget
5.	**J14 ↵**	**Prompt Line:** Data save: R C or ENTER Or you can move the cursor to J14.
6.	**R**	The prompt line clears and the drive activates, saving the sheet as a DIF file.

Load DIF™ Option —/S#L

The Load DIF option of the Storage command (/S#L) loads a worksheet stored on a diskette in the DIF format. The file can be one saved with the Save DIF option of the Storage command, or created by another program.

Unlike a worksheet saved in the normal fashion–which is loaded into the same position it occupied when it was saved–the upper left corner of a worksheet loaded in the DIF format is the position of the cursor when the Storage command is typed.

The following example loads the worksheet stored in a file named BUDGET.DIF on the diskette in drive A starting at location C10:

	Type:	Result:
1.	**/CY**	The worksheet clears and the cursor moves to A1.
2.	**>C10 ↵**	The cursor moves to C10.
3.	**/S#L**	**Prompt Line:** Data load: File to Load
4.	**budget ↵**	**Prompt Line:** Data load: R C or ENTER
5.	**R**	The worksheet appears, starting at C10.

As with the other options of the Storage command, you can display the names of the files on the diskette by typing → in response to the "File to load:" prompt.

To transpose from loading across rows to loading down columns, type "C" instead of "R" at step 4. A worksheet saved in columnar format can be transposed to row format by typing "R" at this point.

The DIF™ Format

The DIF format stores the worksheet in a form accessible to programs other than the VisiCalc program. To accommodate a wide range of languages in which such a program might be written, several simplifying techniques have been used:

- Information about the size of the file is provided at the beginning.
- Records are kept as short as possible.
- The data type (string or number) of each value is explicitly defined.
- Strings are stored one per line.
- Strings that contain special characters are enclosed in quotation marks.
- The file ends with an explicit End-Of-Data record.

Figure B-1 shows a sample worksheet used to describe the format and the programs that work with the DIF format.

Year	1980	1981	1982
-------	-------	-------	-------
Sales	100	110	121
Cost	80	88	97
Profit	20	22	24

Figure B-1. Sample Worksheet

The format stores the worksheet in a series of slices; the worksheet can be sliced either horizontally (by rows) or vertically (by columns). Each of these slices is called a *tuple;* each slice along the other axis is called a *vector.* In Figure B-1, for example, if the worksheet is saved by rows, the first vector is "Year 1980 1981 1982" and the first tuple contains "Year", "– " (one hyphen), "Sales", "Cost", and "Profit"; the entire worksheet is stored in four tuples of five values each.

If the worksheet in Figure B-1 is stored by columns, the first vector is YEAR "– Sales Cost Profit" and the first tuple contains "Year", 1980, 1981, and 1982; the entire worksheet is stored in five tuples of four values each.

A DIF file consists of a series of *header* records that describe the file, followed by one set of *data* records for each tuple, and ends with a pair of *End-Of-Data* records.

Header

The header consists of four sets of three records that give information about the entire file:

TABLE	
0,1	
""	
VECTORS	
0,V	V is the number of vectors in the file.
""	
TUPLES	
0,T	T is the number of tuples in the file.
""	
DATA	
0,0	
""	

Data Records

The data records consist of a pair of header records that identify the beginning of a tuple, and a pair of records for each value in the tuple:

-1,0	
BOT	Beginning Of Tuple records.
T1,N1	
string1	First value of tuple.
T2,N2	
string2	Second value of tuple.
T3,N3	
string3	Third value of tuple.
.	
.	
.	
Tn,Nn	
stringn	Last (nth) value of tuple.

End-Of-Data Records

The End-Of-Data records mark the end of the file:

-1,0
EOD

Programs that Work With the DIF Format

The following three programs demonstrate the use of the DIF format. They perform the following functions:

- Dump a DIF file just as it is stored, record by record.
- Print a worksheet from a DIF file.
- Create a DIF file by prompting for the worksheet entries.

