



Garden Planner

Now you can lay out your garden row by row on your Apple Hi-Res screen. Then calculate the money you save.

The time to start planning your vegetable garden has finally arrived. No doubt you've been poring over those seed catalogs all winter. Remember all the zucchini you had to throw away last summer? Don't you wish you had planted more green peppers? With the Garden Planner you can lay out your garden row by row and analyze the cost benefits you'll receive from growing your own. You can design and print various plans, and select the one that best suits your needs and garden space. A planting guide lists planting depths, maturation times, soil temperatures, row and plant spacings and more, for 45 common garden vegetables. You can save your garden plans to disk, and print garden graphics if you have a graphics-capable printer interface card.

The economic feasibility of any plan can be determined before you actually select your final layout. You can even do an economic evaluation on a row-by-row basis to see where you can economize. In short, the Garden Planner is your guide to an efficient and productive garden.

THE PROGRAM FILES

The Garden Planner consists of four BASIC programs, a machine language cursor selection program, and two shape tables that contain the vegetable shapes and the ASCII text characters used on the Hi-Res screen. GARDEN.PLANNER (Listing 1) does most of the work. PG.GUIDE.CREATE (Listing 2) is a short BASIC program that simply creates and fills the planting guide data file, PG.GUIDE, and PG.NAMES.CREATE (Listing 3) creates the data file PG.NAMES. Listings 2 and 3 need to be run only a single time to create the data files. CONFIGURE (Listing 4) is a short program that creates the system configuration data file GP.CON-

FIG. It is run from GARDEN.PLANNER when GP.CONFIG does not exist or when you want to change the configuration.

The cursor program, CURSOR.BAR.4 (Listing 5), uses a modified form of Jim Ganz's CURSOR.BAR program, which first appeared in the Nibble Stock Analyst program (Vol. 5/No. 12). I've added an Escape function, in addition to support for the four arrow keys and Return. Pressing Escape from any menu (except the main menu) returns you to the previous menu. (Essentially, this function replaces a menu selection that returns to the previous menu.) The shape table VEG.SHAPES contains the 16 vegetable shapes used in the Garden Planner program. The shape table TEXT

FIGURE 1: Main Menu

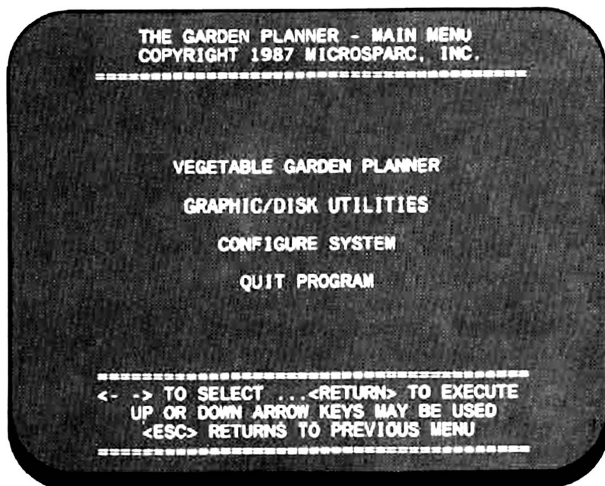




FIGURE 2: Vegetable Graphics

.SHAPES uses the same ASCII shape table that appeared in "Hi-Res Characters," written by Vinay, Vivek and Vijay Pai (Vol. 5/ No. 12).

USING THE PROGRAMS

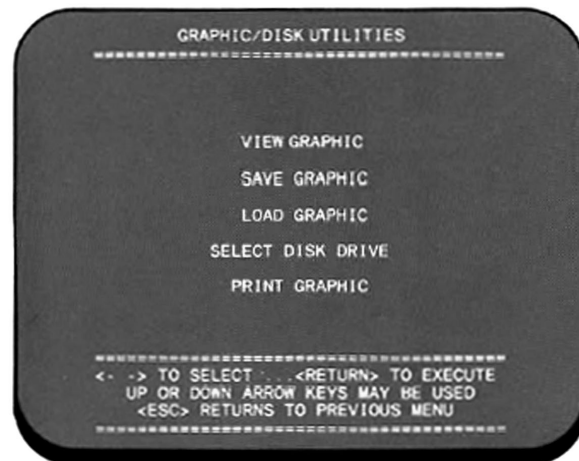
First run PG.GUIDE.CREATE (Listing 2) and make sure it has created the text file PG.GUIDE. Then run PG.NAMES.CREATE (Listing 3) and make sure it has created the file PG.NAMES. To use Garden Planner, make sure the disk is not write-protected and RUN GARDEN.PLANNER. The program relocates itself above graphics page 1, so the initial loading time may be a little longer than usual, as it loads twice — once at the normal load address (\$0801) and again at the new address (\$4001).

SYSTEM CONFIGURATION

Once the data files are confirmed, the program checks for the system configuration file, GP.CONFIG. If it doesn't exist (usually only the first time you run the program), you are offered the chance to quit or run CONFIGURE. When CONFIGURE is run, the default configuration parameters are displayed so you can change them. Indicate whether you have a printer; the slot in which the controller card resides; whether your interface card has text and/or graphic dump capability; and the codes to effect the text and/or graphics dumps. In addition, you can select your monitor type (color or monochrome).

Although the program relies on the interface card for graphics, it includes a generic text screen dump routine in case your card does not have a text screen dump capability. If your card does not

FIGURE 3: Graphic/Disk Utilities Menu



have graphics capability, you must use another graphics printing program. The Graphics/Disk Utilities menu offers a Save Graphic option.

USING GARDEN PLANNER

With your system configured, you're ready to use the system. Figure 1 shows the main menu. When you choose the first selection, Vegetable Garden Planner, four more options are displayed. They are:

GARDEN LAYOUT
PLANTING GUIDE
VIEW SHAPES/DESCRIPTIONS
GARDEN ECONOMICS

View Shapes/Descriptions

To see the shapes that denote the various vegetables, move the inverse cursor bar to the third option and press Return. This choice displays all of the shapes and a brief description of their general intended use (Figure 2). You may return to the Garden Planner menu by pressing Return.

Note: If you are in the middle of creating a garden layout, selecting the View Shapes option will wipe out the existing layout, since both routines use the Hi-Res graphics page. A warning message is displayed, however, before the View Shapes/Descriptions option is implemented.

Planting Guide

To view the planting guide data, select the second option. A short menu appears. From it, you can choose to display a single entry, print the entire guide (if you have a printer) or return to the Garden Planner menu by pressing Escape. After the completion of each function, you may return to the planting guide by pressing Escape. Another Escape keypress will return you to the Garden Planner menu.

Garden Layout

Now select the Garden Layout option from the Garden Planner menu. Since this menu can be accessed from an existing layout screen, a short menu is displayed if a layout is in progress. The choices are:

START NEW LAYOUT
WORK ON CURRENT LAYOUT

Selecting Start New Layout will erase the current layout from memory. To work on a current layout, select the second choice. You will be returned to the garden layout process with all existing plantings intact.

A brief description of the layout requirements is displayed. The garden is assumed to be roughly rectangular and only the row length needs to be specified at first. The number of rows will be dictated by your allotted space and the types of vegetables to be planted. The spacing between rows depends on the types of vegetables you are planting. Row lengths can be a minimum of 6 feet and a maximum of 100 feet.

When a valid row length has been entered, the garden layout screen is displayed. The prompts at the bottom of the screen show your options. You can select the row you wish to "plant" (with the arrow keys) or press Escape to quit the Row Planner and display the Vegetable Garden Planner menu. When you select a row, the corresponding row number blinks to the left of the screen display. When you have the one you want, press Return. The row number stops blinking and new prompts appear at the bottom of the screen, with a short two-item menu:

PLANT ROW M
RE-SELECT ROW

If you select Plant Row M (M is the row number), proceed as described below in Selecting the Vegetable. Select the Re-select Row option to return to the blinking numbers to the left of the screen and re-select the row to plant.

Selecting the Vegetable

If you selected the Plant Row *M* option, pressing the arrow keys will allow you to view each vegetable available to you for planting. As you press the arrow keys, each shape appears in a display window at the bottom of the screen, sequentially. Select one by pressing the Return key. To see a description of any shape, press the question mark key when the vegetable is displayed.

When you have made a vegetable selection, press the Return key to proceed. Now enter the number of feet you wish to plant with this vegetable. The number should not exceed the length remaining in the row. The length remaining is shown in the first line of the prompt area.

After your entry, the appropriate shape is "planted" in the garden and displayed in the corresponding row on the screen. Whenever you re-select a row for planting, the old row is erased and replaced with your new entry. You must replant entire rows at a time, not individual vegetables or partial rows.

Plant as many rows with as great an assortment of vegetables as you wish at this time. When the garden is planted as you like, press Escape to exit the row planner and return to the Vegetable Garden Planner menu. Re-selecting the Garden Layout option will allow you to either continue working on your layout plan or clear the layout and start again, as the menu allows.

The Start New Layout option begins the layout process by requesting a new row length. Your previous layout data is zeroed, but the graphic display remains in memory. It may still be viewed, but your garden cannot be evaluated economically.

The Work on Current Layout option simply allows you to return to the garden layout and start again or edit what you have already done. Pressing Escape takes you back to the Garden Planner menu also. Only the Work on Current Layout option lets you edit or change the existing layout.

PLAYING IN YOUR GARDEN

At this point, return to the main menu by pressing Escape twice. Select the Graphic/Disk Utilities option. Figure 3 shows this menu. From it you may select to view the graphic screen, save it, load another graphic, change disk drives, or print the screen (if you have the interface card capability). For more details, see the section Graphic/Disk Utilities later in the article.

You may execute any of the options except loading another graphic without losing your current layout. Now, return to the main menu by pressing Escape.

HOW MUCH DO YOU SAVE?

From the main menu, select Vegetable Garden Planner. From the Vegetable Garden Planner menu, select Garden Economics. This option allows you to check the value of your expected (or actual) harvest against your costs.

For each vegetable, the number of feet planted is displayed on screen. You must figure out the yield per foot. Perhaps, in the case of tomatoes, you may decide that pounds per foot is a more meaningful measure than tomatoes per foot. For radishes, you may settle on bunches per foot if that's how you buy radishes in the grocery store.

The most common purchase method or the easiest way to measure should determine how you enter your yield. Remember, however, that the next entry requires that you enter the retail cost per unit. So if you have measured radishes in bunches, you should be consistent and enter the retail price per bunch, rather than the price per pound.

An updated total is displayed at the bottom of the screen after each evaluation is made. After values for all planted vegetables have been entered, you will have to enter values for four areas of expenses, which will reduce the economic value of your yield. The cost of your seeds, plants, etc., should be entered. Also, the expenses you incurred preparing the garden (on fertilizer, tilling, watering and insect/pest control) should also be entered. When enter-

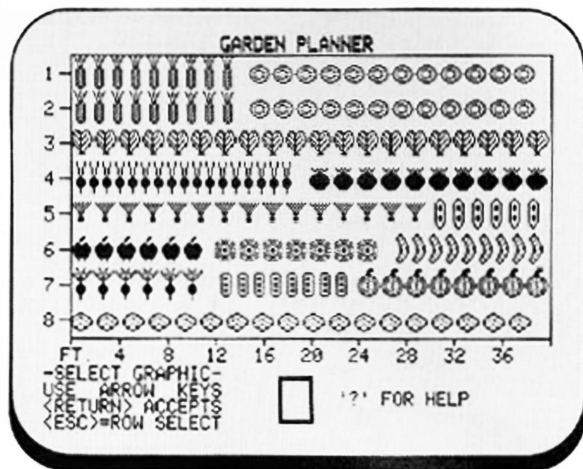


FIGURE 4: Sample Garden Layout

ing the data for this section, pressing Return moves the cursor to the next entry item while Escape returns you to the Garden Planner menu. After you have completed the economic evaluation, the Garden Planner menu is displayed again.

The economic evaluation may reveal that you don't save as much as you think you do. Of course, no economic evaluation can measure the inner satisfaction of growing your own plants. For many, this is the real reason for planting a garden. Who can put a price on the smell of growing things or the challenge of outwitting insect pests?

Garden Planner was designed to offer maximum flexibility in the planning of a garden. You can print screen displays, save them, do step-by-step economic evaluations as you go along, plant and replant rows until you are satisfied, and generally alter any garden as much as you like before deciding on a final plan.

If your garden planning includes more than eight rows, you will have to create another plan to accommodate them, then simply glue the printouts together to make your final plan. Figure 4 shows a sample garden made up of eight rows planted with various vegetables.

GRAPHIC/DISK UTILITIES

Return to the main menu by pressing Escape from the Vegetable Garden Planner menu, and select Graphic/Disk Utilities. This menu (Figure 3) offers the most diverse activities. If you select View Graphic, you can toggle the graphics page and the text page (which contains the menu) by simply pressing the Return key. This gives you a visual confirmation of a screen you might want to save or one you've just loaded.

The Save Graphic and Load Graphic options are very similar in function and appearance. When selecting either, you will see the same submenu selections:

```
ENTER FILE NAME  
CATALOG
```

The second choice simply displays a catalog of the disk in the current drive. You may return to the Graphic/Disk Utilities menu by pressing Escape.

If you select the Enter File Name option, you will need to enter a file name, with or without the file specification .LO (Layout). If you do not append this suffix, the program will. By pressing Return when prompted for a file name, you can exit the routine and return to the Graphic/Disk Utilities menu.

If you enter a valid file name, the SAVE or LOAD is executed. When a LOAD is made, the current garden layout is lost. The SAVE does not affect the layout and you may return at will to the Garden Economics section. The Graphic/Disk Utilities menu is redisplayed after the LOAD/SAVE operations.

The Select Disk Drive option allows you to view the currently active (last accessed) disk drive and change it, if you wish. This lets you easily access a data disk on which you may have saved several garden layouts. Or, if you get a DISK FULL error, you can move data to another disk without difficulty. If you have two disk drives, I'd suggest that you save all of your graphics on a single disk for easy access.

If you have a printer and a graphics capable interface card, and if you have defined the graphics print command using the Configure System option, the Print Graphic option will appear in the menu (see Figure 3). If any of these conditions are not met, the menu choice will not appear, and you will not be able to print the garden graphics directly from the Garden Planner program. Instead you should save it from the Graphics/Disk Utilities menu and print it out later from a separate program.

ENTERING THE PROGRAMS

To enter the programs, start by keying in the program shown in Listing 1. Save it with the command:

SAVE GARDEN.PLANNER

Then enter the program shown in Listing 2 and save it with the command:

SAVE PG.GUIDE.CREATE

Enter the program shown in Listing 3 and save it with the command:

SAVE PG.NAMES.CREATE

Enter the program shown in Listing 4 and save it with the command:

SAVE CONFIGURE

If you have an assembler, enter the source code from Listing 5 and assemble it using CURSOR.BAR.4 for the name of the object file. If you don't have an assembler, enter the Monitor with CALL -151 and key in the hex code. Save the program with the command:

BSAVE CURSOR.BAR.4,A\$300,L\$C9

Enter the Monitor with CALL -151 and key in the hex code from Listing 6. Save the table with the command:

BSAVE VEG.SHAPES,A\$1400,L\$2E4

If you already have the TEXT.SHAPES shape table (i.e., you entered it as part of "Hi-Res Characters" in Vol. 5/No. 12), load it into memory using the command:

BLOAD filename,A\$1000

Then save it onto your Garden Planner disk using the command below.

