

Programming Languages: Communicating With Your Computer

COMPUTE!

\$2.95
November
1985
Issue 66
Vol. 7, No. 11
\$3.75 Canada
02193
ISSN 0194-357X

The Leading Magazine Of Home, Educational, And Recreational Computing

Amiga's Amazing Graphics

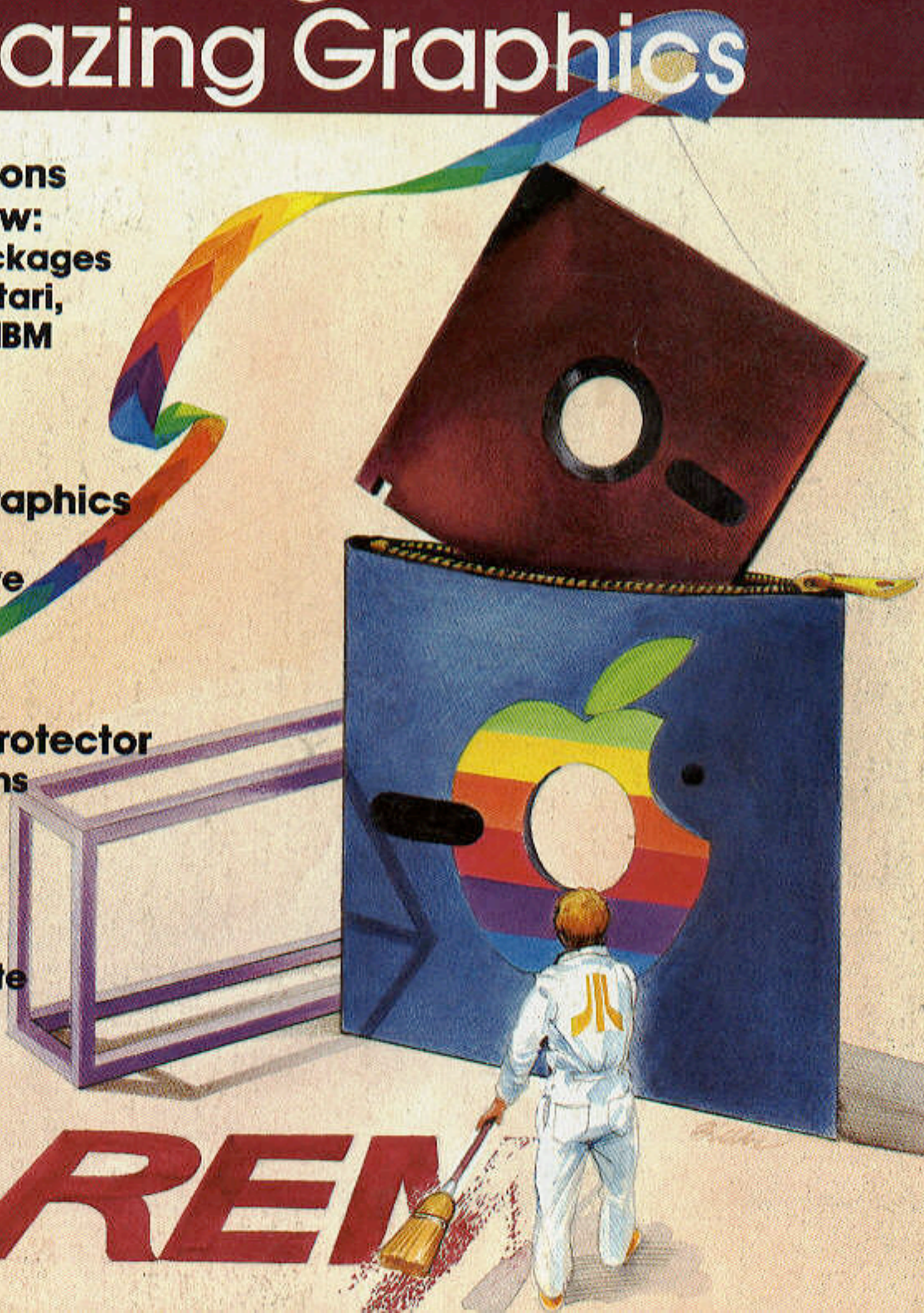
**Telecommunications
Software Overview:
72 Inexpensive Packages
For Commodore, Atari,
Apple, Macintosh, IBM**

**Commodore 64
3-D Animated Graphics
Simple Commands
To Create And Move
Your Own Shapes**

**Apple Program Protector
Shield Your Programs
From Prying Eyes**

**Atari REMover
Automatically Delete
REMs From BASIC**

REM



COMPUTE!

NOVEMBER 1985
VOLUME 7
NUMBER 11
ISSUE 66

FEATURES

- 18 Trends in Telecomputing Selby Bateman
- 22 An Overview of Telecommunications
Software Selby Bateman and Kathy Yakal
- 27 What's New Online? Kathy Yakal
- 30 Programming Languages:
Communicating with Your Computer Todd Heimarck and Kathy Yakal
- 44 Puzzler Mark Tuttle and Kevin Mykytyn
- 62 Skyscape Robert M. Simons

GUIDE TO ARTICLES AND PROGRAMS

64/VIC/+4/16/AT/AP/
PC/PCjr/TI
64/AT/AP/PC/PCjr/TI

REVIEWS

- 80 Crossword Magic Karen McCullough
- 80 Colorasaurus Steve Hudson
- 87 Grolier Online Encyclopedia Dan Gutman
- 88 BASIC Lightning & White Lightning for Commodore 64 Roark Dority
- 90 Gato for Apple and IBM Michael B. Williams
- 90 Atari PaperClip Robert L. Riggs

AP/64/AT/PC/PCjr
AT/64/128

64
AP/PC/PCjr/Mac
AT

COLUMNS AND DEPARTMENTS

- 6 The Editor's Notes Richard Mansfield
- 10 Readers' Feedback The Editors and Readers of COMPUTE!
- 60 HOTWARE: Software Best Sellers
- 131 The Beginner's Page: Variable Accuracy Tom R. Halfhill
- 132 Computers and Society:
Personal Computers and Personal Freedom David D. Thornburg
- 133 Telecomputing Today: Faster Than a Speeding Byte Arlan R. Levitan
- 134 The World Inside the Computer:
The Case of the Phantom Programmers Fred D'Ignazio
- 135 INSIGHT: Atari—Deactivating BASIC Bill Wilkinson
- 136 IBM Personal Computing: A Promise of Things to Come Donald B. Trivette
- 137 Programming the TI: The States of the Union C. Regena

AT
PC/PCjr
TI

THE JOURNAL

- 92 Commodore 64 3-D Animated Graphics Christian-Marc Panneton
- 102 IBM Graphics on a Monochrome Monitor Thomas G. Hanlin, III
- 106 Commodore Dynamic Keyboard, Part 2 Jim Butterfield
- 108 Simple Assembling with IBM DEBUG Tim Victor
- 111 Save-with-Replace: Debugged at Last, Part 2 P. A. Slaymaker
- 115 Atari REMover Jeff Stefanski
- 116 Plus/Term for Commodore 1660 Modem Mark Wood
- 117 Atari Animation with P/M Graphics, Part 3 Robert Powell
- 119 Amiga's Amazing Graphics Charles Brannon
- 125 A Better Way to POKE on the Commodore 64 Matthew MacKenzie
- 127 Adding TIMES to Atari Kenneth S. Szajda
- 129 Apple Program Protector Boris Troyanovsky

64
PC
64/128/V/+4/16/P
PC/PCjr
64/128/V/+4/16/P
AT
64
AT
AM
64
AT
AP

- 139 COMPUTE!'s Guide to Typing in Programs
- 142 MLX Machine Language Entry Program for Commodore 64
- 144 CAPUTE! Modifications or Corrections to Previous Articles
- 145 News & Products
- 160 Advertisers Index

NOTE: See page 139
before typing in
programs.