Dumping a DIF™ File

This program prints the DIF file just as it is stored, record by record. It asks for the name of the file, and whether to print it. If not instructed to print the file, the program displays the file on the screen.

```
100 ' ***********************
110 ' *    INITIALIZATION   *
120 ' ***********************
130 NUL$ = CHR$(34) + CHR$(34)
140 FALSE = 0
150 TRUE = -1
440 '
450 '
460 ' ***********************
470 ' *     MAIN ROUTINE    *
480 ' ***********************
490 '
500 GOSUB 1000'                    * PROMPT FOR ORDERS
510 GOSUB 1200                     * PRINT HEADER
520 GOSUB 1400'                    * PRINT DATA RECORDS
530 GOSUB 1600'                    * END-OF-PROGRAM CLEANUP
540 END
940 '
950 '
960 ' ***********************
970 ' * PROMPT FOR ORDERS   *
980 ' ***********************
990 '
```

```
1000 CLS
1010 INPUT "File name: ",FILENAME$
1020 IF RIGHT$(FILENAME$,4) <> ".DIF" THEN FILENAME$ = FILENAME$ + ".DIF"
1030 INPUT "Print the file (Y or N): ",REPLY$
1040 IF REPLY$ = "Y" OR REPLY$ = "y" THEN HARDCOPY = TRUE
1050 OPEN FILENAME$ FOR INPUT AS #1
1060 IF NOT HARDCOPY THEN CLS
1070 IF HARDCOPY THEN LPRINT FILENAME$: LPRINT " ":LPRINT ELSE PRINT FILENAME$:P
RINT:PRINT
1080 RETURN
1140 '
1150 '
1160 ' ***********************
1170 ' *    PRINT HEADER     *
1180 ' ***********************
1190 '
1200 INPUT #1, TITLE$
1210 INPUT #1, TYPE, NUMBER
1220 INPUT #1, STRNG$
1230 IF HARDCOPY THEN LPRINT TITLE$:LPRINT TYPE; ","; NUMBER ELSE PRINT TITLE$:P
RINT TYPE; ","; NUMBER
1240 IF STRNG$ = "" THEN IF HARDCOPY THEN LPRINT NUL$ ELSE PRINT NUL$ ELSE IF HA
RDCOPY THEN LPRINT STRNG$ ELSE PRINT STRNG$
1250 IF TITLE$ <> "DATA" THEN 1200
1260 IF NOT HARDCOPY THEN GOSUB 2000
1270 RETURN
1340 '
1350 '
1360 ' ***********************
1370 ' * PRINT DATA RECORDS *
1380 ' ***********************
1390 '
1400 INPUT #1, TYPE, NUMBER
1410 INPUT #1, STRNG$
1420 IF CSRLIN > 20 AND NOT HARDCOPY THEN GOSUB 2000
1430 IF HARDCOPY THEN LPRINT TYPE; ","; NUMBER: LPRINT STRNG$ ELSE PRINT TYPE; "
,"; NUMBER: PRINT STRNG$
1440 IF STRNG$ <> "EOD" THEN 1400
1450 RETURN
1530 '
1540 '
1550 ' ***********************
1560 ' *    END-OF-PROGRAM   *
1570 ' *    CLEANUP          *
1580 ' ***********************
1590 '
1600 CLOSE 1
1610 RETURN
1920 '
1930 '
1940 ' ***********************
1950 ' *         PRINT       *
1960 ' * "RETURN FOR MORE"   *
1970 ' * MESSAGE AT BOTTOM   *
1980 ' ***********************
1990 '
2000 LOCATE 24,1
2010 PRINT "RETURN for more";
2020 REPLY$ = INPUT$(1)'                  * WAIT UNTIL ANY KEY IS PRESSED
2030 CLS
2040 RETURN
```