If you don't have TEXT.SHAPES, enter the Monitor with CALL -151 and key in the hex codes from Listing 7. Save the table with the command:

BSAVE TEXT.SHAPES,A\$1000,L\$313

For help with entering *Nibble* listings, see the directions in the Program Listings section.

A FEW PROGRAMMING TRICKS

The GARDEN.PLANNER uses a few tricks that might make nice additions to your programming repertoire. For instance, you might wonder why I used two small shape tables instead of a single, large shape table. The reason is twofold. First, I already had one of the tables (the text shapes), and I didn't want to reinvent the wheel. I only needed to create the vegetable shape table. Second, if I wanted to use either shape table in the future, two separate tables would be easier to use.

But (as you probably know) Applesoft has no provision for multiple shape tables. A single two-byte address holds the beginning address for the only shape table! That pointer is located at \$E8-\$E9 (232-233) in the normal low byte, high byte order. See line 140 in GARDEN.PLANNER for the variable assignments.

The key to using multiple tables was to POKE the address of the shape table to be used next into the pointer address just prior to using that table. That way, any number of tables could be used, as long as the beginning address was POKEd into the pointer address before DRAWing any shape.

Suppose, though, that the same number of shapes were not contained in each table. Couldn't the program try to DRAW a shape that didn't exist? Certainly that is a consideration. But remember that the first two bytes of the shape table contain the number of shapes in the table. So, if the program PEEKs the first two bytes of the address in the pointer, it will find the number of shapes stored there. This value can be stored in a simple numeric variable; this is what was done with the variable NS in line 640.

Since calculations involving variables are faster than calculations with literal values, the decimal equivalent of the high byte of the shape table address is converted to the variable ST (Shape Table) and POKEd into memory whenever the table to be used is changed. You will not notice any slowdown in drawing text or vegetable graphics to the screen. All the drawing from one table is done before the drawing begins from the other table.

The flexibility of this procedure is obvious. You can put your shape tables anywhere in available memory, and if you keep your shapes reasonably short, you can fit them into any gaps in memory. In this case, I kept each table on a page boundary so that only a single variable (the high byte) needed to be POKEd into memory location 233. Now you can have as many shape tables in memory as you want (memory permitting) rather than a single gigantic table.

MORE TRICKS

Because Garden Planner requires a number of separate files on the disk, I avoid later program crashes by verifying the existence of these files at the beginning of the program. If one of them is missing, then a message is displayed and the program stops.

Instead of the VERIFY command, which can be time-consuming under DOS 3.3, I used the UNLOCK command, which works the same under both systems. This speeds the check for the file since when it tries to unlock a file that doesn't exist, a FILE NOT FOUND error is evoked (error code 6). Using the ONERR GOTO error trapping routine, it is simple to intercept that error. When the files are present, each UNLOCK unlocks a file that is already unlocked without evoking an error, and program flow will continue unabated.

CUSTOMIZING THE PROGRAM

Suppose you don't like the picture of a carrot. Or maybe you think the spinach looks like a butterfly. Can you change it? Sure. But it may not be easy.

The VEG.SHAPES table was created using the Font Editor from the Beagle Bros.' Apple Mechanic disk. I preferred the Font Editor over the Shape Editor on the same disk because of the greater number of available shapes. Compressing the table from the 96 shapes in the Font Table to 16 vegetable shapes was a bit tricky. I truncated the unused end of the table and used *Nibble*'s MLE program to edit out references to the nonexistent shapes.

However, any shape editor can be used. Rename the shapes in the DATA statements to match your new ones and, if you have more than 16 shapes, you may want to delete the View Shapes option. More than 16 shapes and their descriptions won't fit easily on the screen.

The VEG.SHAPES table is strategically placed at location \$1400. That leaves 3K of memory between this location and the beginning of graphics page 1 — a pretty big chunk of memory for a shape table. You'd have a hard time filling that much memory with veggies, but it's there if you want to use it.

Happy gardening!



Garden Planner

Article on page 14

Listing 1 for Garden Planner

GARDEN.PLANNER

```
10 REM *****
20 REM * GARDEN.PLANNER *
30 REM * BY MARK R. CRAVEN *
40 REM * COPYRIGHT (C) 1987 *
50 REM * BY MICROSPARC, INC *
60 REM * CONCORD, MA 01742 *
70 REM *****
80 HOME :DS = CHR$ (4)
90 ONERR GOTO 3880
100 PRINT DS"UNLOCK PG.GUIDE": PRINT DS"UNLO
CK PG.NAMES": PRINT DS"UNLOCK VEG.SHAPES
": PRINT DS"UNLOCK TEXT.SHAPES": PRINT D
S"UNLOCK CURSOR.BAR.4": PRINT
110 POKE 216,0: REM NO ONERR
120 IF PEEK (104) < > 64 THEN POKE 103,1:
POKE 104,64: POKE 16384,0: VTAB 10: HTAB
10: PRINT "LOADING PROGRAM FILES": PRINT
: PRINT DS"RUN GARDEN.PLANNER"
130 HIMEM: 38400: DIM VG$(16),VG$(16),T$(8,1
6),EC$(16),EX$(4),PG$(45)
140 SCALE= 1: ROT= 0: HCOLOR= 3: AD = 974: EM =
975: POKE 230,32: TABLE = 233: AS = 16: VG =
20: KD = - 16384: KB = - 16368: CUR = 768
: CALL 62450: HOME
150 VTAB 10: HTAB 10: PRINT "READING PROGRAM
DATA"
160 MS(1) = "VEGETABLE GARDEN PLANNER": MS(2) =
"GRAPHIC/DISK UTILITIES": MS(3) = "CONFIG
URE SYSTEM": MS(4) = "QUIT PROGRAM"
170 DS = CHR$ (4): OP$ = "OPEN ": CL$ = "CLOSE
": RD$ = "READ ": WR$ = "WRITE ": US = "UN
LOCK ": BL$ = "BLOAD ": BS$ = "BSAVE ": VF$
= "VERIFY ": AD$ = ",A$2000": LE$ = ",L$2
000"
180 FOR I = 1 TO 39: L$ = L$ + "=": NEXT
190 EX$(1) = "SEED, SETS, PLANTS, ETC.": EX$(2
) = "FERTILIZER, TILLING, ETC.": EX$(3) =
"INSECTICIDES, PESTICIDES, HERBICIDES": E
X$(4) = "WATERING, HARVESTING"
200 HOME : VTAB 10: HTAB 10: PRINT "LOADING
PG.NAMES"
210 F$ = "PG.NAMES"
220 PRINT DSOP$F$
230 PRINT DS$RDSF$
240 FOR I = 1 TO 45: INPUT PG$(I): NEXT
250 FOR I = 1 TO 16: INPUT VG$(I),NS:VG$(I) =
VAL (NS): NEXT
260 PRINT DSCL$
270 DEF FN V(V) = 19 + 17 * (V - 1): REM GR
APHICS POS'NING FN
280 REM LOAD BINARY FILES
290 ONERR GOTO 4010
300 HOME : VTAB 10: HTAB 10: PRINT "LOADING
BINARY FILES": PRINT
310 PRINT DSBL$ "CURSOR.BAR.4,A$300": REM 768
320 PRINT : PRINT DSBL$ "TEXT.SHAPES,A$1000":
REM 4096
330 PRINT : PRINT DSBL$ "VEG.SHAPES,A$1400": REM
5120
340 POKE 232,0: POKE TABLE,AS: REM TEXT.SHAP
ES ADDRESS ($1000)
350 GOSUB 640
360 REM CONFIGURE
370 F$ = "GP.CONFIG"
380 POKE EM,1
390 ONERR GOTO 3990
400 PRINT : PRINT DSUSF$
410 PRINT DSOP$F$
420 PRINT DS$RDSF$
430 INPUT PT: INPUT SL: INPUT GP: INPUT GP$:
INPUT CL: INPUT SD: INPUT SD$
440 PRINT DSCL$
450 POKE 216,0: REM NO ONERR
460 POKE EM,0: GOTO 1080
470 REM SUBROUTINES
480 VTAB 20: PRINT L$: VTAB 21: HTAB 1: INVERSE
: PRINT "<->": NORMAL : PRINT " ": INVERSE
: PRINT ">": NORMAL : PRINT " TO SELEC
T ...<": INVERSE : PRINT "RETURN": NORMAL
: PRINT "> TO EXECUTE": PRINT " UP OR
DOWN ARROW KEYS MAY BE USED"
490 PRINT " <ESC> RETURNS TO PREVIOUS MEN
U"
500 VTAB 24: PRINT L$: RETURN
510 VTAB 3: HTAB 1: PRINT "PRINTER ": RETURN
520 VTAB 3: HTAB 9: PRINT "IN SLOT #": SL: RETURN
530 VTAB 3: HTAB 1: PRINT "NO PRINTER
": RETURN : REM 8 SPACES
540 VTAB 3: HTAB 21: PRINT "WITH GRAPHICS DU
MP": RETURN
550 VTAB 3: HTAB 21: PRINT " NO GRAPHICS DU
MP": RETURN
560 VTAB 5: HTAB 21: PRINT "WITH TEXT DUMP":
: CALL - 868: RETURN
570 VTAB 5: HTAB 21: PRINT " NO TEXT DUMP":
: CALL - 868: RETURN
580 VTAB 5: HTAB 21: CALL - 868: RETURN
590 VTAB 5: HTAB 1: PRINT "COLOR MONITOR
": RETURN
600 VTAB 5: HTAB 1: PRINT "MONOCHROME MONITO
R": RETURN
610 S$ = " ": FOR I = 1 TO LEN (X$): IF ASC
(MID$ (X$, I, 1)) < 32 THEN S$ = S$ + "^"
+ CHR$ (ASC (MID$ (X$, I, 1)) + 64): NEXT
: GOTO 630
620 S$ = S$ + MID$ (GP$, I, 1): NEXT
630 RETURN
640 NS = PEEK (PEEK (TABLE) + 256): RETURN
: REM # OF SHAPES
650 HCOLOR= 0: FOR B = FN V(R) - 8 TO FN V
(R) + 8: HPLOR R1 + 19, B TO R2 + 17, B: NEXT
: HCOLOR= 3: RETURN
660 DRAW S AT A + 3, C + 2: RETURN : REM DRAW
VEG
670 XDRAW S AT A + 3, C + 2: RETURN : REM XDR
AW VEG
680 POKE 35,24: HOME : POKE - 16301,0: VTAB
22: PRINT "SHAPE REPRESENTS ": INVERSE
: PRINT VG$(S): NORMAL
690 HTAB 1: VTAB 24: PRINT "PRESS <RETURN> T
O CONTINUE .":
700 POKE KB,0: GET X$: POKE - 16302,0: RETURN
710 R = R - 1 + 8 * (R = 1): RETURN : REM UP
ROW
720 R = R + 1 - 8 * (R = 8): RETURN : REM DOW
N ONE
730 ER = 0: IF X$ = "" THEN RETURN
740 IF VAL (X$) < 0 THEN ER = 1: RETURN
750 IF LEN (X$) > 5 THEN ER = 1: RETURN
760 FOR K = 1 TO LEN (X$): IF ASC (MID$ (
X$, K, 1)) < 46 OR ASC (MID$ (X$, K, 1)) >
57 OR ASC (MID$ (X$, K, 1)) = 47 THEN ER
= 1: K = LEN (X$): NEXT : RETURN
770 NEXT : RETURN
780 FOR B = 1 TO LEN (X$): DRAW ASC (MID$
(X$, B, 1)) - 31 AT H + 6 + B, V: NEXT : RETURN
790 FOR I = 1 TO 8: DRAW I AT 1, I + 16 + 17:
NEXT : FOR I = 9 TO NS: DRAW I AT 142, 1
6 + (I - 8) + 17: NEXT : RETURN
800 HCOLOR= 3 + 2 * CL: FOR I = 0 TO 18: DRAW
2 AT I + 14, 10: DRAW 2 AT I + 14, 165: NEXT
: HCOLOR= 3: RETURN
810 REM FIX DECIMALS
820 ER = 0: IF VAL (X$) < .01 AND VAL (X$) >
= 0 THEN X$ = "0.00": RETURN : REM CHEC
K DEC. FLAG
830 FOR J = 1 TO LEN (X$): IF MID$ (X$, J, 1
) = "." THEN ER = 1: X$ = X$ + "00": X$ =
LEFT$ (X$, J + 2): J = LEN (X$): NEXT : RETURN
```