TOLL FREE Subscription Order Line
800-334-0868 (In NC 919-275-9809)

AP Apple, Macintosh, AT
Atari, VIC-20, Commodore 64
+4, Commodore Plus/4, 16, Com-
modore 16, 128 Commodore 128
P PET/CSA, TI Texas Instruments,
PC IBM PC, PCjr IBM PCjr, AM
Amiga, *General interest

COMPUTE! Publications, Inc.

One of the ABC Publishing Companies:
ABC Publishing, President, Robert G. Burton
1330 Avenue of the Americas, New York, New York 10019
Address all inquiries to:
P.O. Box 5406, Greensboro, NC 27403

COMPUTE! The Journal for Progressive Computing (USPS: 537250) is published monthly by COMPUTE! Publications, Inc., P.O. Box 5406, Greensboro, NC 27403 USA. Phone: (919) 275-9809. Editorial Offices are located at 324 West Wendover Avenue, Greensboro, NC 27408. Domestic Subscriptions: 12 issues, \$24. POSTMASTER: Send address changes to: COMPUTE! Magazine, P.O. Box 10955, Des Moines, IA 50950. Second class postage paid at Greensboro, NC, 27403 and additional mailing offices. Entire contents copyright ©1985 by COMPUTE! Publications, Inc. All rights reserved, ISSN 0194-357X.



PUZZLER

Mark Tuttle, Submissions Reviewer

Kevin Mykytyn, Editorial Program

Here's a game that tests your skill in pattern-matching and visualization. It runs on the Commodore 64; unexpanded VIC-20; Plus/4 and 16; IBM PC with color/graphics adapter and BASICA; PCjr with Cartridge BASIC; Apple II-series computers; TI-99/4A with Extended BASIC; and Atari 400/800, XL, and XE computers with at least 16K RAM for tape or 24K for disk. The Commodore 64 and Atari versions also require a joystick.

How good are you at recognizing patterns? Many intelligence tests measure this important conceptual skill. "Puzzler" challenges your ability to find matching patterns in a background of similar shapes. It displays two puzzle grids composed of multicolored blocks (see photos). Both grids contain exactly the same blocks, but those in the left grid have been scrambled. Your job is to rearrange the blocks in the left puzzle grid until they match those on the right. You must solve the puzzle before time runs out.

Because all versions of Puzzler are similar, we've printed general game instructions followed by specific notes for each computer. Read the general instructions as well as the section for your machine, then type in the program listed for your computer. Don't forget to save a copy of the game before you run it.

Puzzle Building

Puzzler begins by letting you choose the size of the puzzle grid. Enter values for the number of rows and columns in the grid. The maximum puzzle size differs among the various versions. Of course, larger puzzles are more difficult to solve than small ones. Next, enter the number of colors the puzzle will use. Two-color puzzles are the easiest. The maximum number of colors depends on which version you're playing. The more colors you choose, the harder your job becomes.

Puzzler then spends a short time building the two grids. Since the blocks are arranged at random, each new puzzle is different from the last. While you try to solve the puzzle, the computer keeps track of the time and alerts you when the puzzle is solved or time runs out. The time limit depends on the size of the puzzle.

Puzzler allows three different operations. You can move within the puzzle grid from one block to another, pick up a block and move it to a new position, or rotate a block in its current position. Use the cursor keys (or joystick in some versions) to move around in the grid. Your position is indicated by a colored cursor (or index arrows in some versions). To pick up a block, press RETURN (or the joystick but-

ton) once. The cursor or arrow changes color to show that you're carrying the piece. Then move to the position where you want to place the block, and press RETURN (or the button) once. The block in the current position trades positions with the block you're carrying.

Each block consists of four colored squares. To rotate a block in its current position, press RETURN (or the joystick button) twice. The block rotates 90 degrees. You may rotate a block as many times as you want.

Continue moving and rotating blocks until both puzzle grids match. Every block must match in color and be turned in the right direction.

Commodore 64 Version

Plug a joystick into port 2. The puzzle may contain as many as seven rows and columns, and up to 16 different colors. The box-shaped cursor shows your position on the puzzle grid. Press the joystick button twice without moving the joystick to rotate the block under the cursor. Press the button once to pick up the piece under the cursor: The cursor changes color to show that you're carrying the block. Now you may move to any other place in the grid. When you find the spot

you want, press the button again to set down the block. It changes places with the block in that position.

VIC-20 Version

VIC-20 Puzzler is played with the cursor keys. The maximum puzzle size is four columns by six rows, with up to six colors. Your position in the grid is shown by two index arrows, normally colored black. Press RETURN twice to rotate a block. Press RETURN once to pick up a block, then move it with the cursor keys and press RETURN to put it down. The arrows turn blue when RETURN is pressed once, and red when it is pressed a second time.

Plus/4 And 16 Version

Puzzler for the Commodore Plus/4 and 16 permits up to seven rows and columns and seven different colors. It is played exactly like the VIC-20 version.

Atari Version

Plug a joystick into port 1. Atari Puzzler lets you build puzzles with as many as eight rows and columns and up to four different colors. Manipulate the joystick as explained in the Commodore 64 instructions.

IBM Version

IBM Puzzler allows grids as large as seven rows and columns with up to seven different colors. Index arrows indicate your position in the grid, as explained in the VIC-20 instructions. Use the cursor keys to move within the grid. Press Enter to move or rotate a block.

TI-99/4A Version

You have the option of playing with either a joystick or keyboard controls. Puzzles can be as large as six rows and six columns with as many as six different colors. The box-shaped cursor shows your position in the puzzle grid and changes colors to indicate when you're carrying a block. When using the keyboard, make sure the Alpha Lock key is down. Move the cursor with the arrow keys and press Enter to rotate or move a block.

Apple Version

Puzzler runs on any Apple II-series computer with either DOS 3.3 or ProDOS. Press the space bar to

move or rotate a block, and press I, J, K, and L to move up, left, down, and right, respectively. Your position in the grid is indicated by small white highlights in the corners of the block.

Program 1: Commodore 64 Puzzler

Version by John Krause, Assistant Technical Editor

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" published bimonthly in COMPUTE!.

```

100 GOSUB460 :rem 171
110 IFT<TI/60THENPOKE53269,0:GOTO790 :rem 185
120 A=INT(T-TI/60+.5):B=INT(A/60) :rem 89
130 PRINT"{HOME}{3 DOWN}"SPC(17)B"{LEFT}:" :rem 226
140 Z$=STR$(A-60*B):Z$=RIGHT$(Z$,LEN(Z$)-1):IFLEN(Z$)=1THENPRINT"0": :rem 236
150 PRINTZ$ :rem 161
160 A=NOTPEEK(56320) :rem 124
170 R=R+SGN((AAND2)-(AAND1)) :rem 55
180 C=C+SGN((AAND8)-(AAND4)) :rem 35
190 IFR<0THENR=0 :rem 213
200 IFR>=R3THENR=R3-1 :rem 20
210 IFC<0THENC=0 :rem 176
220 IFC>=C3THENC=C3-1 :rem 218
230 POKE53248,CS+16*C:POKE53249,RS+16*R :rem 218
240 IF(AAND16)=0THEN110:rem 31
250 IFF=0THENF=1:POKE53287,14:RR=R:CC=C:WAIT56320,16:GOTO110 :rem 115
260 F=0:IFRR=RANDCC=CTHENGOSUB300:GOTO280 :rem 111
270 GOSUB330 :rem 175
280 POKE53287,15:WAIT56320,16:IFA$<>B$THEN110 :rem 53
290 GOTO800 :rem 108
300 B=C1+80*R+C+C+41:GOSUB420 :rem 81
310 POKEB,PEEK(A):POKEB+1,PEEK(A+1) :rem 46
320 POKEB+40,PEEK(A+NC):POKEB+41,PEEK(A+NC+1):RETURN :rem 132
330 GOSUB360:B=C1+80*R+2*C+41:GOSUB310 :rem 147
340 B=C1+80*RR+2*CC+41:A=AA:GOTO310 :rem 201
350 REM *** MOVE :rem 49
360 AA=SS+2*NC*RR+2*CC:A=SS+2*NC*R+C+C :rem 191
370 D=PEEK(A):POKEA,PEEK(AA):POKEAA,D :rem 251
380 D=PEEK(A+1):POKEA+1,PEEK(A+1):POKEAA+1,D :rem 108
390 D=PEEK(A+NC):POKEA+NC,PEEK(AA+NC):POKEAA+NC,D :rem 237
400 D=PEEK(A+NC+1):POKEA+NC+1,PEEK(AA+NC+1):POKEAA+NC+1,D:RETURN :rem 111
410 REM *** ROTATE :rem 198
420 A=SS+2*NC*R+C+C :rem 42
430 D=PEEK(A):POKEA,PEEK(A+NC) :rem 24
440 POKEA+NC,PEEK(A+NC+1) :rem 191
450 POKEA+NC+1,PEEK(A+1):POKEA