Printing A Worksheet From A DIF™ File

The following program prints the worksheet as it would appear on the screen. It asks for the name of the file in which the worksheet was saved in the DIF format, the width of columns to be printed, and whether the worksheet was saved by rows or by columns:

```
60 ' **********************
70 ' *    INITIALIZATION    *
80 ' **********************
90 '
100 DIM WORKSHEET$(50,50)
110 FALSE = 0
120 TRUE = -1
130 BYROWS = FALSE
440 '
450 '
460 ' **********************
470 ' *     MAIN ROUTINE     *
480 ' **********************
490 '
500 GOSUB 1000'                          * PROMPT FOR ORDERS
510 GOSUB 1200'                          * READ HEADER
520 GOSUB 1400'                          * READ DATA RECORDS
530 IF NOT FILEBAD THEN GOSUB 1600'      * PRINT THE WORKSHEET
540 GOSUB 1800'                          * END-OF-PROGRAM CLEANUP
550 END
940 '
950 '
960 ' **********************
970 ' * PROMPT FOR ORDERS  *
980 ' **********************
990 '
1000 CLS
1010 INPUT "File name: ", FILENAME$
1020 IF RIGHT$(FILENAME$,4) <> ".DIF" THEN FILENAME$ = FILENAME$ + ".DIF"
1030 INPUT "Column width: ",COLUMNWIDTH
1040 INPUT "Saved by row or column (R or C): ",REPLY$
1050 IF REPLY$ = "R" OR REPLY$ = "r" THEN BYROWS = TRUE
1060 INPUT "Print the worksheet (Y or N): ",REPLY$
1070 IF REPLY$ = "Y" OR REPLY$ = "y" THEN HARDCOPY = TRUE
1080 OPEN FILENAME$ FOR INPUT AS #1
1090 RETURN
1140 '
1150 '
1160 ' **********************
1170 ' *     READ HEADER      *
1180 ' **********************
1190 '
1200 INPUT #1, TITLE$
1210 INPUT #1, TYPE, NUMBER
1220 INPUT #1, STRNG$
1230 IF TITLE$ = "VECTORS" THEN VECTORS = NUMBER
1240 IF TITLE$ = "TUPLES" THEN TUPLES = NUMBER
1250 IF TITLE$ = "DATA" THEN RETURN
1260 GOTO 1200
1340 '
1350 '
1360 ' **********************
1370 ' * READ DATA RECORDS  *
1380 ' **********************
1390 '
1400 FOR ROW = 1 TO TUPLES
1410     INPUT #1, TYPE, NUMBER
1420     INPUT #1, STRNG$
1430     IF TYPE <> -1 OR STRNG$ <> "BOT" THEN
            GOSUB 2000:
            RETURN
```

```
1440      FOR COL = 1 TO VECTORS
1450           INPUT #1, TYPE, NUMBER