Listing 1 for Garden Planner

GARDEN.PLANNER (continued)

```

840 NEXT : IF ER = 1 THEN ER = 0: RETURN
850 X$ = X$ + ".00": RETURN
860 IF LEFT$(X$,1) = " " THEN X$ = RIGHT$(
(X$, LEN (X$) - 1): GOTO 860
870 RETURN
880 FOR J = I TO I + 39: A = PEEK (J): A = A +
(A < 32) * 192: A = A + (A < 64) * 128: A =
A + (A < 96) * 64: A = A + (A < 128) * 64
: A = A + (A < 160) * 64: PRINT CHR$(A)
:
890 IF PEEK (KD) = 155 THEN J = I + 39: NEXT
: POP : GOTO 1050: REM STOP PRINTING
900 NEXT J: PRINT CHR$(13):
910 RETURN
920 VTAB 22: PRINT L$: PRINT : VTAB 23: PRINT
TAB(6)"PRESS <RETURN> TO CONTINUE...":
: IF PT THEN PRINT : PRINT TAB(7)"PRE
SS 'P' TO PRINT SCREEN":
930 RETURN
940 VTAB 24: PRINT " <ESC> TO EXIT ROUTI
NE TO MENU": RETURN
950 FOR I = 1 TO 16: EC%(I) = 0: NEXT : RETURN
: REM ZERO FOOTAGES
960 ER = PEEK (222): TEXT : HOME : POKE 216,
0: RETURN : REM NO ONERR
970 LINE = ( PEEK (4) + 1) / 2: RETURN
980 PRINT CHR$(4)"OPEN PG.GUIDE.L60": PRINT
CHR$(4)"READ PG.GUIDE.R": I
990 INPUT DP$,DS$,DR$,GM$,ST$,MT$: PRINT DS"
CLOSE": RETURN
1000 PRINT DS"PR#": SL: PRINT SD$: GOTO 1050
1010 PRINT DS"PR#": SL: PRINT CHR$(9)"80N"
1020 FOR I = 1024 TO 1920 STEP 128: GOSUB 88
0: NEXT I
1030 FOR I = 1064 TO 1960 STEP 128: GOSUB 88
0: NEXT I
1040 FOR I = 1104 TO 1616 - 384 + (NR < 6) STEP
128: GOSUB 880: NEXT I
1050 PRINT : PRINT DS"PR#": RETURN
1060 INVERSE : VTAB 15: HTAB 3: PRINT " TH
IS OPTION WILL DESTROY ANY " : HTAB 3:
PRINT " EXISTING LAYOUT
": NORMAL : RETURN : REM 3/4/10/10 S
PACES
1070 REM MAIN MENU
1080 TEXT : HOME : PRINT TAB(5)"THE GARDEN
PLANNER - MAIN MENU": PRINT : PRINT "
COPYRIGHT 1987 BY MICROSPARC, INC.": PRINT
L$
1090 VTAB 8: HTAB 8: PRINT M$(1): VTAB 10: HTAB
9: PRINT M$(2): VTAB 12: HTAB 12: PRINT
M$(3): VTAB 14: HTAB 14: PRINT M$(4)
1100 GOSUB 480
1110 CALL CUR,8,18,7,32
1120 GOSUB 970: POKE AD,LINE
1130 ON LINE GOTO 1220,2630,3750,1150
1140 GOTO 1080
1150 TEXT : HOME : VTAB 10: HTAB 5: PRINT "Q
UIT PROGRAM? (Y/N) ": POKE KB,0: GET X$:
: IF X$ = "Y" OR X$ = CHR$(121) THEN 1
170
1160 GOTO 1080
1170 TEXT : HOME : END
1180 HOME : CALL 62450: HGR : POKE 49234,0: POKE
TABLE,AS:V = 5: GOSUB 780: POKE TABLE,ST
: GOSUB 640: GOSUB 800: GOSUB 790: POKE
TABLE,AS: GOSUB 640: RETURN
1190 POKE TABLE,AS: X$ = "PRESS <RETURN> TO C
ONTINUE...": V = 188: H = 10: GOSUB 780: PRINT
CHR$(7): POKE KB,0: GET X$: HOME : TEXT
1200 RETURN
1210 REM PLAN GARDEN
1220 TEXT : HOME : VTAB 1: HTAB 9: PRINT M$(
PEEK (AD)): PRINT : PRINT L$: GOSUB 480
1230 VTAB 8: HTAB 13: PRINT "GARDEN LAYOUT"
: VTAB 10: PRINT TAB(13)"PLANTING GUID
E": VTAB 12: PRINT TAB(8)"VIEW SHAPES/
DESCRIPTIONS": VTAB 14: PRINT TAB(12)"
GARDEN ECONOMICS"
1240 CALL CUR,8,14,7,32
1250 GOSUB 970
1260 ON LINE GOTO 1810,1370,1280,3130
1270 GOTO 1080: REM MAIN MENU (ESC)
1280 IF LN < 6 THEN LN = 1320
1290 HOME : VTAB 10: PRINT "YOU HAVE AN ACTI
VE LAYOUT IN PROGRESS.": PRINT : PRINT "
VIEWING THE VEGETABLE GRAPHICS": PRINT "
WILL DESTROY THE EXISTING LAYOUT.": PRINT
: PRINT "DO YOU WISH TO CONTINUE ANYWAY?
N": HTAB 33
1300 POKE KB,0: GET Z$: IF Z$ = "Y" OR Z$ =
CHR$(121) THEN LN = 0: GOTO 1320
1310 GOTO 1220
1320 LN = 0: X$ = "VEGETABLE GRAPHICS": H = 80:
ST = VG: GOSUB 1180
1330 FOR J = 1 TO 8: X$ = VG$(J): H = 10: V = 1
6 * J + 25: GOSUB 780: NEXT : FOR J = 9 TO
16: X$ = VG$(J): H = 150: V = 16 + (J - 8) +
25: GOSUB 780: NEXT
1340 GOSUB 1190
1350 GOTO 1220
1360 REM PLANTING GUIDE
1370 TEXT : HOME : PRINT TAB(13)"PLANTING
GUIDE": PRINT L$: PRINT
1380 VTAB 8: PRINT TAB(8)"DISPLAY SINGLE V
EGETABLE": IF PT = 1 THEN PRINT : PRINT
TAB(6)"PRINT ENTIRE PLANTING GUIDE"
1390 GOSUB 480: REM PROMPTS
1400 CALL CUR,8,12,5,34
1410 GOSUB 970
1420 ON LINE + (LINE = 2 AND PT = 0) GOTO 14
40,1550
1430 GOTO 1220
1440 TEXT : HOME : HTAB 12: PRINT "SELECT VE
GETABLE": PRINT L$: PRINT
1450 GOSUB 480: J = 1: POKE 34,2: POKE 35,19:
REM SET WINDOW
1460 PRINT : VTAB 3: FOR I = J TO J + 14: PRINT
TAB(20 - LEN (PG$(I)) / 2)PG$(I): NEXT
: HTAB 12: PRINT "SEE MORE CHOICES": CALL
CUR,3,18,8,32
1470 LINE = PEEK (4): IF LINE < > 16 THEN 1
500
1480 J = J + 15: IF J = 46 THEN J = 1
1490 HOME : GOTO 1460
1500 IF LINE = 0 THEN NEXT 1370
1510 R = LINE + J - 1: REM FILE RECORD #
1520 I = R: GOSUB 980: REM READ RECORD
1530 GOTO 1700: REM SINGLE RECORD
1540 REM PRINT WHOLE GUIDE
1550 HOME : VTAB 10: PRINT TAB(5)"PRESS <E
SC> TO STOP PRINTING": PRINT : PRINT CHR$(
4)"PR#": SL: PRINT CHR$(9)"80N": REM T
URN ON PRINTER
1560 PRINT SPC(28)"VEGETABLE PLANTING GUID
E"
1570 PRINT L$L$=="
1580 PRINT SPC(30)"DISTANCE DISTANCE
DAYS TO" SPC(9)"DAYS TO"
1590 PRINT SPC(20)"DEPTH TO BETWEEN
BETWEEN GERMI- SOIL MATURITY": REM
SPACES = 3,6,4,2,3
1600 PRINT SPC(22)"PLANT PLANTS
ROWS TION TEMP.*": REM SPACES =
4,8,7,3
1610 PRINT " VEGETABLE SEEDS (I
NCHES) (INCHES)": REM SPACES=3,10,3,
5
1620 PRINT L$L$=="
1630 FOR I = 1 TO 45: GOSUB 980: REM READ RE
CORD
1640 IF PEEK (KD) = 155 THEN 1690: REM ESC
OUT
1650 PRINT PG$(I) SPC(20 - LEN (PG$(I)))DP
$ SPC(10 - LEN (DP$))DS$ SPC(13 - LEN
(DS$))DR$ SPC(12 - LEN (DR$))GM$ SPC(
10 - LEN (GM$))ST$ SPC(7 - LEN (ST$))
MT$
1660 NEXT
1670 PRINT L$L$=="
1680 PRINT " + 1=NEEDS COOL SOIL": PRINT
" 2=TOLERATES COOL SOIL": PRINT "
3=NEEDS WARM SOIL": REM SPACES=3,2,6
,6
1690 PRINT : PRINT DS"PR#": GOTO 1370
1700 TEXT : HOME : PRINT TAB(13)"PLANTING
GUIDE": HTAB 20 - LEN (PG$(I)) / 2: PRINT
PG$(I): PRINT L$

```

```

1710 VTAB 4: HTAB 1: PRINT " PLANTING DI
STANCE DISTANCE": PRINT " DEPTH
BETWEEN BETWEEN": PRINT " (IN.)
PLANTS ROWS": REM SPACES=3,3,3
,5,5,4,5,5,6
1720 PRINT L$
1730 HTAB 8 - LEN (DP$) / 2: PRINT DP$: HTAB
19 - LEN (DS$) / 2: PRINT DS$: HTAB 31
- LEN (DR$) / 2: PRINT DR$
1740 VTAB 12: HTAB 1: PRINT L$: PRINT " #
OF DAYS 1=COOL SOIL DAYS": PRINT
" TO 2=COOL OK TO": PRINT
" GERMINATION 3=WARM SOIL MATURITY
": REM SPACES=3,4,6,7,7,9,2,3,4
1750 PRINT L$
1760 HTAB 8 - LEN (GM$) / 2: PRINT GM$: HTAB
23: PRINT ST$: HTAB 35 - LEN (MT$) / 2
: PRINT MT$
1770 GOSUB 920: GOSUB 700: REM PROMPTS
1780 IF PT = 1 AND (X$ = "P" OR X$ = CHR$ (
112)) THEN VTAB 20: HTAB 1: CALL - 958
: PRINT L$: VTAB 22: PRINT : ON SD GOSUB
1000: ON SD GOTO 1370: VTAB 23: PRINT TAB(
8)"PRESS <ESC> TO CANCEL PRINT": GOSUB 1
010: REM SCREEN DUMP
1790 GOTO 1440: REM RESELECT VEGGIE
1800 REM GRAPHIC LAYOUT
1810 HOME : HTAB 14: PRINT "GARDEN LAYOUT": PRINT
L$
1820 IF LN < 6 THEN 1870: REM NO LAYOUT
1830 GOSUB 480: VTAB 10: HTAB 12: PRINT "STA
RT NEW LAYOUT": VTAB 12: HTAB 9: PRINT "
WORK ON CURRENT LAYOUT"
1840 CALL CUR,10,12,6,33: GOSUB 970: ON LINE
GOTO 1870,1860
1850 GOTO 1220
1860 POKE - 16297,0: POKE - 16304,0: POKE
- 16301,0: GOTO 2070
1870 VTAB 4: PRINT "THIS LAYOUT ASSUMES A RE
CTANGULAR,": PRINT "SQUARE, OR NEARLY RE
CTANGULAR GARDEN,": CALL - 958: GOSUB 1
060
1880 VTAB 22: HTAB 1: PRINT L$: VTAB 23: HTAB
1: PRINT "<RETURN> TO CONTINUE..ANY OTHE
R TO QUIT": GET X$: IF X$ < > CHR$ (1
3) THEN 1220
1890 GOSUB 950
1900 VTAB 4: HTAB 1: CALL - 958: VTAB 5: HTAB
1: PRINT "ENTER THE LONGEST DIMENSION (L
ENGTH)": VTAB 21: PRINT L$: PRINT TAB(
11)"RANGE = 6-100 FT,": PRINT TAB(5)"<
RETURN> WITHOUT ENTRY EXITS"
1910 VTAB 7: PRINT " LENGTH (IN FEET) = ": INPUT
": LN$: IF LN$ = "" OR LN$ = CHR$ (27) THEN
1220
1920 IF VAL (LN$) > 100 OR VAL (LN$) < 6 THEN
PRINT CHR$ (7): VTAB 7: HTAB 19: CALL
- 958: GOTO 1900
1930 CALL - 958
1940 LN = INT ( VAL (LN$)):L1 = LN
1950 IF LN / 10 < > INT (LN / 10) THEN LN =
LN + 1: GOTO 1950: REM MAKE MULTIPLE OF
10
1960 F1 = LN / 10:F2 = 2 + 2 * (F1 = 2) + 3 *
(F1 = 3) + 2 * (F1 = 4) + 3 * (F1 = 5) +
8 * (F1 > 5):SC = 10 * F1 / F2: REM SCAL
E X-AXIS
1970 CALL 62450: HOME : POKE - 16300,0: POKE
- 16301,0: POKE - 16297,0: POKE - 163
04,0: REM GRAFIX PG1
1980 ST = 16: GOSUB 640:X$ = "GARDEN PLANNER"
:H = (((260 * L1) / LN) - 19) / 2 - 3 *
LEN (X$) + 19:V = 4: GOSUB 780:V = 5: GOSUB
780
1990 HPLLOT 19,10 TO 19 + (260 * L1) / LN,10 TO
19 + (260 * L1) / LN,147 TO 19,147 TO 19
,10
2000 X$ = "FT,":H = 10:V = 155: GOSUB 780
2010 FOR I = 19 TO 19 + (260 * L1) / LN STEP
260 / SC: HPLLOT I,147 TO I,150: NEXT
2020 FOR J = 1 TO (L1 / F2) - (L1 = LN):X$ =
STR$ (J * F2):H = 19 + J * (260 / SC) -
9 + 3 * ( LEN (X$) = 1):V = 155: GOSUB 7
80: NEXT
2030 FOR I = 19 TO 147 STEP 17: HPLLOT 13,I TO
19,I: NEXT
2040 FOR I = 1 TO 8:X$ = STR$ (I):H = 1:V =
FN V(I): GOSUB 780: NEXT
2050 R = 1: FOR I = 1 TO 16:EC$(I) = 0: NEXT
: REM START AT ROW #1,ZERO ECONOMICS COU
NTER
2060 FOR I = 1 TO 16:T%(R,I) = 0: NEXT : REM
ASCII SHAPES,ZERO TEMP. ECONOMICS COUNT
ER
2070 POKE TABLE,AS: GOSUB 640
2080 HOME : VTAB 22: PRINT " SELECT ROW TO P
LANT -- USE <- -> KEYS": VTAB 23: PRINT
" <RETURN> TO ACCEPT -- <ESC> TO QUIT":
POKE 35,21: HOME
2090 DRAW R + 17 AT 7, FN V(R): IF PEEK (KD
) > 128 THEN POKE KB,0: GOTO 2110
2100 XDRAW R + 17 AT 7, FN V(R): GOTO 2090
2110 IF PEEK (KD) = 27 THEN HOME : TEXT : GOTO
1220: REM <ESC>
2120 IF PEEK (KD) = 13 THEN 2210: REM ROW I
S CHOSEN AND=R
2130 IF PEEK (KD) < > 8 AND PEEK (KD) < >
11 THEN 2150
2140 GOSUB 710: GOTO 2090: REM UP
2150 IF PEEK (KD) < > 10 AND PEEK (KD) <
> 21 THEN 2180: REM DOWN
2160 GOSUB 720: GOTO 2090: REM DOWN ONE
2170 GOTO 2090
2180 PRINT CHR$ (7): GOTO 2090
2190 IF LN < 6 THEN TEXT : HOME : VTAB 10: HTAB
10: PRINT "NO LAYOUT IN PROGRESS!": GOSUB
690: GOTO 1220
2200 POKE - 16301,0: POKE - 16304,0: POKE
- 16300,0: HOME : GOTO 2070
2210 POKE 35,24: HOME : VTAB 22: PRINT " SEL
ECT CHOICE : PLANT ROW "R: VTAB 23: PRINT
" USE ARROW KEYS" TAB(18) " RE-SELECT R
OW": PRINT "<RETURN> ACCEPTS <ESC>=RE-
SELECT": CALL CUR,22,23,19,35:LINE = PEEK
(4)
2220 ON LINE GOTO 2240
2230 HOME : GOTO 2070
2240 FOR I = 1 TO 16: IF T%(R,I) > 0 THEN I =
16: NEXT : GOTO 2260
2250 NEXT : GOTO 2280
2260 FOR I = 1 TO 16: IF T%(R,I) > 0 THEN EC
$(I) = EC$(I) - T%(R,I):T%(R,I) = 0: NEXT
: GOTO 2280
2270 NEXT
2280 RW = 0:R1 = 1:R2 = (260 * L1) / LN: GOSUB
650: REM ERASE ROW FOR PLANTING & ZERO P
LANTED LENGTH
2290 HOME : POKE TABLE,AS: GOSUB 640: REM AS
CII SHAPES
2300 POKE - 16302,0: REM FULL GRAPHICS
2310 X$ = "-SELECT GRAPHIC-":V = 164:H = 1: GOSUB
780:X$ = "USE ARROW KEYS":V = 172: GOSUB
780:X$ = "<RETURN> ACCEPTS":V = 180: GOSUB
780:X$ = "<ESC>=ROW SELECT":V = 188: GOSUB
780
2320 X$ = "'?' FOR HELP":V = 176:H = 160: GOSUB
780
2330 A = 132:B = A + 16:C = 167:D = C + 20
2340 HPLLOT A,C TO B,C TO B,D TO A,D TO A,C: HPLLOT
A - 1,C - 1 TO B + 1,C - 1 TO B + 1,D +
1 TO A - 1,D + 1 TO A - 1,C - 1: REM DRA
W SHAPE BOX
2350 POKE TABLE,VG: GOSUB 640: REM VEG SHAPE
S
2360 S = 1: GOSUB 660: REM FIRST SHAPE
2370 POKE KB,0: GET X$: IF X$ = CHR$ (27) THEN
GOSUB 670: HOME : POKE - 16301,0: GOTO
2070
2380 IF X$ = CHR$ (11) OR X$ = CHR$ (8) OR
X$ = CHR$ (21) OR X$ = CHR$ (10) THEN
GOSUB 670: REM ARROWS
2390 IF X$ = CHR$ (63) THEN GOSUB 680: GOTO
2370: REM HELP (?)
2400 IF X$ < > CHR$ (8) AND X$ < > CHR$
(11) THEN 2420
2410 S = S - 1 + 16 * (S = 1): GOTO 2440: REM
DOWN ONE OR WRAP AROUND
2420 IF X$ < > CHR$ (21) AND X$ < > CHR$
(10) THEN 2450
2430 S = S + 1 - 16 * (S = 16): REM UP ONE OR
WRAP