```

```

+1,D:RETURN :rem 240
460 POKE53269,0:A$="" :POKE53280,6:POKE53281,6 :rem 233
470 PRINT"{CLR}"CHR$(14)SPC(16)"{2 DOWN}{WHT}PUZZLER":PRINTSPC(16)"[7 T]" :rem 153
480 FORT=54272TO54295:POKET,0:NEXT:POKE54296,15 :rem 91
490 INPUT"{HOME}{7 DOWN}NUMBER OF ROWS (2-7)":R3:rem 203
500 IFR3<2ORR3>7THEN490 :rem 126
510 INPUT"{HOME}{10 DOWN}NUMBER OF COLUMNS (2-7)":C3 :rem 190
520 IFC3<2ORC3>7THEN510:rem 91
530 INPUT"{HOME}{13 DOWN}NUMBER OF COLORS (2-14)":C0 :rem 238
540 IFCO<2ORCO>14THEN530 :rem 197
550 PRINT"{2 DOWN}PLEASE WAIT {SPACE}..." :rem 134
560 S1=1473-40*R3-C3:C1=S1+54272:S2=S1+20:C2=C1+20:NR=2*R3:NC=2*C3 :rem 120
570 FORA=1TONR*NC:A$=A$+CHR$(RAND(1)*CO):NEXT:B$=A$ :rem 203
580 A=256*PEEK(46)+PEEK(45) :rem 204
590 SS=256*PEEK(A+4)+PEEK(A+3) :rem 158
600 FORR=0TOR3-1:FORC=0TOC3-1:B=INT(RND(1)*4) :rem 195
610 IFBTHENGOSUB420:B=B-1:GOTO610 :rem 16
620 NEXT:NEXT :rem 80
630 FORR=0TOR3-1:FORC=0TOC3-1 :rem 13
640 RR=INT(RND(1)*R3):CC=INT(RND(1)*C3):GOSUB360:NEXT:NEXT :rem 80
650 PRINT"{CLR}"SPC(17)"{DOWN}PUZZLER :rem 141
660 FORA=1TONR:FORB=1TONC:POKEC1+40*A+B,PEEK(SS+E) :rem 118
670 POKEC1+40*A+B,160:E=E+1:NEXT:NEXT :rem 201
680 FORA=1TONR:FORB=1TONC:POKEC2+40*A+B,ASC(MID$(B$,G+1)) :rem 153
690 POKEC2+40*A+B,160:G=G+1:NEXT:NEXT :rem 208
700 POKE2040,14:POKE53287,15:POKE53277,1:POKE53271,1 :rem 183
710 FORA=896TO924:READB:POKEA,B:NEXT :rem 15
720 FORA=925TO958:POKEA,0:NEXT :rem 102
730 RS=144-4*NR:CS=102-4*NC:R=0:C=0 :rem 223
740 T=NR*NC*3:POKE53269,1:TI$="000000":RETURN :rem 105
750 DATA255,192,0,128,64,0,128,64,0 :rem 232
760 DATA128,64,0,128,64,0,128,64,0 :rem 182
770 DATA128,64,0,128,64,0,128,64,0 :rem 183
780 DATA255,192 :rem 29
790 Z1=50:Z2=10:Z3=-2:GOSUB830:PRINT"{HOME}{DOWN}"SPC(15)"{YEL}TIME'S UP":GOTO820 :rem 114
800 Z1=10:Z2=50:Z3=2:GOSUB830 :rem 180
810 PRINT"{HOME}{DOWN}"SPC(13)"{YEL}YOU SOLVED IT!" :rem 19

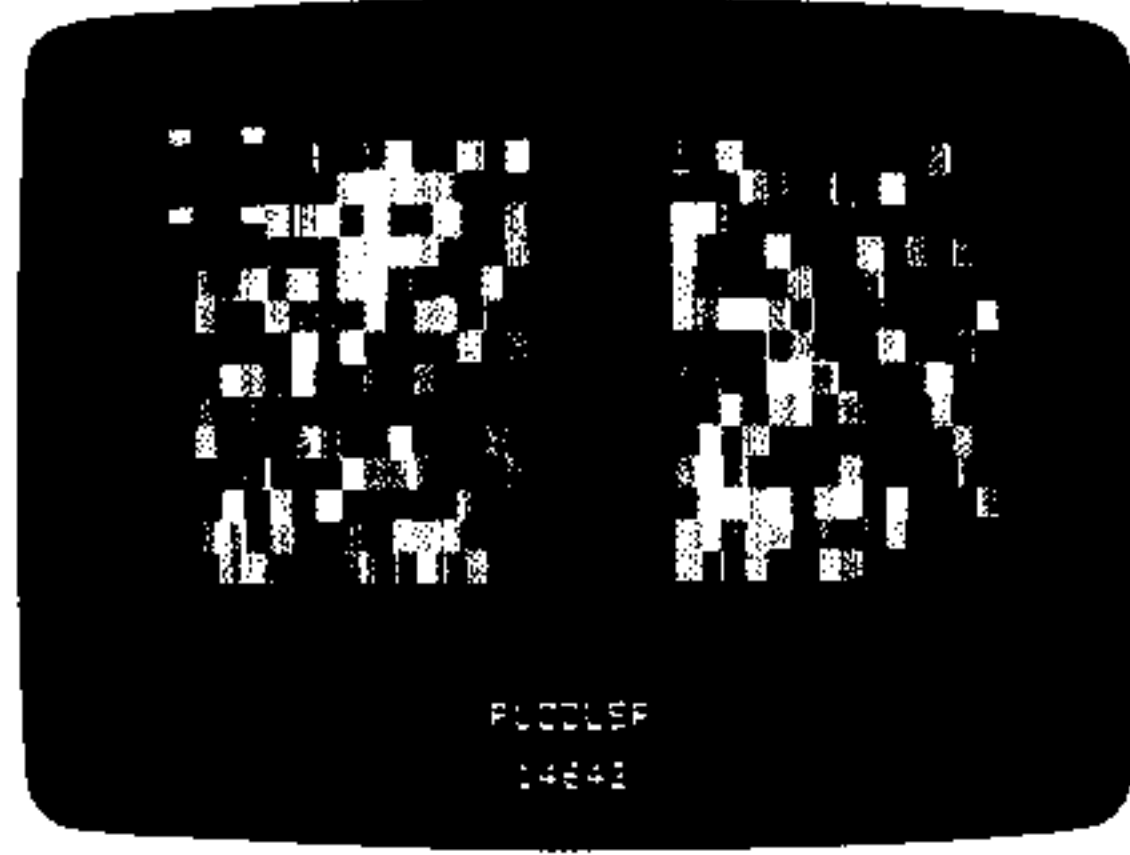
```



```

420 :: F=0 :: GOSUB
470 :: J=TJ :: I=TI :
: RETURN
470 CALL COLOR(#1,CS(F)):
: RETURN
480 IF R1=R2 THEN RETURN
:: REM TRANSPOSE
490 IF R2>R1 THEN A=R1 ::
B=R2 ELSE A=R2 :: B=
R1
500 A$=SEG$(A$,1,A-1)&SEG
$(A$,B,4)&SEG$(A$,A+4
,B-A-4)&SEG$(A$,A,4)&
SEG$(A$,B+4,LEN(A$)-B
+5):: RETURN
510 T=J*4+1+INT((I*2+1)/2
)*C*4 :: RETURN :: RE
M CALC STRING POINTER
520 TEM$=SEG$(TEM$,3,1)&S
EG$(TEM$,1,1)&SEG$(TE
M$,4,1)&SEG$(TEM$,2,1
):: RETURN :: REM ROT
ATE
530 A$=SEG$(A$,1,T-1)&TEM
$&SEG$(A$,T+4,LEN(A$)
-T-3):: RETURN :: REM
SUBSTITUTE ROTATED S
UBSTRING