1460           INPUT #1, STRNG$
1470           IF TYPE <> 0 AND TYPE <> 1 THEN
                  GOSUB 2000: RETURN
1475           GOSUB 2200
1480           IF BYROWS AND TYPE = 0 THEN WORKSHEET$(COL,ROW) = NUMBER$:
               GOTO 1500
1490           IF BYROWS THEN WORKSHEET$(COL,ROW) = STRNG$
1500           IF NOT BYROWS AND TYPE = 0 THEN WORKSHEET$(ROW,COL) = NUMBER$:
               GOTO 1520
1510           IF NOT BYROWS THEN WORKSHEET$(ROW,COL) = STRNG$
1520           NEXT COL
1530      NEXT ROW
1540 RETURN
1550 '
1560 ' **********************
1570 ' * PRINT THE WORKSHEET*
1580 ' **********************
1590 '
1600 IF BYROWS THEN WDTH = TUPLES: DEPTH = VECTORS ELSE WDTH = VECTORS: DEPTH =
TUPLES
1610 FOR ROW = 1 TO DEPTH
1620      FOR COL = 1 TO WDTH
1630           IF HARDCOPY THEN LPRINT WORKSHEET$(ROW,COL); ELSE PRINT WORKSHEET$(
ROW,COL);
1640           NEXT COL
1650      IF HARDCOPY THEN LPRINT " " ELSE PRINT
1660      NEXT ROW
1670 RETURN
1730 '
1740 '
1750 ' **********************
1760 ' *   END-OF-PROGRAM   *
1770 ' *   CLEANUP          *
1780 ' **********************
1790 '
1800 CLOSE 1
1810 RETURN
1940 '
1950 '
1960 ' **********************
1970 ' *   ERROR IN FILE    *
1980 ' **********************
1990 '
2000 PRINT
2010 BEEP: PRINT "ERROR IN FILE . . . "
2020 PRINT TAB(5); "TYPE ="; TYPE
2030 PRINT TAB(5); "NUMBER ="; NUMBER
2040 PRINT TAB(5); "STRING = "; STRNG$
2050 FILEBAD = TRUE
2060 RETURN
2140 '
2150 ' *********************
2160 ' *      OFFSET       *
2170 ' * LABELS & NUMBERS  *
2180 ' *********************
2190 '
2200 IF TYPE <> 0 THEN 2300
2210 NUMBER$ = STR$(NUMBER)
2220 IF LEN(NUMBER$) > COLUMNWIDTH - 1 THEN NUMBER$ = " " + LEFT$(NUMBER$,COLUMN
WIDTH - 1):RETURN
2230 BLANK$ = "                                   "
2240 BN = COLUMNWIDTH - LEN(NUMBER$)
2250 NUMBER$ = LEFT$(BLANK$,BN) + NUMBER$
2260 RETURN
2300 IF LEN(STRNG$) > COLUMNWIDTH THEN STRNG$ = LEFT$(STRNG$,COLUMNWIDTH):
      RETURN
2310 BLANK$ = "                                   "
2320 BN = COLUMNWIDTH - LEN(STRNG$)
2330 STRNG$ = STRNG$ + LEFT$(BLANK$,BN)
2340 RETURN
```