```


Listing 1 for Garden Planner

GARDEN.PLANNER (continued)

```

2440 GOSUB 660: GOTO 2370: REM DRAW & GET AN
      OTHER
2450 IF X$ < > CHR$(13) THEN PRINT CHR$(
      7): GOTO 2370
2460 POKE 35,24: HOME: POKE - 16301,0: VTAB
      24: PRINT "PRESS <RETURN> TO EXIT WITHOU
      T ENTRY": HTAB 1
2470 VTAB 22: PRINT "PLANT HOW MANY FEET? (0
      -"L1 - RW)": INPUT "": X$: IF X$ = "" THEN
      POKE - 16302,0: GOTO 2370
2480 IF VAL(X$) < = 0 THEN PRINT CHR$(
      7): GOTO 2460: REM NOT NUMBERS
2490 IF VAL(X$) > L1 - RW THEN HOME: VTAB
      22: PRINT CHR$(7): PRINT "ENTRY LONGE
      R THAN GARDEN ROW!": VTAB 24: PRINT "PRE
      SS <RETURN> TO CONTINUE..": POKE KB,0: GET
      X$: GOTO 2460
2500 HOME: R1 = RW: RW = RW + INT(VAL(X$)
      ): R2 = RW: T%(R,S) = T%(R,S) + INT(VAL
      (X$)): REM ADD NEW LENGTH OF ROW
2510 FOR I = (260 * R1) / LN TO (260 * R2) /
      LN STEP VG$(S): IF I > 260 * R2 / LN - 1
      2 THEN NEXT: GOTO 2530
2520 DRAW S AT I + 19, FN V(R) - 8: NEXT
2530 R3 = R1: R4 = R2: R1 = (260 * RW) / LN: R2 =
      (260 * L1) / LN: REM ERASE TO END OF ROW
2540 IF R2 = R1 THEN 2560: REM END OF ROW
2550 GOSUB 650
2560 R1 = R3: R2 = R4: REM RESTORE ROW PARMS
2570 IF L1 - RW > 0 THEN RW = R2: POKE - 16
      302,0: GOSUB 2590: GOTO 2370: REM MORE R
      OW TO FILL
2580 GOSUB 670: GOSUB 2590: GOSUB 720: GOTO
      2070: REM ADD ROW TOTALS AND RESELECT RO
      W
2590 EC$(S) = EC$(S) + T%(R,S): RETURN
2600 REM
2610 REM DISK/GRAPHICS UTILITIES
2620 REM
2630 TEXT: HOME: PRINT TAB(9)"GRAPHIC/DI
      SK UTILITIES": PRINT L$: PRINT
2640 POKE EM,2
2650 ONERR GOTO 3770
2660 VTAB 6: PRINT TAB(11)"VIEW GARDEN LA
      YOUT ": PRINT: PRINT TAB(11)"SAVE GA
      RDEN LAYOUT "
2670 PRINT: PRINT TAB(11)"LOAD GARDEN LA
      YOUT "
2680 PRINT: PRINT TAB(12)"SELECT DISK DRI
      VE"
2690 IF (GP = 1 AND GP$ < > "") THEN PRINT
      : PRINT TAB(11)"PRINT GARDEN LAYOUT "
2700 GOSUB 480
2710 CALL CUR,6,17,10,30
2720 GOSUB 970
2730 S% = LINE: REM LOAD FILE FLAG
2740 ON LINE + (LINE = 6 AND GP = 0) GOTO 27
      60,2810,2990,2770,3070,3110
2750 GOTO 1080: REM MAIN MENU
2760 HOME: POKE - 16297,0: POKE - 16301,0
      : POKE - 16300,0: POKE - 16304,0: GOSUB
      690: GOTO 2630: REM SEE GRAPHIC
2770 HOME: PRINT TAB(12)"SELECT DISK DRIV
      E": PRINT L$: GOSUB 480: VTAB 6: HTAB 13
      : PRINT "ACTIVE DRIVE=": INVERSE: PRINT
      PEEK(43624): NORMAL
2780 VTAB 11: HTAB 16: PRINT "DRIVE #1": PRINT
      : HTAB 16: PRINT "DRIVE #2"
2790 CALL CUR,11,13,15,24: GOSUB 970: IF LIN
      E < 1 THEN 2630: REM <ESC>
2800 POKE 43624,LINE:F$ = "": GOTO 2630: REM
      NEW DRIVE/NULL FILE NAME
2810 HOME: VTAB 11: PRINT TAB(10)"SAVE GA
      RDEN LAYOUT": PRINT L$: PRINT: GOSUB 48
      0
2820 VTAB 9: HTAB 12: PRINT "ENTER LAYOUT NA
      ME": PRINT: HTAB 17: PRINT "CATALOG"
2830 CALL CUR,9,12,11,29
2840 GOSUB 970: ON LINE GOTO 2860,2940
2850 GOTO 2630: REM <ESC>
2860 VTAB 3: CALL - 958: VTAB 23: PRINT "<R
      ETURN> WITHOUT ENTRY EXITS ROUTINE":
2870 X$ = "": VTAB 10: HTAB 11: PRINT "ENTER
  
```

```

LAYOUT NAME": PRINT TAB(6)"(10 CHARACT
      ERS MAX + '.LO)": VTAB 12: FOR I = 1 TO
      10: X$ = X$ + CHR$(95): NEXT
2880 PRINT: PRINT "NAME -> ": X$: VTAB 13: HTAB
      9: INPUT "": N$: IF LEN(N$) > 10 + 4 *
      (S% = 3) + 4 * (RIGHT$(N$,4) = ".LO" AND
      S% = 2) THEN PRINT CHR$(7): GOTO 286
      0
2890 IF N$ = "" THEN 2630
2900 IF ASC(N$) < 65 OR ASC(N$) > 90 THEN
      PRINT CHR$(7): GOTO 2870
2910 IF S% = 3 THEN S% = 0: GOTO 3000
2920 VTAB 15: IF RIGHT$(N$,3) < > ".LO" THEN
      N$ = N$ + ".LO"
2930 X$ = N$: PRINT "FILE NAME=": X$: PRINT: PRINT
      D$B$X$A$D$S$: GOTO 2630: REM SAVE LAYOU
      T
2940 HOME: PRINT D$"CATALOG": PRINT: PRINT
      : CALL - 958: GOSUB 690
2950 IF S% = 2 THEN 2810
2960 IF S% = 3 THEN 2990
2970 GOTO 2810
2980 HOME: GOTO 2630
2990 HOME: HTAB 11: PRINT "LOAD LAYOUT FILE
      ": PRINT L$: GOSUB 480: GOSUB 1060: GOSUB
      950: GOTO 2820
3000 VTAB 15: PRINT: X$ = N$: IF RIGHT$(X$
      ,3) < > ".LO" THEN X$ = X$ + ".LO"
3010 PRINT "FILE NAME=": X$: PRINT: PRINT D$
      B$X$A$D$S$: S = 0: R = 0: GOTO 2630: REM LOA
      D FILE,ZERO FLAGS
3020 VTAB 20: INVERSE: PRINT "DISK ERROR OR
      BAD FILE NAME-TRY AGAIN": NORMAL: GOSUB
      690: POKE 216,0: GOTO 2810
3030 HOME: TEXT: IF LN < 6 THEN VTAB 10: HTAB
      10: PRINT "NO LAYOUT IN PROGRESS!": GOSUB
      690: GOTO 2630
3040 IF R > 0 AND S > 0 THEN 1220
3050 PRINT: VTAB 10: PRINT "NO GARDEN BEING
      PLANTED!": GOSUB 690: GOTO 2630
3060 REM PRINT GRAPHICS
3070 IF GP = 0 THEN 1080: REM MAIN MENU
3080 HOME: VTAB 10: PRINT TAB(5)"BE SURE
      PRINTER IS ON.."
3090 VTAB 24: PRINT "<ESC>= STOP PRINTER <R
      ETURN>= CONTINUE": GOSUB 700: IF X$ = CHR$(
      27) THEN 2630
3100 HOME: HCOLOR= 0: FOR I = 161 TO 191: HPL0T
      0,I TO 279,1: NEXT: HCOLOR= 3: PRINT: PRINT
      D$"PR#": SL: PRINT GP$: PRINT: HOME: PRINT
      D$"PR#": GOTO 2630
3110 GOTO 1080: REM MAIN MENU
3120 REM GARDEN ECONOMICS
3130 U = 0: U$ = "": P = 0: P$ = "": T = 0: LB = 0
      : LB$ = "": T$ = "": FOR I = 1 TO 4: S$(I) =
      "": NEXT: HOME: FOR I = 1 TO 16: IF EC
      $(I) < > 0 THEN I = 16: NEXT: GOTO 316
      0: REM AT LEAST ONE ENTRY
3140 NEXT
3150 VTAB 10: PRINT "NOTHING PLANTED-NO ESTI
      MATE POSSIBLE": GOSUB 690: GOTO 1220
3160 HOME: VTAB 22: PRINT L$: HTAB 1: VTAB
      23: PRINT " <RETURN> FOR NO ENTRY OR TO
      ADVANCE": GOSUB 940: POKE 35,21: HOME
3170 FOR I = 1 TO 16: IF EC$(I) = 0 THEN NEXT
      : GOTO 3440
3180 POKE 35,20: HOME: POKE 35,22
3190 VTAB 5: HTAB 1: INVERSE: PRINT EC$(I):
      : NORMAL: PRINT " FT. OF "VG$(I)" PLANT
      ED"
3200 VTAB 12: PRINT "E.G. 5 ONIONS/FT @ .10/
      ONION": PRINT "OR 2 LBS TOMATOES/FT @ .
      29/LB": U$ = ""
3210 HTAB 1: VTAB 10: PRINT "EST. UNITS/FT =
      ":
3220 GET X$: PRINT X$: IF X$ = CHR$(27) THEN
      1220
3230 IF X$ = CHR$(8) AND LEN(U$) < 2 THEN
      POKE KB,0: U$ = "": PRINT CHR$(7): GOTO
      3210
3240 IF X$ = CHR$(8) THEN U$ = LEFT$(U$,
      LEN(U$) - 1): PRINT " ": CHR$(8): GOTO
      3220
3250 IF X$ = CHR$(13) AND U$ = "" THEN U =
      0: GOTO 3390
3260 IF X$ = CHR$(13) THEN 3280
  