```



"Puzzler" runs on any Apple II-series computer.

Program 7: Apple Puzzler
Version by Kevin Martin, Editorial Programmer

For instructions on entering this listing, please refer to "COMPUTE!'s Guide to Typing In Programs" published bimonthly in COMPUTE!.

```

29 100 A$ = "": IF PEEK (24576)
= 162 THEN 140
35 110 FOR I = 24576 TO 24872
32 120 READ A: POKE I,A
FE 130 NEXT
C5 140 HIMEM: 24576
50 150 GOSUB 550
50 160 IF T = 0 THEN VTAB 21: PR
INT TAB(14)"OUT OF TIME"
: GOTO 380
DB 170 HTAB 17: VTAB 23: PRINT T
: " "
47 180 T = T - 1
2A 190 IF PEEK (-16384) < 128
THEN 160
90 200 GET C$: IF (C$ < "I" OR C
$ > "L") AND C$ < > " " T
HEN 160
B1 210 R = R - (C$ = "I") + (C$
= "K")
59 220 C = C - (C$ = "J") + (C$
= "L")
7E 230 IF R < 0 THEN R = 0
9E 240 IF R > = R3 THEN R = R3 -
1
AF 250 IF C < 0 THEN C = 0

```

```

DE 260 IF C > = C3 THEN C = C3 -
1
61 270 POKE 773,X1 + C * 2 - 1:
POKE 772,Y1 + R * 2 - 1:
CALL 24671
C0 280 IF C$ < > " " THEN 160
46 290 IF F = 0 THEN 440
0A 300 F = 0: IF RR = R AND CC =
C THEN GOSUB 510: GOTO 3
20
4B 310 GOSUB 460
D2 320 CALL 24691
C4 330 POKE 768,X1: POKE 769,Y1:
CALL 24576
B0 340 POKE 773,X1 + C * 2 - 1:
POKE 772,Y1 + R * 2 - 1:
POKE 774,255: CALL 24753
C3 350 IF A$ < > B$ THEN 160
DA 360 CALL 24691
74 370 HOME : PRINT TAB(16);"CO
RRECT!"
59 380 HTAB 13: VTAB 22: PRINT "
PRESS ANY KEY."
1C 390 HTAB 17: VTAB 23: PRINT T
0F 400 POKE -16368,0
0F 410 IF PEEK (-16384) < 128
THEN 410
D4 420 GET A$
AC 430 RUN
3D 440 F = 1:RR = R:CC = C: POKE
773,X1 + C * 2 - 1: POKE
772,Y1 + R * 2 - 1: POKE
774,119: CALL 24671
9D 450 GOTO 160
8B 460 AA = SS + 2 * NC * RR + 2
* CC:A = SS + 2 * NC * R
+ C * 2
04 470 D = PEEK (A): POKE A, PEE
K (AA): POKE AA,D
C0 480 D = PEEK (A + 1): POKE A
+ 1, PEEK (AA + 1): POKE
AA + 1,D
9B 490 D = PEEK (A + NC): POKE A
+ NC, PEEK (AA + NC): PO
KE AA + NC,D
A4 500 D = PEEK (A + NC + 1): PO
KE A + NC + 1, PEEK (AA +
NC + 1): POKE AA + NC +
1,D: RETURN
4E 510 A = SS + 2 * NC * R + C *
2
4B 520 D = PEEK (A): POKE A, PEE
K (A + NC)
4E 530 POKE A + NC PEEK (A + NC
+ 1)
6F 540 POKE A + NC + 1, PEEK (A
+ 1): POKE A + 1,D: RETUR
N
62 550 TEXT : HOME
0B 560 PRINT TAB(16);"PUZZLER"
F2 570 INPUT "NUMBER OF ROWS (2-
7):";R3
8B 580 IF R3 < 2 OR R3 > 7 THEN
570
77 590 INPUT "NUMBER OF COLUMNS
(2-7):";C3
6F 600 IF C3 < 2 OR C3 > 7 THEN
590
8C 610 INPUT "NUMBER OF COLORS (
2-15):";C0
FD 620 IF C0 < 2 OR C0 > 15 THEN
610
0D 630 PRINT "PLEASE WAIT..."
FC 640 NR = 2 * R3:NC = 2 * C3
7E 650 FOR A = 1 TO NR * NC:B =
INT (RND (1) * C0 + 1):A
$ = A$ + CHR$ (B + B * 16
): NEXT :B$ = A$
A2 660 A = PEEK (105) + PEEK (10
6) * 256
CF 670 SS = PEEK (A + 3) + PEEK
(A + 4) * 256
54 680 X1 = 10 - C3:Y1 = 9 - R3:
X2 = X1 + 20
50 690 POKE 24600, PEEK (A + 3):

```

```

POKE 24601, PEEK (A + 4)
96 700 POKE 768,X2: POKE 769,Y1:
POKE 770,NC: POKE 771,NR
+ Y1
49 710 GR
F4 720 CALL 24576
ED 730 FOR R = 0 TO R3 - 1: FOR
C = 0 TO C3 - 1:B = INT (
RND (1) * 4)
4B 740 IF B THEN GOSUB 510:B = B
- 1: GOTO 740
CA 750 NEXT : NEXT
42 760 FOR R = 0 TO R3 - 1: FOR
C = 0 TO C3 - 1
84 770 RR = INT (RND (1) * R3):
CC = INT (RND (1) * C3):
GOSUB 460: NEXT : NEXT
D2 780 POKE 768,X1: POKE 769,Y1:
CALL 24576
5E 790 HOME : PRINT TAB(16);"PU
ZZLER"
20 800 POKE 772,Y1 - 1: POKE 773
,X1 - 1: POKE 774,255: CA
LL 24753
B3 810 R = 0:C = 0:T = NR * NC *
75: RETURN
0E 820 DATA 162,0,172,1,3,185
C9 830 DATA 47,96,24,109,0,3
93 840 DATA 133,251,185,71,96,10
5
72 850 DATA 0,133,252,160,0,189
8B 860 DATA 140,89,145,251,232,2
00
AF 870 DATA 204,2,3,208,244,238
7B 880 DATA 1,3,173,1,3,205
DD 890 DATA 3,3,208,212,96,0
7E 900 DATA 128,0,128,0,128,0
39 910 DATA 128,40,168,40,168,40
F5 920 DATA 168,40,168,80,208,80
05 930 DATA 208,80,208,80,208,4
4A 940 DATA 4,5,5,6,6,7
21 950 DATA 7,4,4,5,5,6
1E 960 DATA 6,7,7,4,4,5
3E 970 DATA 5,6,6,7,7,32
CA 980 DATA 115,96,76,177,96,24
E3 990 DATA 121,47,96,133,251,18
5
4C 1000 DATA 71,96,105,0,133,252
08 1010 DATA 96,172,7,3,173,8
FA 1020 DATA 3,32,101,96,160,0
67 1030 DATA 162,0,189,9,3,145
D1 1040 DATA 251,232,200,200,200
,189
A5 1050 DATA 9,3,145,251,232,173
8C 1060 DATA 7,3,24,105,3,141
31 1070 DATA 7,3,168,173,8,3
72 1080 DATA 32,101,96,160,0,189
9C 1090 DATA 9,3,145,251,232,200
CD 1100 DATA 200,200,189,9,3,145
BD 1110 DATA 251,232,96,172,4,3
CA 1120 DATA 140,7,3,173,5,3
09 1130 DATA 141,8,3,32,101,96
27 1140 DATA 160,0,162,0,177,251
67 1150 DATA 157,9,3,232,41,15
BD 1160 DATA 145,251,173,6,3,41
51 1170 DATA 240,17,251,145,251,
200
1F 1180 DATA 200,200,177,251,157
,9
FE 1190 DATA 3,232,41,15,145,251
11 1200 DATA 173,6,3,41,240,17
3A 1210 DATA 251,145,251,173,4,3
FB 1220 DATA 24,105,3,141,4,3
AF 1230 DATA 168,173,5,3,32,101
2F 1240 DATA 96,160,0,177,251,15
7
DF 1250 DATA 9,3,232,41,240,145
31 1260 DATA 251,173,6,3,41,15
0F 1270 DATA 17,251,145,251,200,
200
EE 1280 DATA 200,177,251,157,9,3
96 1290 DATA 232,41,240,145,251,
173
C0 1300 DATA 6,3,41,15,17,251
23 1310 DATA 145,251,96