Creating A DIF™ File

The following program prompts for worksheet entries (by row-column coordinate), then writes the entries on a diskette in a DIF file. Either a string or number (integer or real) can be entered. To enter a label that starts with a number, type a quotation mark (") as the first character of the label. To end a row, type ESC ↵; to end the worksheet, type ESC ESC ↵.

The program assumes the coordinate of the lower-right corner of the worksheet is the row-column coordinate of the location immediately to the left of the coordinate where ESC-ESC is typed, so the last row should be at least as wide as all preceding rows. The worksheet is saved by rows:

```
60 ' **********************
70 ' *    INITIALIZATION   *
80 ' **********************
90 '
100 ESC$ = CHR$(27)
110 LASTINROW$ = ESC$
120 LASTONSHEET$ = ESC$ + ESC$
130 QUOTE$ = CHR$(34)
140 NUL$ = QUOTE$ + QUOTE$
150 DIM WORKSHEET$(50,50)
160 ROW = 1
170 COL = 1
180 FALSE = 0
190 TRUE = -1
440 '
450 '
460 ' **********************
470 ' *    MAIN ROUTINE     *
480 ' **********************
490 '
500 GOSUB 1000'                          * PROMPT FOR ORDERS
510 GOSUB 1200'                          * PROMPT FOR ENTRIES
520 GOSUB 1400'                          * WRITE FILE
530 GOSUB 2000'                          * END-OF-PROGRAM CLEANUP
540 END
940 '
950 '
960 ' **********************
970 ' * PROMPT FOR ORDERS  *
980 ' **********************
990 '
1000 CLS
1010 INPUT "Write the file (Y or N): ",REPLY$
1020 IF REPLY$ = "Y" OR REPLY$ = "y" THEN DISKCOPY = TRUE
1030 IF DISKCOPY
        THEN INPUT "File name: ",FILENAME$:
              IF RIGHT$(FILENAME$,4) <> ".DIF" THEN FILENAME$ = FILENAME$ + ".DIF
"
1040 RETURN
1140 '
1150 '
1160 ' **********************
1170 ' * PROMPT FOR ENTRIES *
1180 ' **********************
1190 '
```

```
1200 CLS
1210 WORKSHEET$(ROW,COL) = ""
1220 PRINT "Row"; ROW; ", Column ";CHR$(64+COL); ": ";
1230 REPLY$ = INPUT$(1)
1240 IF REPLY$ = CHR$(8) AND LEN(WORKSHEET$(ROW,COL)) - 1 = 0 THEN PRINT " ":
     GOTO 1210
1250 IF REPLY$ = CHR$(8) THEN WORKSHEET$(ROW,COL) = LEFT$(WORKSHEET$(ROW,COL),LE
N(WORKSHEET$(ROW,COL)) - 1): PRINT " "; WORKSHEET$(ROW,COL);:GOTO 1230
1260 IF REPLY$ <> CHR$(13)
      THEN PRINT REPLY$;: WORKSHEET$(ROW,COL) = WORKSHEET$(ROW,COL) + REPLY$:
GOTO 1230
1270 PRINT
1280 IF WORKSHEET$(ROW,COL) = LASTONSHEET$
      THEN WDTH = COL - 1: DEPTH = ROW: RETURN
1290 IF WORKSHEET$(ROW,COL) = LASTINROW$
      THEN ROW = ROW + 1: COL = 1: PRINT:GOTO 1210
1300 COL = COL + 1
1310 GOTO 1210
1320 '
1330 ' **********************
1340 ' *     WRITE FILE     *
1350 ' **********************
1360 '
1370 ' ---------------------
1380 '          HEADER
1390 ' ---------------------
1400 IF NOT DISKCOPY THEN RETURN
1404 OPEN FILENAME$ FOR OUTPUT AS #1
1410 PRINT #1, "TABLE"
1420 PRINT #1, 0; ","; 1
1430 PRINT #1, NUL$
1440 PRINT #1, "VECTORS"
1450 PRINT #1, 0; ","; DEPTH
1460 PRINT #1, NUL$
1470 PRINT #1, "TUPLES"
1480 PRINT #1, 0; ","; WDTH
1490 PRINT #1, NUL$
1500 PRINT #1, "DATA"
1510 PRINT #1, 0; ","; 0
1520 PRINT #1, NUL$
1540 '
1550 ' ---------------------
1560 '      DATA RECORDS
1570 ' ---------------------
1580 FOR COL = 1 TO WDTH
1590     PRINT #1, -1; ","; 0
1600     PRINT #1, "BOT"
1610     FOR ROW = 1 TO DEPTH
1620         IF VAL(WORKSHEET$(ROW,COL)) <> 0 OR WORKSHEET$(ROW,COL) = "0" THEN
              PRINT #1, 0;","; VAL(WORKSHEET$(ROW,COL)):PRINT #1, "V":GOTO 1660

1630         IF LEFT$(WORKSHEET$(ROW,COL),1) = QUOTE$ THEN
               WORKSHEET$(ROW,COL) = MID$(WORKSHEET$(ROW,COL),2)
1640         PRINT #1, 1; ","; 0
1650         PRINT #1, QUOTE$; WORKSHEET$(ROW,COL); QUOTE$
1660         NEXT ROW
1670     NEXT COL
1680 '
1690 ' ---------------------
1700 '      END-OF-DATA
1710 ' ---------------------
1720 PRINT #1, -1; ","; 0
1730 PRINT #1, "EOD"
1740 RETURN
1930 '
1940 '
1950 ' **********************
1960 ' *   END-OF-PROGRAM   *
1970 ' *   CLEANUP          *
1980 ' **********************
1990 '
2000 CLOSE 1
2010 RETURN
```

APPENDIX C

PRINTERS AND ADAPTERS

The VisiCalc IV program supports and has been tested with the following printers. Some of these printers require special interface cables to connect to the computer. See your dealer for information on connecting your printer.