```

```

3270 US = US + X$: GOTO 3220
3280 X$ = US: GOSUB 730: US = X$: IF ER THEN E
R = 0: PRINT CHR$(7): GOTO 3180: REM E
RROR IN INPUT
3290 U = VAL (U$): REM TOTAL UNITS
3300 HTAB 1: VTAB 18: PRINT TAB(5)"ENTER D
ECIMAL NUMBERS ONLY *
3310 VTAB 16: CALL - 868: HTAB 1: PRINT "ES
T, RETAIL COST PER UNIT =": P$ = ""
3320 GET X$: PRINT X$: IF X$ = CHR$(27) THEN
1220
3330 IF X$ = CHR$(8) AND LEN(P$) < 2 THEN
POKE KB,0: P$ = "": PRINT CHR$(7): GOTO
3310
3340 IF X$ = CHR$(8) THEN P$ = LEFT$(P$,
LEN(P$) - 1): PRINT " ": CHR$(8): GOTO
3320
3350 IF X$ = CHR$(13) AND P$ = "" THEN P =
0: GOTO 3390
3360 IF X$ = CHR$(13) THEN 3380
3370 P$ = P$ + X$: GOTO 3320
3380 X$ = P$: GOSUB 730: P$ = X$: IF ER THEN E
R = 0: PRINT CHR$(7): GOTO 3300
3390 P = VAL (P$): T = EC$(1) * U * P + T: T =
INT (100 * T + .5) / 100 + .001: T$ = LEFT$(
STR$(T), LEN (STR$(T)) - 1)
3400 IF T < .01 THEN T$ = "0.00"
3410 X$ = T$: GOSUB 820: T$ = X$
3420 VTAB 21: HTAB 7: PRINT " RETAIL VALUE=$
": T$: CALL - 868
3430 NEXT
3440 POKE 35,20: HOME
3450 FOR I = 1 TO 4: REM GARDEN EXPENSES
3460 VTAB 3: HTAB 1: CALL - 868: HTAB 12: PRINT
"GARDEN ASSETS = $": HTAB 33 - (T = 0) -
(T = > 1) - (T = > 10) - (T = > 100) -
(T = > 1000)
3470 IF VAL (T$) < .01 THEN T$ = "0.00"
3480 PRINT T$
3490 VTAB 8: HTAB 1: CALL - 958: HTAB 1: PRINT
"INPUT COST OF :
3500 PRINT : INVERSE: PRINT EX$(I): NORMAL
3510 PRINT : PRINT TAB(16)"$ ": FOR K = 1
TO 6: PRINT CHR$(95): NEXT : HTAB 18
: GET X$: PRINT X$: IF X$ = CHR$(27) THEN
S = 0: S$ = "": GOTO 1220
3520 IF X$ = CHR$(13) THEN NEXT : GOTO 35
90
3530 INPUT "": S$(I): S$(I) = X$ + S$(I)
3540 X$ = S$(I): GOSUB 730: IF ER THEN ER = 0
: PRINT CHR$(7): S$(I) = "": GOTO 3490
3550 LB = VAL (S$(I)) + VAL (LB$): LB = INT
(100 * LB + .5) / 100 + .001: LB$ = STR$(
LB)
3560 X$ = LB$: GOSUB 820: LB$ = X$
3570 IF VAL (LB$) < .01 THEN LB$ = "0.00"
3580 VTAB 4: HTAB 1: CALL - 868: HTAB 12: PRINT
"EXPENSES = $": HTAB 33 - (LB = 0)
- (LB = > 1) - (LB = > 10) - (LB = >
100) - (LB = > 1000): PRINT LB$: CALL
- 868: NEXT : REM 6 SPCS
3590 POKE 35,24: VTAB 7: CALL - 958
3600 NT = VAL (T$) - VAL (LB$): NT = INT (1
00 * NT + .5) / 100 + .001 - (NT < 0) +
.002: NT$ = STR$(NT): X$ = NT$: GOSUB 82
0: NT$ = X$
3610 IF NT < 0 THEN 3630
3620 IF VAL (NT$) < .01 AND VAL (NT$) = >
0 THEN NT$ = "0.00"
3630 VTAB 5: HTAB 29: FOR I = 1 TO 8: PRINT
CHR$(95): NEXT : VTAB 6: HTAB 15: PRINT
"GARDEN NET = $": HTAB 33 - 2 * (NT <
0) - (NT < - 10) - (NT < - 100) - (NT =
0) - (NT = > 1) - (NT > = 10) - (NT >
= 100) - (NT > = 1000): PRINT NT$: CALL
- 958
3640 PRINT : VTAB 10: PRINT EX$(1): VTAB 12:
PRINT EX$(2): VTAB 14: PRINT "INSECTICI
DES, ETC.": VTAB 16: PRINT EX$(4)
3650 FOR I = 1 TO 4: X$ = S$(I): GOSUB 820: S$(
I) = X$: VTAB 8 + 2 * I
3660 HTAB 28: PRINT "$":
3670 HTAB 33 - ( VAL (S$(I)) = 0) - ( VAL (S
$(I)) = > 1) - ( VAL (S$(I)) > = 10) -
( VAL (S$(I)) > = 100) - ( VAL (S$(I)) >
= 1000): PRINT S$(I): NEXT
3680 GOSUB 920
3690 GOSUB 700: IF PT = 1 AND (X$ = "P" OR X
$ = CHR$(112)) THEN VTAB 18: HTAB 1: CALL
- 958: PRINT L$: PRINT : ON SD GOTO 371
0: VTAB 23: PRINT TAB(8)"PRESS <ESC> T
O CANCEL PRINT": GOTO 3730: REM TYPE OF
SCREEN PRINT
3700 GOTO 1220: REM NO PRINT-BACK TO PLANNER
MENU
3710 IF SD = 1 THEN VTAB 17: CALL - 958: PRINT
: GOSUB 1000: GOTO 3740
3720 REM GENERIC SCREEN DUMP
3730 GOSUB 1010
3740 TEXT : HOME : GOSUB 690: GOTO 1220: REM
PLANNER MENU
3750 HOME : VTAB 12: PRINT "<RETURN> TO RUN
CONFIGURE": PRINT : PRINT "<ESC> FOR MAI
N MENU": GET Z$: PRINT : ON Z$ = CHR$(
27) GOTO 1080: PRINT D$"RUN CONFIGURE"
3760 REM ERROR TRAP
3770 GOSUB 960: REM NO ONERR
3780 IF ER = 6 THEN X$ = "GRAPHIC NOT ON THI
S DISK": Y$ = "CHECK NAME OR SELECT OTHER
DISK": GOTO 3830
3790 IF ER = 4 THEN X$ = "DISK WRITE PROTECT
ED": Y$ = "REMOVE WRITE-PROTECT TAB": GOTO
3830
3800 IF ER = 8 THEN X$ = "DISK I/O PROBLEM":
Y$ = "CHECK DRIVE DOOR OR DISK POSITION"
: F$ = "": GOTO 3830
3810 IF ER = 9 THEN X$ = "THIS DISK IS FULL"
: Y$ = "INSERT OTHER DISK OR CHANGE DRIVE
S": GOTO 3830
3820 VTAB 10: PRINT " ERROR #": PEEK (222)"
IN LINE": PEEK (219) + 256 + PEEK (218
): PRINT : PRINT "TAKE APPROPRIATE ACTIO
N": GOTO 3850
3830 PRINT TAB(10)"*** DISK ERROR ***": PRINT
L$
3840 VTAB 6: INVERSE: PRINT "PROBLEM.": NORMAL
: PRINT : PRINT X$: VTAB 13: INVERSE: PRINT
"POSSIBLE SOLUTION.": NORMAL : PRINT : PRINT
Y$
3850 GOSUB 690
3860 HOME : I = PEEK (EM): POKE EM,0: ON I GOTO
3980,2630,390,3750
3870 GOTO 1080: REM MAIN MENU
3880 GOSUB 960: IF ER < > 6 THEN 3790
3890 VTAB 5: VTAB 1: PRINT "THESE FILES MUST
BE ON THE DISK.": VTAB 8: HTAB 1: PRINT
"PG.GUIDE, PG.NAMES, CURSOR.BAR.4": PRINT
"TEXT.SHAPES & VEG.SHAPES": VTAB 11: PRINT
"NOT ALL ARE WERE FOUND."
3900 VTAB 19: PRINT "PRESS <RETURN> TO QUIT
PROGRAM.": POKE KB,0: GET X$: PRINT : END
3910 HOME : GOTO 3890
3920 REM NO CONFIG. FILE
3930 ONERR GOTO 4030
3940 HOME : VTAB 12: HTAB 1: PRINT "NO GP.CO
NFIG FILE FOUND": PRINT "<RETURN> TO RUN
CONFIGURE": PRINT : PRINT "<ESC> TO QUI
T": GET Z$: PRINT : IF Z$ < > CHR$(2
7) THEN PRINT D$"RUN CONFIGURE"
3950 END
3960 GOTO 3760: REM OTHER ERROR
3970 REM ONERR FROM DISK/GRAPHICS AREA
3980 PRINT : GOTO 370: REM RE- CONFIG
3990 ER = PEEK (222): IF ER = 6 THEN 3930
4000 GOTO 3770
4010 POKE EM,3: ER = PEEK (222): IF ER = 6 THEN
TEXT : HOME : VTAB 10: PRINT "FILE NOT
FOUND ON DISK": PRINT : PRINT "FATAL ERR
OR--CANNOT CONTINUE": END
4020 GOTO 3770: REM OTHER ERROR
4030 ER = PEEK (222): IF ER = 6 THEN HOME :
VTAB 12: PRINT "CONFIGURE NOT ON THIS D
ISK": PRINT : PRINT "RETURN TO TRY AGAIN
, <ESC> TO QUIT": GET Z$: PRINT : ON Z$
= CHR$(27) GOTO 3950: PRINT D$"RUN CO
NFIGURE"
4040 GOTO 3770
END OF LISTING 1

```


KEY PERFECT 5.0
RUN ON
GARDEN PLANNER

CODE-5.0	LINE#	LINE#	CODE-4.0
37A2CFB2	10	100	86E5
AF2B826	110	200	016D3F
56ED154D	210	300	5C53
A4E136F6	310	400	6696
7698E7F8	410	500	848D
C4689104	510	600	8815
7DACB4AE	610	700	A7BC
5272F32E	710	800	ABE4
0330CD71	810	900	B2F3
C0377ECF	910	1000	9F44
4BD6D719	1010	1100	B80C
D32B9087	1110	1200	9FE2
8A7A23B7	1210	1300	D0BC
55B4F72C	1310	1400	96EB
1FF7D6BF	1410	1500	81D8
29CE4AF3	1510	1600	B099
C1515EAB	1610	1700	B935
202ABAB1	1710	1800	010194
1B77FD90	1810	1900	ED93
5084E2AA	1910	2000	F7BD
13F41D93	2010	2100	E393
C8FAF623	2110	2200	8712
009CE71C	2210	2300	C1D0
C33F793B	2310	2400	E991
8FE6DFAC	2410	2500	FB18
B362C8F8	2510	2600	A554
21599339	2610	2700	746A
C123D5BE	2710	2800	BB3D
68715368	2810	2900	D3F0
8FB3A7A8	2910	3000	8AC0
A85714B1	3010	3100	D567
8FDF82F6	3110	3200	D496
44166BB6	3210	3300	9A48
828DF647	3310	3400	AF35
F6DB15D4	3410	3500	7697
C0ACE937	3510	3600	E5E7
C3366503	3610	3700	010B9E
7333DA23	3710	3800	D578
FA3F0FD2	3810	3900	F68B
91F49F2B	3910	4000	7656
6B17EF8F	4010	4040	813C
88BED3C8	PROGRAM TOTAL =		3DF3

Listing 2 for Garden Planner
PG.GUIDE.CREATE

```

10 REM .....
20 REM + PG.GUIDE.CREATE +
30 REM + BY MARK R. CRAVEN +
40 REM + COPYRIGHT (C) 1987 +
50 REM + BY MICROSPARC, INC +
60 REM + CONCORD, MA 01742 +
70 REM .....
80 DIM PGS(45,6): HOME = VTAB 12: PRINT "PRE
SS RETURN TO CREATE PG.GUIDE": PRINT "FI
LE ON CURRENT DISK": GET Z$: PRINT : ONERR
GOTO 420
90 FOR I = 1 TO 6: FOR J = 1 TO 45: READ PGS
(J,I): NEXT J: NEXT I
100 REM PLANTING DEPTH - PGS(J,2)
110 DATA 1/2,3/2-2,3/2-2,3/2-2,1/2-1,
1/2,1/2,1/2,1/4,1/2,1/8,1,1/2,1/4,2
120 DATA 1,1/4-1/2,1,1/2,1/2-1,1/4-1/2,1/4-
1/2,1,1,1-2,2-3,1/2,1/4-1/2,1/2,2,1/4,4,
1-3/2,1/2
130 DATA CROWN,1/2,1/2,1,1,PLANTS,1/2,1/2,1
/2,1
140 REM DISTANCE BETWEEN PLANTS PGS(J,3)
150 DATA 60,2-3,4-6,3-6,6-10,2,14-18,12-18,
12-20,1-2,18,8,4-8,8,10-15,10-14,12,18,2
-4
160 DATA 3-4,2-4,12-14,4-6,12,15-18,2-3,2-3
,2-3,3-6,3-4,2-3,18-24,12,30,1-2,24-30,8
-12,2-4,16-24,24-48
170 DATA 12-18,24,18-36,1-3,12-16
180 REM DISTANCE BETWEEN ROWS PGS(J,4)
190 DATA 72,18-30,36-48,24-30,30-36,12-18,2
4-30,24-30,24-30,14-24,30-36,24-30,18-24
,10-16,24-30,30-36,48-72

```

```

200 DATA 36,12-18,18-24,12-18,18-24,12-18,4
8-72,28-36,12-24,12-24,12-24,12-20,16-24
,18-30
210 DATA 24-36,24-36,72-120,6-12,36,18-24,1
2-14,36-60,72-120,36-48,36,36-60,15-18,6
0
220 REM DAYS TO GERMINATION PGS(J,5)
230 DATA 7-14,6-14,6-14,7-12,7-12,7-10,3-10,
3-10,4-10,10-17,4-10,9-21,7-10,8-12,4-10
,6-10,6-10,7-14,6-10,3-10,7-12
240 DATA 4-10,4-10,4-8,7-14,N/A,N/A,7-12,14
-28,15-25,6-15,10-20,8-16,6-10,3-10,N/A,
3-10
250 DATA 6-14,3-12,6-10,N/A,6-13,6-14,3-10,
3-12
260 REM SOIL TYPE COOL.TOLERATES COOL,WARM (
1,2,3) PGS(J,6)
270 DATA 2,3,3,3,3,2,2,2,2,2,1,2,2,2,3,3,
3,2,2,2,1,1,3,3,1,1,1,2,2,1,3,2,3,1,2,2,
1,3,3,3,3,1,3
280 REM DAYS TO MATURITY PGS(J,7)
290 DATA 365,45-65,60-70,60-80,85-90,55-65,
60-80,80-90,65-95,60-80,55-65,90-120,55-
65,80-90
300 DATA 65-85,60-90,55-65,75-95,90,60-70,8
0-90,55-80,45-60,75-100,50-60,95-120,95-
120,100-165,85-90,100-120
310 DATA 65-85,60-80,90-105,70-110,20-50,72
0,80-90,40-65,50-60,85-120,120,90-100,55
-90,45-60,80-100
320 REM CREATE RANDOM FILE
330 DS = CHR$(4):FS = "PG.GUIDE": ONERR GOTO
420
340 PRINT DS"OPEN"FS: PRINT DS"CLOSE"FS: PRINT
DS"DELETE"FS
350 PRINT DS"OPEN"FS",L60"
360 FOR I = 1 TO 45
370 PRINT DS"WRITE"FS",R" I
380 FOR J = 1 TO 6: PRINT PGS(I,J): NEXT J
390 NEXT I
400 PRINT DS"CLOSE"FS
410 HOME = VTAB 10: HTAB 1: PRINT "BE SURE Y
OU HAVE SAVED PG.GUIDE.CREATE": PRINT "T
HEN RUN GARDEN.PLANNER": END
420 E = PEEK(222):EL = PEEK(218) + 256 +
PEEK(219): POKE 216,0
430 IF E = 4 THEN AS = "DISK WRITE-PROTECTED
": GOTO 490
440 IF E = 8 THEN AS = "I/O ERROR--CHECK DRI
VE DOOR": GOTO 490
450 IF E = 9 THEN AS = "DISK FULL--TRY ANOTH
ER": GOTO 490
460 IF E = 16 THEN AS = "SYNTAX ERROR IN LIN
E " + STR$(EL): GOTO 490
470 IF E = 42 THEN AS = "OUT OF DATA ERROR I
N LINE " + STR$(EL): GOTO 490
480 AS = "ERROR " + STR$(E) + " IN LINE " +
STR$(EL)
490 HOME = VTAB 12: PRINT AS$
500 VTAB 22: HTAB 1: IF E < 16 THEN PRINT "
RETURN TO CONTINUE, ESCAPE TO QUIT": GET
Z$: PRINT : ON Z$ < > CHR$(27) GOTO 3
30: END
510 PRINT "PRESS RETURN TO QUIT PROGRAM": GET
Z$: PRINT : END
END OF LISTING 2