```

Skyscape

Robert M. Simons

This unique program, written by a planetarium director, presents the sky as it can be viewed at any date and time from the year 1977 onward—including zodiac constellations and all the visible planets. It also calculates planet tables, positions of the sun, and phases of the moon for any date and time from 1977 into the future. As an extra (and timely) bonus, it can even display Halley's Comet, due to become visible in late 1985 and early 1986. "Skyscape" is both educational and entertaining. The original version is for the Commodore 64, and we've written additional versions for Apple II-series computers with DOS 3.3 or ProDOS; the TI-99/4A with Extended BASIC; the IBM PC with color/graphics adapter; the PCjr with Cartridge BASIC; and Atari 400/800, XL, and XE computers with at least 24K RAM for tape or 32K for disk.

For thousands of years the sun, moon, and planets in our solar system have excited human imagination. In ancient times they were regarded as gods whose distant motions influenced the course of earthly events. Though we now understand more about the true nature of celestial objects, many facts remain unknown, and a brilliant nighttime sky still presents an inspiring spectacle.

Whether you're seriously interested in the sky or just casually curious, "Skyscape" is a convenient tool for extending your knowledge. It opens a movable window on the heavens, displaying the position of our sun, moon, and neighboring planets from almost any location on Earth, at any point in time from 1977 into the distant future. Since it performs all the necessary calculations, you can enjoy and learn from this program even if you're not an expert in astronomy. In addition to providing data about the position of celestial objects, it draws a sky map on the screen, showing each object as it would appear to you at the chosen location and time.

To get started, type in the appropriate version of Skyscape for your computer and save a copy before running it.

Past, Present, Or Future

Skyscape begins by asking you to answer several questions. Enter the year, choosing any year from 1977 forward. In some ways this is the most important input of all, since objects in our solar system move significantly from one year to the next. After you choose the year, Skyscape allows you to enter the month and day.

Next you must enter the latitude (north/south position on Earth) from which you wish to view

the sky. Latitude 0 places you, the observer, at the equator. Latitudes 1-90 place you in the northern hemisphere (north of the equator). To choose a southern latitude (south of the equator), enter a negative number from -1 to -90. Skyscape generally represents southerly locations with negative values.

Whenever Skyscape asks for information, it checks your entry to make sure it's in the acceptable range. If you enter an illegal value, the program displays an error message and gives you another chance.

The Sun And Moon

Though very different in size and composition, the sun and moon are alike in being the largest celestial objects visible from Earth. After you enter the date and latitude, Skyscape displays a table of data for the sun and moon. In addition to the date, day of the year, and latitude north or south, you'll see the following information:

- Sun's geocentric angle. This figure represents the sun's position as a number of degrees relative to the vernal equinox. The vernal equinox is where the sun is located when spring begins in the northern hemisphere (the same time that autumn begins in the southern hemisphere).

- Sun's declination. The number of degrees north or south of the equator. Negative values indicate a southerly location.

- Sun's altitude at noon. The location of the sun in degrees from the northern or southern horizon at noon.

- Sun's right ascension. Just as longitude and latitude indicate locations on the Earth, *right ascension* and *declination* are used to pinpoint locations in the sky. For this purpose the sky is visualized as a gigantic sphere surrounding the Earth. Declination locates a point vertically in the celestial sphere and right ascension locates it horizontally. Right ascension values are given in *hours* and *minutes* in the range 0:00-23:59. Right ascension 0:00 is exactly at the vernal equinox. Larger right ascension values lie to the east of smaller ones.

- Right ascension at 9 p.m. The right ascension which would be on the meridian at 9 p.m. This coordinate system would be found on star charts. By comparing this number with those charts, you can tell what stars and constellations would be visible at that time.

- Moon's age. The number of days since the last new moon.

- Moon's elongation. The location of the moon in degrees east or west of the sun.

- Moon's phase. The phase of the moon on this particular day.

The Planet Table

After viewing the sun and moon display, press P to continue to the next display screen, which contains the planet table. (Press D if you wish to enter a new date.) The planet table shows vital information about the visible planets (through Uranus, which is at the limit of our visibility). The table shows the position of each planet in right ascension and degrees east or west of the sun. It also shows the distance of each planet from Earth in millions of miles.

If you'd rather see the distance in kilometers, modify the program to change the value of ES=93 to ES=149.6 (the program line which defines the value of ES varies with the version of Skyscape: Commodore 64—line 220; Atari—line 190; IBM—line 130; Apple—line 80;

TI—line 150).

Some planets have an asterisk to the left of the right ascension figure. This signifies that they are visible at 9 o'clock this evening. For reference, the planet table also includes the sun's present right ascension and its right ascension at 9 p.m. Press D to input a new date or S to view a graphics display of the sky at any time in the current day.

The Visible Skyscape

After selecting the sky display, you must enter the hour when you wish to view the sky. The hour value should be a whole number from 0-23 (enter 22 for 10 p.m., etc.). You'll also need to enter the minutes (0-59). Skyscape then displays the time and offers you a chance to enter different values. Press RETURN or Enter when you're satisfied with the time.

Skyscape now displays the sky as it would appear at the chosen latitude, date, and time. Since the sky looks very different from different places on Earth, the latitude affects the display considerably. If your latitude is in the range 24-90 degrees north or south, the sky shows a dashed line representing the position of the celestial equator, along with symbols representing the sun, moon, and planets visible at that time. If your latitude is in the tropical region—from 23½ degrees north to 23½ degrees south—the dashed line indicates a position directly overhead.

If you're viewing in the northern hemisphere, north is above the dashed line and south is below it. In the southern hemisphere these directions are reversed. Below the sky display is a key that interprets the symbols used to represent celestial objects. If more than one object is positioned at the same spot, the symbols are displayed above each other.

At the bottom of the sky you may see two-letter abbreviations. These represent zodiac constellations that would be visible from your chosen vantage point. Skyscape uses the abbreviations AR (Aries), PI (Pisces), AQ (Aquarius), CP (Capricorn), SA (Sagittarius), SC (Scorpio), LI (Libra), VI (Virgo), LE (Leo), CA (Cancer), GE (Gemini) and TA (Taurus). Each constellation is located above the spot where its

abbreviation appears. In northern latitudes, the border of each constellation's zone begins at its abbreviation and extends left. In southern latitudes, the constellation extends right from the position of its abbreviation.