- IBM® 80 CPS Matrix Printer
- IBM 80 CPS Matrix Graphics Printer
- Centronics® 739 printer
- Epson MX, FX, and RX printers (80 or 100)
- NEC Spinwriter™ models 3530, 5520, 5530, 7710, or 7730
- Anadex® DP-8000 printer
- Diablo® 630 printer
- Qume® Sprint 5 printer
- IDS Paper Tiger™ 440, 460, and 560 printers
- Okidata (Microline) 82A and 83A printers
- Okidata (Microline) 84, 92, and 93 printers
- Gemini 10 and 10x printers
- C. Itoh Prowriter

GRAPHICS PRINTERS

The Star* Graph command (/*G) allows you print your graphs on several kinds of graphics printers. But the program must know what kind of printer you have attached to your computer. The Star* Devices command (/*D) tells the program what kind of graphics printer you have.

The following is a list of the supported graphics printers and their class.

Class 1

IBM 80 CPS Matrix Graphics Printer
Epson MX-80 or MX-100 printers with Graftrax or Graftrax Plus
Epson FX or RX printers
Gemini 10 and 10x printers
Okidata (Microline) with Plug'N Play

Class 2

C. Itoh Prowriter

Class 3

Okidata Microline 82A, 83A, 84, 92 and 93 with Okigraph™

If you do not use the Star* Devices command to tell the program what kind of printer you have, the program assumes that you have a printer from Class 1.

If you are using a Prowriter, do not use the buttons on the printer to position the paper when the printer is printing graphs. If you use the printer's buttons, you will destroy the line count maintained by the program. This can cause the program to think that your paper is positioned at the Top of Form when it actually is not.

You can avoid this problem by using the F and L options when your graph is displayed on the monitor. These options allow you to position the paper, but they do not destroy the line count. For more information about these options, see the "Printing Your Graphs" section of Chapter 4.

ADAPTERS

The following monitor adapters have been tested with the VisiCalc IV program:

- IBM Color/Graphics adapter
- IBM Monochrome adapter

Entries are capitalized as they appear in text.

Special Characters

A

B

C

D

E

F

L

M

S

W

Reader's Critique

VisiCorp's Publications Department wants to provide documents that meet the needs of all our product users. Your feedback helps us produce better manuals.

Please tell us how you rate this manual in the following areas:

	Excellent (✓)	Good (✓)	Average (✓)	Fair (✓)	Poor (✓)
Completeness of information.					
Clarity of information.					
Organization of manual.					
Appearance of manual.					
Quantity of examples.					
Usefulness of examples.					
Quantity of photographs and illustrations.					
Appropriateness of photographs and illustrations.					
Degree to which this manual meets your expectations and needs.					
Appropriateness of the manual level to a person of your experience.					
Overall manual.					

Your computer experience: ☐ 2 yrs or less ☐ 2-5 yrs ☐ 5 yrs +

Did you use the manual to learn the program? ☐ Yes ☐ No

I read ☐ 25% ☐ 50% ☐ 75% ☐ 100% of this manual.

Please specify, by page, any errors you found in this manual.

Please tell us for what application(s) you are using this program.

What one thing could we do to improve this manual for you?

Name ______________________ Title ______________________

Company ______________________

Address ______________________

City ______________________ State ________ Zip __________

Phone: (Office) ______________________ (Home) ______________________

No Postage
Necessary
If Mailed
in the
United States

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 7066 SAN JOSE, CA 95134

POSTAGE WILL BE PAID FOR BY ADDRESSEE

VISICORP
2895 Zanker Road
San Jose, CA 95134

2895 Zanker Road
San Jose, CA 95134
Phone: 408/946-9000

VISICORP™

IBM Personal Computer
VisiCalc® IV
60917-2210 11/83