```

KEY PERFECT 5.0
RUN ON
PG.GUIDE.CREATE

CODE-5.0	LINE#	LINE#	CODE-4.0
C6C50A34	10	100	8CAA
1BC05E6B	110	200	014669
290DE049	210	300	015430
708A3ABD	310	400	8023
CFD48F02	410	500	DCC2
3FEE591F	510	510	14D3
F8BB7A96	PROGRAM TOTAL =		099A

Listing 3 for Garden Planner

PG.NAMES.CREATE

```

10 REM *****
20 REM * PG.NAMES.CREATE *
30 REM * BY MARK R. CRAVEN *
40 REM * COPYRIGHT (C) 1987 *
50 REM * BY MICROSPARC, INC *
60 REM * CONCORD, MA 01742 *
70 REM *****
80 HOME : VTAB 12: PRINT "PRESS RETURN TO CR
    EATE PG.NAMES": PRINT " FILE ON CURRENT
    DISK": GET Z$: PRINT
90 ONERR GOTO 240
100 D$ = CHR$(4):F$ = "PG.NAMES"
110 PRINT D$"OPEN" F$: PRINT D$"CLOSE" F$: PRINT
    D$"DELETE" F$: REM REMOVE EXISTING FILE,
    IF ANY
120 PRINT D$"OPEN" F$: PRINT D$"WRITE" F$
130 RESTORE
140 FOR I = 1 TO 45
150 READ A$
160 PRINT A$
170 NEXT I
180 FOR I = 1 TO 16
190 READ A$, B$
200 PRINT A$: PRINT B$
210 NEXT
220 PRINT D$"CLOSE"
230 HOME : VTAB 12: PRINT "FILE CREATION COM
    PLETE": END
240 E = PEEK(222):EL = PEEK(218) + 256 +
    PEEK(219): POKE 216
250 HOME : VTAB 12: PRINT "TROUBLE CREATING
    PG.NAMES FILE"
260 IF E = 4 THEN A$ = "DISK IS WRITE-PROTEC
    TED": GOTO 300
270 IF E = 8 THEN A$ = "I/O ERROR--CHECK DRI
    VE DOOR": GOTO 300
280 IF E = 9 THEN A$ = "DISK FULL--TRY ANOTH
    ER": GOTO 300
290 A$ = "ERROR " + STR$(E) + " IN LINE " +
    STR$(EL)
300 VTAB 14: HTAB 1: PRINT A$
310 VTAB 22: HTAB 1: PRINT "RETURN TO TRY AG
    AIN, ESCAPE TO QUIT": GET Z$: PRINT : IF
    Z$ < > CHR$(27) GOTO 80
320 END
330 REM PLANTING GUIDE DATA
340 DATA ARTICHOKE, BEANS-SNAP-BUSH, BEANS-SNA
    P-POLE, BEANS-LIMA-BUSH, BEANS-LIMA-POLE, B
    EETS, BROCCOLI, BRUSSELS SPROUTS, CABBAGE, C
    ARROT, CAULIFLOWER, CELERY, CHARD-SWISS, CHI
    VES, COLLARDS, CORN
350 DATA CUCUMBER, EGGPLANT, GARLIC, KOHLRABI, L
    EEKS, LETTUCE-HEAD, LETTUCE-LEAF, MUSKMELON
    , OKRA, ONION SETS, ONION PLANTS, ONION SEED
    , PARSLEY, PARSNIPS, PEAS, PEPPERS, POTATO, PU
    MPKIN, RADISH
360 DATA RHUBARB, RUTABAGA, SPINACH, SQUASH-SUM
    MER, SQUASH-WINTER, SWEET POTATO, TOMATILLO
    , TOMATO, TURNIP, WATERMELON
370 DATA "CARROTS/PARSNIPS", 10, "TOMATOES", 13
    , "CORN/STALK VEGETABLES", 10, "LETTUCE/CAB
    BAGE/ETC.", 13
380 DATA "MELONS/CANTALOUPE", 13, "CUCUMBER/Z
    UCCHINI", 9, "SQUASH/PUMPKIN/GOURDS", 13, "T
    URNIPS/LARGE ROOTS", 11
390 DATA "PEAS/POD VEGETABLES", 9, "RADISH/S
    MALL ROOTS", 12, "PEPPERS/EGGPLANT", 12, "
    BEANS/OKRA/BIG PODS", 10
400 DATA "GREENS/PARSLEY", 13, "POTATOES/TUB
    ERS", 14, "ONIONS/SCALLIONS", 7, "SPINACH/
    LEAFY GREENS", 13
END OF LISTING 3

```

KEY PERFECT 5.0
RUN ON
PG.NAMES.CREATE

CODE-5.0	LINE#	- LINE#	CODE-4.0
DDC6980D	10	- 100	78A8
8B4422B9	110	- 200	455F
89F5783D	210	- 300	947A
AF6E1E5D	310	- 400	01BA3D
ABE8C787 = PROGRAM TOTAL =			0677

Listing 4 for Garden Planner

CONFIGURE

```

10 REM *****
20 REM * CONFIGURE *
30 REM * BY MARK R. CRAVEN *
40 REM * COPYRIGHT (C) 1987 *
50 REM * BY MICROSPARC, INC *
60 REM * CONCORD, MA 01742 *
70 REM *****
80 HOME : D$ = CHR$(4):AD = 974:EM = 975:KB
    = - 16368:KD = - 16384:CUR = 768:OP$ =
    "OPEN":CL$ = "CLOSE":RD$ = "READ":WR$ =
    "WRITE":US = "UNLOCK":VF$ = "VERIFY"
90 ONERR GOTO 1260
100 PRINT D$"BLOAD CURSOR.BAR.4"
110 GOTO 410
120 ONERR GOTO 3720
130 PRINT : PRINT D$US$F$
140 PRINT D$OP$F$
150 PRINT D$RD$F$
160 INPUT PT: INPUT SL: INPUT GP: INPUT GP$:
    INPUT CL: INPUT SD: INPUT SD$
170 PRINT D$CL$
180 POKE 216,0: REM NO ONERR
190 POKE EM,0: GOTO 370
200 REM SUBROUTINES
210 VTAB 20: PRINT L$: VTAB 21: HTAB 1: INVERSE
    : PRINT "<-": NORMAL : PRINT " ": INVERSE
    : PRINT "->": NORMAL : PRINT " " TO SELEC
    T ...<": INVERSE : PRINT "RETURN": NORMAL
    : PRINT "> TO EXECUTE": PRINT " UP OR
    DOWN ARROW KEYS MAY BE USED"
220 PRINT " <ESC> RETURNS TO PREVIOUS MEN
    U"
230 VTAB 24: PRINT L$: RETURN
240 VTAB 3: HTAB 1: PRINT "PRINTER " : RETURN
250 VTAB 3: HTAB 9: PRINT "IN SLOT #": SL: RETURN
260 VTAB 3: HTAB 1: PRINT "NO PRINTER
    ": RETURN : REM 8 SPACES
270 VTAB 3: HTAB 21: PRINT "WITH GRAPHICS DU
    MP": RETURN
280 VTAB 3: HTAB 21: PRINT " NO GRAPHICS DU
    MP": RETURN
290 VTAB 5: HTAB 21: PRINT "WITH TEXT DUMP":
    : CALL - 868: RETURN
300 VTAB 5: HTAB 21: PRINT " NO TEXT DUMP":
    : CALL - 868: RETURN
310 VTAB 5: HTAB 21: CALL - 868: RETURN
320 VTAB 5: HTAB 1: PRINT "COLOR MONITOR
    ": RETURN
330 VTAB 5: HTAB 1: PRINT "MONOCHROME MONITO
    R": RETURN
340 S$ = " ": FOR I = 1 TO LEN(X$): IF ASC
    ( MID$(X$, I, 1) ) < 32 THEN S$ = S$ + "^"
    + CHR$( ASC ( MID$(X$, I, 1) ) + 64): NEXT
    : GOTO 360
350 S$ = S$ + MID$(GP$, I, 1): NEXT
360 RETURN
370 ONERR GOTO 1380
380 HOME : VTAB 12: HTAB 1: PRINT "RETURN TO
    RUN GARDEN.PLANNER": PRINT "ESCAPE TO Q
    UIT": GET Z$: PRINT : IF Z$ < > CHR$(
    27) THEN PRINT D$" RUN GARDEN.PLANNER"
390 END
400 REM CONFIGURE SYSTEM
410 TEXT : HOME : PRINT TAB(12)"CONFIGURE
    SYSTEM": PRINT L$: REM HEADING
420 VTAB 6: PRINT L$
430 IF PT THEN GOSUB 240: GOSUB 250: GOTO 4
    50
440 GOSUB 260: GOSUB 280: GOSUB 300: GOTO 47
    0
450 IF GP THEN GOSUB 270: GOTO 470
460 GOSUB 280: GOTO 480
470 IF CL THEN GOSUB 320: GOTO 490
480 GOSUB 330
490 IF SD THEN GOSUB 290: GOTO 510
500 GOSUB 300
510 HTAB 1: VTAB 8: PRINT "PRESS <": INVERSE
    : PRINT "ESC": NORMAL : PRINT "> TO ACC
    EPT ABOVE DEFAULTS."
520 HTAB 1: VTAB 10: PRINT "PRESS <RETURN> T
    O ENTER NEW PARAMETERS": CALL - 958: POKE
    KB,0: GET X$: PRINT

```


Listing 4 for Garden Planner

CONFIGURE (continued)

```

530 IF X$ = CHR$( 27) THEN 1060: REM WRITE
    DISK
540 IF X$ < > CHR$( 13) THEN 520
550 VTAB 7: HTAB 1: CALL - 958
560 GOSUB 210: HTAB 1: VTAB 7: PRINT "DO YOU
    HAVE A PRINTER CONNECTED? ": VTAB 7: HTAB
    36: PRINT "YES": HTAB 36: PRINT "NO"
570 A = 7: B = 8: C = 36: D = 38: GOSUB 970: GOSUB
    980
580 ON LINE GOTO 620,650
590 PT = 0: SL = 0: GP = 0: CL = 0: GP$ = "": SD$ =
    "": SD = 0
600 IF A = 7 THEN 370
610 GOTO 410
620 IF LINE = 1 THEN PT = 1: GOSUB 240: GOSUB
    1000: VTAB 7: CALL - 868: PRINT : CALL
    - 868: VTAB 7: POKE KB,0: PRINT "PRINTE
    R IS IN SLOT ": GET SL$: IF SL$ < CHR$(
    49) THEN SL = 0: PT = 0: GOSUB 260: VTAB
    7: CALL - 868: CALL - 868: GOTO 560
630 IF VAL (SL$) < 1 OR VAL (SL$) > 7 THEN
    PRINT CHR$( 7):: GOTO 620
640 PRINT SL$: SL = VAL (SL$): GOSUB 250: GOTO
    660
650 PT = 0: GP = 0: GP$ = "": GOSUB 260: GOSUB
    280: GOSUB 300: GOTO 930: REM SKIP CARD
    QUES.
660 GOSUB 990: VTAB 9: PRINT "PRINTER CARD H
    AVE GRAPHICS DUMP ? YES": HTAB 37: PRINT
    "NO"
670 A = 9: B = 10: GOSUB 970: GOSUB 980: IF LI
    NE = 0 THEN 590
680 IF LINE = 1 THEN GP = 1: GOSUB 270: GOTO
    700
690 GP = 0: GOSUB 280: GOTO 790: REM NO GRAPH
    IX DUMP CARD
700 GOSUB 1000: GP$ = "": VTAB 10: HTAB 1: PRINT
    "ENTER CODE FOR PAGE 1 GRAPHICS DUMP": PRINT
    "TERMINATE WITH '^' ": HTAB 20
710 POKE KB,0: GET X$: IF X$ < CHR$( 1) THEN
    710
720 IF X$ = CHR$( 94) AND LEN (GP$) = 0 THEN
    GP = 0: GOTO 760
730 IF ASC (X$) = 94 THEN 770
740 IF ASC (X$) < 32 THEN PRINT "^": CHR$(
    ASC (X$) + 64):: GP$ = GP$ + X$: GOTO 7
    10
750 PRINT X$: GP$ = GP$ + X$: GOTO 710
760 GOSUB 280: GOSUB 990: GOTO 790
770 X$ = GP$: GOSUB 340: X$ = "": GOTO 790
780 S$ = S$ + MID$( GP$, I, 1): NEXT
790 GOSUB 990: HTAB 1: VTAB 13: PRINT "PRINT
    ER CARD HAVE TEXT DUMP ? YES": HTAB
    37: PRINT "NO": REM 6 SPCS
800 A = 13: B = 14: GOSUB 970: GOSUB 980: IF L
    INE = 0 THEN 590
810 IF LINE = 1 THEN SD = 1: GOSUB 290: GOTO
    830
820 SD = 0: GOTO 930: REM NO TEXT DUMP ON CAR
    D
830 GOSUB 1000: SD$ = "": VTAB 14: HTAB 1: PRINT
    "ENTER CODE FOR PAGE 1 TEXT DUMP": PRINT
    "TERMINATE WITH '^' ": HTAB 20
840 POKE KB,0: GET X$: IF X$ < CHR$( 1) THEN
    840
850 IF X$ = CHR$( 94) AND LEN (SD$) = 0 THEN
    SD = 0: GOTO 890
860 IF ASC (X$) = 94 THEN 900
870 IF ASC (X$) < 32 THEN PRINT "^": CHR$(
    ASC (X$) + 64):: SD$ = SD$ + X$: GOTO 8
    40
880 PRINT X$: SD$ = SD$ + X$: GOTO 840
890 GOSUB 300: GOTO 930
900 X$ = SD$: GOSUB 340: X$ = "": GOTO 920
910 S$ = S$ + MID$( SD$, I, 1): NEXT
920 GOSUB 290
930 GOSUB 1000: GOSUB 990: VTAB 17: HTAB 1: PRINT
    "WHICH TYPE OF MONITOR? " TAB( 34)"COLOR
    ": PRINT TAB( 35)"MONO"
940 A = 17: B = 18: C = 34: D = 38: GOSUB 970: GOSUB
    980: IF LINE = 0 THEN 590
950 IF LINE = 1 THEN CL = 1: GOSUB 320: GOTO
    1010
960 CL = 0: GOSUB 330: GOTO 1010
970 CALL CUR,A,B,C,D: LINE = PEEK (4): RETURN
980 VTAB A: HTAB C: CALL - 868: VTAB B: HTAB
    C: CALL - 868: RETURN
990 VTAB 23: HTAB 8: PRINT "<ESC> RE-STARTS
    CONFIGURE": CALL - 868: PRINT L$: HTAB
    1: RETURN : REM 7 SPCS
1000 VTAB 23: HTAB 1: CALL - 868: HTAB 1: RETURN
    : REM CLEAR LINE 23
1010 VTAB 7: CALL - 958: GOSUB 210: GOSUB 1
    000: GOSUB 990
1020 VTAB 10: HTAB 5: PRINT "IS SET-UP OK? "
    : VTAB 14: HTAB 19: PRINT "YES": HTAB 19
    : PRINT "NO"
1030 CALL CUR,14,15,18,22
1040 LINE = PEEK (4): IF LINE = 0 THEN 590
1050 IF LINE = 2 THEN HOME : GOTO 410
1060 HOME : VTAB 10: PRINT TAB( 5)"DISK WRI
    TE - DO NOT INTERRUPT"
1070 POKE EM,4: REM ONERR GOTO 4720
1080 T$ = FS: F$ = "GP.CONFIG"
1090 PRINT DSOP$F$: PRINT DSCL$F$: PRINT D$
    DELETE"F$: PRINT DSOP$F$
1100 PRINT D$WR$F$
1110 PRINT PT: PRINT SL: PRINT GP: PRINT GP$
    : PRINT CL: PRINT SD: PRINT SD$: REM PRI
    NTER.CARD,GRAPHICS.STRING.MONITOR.TEXT S
    TRING
1120 PRINT D$
1130 PRINT D$CL$
1140 F$ = T$: POKE 216,0: GOTO 370: REM MAIN
    MENU
1150 REM ERROR TRAP
1160 ER = PEEK (222): POKE 216,0: HOME
1170 IF ER = 6 THEN X$ = "GRAPHIC NOT ON THI
    S DISK": Y$ = "CHECK NAME OR SELECT OTHER
    DISK": GOTO 1220
1180 IF ER = 4 THEN X$ = "DISK WRITE PROTECT
    ED": Y$ = "REMOVE WRITE-PROTECT TAB": GOTO
    1220
1190 IF ER = 8 THEN X$ = "DISK I/O PROBLEM":
    Y$ = "CHECK DRIVE DOOR OR DISK POSITION"
    : F$ = "": GOTO 1220
1200 IF ER = 9 THEN X$ = "THIS DISK IS FULL"
    : Y$ = "INSERT OTHER DISK OR CHANGE DRIVE
    S": GOTO 1220
1210 VTAB 10: PRINT " ERROR #": PEEK (222)"
    IN LINE ": PEEK (219) + 256 + PEEK (218
    ): PRINT : PRINT "TAKE APPROPRIATE ACTIO
    N ": GOTO 1240
1220 PRINT TAB( 10)"*** DISK ERROR ***": PRINT
    L$
1230 VTAB 6: INVERSE : PRINT "PROBLEM:": NORMAL
    : PRINT : PRINT X$: VTAB 13: INVERSE : PRINT
    "POSSIBLE SOLUTION:": NORMAL : PRINT : PRINT
    Y$
1240 VTAB 24: HTAB 1: PRINT "PRESS <RETURN>
    TO CONTINUE": GET Z$: PRINT
1250 GOTO 370: REM MAIN MENU
1260 ER = PEEK (222): HOME : POKE 216,0: IF
    ER < > 6 THEN 1180
1270 VTAB 12: HTAB 1: PRINT "CURSOR.BAR.4 NO
    T ON THIS DISK"
1280 VTAB 23: HTAB 1: PRINT "<RETURN> TO TRY
    AGAIN, <ESC> TO QUIT": POKE KB,0: GET
    X$: PRINT : ON X$ < > CHR$( 27) GOTO 9
    0
1290 END
1300 HOME : GOTO 1270
1310 REM NO CONFIG. FILE
1320 IF PEEK (222) = 6 THEN POKE 216,0: GOTO
    410: REM NOT CONFIGURED
1330 GOTO 1150: REM OTHER ERROR
1340 ER = PEEK (222): IF ER = 6 THEN 1320
1350 GOTO 1160
1360 POKE EM,3: ER = PEEK (222): IF ER = 6 THEN
    TEXT : HOME : VTAB 10: PRINT "FILE NOT
    FOUND ON DISK": PRINT : PRINT "FATAL ERR
    OR--CANNOT CONTINUE": END
1370 GOTO 1160: REM OTHER ERROR
1380 ER = PEEK (222): IF ER = 6 THEN HOME :
    VTAB 12: PRINT "GARDEN.PLANNER NOT ON T
    HIS DISK": PRINT : PRINT "<RETURN> TO TR
    Y AGAIN, <ESC> TO QUIT": GET Z$: PRINT
    : ON Z$ = CHR$( 27) GOTO 1290: PRINT D$
    "RUN GARDEN.PLANNER"
1390 GOTO 1160: REM OTHER ERROR
    END OF LISTING 4