Daytime skies are shown in blue and nighttime skies in black. Skyscape does not calculate the actual rising or setting time of the sun. Average rising and setting times of 6 a.m. and 6 p.m. are used in every case. You may obtain exact rising and setting times from local newspapers. However, keep in mind that there is usually about an hour of twilight before sunrise and after sunset.

Halley's Comet

In addition to permanent objects, Skyscape's graphics display includes Halley's Comet, which should be visible during late 1985 and early 1986. If you choose a date from November 1, 1985 to May 29, 1986, Skyscape calculates the position of Halley's Comet and includes it in the graphics display (if it would be visible at the place and time you select). The comet's position is based on the best predictions available at the time of this writing (summer 1985). These positions may differ slightly from the comet's actual position when it finally makes its appearance.

While Skyscape is generally accurate, it bases most position calculations on circular orbits. This introduces a certain element of error, since no object in our solar system has a perfectly circular orbit. The position error is most pronounced for Mercury and Mars (whose orbits are quite elliptical), but does not significantly affect other objects. I've found Skyscape accurate enough for my own purposes, which include planning astronomy classes and planetarium displays.

For instructions on entering these listings, please refer to "COMPUTE!s Guide to Typing In Programs" published bimonthly in COMPUTE!

Program 1: Commodore 64 Skyscape

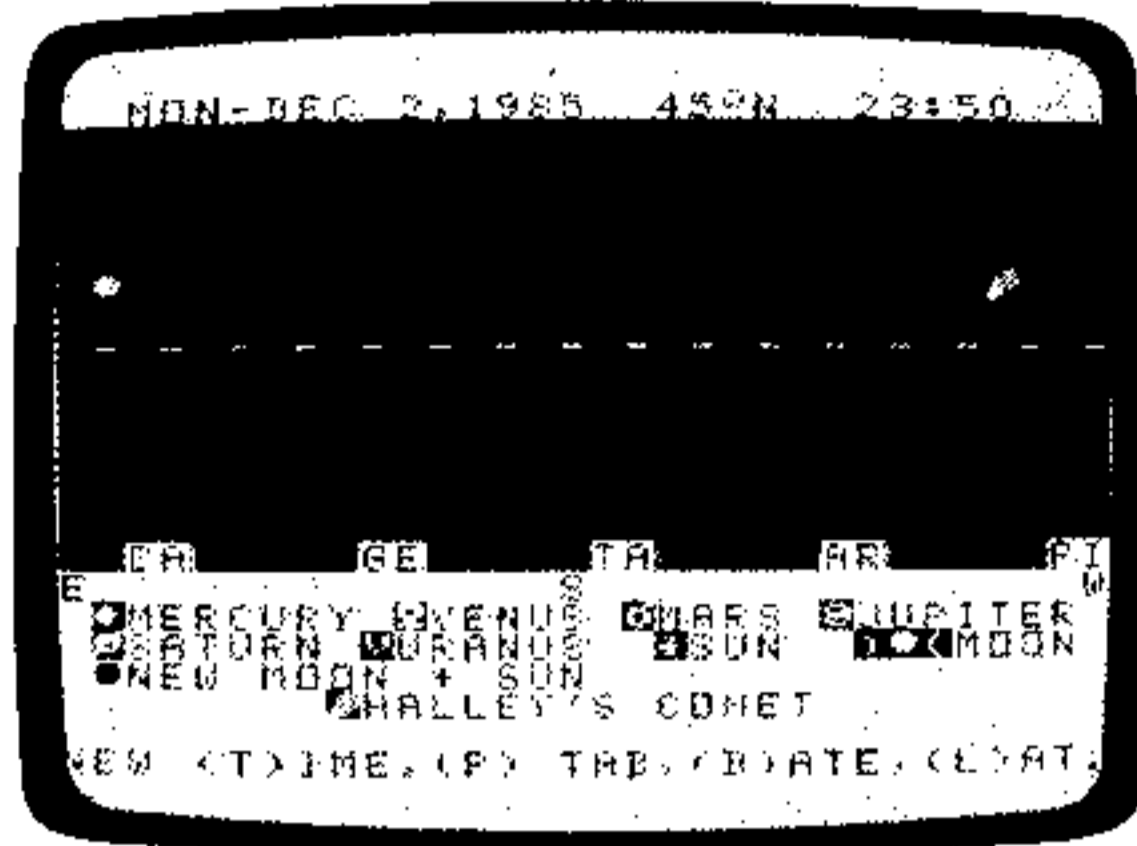
```
100 POKE56,56:POKE55,0:CLR:FOR
I=828TO909:READA:X=X+A:POK
EI,A:NEXTI:PRINTCHR$(8)
:rem 246
110 IFX<>9923THENPRINT"ERROR I
N DATA STATEMENTS.":STOP
:rem 187
```



```

+ 1: GOTO 5040
85 5030 C1$ = MID$(CD$,CI,2):C2
  $ = C1$ + C2$:CI = CI +
    2
88 5040 IF CI < 41 THEN 5010
58 5050 CD$ = C2$: RETURN

```



The TI-99/4A version of "Skyscape."

Program 5: TI-99/4A Skyscape

Version by Patrick Parrish,
Programming Supervisor

```

100 GOTO 130
110 PK=PK-1023 :: PKROW=I
  NT(PK/40)-1 :: PKCOL=
  PK-(PKROW+1)*40 :: RE
  TURN
120 FOR I=1 TO LEN(QQ$)::
  CALL HCHAR(ROW,COL+I
  ,ASC(SEG$(QQ$,I,1))):
  : NEXT I :: RETURN
130 MM$="098108099" :: CA
  LL CLEAR :: CALL SCRE
  EN(15):: DISPLAY AT(1
  1,6):"*** SKYSCAPE *
  ***" :: DISPLAY AT(22
  ,8):"INITIALIZING..."
140 D$="00003105909012015
  1181212243273304334"
  :: K1=1440 :: DIM HC(
  22):: M$="28631734501
  104107210213316419422
  5255"
150 ES=93 :: D1$(1)="S" :
  : D1$(2)="N"
160 A$="JANFEBMARAPR MAYJU
  NJULAUGSEPOCTNOVDEC"
  :: OQ$="OUT OF RANGE!
  !" :: MD$="3128313031
  30313130313031" :: D9
  =PI/180 :: READ EE,M9
170 DIM P(6,6):: DEF R(X)
  =INT(X*100+.5)/100 ::
  DEF S(X)=INT(X*10+.5
  )/10
180 FOR Y=1 TO 2 :: FOR X
  =1 TO 6 :: READ P(X,Y
  ):: NEXT X :: NEXT Y
  :: Y=0
190 FOR X=1 TO 6 :: READ
  P$(X),P(X,3):: NEXT X
200 FOR X=1 TO 7 :: PP(X)
  =X+99 :: NEXT X
210 J$="SAT SUN MONTUEWEDTH
  UFRI" :: CALL SCREEN(
  12):: FOR X=1 TO 12 :
  : READ F$
220 CC$=CC$&RPT$(CHR$(128
  ),5)&F$ :: NEXT X ::
  CC$=CC$&CC$ :: F$=SEG
  $(CC$,LEN(CC$)-8,9)::
  CC$=F$&CC$
230 FOR X=1 TO 8 :: READ

```

```

PH$(X):: NEXT X :: FO
R X=1 TO 22 :: READ H
C(X):: NEXT X :: GOSU
B 2300 :: GOTO 830
240 CC=MT-720 :: IF CC<0
  THEN CC=CC+K1
250 CC=CC/120 :: CD=CC-IN
  T(CC):: CC=INT(CC)::
  CD=INT(CD*7+.2):: CC=
  81-(CC*7+CD)
260 GOSUB 1890 :: QQ$=CD$
  :: ROW=16 :: COL=0 :
  : GOSUB 120
270 IF LL>=0 THEN RETURN
280 FOR I=1 TO 16 :: CALL
  GCHAR(16,I,Z):: CALL
  GCHAR(16,33-I,Z1)::
  CALL HCHAR(16,I,Z1)::
  CALL HCHAR(16,33-I,Z
  ):: NEXT I
290 FOR I=1 TO 31 :: CALL
  GCHAR(16,I,Z):: IF Z
  =128 THEN 310
300 CALL GCHAR(16,I+1,Z1)
  :: CALL HCHAR(16,I,Z1
  ):: CALL HCHAR(16,I+1
  ,Z):: I=I+1
310 NEXT I :: RETURN
320 CALL CLEAR :: DISPLAY
  AT(2,9):"*** DAYS SKY
  ***" :: Q=1 :: GOSUB
  1680
330 DISPLAY AT(6,1):"INPU
  T THE TIME:" :: DISPL
  AY AT(7,1):"-----
  -----" :: T1,T2=0
340 DISPLAY AT(9,4):"HOUR
  (0-23) ?" :: ACCEPT
  AT(9,18):T1 :: IF T1<
  0 OR T1>23 THEN Q=10
  :: GOSUB 2290 :: GOTO
  340
350 DISPLAY AT(11,4):"MIN
  UTE (0-59) ?" :: ACCE
  PT AT(11,20):T2 :: IF
  T2<0 OR T2>59 THEN Q
  =12 :: GOSUB 2290 ::
  GOTO 350
360 R$=STR$(T1):: T$=STR$
  (T2):: IF LEN(T$)=1 T
  HEN T$="0"&T$
370 DISPLAY AT(15,1):"TIM
  E-- ";R$;" ";T$
380 GOSUB 2050 :: IF Z$="
  R" THEN 320
390 CALL CLEAR :: T3=T1*6
  0+T2+AA-720 :: IF T3<
  0 THEN T3=T3+K1
400 IF T3>K1 THEN T3=T3-K
  1
410 MT=T3-360 :: IF MT<0
  THEN MT=MT+K1
420 PT=T3+360 :: IF PT>K1
  THEN PT=PT-K1
430 DISPLAY AT(1,1):K$;"-
  ";TEM$;STR$(Y);TAB(17
  );STR$(ABS(LL));LL$;"
  ";R$;" ";T$;
440 CALL COLOR(9,1,5,10,1
  ,5):: TM=VAL(R$&"."&T
  $):: IF TM<6 OR TM>18
  THEN CALL COLOR(9,1,
  2,10,1,2)
450 FOR X=2 TO 15 :: CALL
  HCHAR(X,1,107,32)::
  NEXT X :: XX=7+LC ::
  FOR I=2 TO 32 STEP 2
  :: CALL HCHAR(XX+1,I,
  96):: NEXT I
460 GOSUB 240 :: ROW=17 :
  : COL=0 :: IF LL<0 TH
  EN 490
470 IF LL>24 THEN QQ$="E

```

```

(14 SPACES)S
(15 SPACES)W" :: GOSUB
  120 :: GOTO 510
480 QQ$="UP-N(6 SPACES)-O
  VERHEAD-(6 SPACES)DOW
  N-S" :: GOSUB 120 ::
  GOTO 510
490 IF ABS(LL)>24 THEN QQ
  $="W(14 SPACES)N
  (15 SPACES)E" :: GOSUB
  120 :: GOTO 510
500 QQ$="UP-S(6 SPACES)-O
  VERHEAD-(6 SPACES)DOW
  N-S" :: GOSUB 120
510 T4=AA :: GOSUB 710 ::
  Y8=888 :: IF Y9=999
  THEN 550
520 Y8=Y9 :: GOSUB 2380 :
  : IF A1<0 THEN 550
530 IF PK>1703 OR PK<1144
  THEN 550
540 GOSUB 110 :: IF PKCOL
  >4 AND PKCOL<37 THEN
  CALL HCHAR(PKROW,PKCO
  L-4,97)
550 T4=AA+M2*K1 :: IF T4>
  K1 THEN T4=T4-K1
560 GOSUB 710 :: IF Y9=99
  9 THEN 600
570 MM=INT(M1/9.83333)+1
  :: GOSUB 810 :: IF Y9
  =999 THEN 600
580 GOSUB 2380 :: IF PK>1
  703 OR PK<1144 THEN 6
  00
590 GOSUB 110 :: IF PKCOL
  >4 AND PKCOL<37 THEN
  CALL HCHAR(PKROW,PKCO
  L-4,MM):: IF ABS(Y8-Y
  9)<=.5 THEN CALL HCHA
  R(PKROW,PKCOL-4,108)
600 FOR X=1 TO 7 :: IF X=
  7 THEN 2170
610 T4=P(X,6):: GOSUB 710
  :: IF Y9=999 THEN 67
  0
620 U9=SIN(P(X,6)*D9/4)::
  U9=-3*U9+.5 :: U9=IN
  T(U9):: U(X)=U9*40
630 PK=1423-Y9+U(X)+LB ::
  GOSUB 2390 :: IF PK>
  1703 OR PK<1144 THEN
  670
640 GOSUB 110
650 IF PKCOL>4 AND PKCOL<
  37 THEN CALL GCHAR(PK
  ROW,PKCOL-4,Z):: IF Z
  <>107 AND Z<>96 THEN
  PK=PK+1023+SGN(LL)*40
  +(LL=0)*40 :: GOTO 64
  0
660 IF PKCOL>4 AND PKCOL<
  37 THEN CALL HCHAR(PK
  ROW,PKCOL-4,PP(X))
670 NEXT X :: QQ$="dMERCU
  RY eVENUS fMARS gJUPI
  TER" :: ROW=18 :: COL
  =1 :: GOSUB 120
680 QQ$="hSATURN iURANUS
  aSUN b1cMOON" :: RO
  W=19 :: GOSUB 120 ::
  QQ$="mNEW MOON + SUN
  " :: ROW=20 :: GOSUB
  120
690 IF B$<>" THEN QQ$=B$
  :: ROW=21 :: COL=8 :
  : GOSUB 120
700 QQ$="NEW (T)IME, (P) T
  AB, (D)ATE, (L)AT." ::
  ROW=23 :: COL=0 :: GO
  SUB 120 :: GOTO 1810
710 Y9=999 :: IF MT<PT TH
  EN 760
720 IF T4>=MT OR T4<=PT T
  HEN 740