```


KEY PERFECT 5.0
RUN ON
CONFIGURE

CODE-5.0	LINE#	LINE#	CODE-4.0
01E2A5AD	10	100	9466
D8B25407	110	200	3EFD
B9B771F8	210	300	B20B
C17E28C0	310	400	9936
4A42CE9D	410	500	4E42
614C0AD7	510	600	A6FF
A1D98154	610	700	ED7A
E24B619C	710	800	97D5
6CA8E39F	810	900	8CB6
25FAF9E9	910	1000	A224
6410E4D9	1010	1100	8348
8005BB24	1110	1200	D22C
9652CF9A	1210	1300	BBA8
2D981917	1310	1390	ACDD
16997E4C	= PROGRAM TOTAL =		1233

Listing 5 for Garden Planner
CURSOR.BAR.4

```

SOURCE FILE: CURSOR.BAR.4.S
0000: 1 .....
0000: 2 : CURSOR.BAR.4V .....
0000: 3 : BY MARK, R. CRAVEN .....
0000: 4 : .....
0000: 5 : MODIFIED FROM .....
0000: 6 : CURSOR.BAR .....
0000: 7 : BY JIM GANZ .....
0000: 8 : COPYRIGHT (C) 1984 .....
0000: 9 : BY MICROSPARC, INC. ....
0000: 10 : CONCORD, MA 01742 .....
0000: 11 : .....
0000: 12 : FROM NIBBLE DECEMBER 1984 .....
0000: 13 : SYNTAX .....
0000: 14 : CALL MEM.TOPLM, BOTLM, .....
0000: 15 : LMRGIN, RMRGIN .....
0000: 16 : .....
0000: 17 : TOPLM=(1-24) BOTLM=(1-24) .....
0000: 18 : LMRGIN=(1-40) RMRGIN=(1-40) .....
0000: 19 : .....
0000: 20 : Returns with LINE = PEEK(4) .....
0000: 21 : .....
0000: 22 : e.g. CALL 768.2.18.10.30 .....
0000: 23 : .....
0000: 24 : APPLESOFT TOOLKIT ASSEMBLER .....
0000: 25 : .....
0000: 26 : .....
0000: 27 : .....
0000: 28 : .....
0000: 29 : Equates: .....
0000: 30 : .....
0000: 31 : .....
0000: 32 : .....
FB01: 33 BASCALC EQU $FBC1 .....
0028: 34 BASL EQU $28 .....
0003: 35 BOTLM EQU $03 .....
DEBE: 36 CHKCOM EQU $DEBE .....
C010: 37 CLRKBD EQU $C010 .....
0067: 38 FRMNUM EQU $0D67 .....
ESFB: 39 CONINT EQU $E6FB .....
C000: 40 KYBD EQU $C000 .....
0004: 41 LINE EQU $04 .....
0300: 42 MEM EQU $300 .....
0002: 43 TOPLM EQU $02 .....
0000: 44 : .....
0006: 45 LMRGIN EQU $06 :LEFT MARGIN <-- NEW LINE
0007: 46 RMRGIN EQU $07 :RIGHT MARGIN <-- NEW LINE
0000: 47 : .....
----- NEXT OBJECT FILE NAME IS CURSOR.BAR.4
0300: 48 ORG MEM .....
0300: 49 : .....
0300: 50 : .....
0300: 51 : Initialize limits .....
0300: 52 : .....
0300: 53 : .....
0300: 20 BE DE 54 JSR CHKCOM :Advance TXTPTR past comma
0300: 20 67 DD 55 JSR FRMNUM :Get variable/expression
0300: 20 FB E6 56 JSR CONINT :Convert to integer in X
0300: CA 57 DEX .....
030A: 86 02 58 STX TOPLM .....
030C: 86 04 59 STX LINE .....
030E: 20 BE DE 60 JSR CHKCOM .....
0311: 20 67 DD 61 JSR FRMNUM .....
0314: 20 FB E6 62 JSR CONINT .....
0317: CA 63 DEX .....
0318: 86 03 64 STX BOTLM .....
031A: 20 BE DE 65 JSR CHKCOM :<-- NEW LINES 65-74
031D: 20 67 DD 66 JSR FRMNUM .....
0320: 20 FB E6 67 JSR CONINT .....
0323: CA 68 DEX .....
0324: 86 06 69 STX LMRGIN :Left Margin Value
0326: 20 BE DE 70 JSR CHKCOM .....
0329: 20 67 DD 71 JSR FRMNUM .....
032C: 20 FB E6 72 JSR CONINT .....
032F: CA 73 DEX .....
0330: 86 07 74 STX RMRGIN :Right Margin Value
0332: 75 : .....
0332: 76 : .....
0332: 77 : Highlight cursor line .....
0332: 78 : (change NORMAL to INVERSE) .....
0332: 79 : .....

```

```

0332: 80 : .....
0332: A5 04 81 START LDA LINE .....
0334: 20 C1 FB 82 JSR BASCALC :Set BASL.H
0337: A4 07 83 LDY RMRGIN :Right Margin <-- NEW LINE
0339: B1 28 84 CLOOP1 LDA (BASL),Y :Get a Character
033B: C9 A0 85 CMP #5A0 :Is it a SPACE?
033D: D0 05 86 BNE CBAR .....
033F: B8 87 DEY .....
0340: 10 F7 88 BPL CLOOP1 :Test for blank line
0342: 30 13 89 BMI LEFT .....
0344: A4 07 90 CBAR LDY RMRGIN .....
0346: B1 28 91 CLOOP4 LDA (BASL),Y .....
0348: 29 3F 92 AND #3F :Change it to inverse
034A: 91 28 93 STA (BASL),Y :and put it back.
034C: 88 94 DEY .....
034D: C4 06 95 CPY LMRGIN :Compare to left margin <--NEW LINE
034F: 10 F5 96 BPL CLOOP4 :Loop til end of line
0351: 97 : .....
0351: 98 : .....
0351: 99 : Wait for keypress .....
0351: 100 : .....
0351: 101 : .....
0351: AD 10 C0 102 LDA CLRKBD :Clear keyboard
0354: AE 00 C0 103 KLOOP LDX KYBD :Get keypress
0357: E0 88 104 LEFT CPX #5B8 :Is it a LEFT ARROW?
0359: F0 38 105 BEQ CONT .....
035B: E0 88 106 CPX #5B8 :Is it a DOWN ARROW? <-- NEW LINE
035D: F0 34 107 BEQ CONT .....
035F: E0 95 108 CPX #595 :Is it a RIGHT ARROW?
0361: F0 30 109 BEQ CONT .....
0363: E0 8A 110 CPX #58A :Is it an UP ARROW? <-- NEW LINE
0365: F0 2C 111 BEQ CONT .....
0367: E0 8D 112 CPX #58D :Is it a RETURN char?
0369: F0 0E 113 BEQ CALLER :Go exit if RETURN
036B: E0 9B 114 CPX #59B :Is it the ESC Character?
036D: F0 02 115 BNE EXIT :Exit without a choice
036F: D0 E3 116 BNE KLOOP :None of the above.
0371: A9 00 117 EXIT LDA#0 :Put zero in accumulator
0373: 85 04 118 STA LINE :Put a zero in line counter
0375: AD 10 C0 119 LDA CLRKBD .....
0378: 60 120 RTS :Return with no line selected
0379: 121 : .....
0379: 122 : .....
0379: 123 : Return to caller .....
0379: 124 : .....
0379: 125 : .....
0379: A4 07 126 CALLER LDY RMRGIN :Reset last line to NORMAL <--NEW LINE
037B: B1 28 127 CLOOP3 LDA (BASL),Y .....
037D: 09 80 128 ORA #580 .....
037F: 91 28 129 STA (BASL),Y .....
0381: 88 130 DEY .....
0382: C4 06 131 CPY LMRGIN :Check left margin <-- NEW LINE
0384: 10 F5 132 BPL CLOOP3 .....
0386: A5 04 133 LDA LINE .....
0388: 38 134 SEC .....
0389: E5 02 135 SBC TOPLM .....
038B: 85 04 136 STA LINE .....
038D: E6 04 137 JNC LINE .....
038F: AD 10 C0 138 LDA CLRKBD .....
0392: 60 139 RTS :Exit
0393: 140 : .....
0393: 141 : .....
0393: 142 : Lowlight cursor line .....
0393: 143 : (Change INVERSE to NORMAL) .....
0393: 144 : .....
0393: 145 : .....
0393: A4 07 146 CONT LDY RMRGIN :Load right margin value <--NEW LINE
0395: B1 28 147 CLOOP2 LDA (BASL),Y .....
0397: 09 80 148 ORA #580 :Change to NORMAL
0399: 91 28 149 STA (BASL),Y .....
039B: 88 150 DEY .....
039C: C4 06 151 CPY LMRGIN :Check left margin <-- NEW LINE
039E: 10 F5 152 BPL CLOOP2 .....
03A0: A5 04 153 LDA LINE .....
03A2: E0 95 154 CPX #595 .....
03A4: F0 12 155 BEQ RCONT .....
03A6: E0 8A 156 CPX #58A :Right arrow
03A8: F0 0E 157 BEQ RCONT .....
03AA: 158 : .....
03AA: 159 : .....
03AA: 160 : Move cursor up one line .....
03AA: 161 : .....
03AA: 162 : .....
03AA: C5 02 163 CMP TOPLM :Are we above the top line?
03AC: D0 06 164 BNE LCONT :No, so move up one line.
03AE: A5 03 165 LDA BOTLM :Wraparound to bottom line
03B0: 85 04 166 STA LINE .....
03B2: D0 12 167 BNE JUMP :Always.
03B4: C6 04 168 LCONT DEC LINE .....
03B6: 10 0E 169 BPL JUMP .....
03B8: 170 : .....
03B8: 171 : .....
03B8: 172 : Move cursor down one line .....
03B8: 173 : .....
03B8: 174 : .....
03B8: C5 03 175 RCONT CMP BOTLM :Are we below the bottom line?
03BA: D0 06 176 BNE RCONT1 :No, so move down one line
03BC: A5 02 177 LDA TOPLM :Wraparound to top line.
03BE: 85 04 178 STA LINE .....
03C0: 10 04 179 BPL JUMP .....
03C2: E6 04 180 RCONT1 INC LINE .....
03C4: D0 00 181 BNE JUMP .....
03C6: 4C 32 03 182 JUMP JMP START .....
03C9: 183 : .....
03C9: 184 : .....
03C9: 185 : LENGTH=5AC (172) .....
03C9: 186 : .....

```

... SUCCESSFUL ASSEMBLY: NO ERRORS
END OF LISTING 5

Listing 6 for Garden Planner
VEG.SHAPES

1400- 10 02 39 00 5C 00 94 00 00
 1408- BE 00 F6 00 21 01 46 01 01
 1410- 7D 01 AC 01 CC 01 F8 01 01
 1418- 26 02 48 02 6B 02 96 02 02
 1420- B4 02 00 00 00 00 00 00 00
 1428- 00 00 00 00 00 00 00 00 00
 1430- 00 00 00 00 00 00 00 00 00
 1438- 00 49 49 49 FA 1F 1F 1F 1F
 1440- 0E 0D 0D 0D 1E 1F 1F 0E 0E
 1448- 0D 3E 77 15 3F 2E 35 3F
 1450- 2E 35 3F 2E 35 3F 0E 4E 4E
 1458- 49 01 00 01 52 49 49 89 89
 1460- DF 1F DF 0E 6D 0D F5 3B 3B
 1468- FF 2A 2D 2D AD 3F 3F 3F 3F
 1470- 3F 1F 2D 2D 2D 2D 2D 3E 3E
 1478- 3F 3F 3F 3F 37 2D 2D 2D 2D
 1480- 2D 2D 1E 3F 3F 3F 3F 0E 0E
 1488- 2D 2D 2D 1E 3F 3F 56 49 49
 1490- 49 09 00 29 09 4D 4D 1E 1E
 1498- 1F 1F 6E 0D 1E 1F 6E 3E 3E
 14A0- BF 0D 0D 3E 1F 37 0D 0D 0D
 14A8- 3E 1F 37 0D 0D 3E 1F 37 37
 14B0- 0D 0D 3E 1F 37 0D 0D 1E 1E
 14B8- 3F 4D 49 00 00 1F 52 49 49
 14C0- 49 FA 1F 9F 4D 0D 4D 15 15
 14C8- 1F 1F 3F 1F 1F 0E 0D 4D 4D
 14D0- 69 15 1F DF DF 1F 4E 0D 0D
 14D8- 2D 0D 8D 1F DF DF 1F 0E 0E
 14E0- 0D 4D 69 15 1F 1F 3F 1F 1F
 14E8- 1F 0E 4D 0D 4D DE 1F 1F 1F
 14F0- 56 49 49 09 00 3F 92 4A 4A
 14F8- 09 2D 6D 3A DF 1B BF 4D 4D
 1500- 2D 4D 31 1F FF 1B 1F 17 17
 1508- 4D 49 49 09 1E 1F FF 1B 1B
 1510- 1F 6E 29 6D 09 1E FF DB DB
 1518- 77 29 2D 96 49 49 00 00 00
 1520- 09 49 A9 3F 17 4D 31 1F 1F
 1528- 1F 0E 4D 31 1F 1F 0E 4D 4D
 1530- 31 FF 33 6D 31 DF BB 0D 0D
 1538- 0D FE 1B 17 0D 0D 1E DF DF
 1540- 0E 75 49 09 00 3F 49 49 49
 1548- 2D 3E 3F 2E 15 3F FF 2A 2A
 1550- 4D 0D AD 1F DF 1F 1F 6E 6E
 1558- 4D 0D 0D 15 1F DF FB FB FB
 1560- 6E 4D 69 69 FE DF FB FB FB
 1568- 0E 0D 4D 0D 0D FE DF 1F 1F
 1570- 1F 0E 6D 69 F5 3B 3F 56 56
 1578- 49 49 09 00 2D 4D 4D 09 09
 1580- 8D DF DF FB 6E 69 4D F1 F1
 1588- DF FB 73 0D 4D 1E FF 0E 0E
 1590- F5 17 2D 15 3F 3F 17 2D 2D
 1598- 2D 2D 3E 3F 3F 37 2D 2D 2D
 15A0- 2D 1E 3F 3F 0E 2D 1E 4D 4D
 15A8- 49 01 00 3E 52 49 2D AD AD
 15B0- DF 1B 6E 6D FE FF 6E 49 49
 15B8- FE FF 6E 6D FE DB 6E 6D 6D
 15C0- FE FF 6E 49 1E 3F B7 49 49
 15C8- 49 00 00 2D 49 4D 69 11 11
 15D0- DF FB DF 33 0D 0D 0D 0D 0D
 15D8- 0D 1E 1F 1F 1F 1F 0E 4D 4D
 15E0- 4D 1E 3F 3F 4E BE 2D 15 15
 15E8- 3F 3F 2E 2D 35 3F 3F 0E 0E
 15F0- 2D 1E 36 4D 49 01 00 37 37
 15F8- 52 49 09 F5 37 4D 3A DF DF
 1600- BF 2D 2D 2D AD 3F 3F 3F 3F
 1608- 3F 3F 2D 2D 2D 2D 35 3F
 1610- 3F 3F 3F 77 2D 2D 2D 35 35
 1618- 3F 3F 3F 77 2D 2D F5 DF
 1620- 56 49 49 01 00 3E 49 49 49
 1628- 3A 17 4D 15 DF 1B 6E 6D 6D
 1630- FE FF 6E 6D FE DB 6E 49 49
 1638- FE FF 6E 6D FE FF 6E 49 49
 1640- 1E DF 0E 75 49 00 00 1F 1F
 1648- 52 49 49 49 FA 1F 1F 1F 1F

1650- 1F 0E 0D 0D 0D 0D 1E 1F 1F
 1658- 1F 1F 0E 0D 0D 0D 1E 1F 2E 2E
 1660- F5 17 2D 3E B7 52 49 49 49
 1668- 01 00 35 92 4A 49 11 3F 3F
 1670- 9F 6D 09 6D 3A DF FB 1B 1B
 1678- 17 4D 4D 4D 09 FE FB DB DB
 1680- 1B 37 4D 09 4D 09 F5 FB FB
 1688- DF FB 0E 6D 09 F5 3B 87 87
 1690- 49 49 49 00 00 35 49 4D 4D
 1698- 31 DF 33 4D F1 1F 6E FE FE
 16A0- 6E 1E 17 2D 15 3F 3F 2E 2E
 16A8- 2D 35 3F 3F 0E 2D 1E 36 36
 16B0- 4D 01 00 35 52 29 6D 29 29
 16B8- AD DF 3B FF 33 6D 69 69 69
 16C0- 15 DF FB 1F DF 2E 4D 2D 2D
 16C8- 4D 0D FE FB FB FB 0E 0D 0D
 16D0- 0D 0D 4D 3E DF 3F 1F 77 77
 16D8- 69 4D 1E 1F 77 AD 3F B7 B7
 16E0- 49 49 49 00 00 00 00 00

END OF LISTING 6

KEY PERFECT 5.0
RUN ON
VEG. SHAPES

CODE-5.0	ADDR#	ADDR#	CODE-4.0
0400B29A	1400	144F	23FE
9388802F	1450	149F	28EF
B09C0B1D	14A0	14EF	2AF1
F8B8F4DC	14F0	153F	26CB
2B78D3C6	1540	158F	2697
79846CC0	1590	15DF	2376
60AB98FE	15E0	162F	299C
F68BA455	1630	167F	26F0
5DFBC322	1680	16CF	2504
BFEC9788	16D0	16E3	08C0
E2A41681	PROGRAM TOTAL		02E4

Listing 7 for Garden Planner
TEXT.SHAPES

1000- 40 00 82 00 84 00 8B 00 00
 1008- 91 00 9F 00 AD 00 BA 00 00
 1010- C6 00 C9 00 D4 00 DD 00 00
 1018- EC 00 F3 00 F8 00 FD 00 00
 1020- 00 01 07 01 15 01 1D 01 01
 1028- 28 01 35 01 42 01 4E 01 01
 1030- 5A 01 65 01 71 01 7D 01 01
 1038- 80 01 84 01 8E 01 96 01 01
 1040- A0 01 AA 01 B7 01 C3 01 01
 1048- CF 01 DC 01 E8 01 F4 01 01
 1050- 00 02 0C 02 18 02 22 02 02
 1058- 2A 02 37 02 40 02 4D 02 02
 1060- 5A 02 66 02 70 02 7E 02 02
 1068- 8C 02 97 02 A0 02 AB 02 02
 1070- B6 02 C2 02 CE 02 D8 02 02
 1078- E5 02 F3 02 FB 02 08 03 03
 1080- 0E 03 00 00 B6 04 40 18 18
 1088- 24 04 00 18 24 0D 36 04 04
 1090- 00 83 24 6C 36 FF 16 2D 2D
 1098- 25 0C 16 17 FE 24 00 24 24
 10A0- 74 39 3F 17 0E 0D 0E 1E 1E
 10A8- 27 1E 77 21 00 18 38 2C 2C
 10B0- 56 09 88 17 17 17 4D 35 35
 10B8- 27 00 20 1C 17 76 1E 76 76
 10C0- 65 1C 8C B1 04 00 20 24 24
 10C8- 00 1B 40 18 09 17 1E 36 36
 10D0- 0E 0E 04 00 40 18 0E 0E 0E
 10D8- 3E 1E 1E 04 00 24 34 50 50
 10E0- F1 1E 18 1C 96 62 0D 0E 0E
 10E8- 1F B4 04 00 2D DF 27 48 48
 10F0- B6 26 00 12 30 1E 04 00 00
 10F8- 1B 2D 2D 04 00 92 04 00 00
 1100- 40 B9 17 17 17 04 00 0C 0C
 1108- 0C 1C 3F 1E 36 2E 1E 0E 0E
 1110- 2D 0C 24 24 00 24 BC 96 96
 1118- 31 3E 0D 04 00 65 E4 3F 3F
 1120- 1E 96 F1 CE 2D 2D 04 00 00
 1128- 25 05 20 3F 3F 96 4A 09 09

1130- F6 3F 1C 04 00 2A 36 04 04
 1138- 28 07 20 24 17 17 17 2E 2E
 1140- 04 00 28 1F 27 2C 2D B5 B5
 1148- 32 F6 3F 1C 04 00 39 3F 3F
 1150- 2C 60 2D 96 32 1E 3F 1C 1C
 1158- 24 00 1A 0C 0C 0C 3C 3F 3F
 1160- B7 92 31 04 00 39 E7 2C 2C
 1168- 28 75 B6 F6 3F 07 20 04 04
 1170- 00 2D 24 07 38 F7 76 4E 4E
 1178- F1 1E 3F 04 00 B0 04 00 00
 1180- B0 F6 04 00 1B 0C 0C 0C 0C
 1188- 96 92 1C 1C 04 00 D8 2D 2D
 1190- 2D D6 39 3F 27 00 09 07 07
 1198- 38 E0 96 4A 1E 1E 04 00 00
 11A0- 92 04 20 0C 0C 1C 3F 1E 1E
 11A8- 04 00 30 2E 2C 24 1C 3F 3F
 11B0- 1E 36 36 0E 2D 25 00 2A 2A
 11B8- 25 3C 38 B8 17 36 F5 6E 6E
 11C0- 09 24 00 3F 24 2C 2D 15 15
 11C8- F6 0E F6 3F 27 24 00 09 09
 11D0- 40 03 1C 3F 1E 36 36 15 15
 11D8- 2D 0C 04 00 1B 24 2C 2D 2D
 11E0- 0E 36 36 17 3F 27 24 00 00
 11E8- 25 40 3F 3F 36 2E 1E 36 36
 11F0- 2D 2D 04 00 19 2D 40 18 18
 11F8- 3F 3F 36 2E 1E 36 04 00 00
 1200- 0A 46 36 3F 3F 20 24 2C 2C
 1208- 28 2D 04 00 2D 24 FC 1B 1B
 1210- 36 2E 1E 36 4D 21 24 00 00
 1218- 40 18 2B F5 36 36 3E 0D 0D
 1220- 04 00 93 73 2D 0C 24 24 24
 1228- 24 00 63 0C 0C DF 33 36 36
 1230- 36 6E 09 07 38 20 00 1B 1B
 1238- 24 B4 12 36 2D 2D 04 00 00
 1240- 64 05 30 36 36 FE 1B 24 24
 1248- 24 24 0E 04 00 18 0E 0E 0E
 1250- 56 24 24 24 DF 33 36 36 36
 1258- 26 00 1B 24 05 28 75 36 36
 1260- 36 1E 3F 1C 24 00 65 3C 3C
 1268- 38 3F 36 2E 1E 36 04 00 00
 1270- 8A 11 1C 07 68 24 3C 38 38
 1278- F7 36 36 0E 25 00 65 3C 3C
 1280- 38 37 18 36 2E 0E 0E 0E 0E
 1288- DF 23 24 00 75 F6 3F 1C 1C
 1290- 44 1C 64 2D 0E 04 00 24 24
 1298- 3C 6F 29 96 DA 36 04 00 00
 12A0- 1B 24 6C 09 36 36 F6 3F 3F
 12A8- 1C 24 00 92 0C 0C 24 24 24
 12B0- DF 33 36 AE 04 00 76 0E 0E
 12B8- 24 24 24 DF 33 36 36 66 66
 12C0- 04 00 0C 0C FC 1B AE 16 16
 12C8- 17 6E 09 E4 04 00 07 68 68
 12D0- 0C FC 1B B6 4A 36 04 00 00
 12D8- 17 1E 2E 2D 25 40 18 2B 2B
 12E0- 20 3F 3F 04 00 1B 24 2C 2C
 12E8- 2D F5 1B 36 36 2E 2D DF DF
 12F0- 23 24 00 19 1C 1E 96 49 49
 12F8- 0E 04 00 21 24 3F 6F 09 09
 1300- 36 36 36 27 B4 3B 27 00 00
 1308- B8 17 4D E1 04 00 92 1F 1F
 1310- 6D 25 00 00 00 00 00 00

END OF LISTING 7

KEY PERFECT 5.0
RUN ON
TEXT. SHAPES

CODE-5.0	ADDR#	ADDR#	CODE-4.0
C1B0200C	1000	104F	22BB
6A72DB3F	1050	109F	2445
38A47BB3	10A0	10EF	2844
210C72D9	10F0	113F	2708
182A58C9	1140	118F	292F
FEB21539	1190	11DF	271E
6A080EFF	11E0	122F	251A
43AA86A5	1230	127F	25FF
9E87CAF7	1280	12CF	253C
CBAB8627	12D0	131F	220F
560BAB15	PROGRAM TOTAL		0313