```

```

730 RETURN
740 IF T4>=MT AND T4<=K1
THEN 780
750 T4=T4+K1 :: GOTO 780
760 IF T4>=MT AND T4<=PT
THEN 780
770 RETURN
780 Y9=INT((T4-MT)/18+.5)
:: IF Y9=40 THEN Y9=39
790 RETURN
800 U9=SIN(T4/4*D9):: U9=
INT(-3*U9+.5)*40 :: R
ETURN
810 MM=VAL(SEG$(MM$,3*MM-
2,3)):: IF LL<0 AND M
M<>100 THEN MM=197-MM
820 RETURN
830 Q=1
840 CALL CLEAR :: DISPLAY
AT(2,6):"*** SKYSCA
PE ***" :: DISPLAY A
T(4,1):"DATE INPUT" :
: DISPLAY AT(5,1):"--
-----" :: S1=0
850 IF Y<>0 THEN GOSUB 16
80
860 DISPLAY AT(Q+5,1):"YE
AR?" :: ACCEPT AT(Q+5
,7):Y :: IF Y>=1977 T
HEN 880
870 DISPLAY AT(Q+5,14):"M
UST BE >1977" :: FOR
I=1 TO 250 :: NEXT I
:: GOTO 860
880 GOSUB 1730 :: DISPLAY
AT(Q+7,1):"MONTH (1-
12)?" :: ACCEPT AT(Q+
7,15):M :: IF M<1 OR
M>12 THEN Q=Q+8 :: GO
SUB 2290 :: Q=Q-8 ::
GOTO 880
890 DI=VAL(SEG$(M$,2*M-1
,2)):: DI=DI-(M=2)*LY
:: DI$=STR$(DI)
900 DISPLAY AT(Q+9,1):"DA
Y (1-";DI$;"?)" :: AC
CEPT AT(Q+9,13):D ::
IF D<1 OR D>DI THEN Q
=Q+10 :: GOSUB 2290 :
: Q=Q-10 :: GOTO 900
910 H$=SEG$(A$,M*3-2,3)
920 DISPLAY AT(Q+11,1):"L
ATITUDE (-90 TO 90)?"
:: ACCEPT AT(Q+11,23
):LL :: IF ABS(LL)>90
THEN Q=Q+12 :: GOSUB
2290 :: Q=Q-12 :: GO
TO 920
930 GOSUB 2410
940 TEM$=H$&" "&STR$(D)&"
," :: DISPLAY AT(Q+14
,8):TEM$:Y :: GOSUB 2
050 :: IF Z$="R" THEN
Q=4 :: GOTO 840
950 D2=VAL(SEG$(M$,M*3-2,
3))+D :: GOSUB 1760 :
: IF M>2 THEN D1=D1+L
Y :: Y1=Y1+LY
960 D3=D2-185 :: IF M=3 A
ND D<20 THEN D2=D2+LY
:: D3=D3+LY
970 S5=0 :: IF D3<=0 THEN
A=180*D2/185 :: GOTO
990
980 A=180*D3/(180+ZY)+180
990 IF A<>180 THEN S5=23.
43333333*SIN(D9*D2*18
0/185)
1000 IF A>180 THEN S5=-23
.43333333*SIN(D9*D3)
1010 IF A>=360 THEN A=A-3
60
1020 A=R(A):: S5=R(S5)::

```

```

A1=(SGN(LL)-(LL=0))*
S5+90-ABS(LL):: A1=R
(A1):: GOSUB 1380 ::
GOSUB 1330
1030 W=1-(SGN(LL)<0):: IF
A1>90 THEN A1=180-A
1 :: W=ABS(W-3)
1040 CALL CLEAR :: PRINT
:: PRINT K$;"-";TEM$
;Y;TAB(19);ABS(LL);L
L$ :: PRINT RPT$("-"
,28)
1050 PRINT :: PRINT "DAY
OF THE YEAR--- ";ST
R$(D1):: PRINT :: PR
INT "SUN'S DATA:" :
:
1060 PRINT "GEOCENTRIC AN
GLE-- ";STR$(A);"@"
1070 PRINT "DECLINATION--
----- ";STR$(S5);"@"
"
1080 PRINT "ALTITUDE AT N
OON-- ";STR$(A1);"@"
";D1$(W)
1090 PRINT "RIGHT ASCENSI
ON--- ";A3$
1100 PRINT "R.A. AT 9:00
PM--- ";A5$ :: PRIN
T :: PRINT "MOON'S D
ATA:" :
:
1110 PRINT "AGE-----
----- ";STR$(M1);"
";"DY";
1120 PRINT "ELONGATION---
----- ";STR$(M8);"@"
";L$
1130 PRINT "PHASE - ";PH
$(M3): : :
1140 PRINT "(P)LANET TABL
E OR NEW (D)ATE" ::
PRINT :: GOTO 1810
1150 CALL CLEAR :: PRINT
TAB(6);"** PLANET TA
BLE **" :: PRINT ::
PRINT K$;"-- ";TEM$;
Y;TAB(20);STR$(ABS(L
L));LL$ :: PRINT ::
S1=1
1160 PRINT "PLANET DIST.
ANG.W/SUN R.A." ::
PRINT RPT$("- ",28):
:
1170 FOR X=1 TO 6 :: A2=Y
1/P(X,2)-INT(Y1/P(X,
2)):: Q3=1
1180 A2=A2*360+P(X,1):: I
F A2>360 THEN A2=A2-
360
1190 E=180+A :: IF E>360
THEN E=E-360
1200 E1=ABS(E-A2):: IF E1
>180 THEN E1=360-E1
1210 GOSUB 1440 :: E1=E1*
D9 :: P5=P(X,3):: IF
X=3 THEN GOSUB 1870
1220 P(X,4)=SQR(1+P5^2-2*
P5*COS(E1)):: XX=(P5
^2-1-P(X,4)^2)/(-2*P
(X,4))
1230 P(X,5)=-ATN(XX/SQR(-
XX*XX+1))+PI/2 :: P(
X,4)=INT(P(X,4)*ES+.
5):: P(X,5)=P(X,5)/D
9
1240 P(X,5)=S(P(X,5)):: Q
1$=STR$(P(X,4)):: Q2
$=STR$(P(X,5))
1250 Q1=LEN(Q1$):: Q2=LEN
(Q2$):: GOSUB 1540
1260 PRINT P$(X);TAB(13-Q
1);Q1$;TAB(20-Q2);Q2
$:: IF Q3=-1 THEN P
RINT "0W";
1270 IF Q3=1 THEN PRINT "
0E";

```

```

1280 GOSUB 1590 :: Q4$=ST
R$(Q4):: Q5$=STR$(Q5
):: IF Q5<10 THEN Q5
$="0"&Q5$
1290 Q4$=Q4$&" "&Q5$ :: Z
=LEN(Q4$)
1300 PRINT TAB(22);Q4$;TA
B(29-Z);Q4$:: NEXT
X :: PRINT :: PRINT
:: PRINT :: PRINT "
- VISIBLE AT 9 P.M.
"
1310 PRINT :: PRINT :: PR
INT "SUN'S R.A. ----
----- ";A3$ :: PRI
NT "R.A. AT 9:00 P.M
. --- ";A5$
1320 PRINT :: PRINT TAB(3
);"DAYS (S)KY
(3 SPACES)NEW (D)ATE
" :: GOTO 1810
1330 A2=K1*A/360 :: IF A2
>K1 THEN A2=A2-K1
1340 A3=INT(A2/60):: A4=A
2-A3*60 :: A5=A3+9 :
: IF A5>23 THEN A5=A
5-24
1350 A4=INT(A2-A3*60+.5):
: IF A4=60 THEN A4=0
:: A3=A3+1
1360 IF A3=24 THEN A3=0
1370 AA=A3*60+A4 :: GOTO
1690
1380 M1=(Y1/M9-INT(Y1/M9
))*M9+10 :: IF M1>M9
THEN M1=M1-M9
1390 GOSUB 2080 :: M8=360
*M2 :: IF M8>180 THE
N L$="W"
1400 IF M8<=180 THEN L$="
E"
1410 IF M8>180 THEN M8=36
0-M8
1420 M1=R(M1):: M8=R(M8):
: YY=INT(7*(Y1/7-INT
(Y1/7))+.2):: IF YY=
0 THEN YY=7
1430 K$=SEG$(J$,YY*3-2,3)
:: RETURN
1440 Q3=0 :: Q1=E+180 ::
IF Q1>360 THEN 1480
1450 IF A2>E AND A2<Q1 TH
EN 1470
1460 Q3=1 :: RETURN
1470 Q3=-1 :: RETURN
1480 Q1=Q1-360 :: IF A2<=
360 AND A2>E THEN 14
70
1490 IF Q3<>0 THEN RETURN
1500 IF A2>0 AND A2<=Q1 T
HEN 1470
1510 IF Q3<>0 THEN RETURN
1520 IF A2>Q1 THEN 1460
1530 RETURN
1540 Q5=Q3*P(X,5)*4+AA ::
IF Q5<0 THEN Q5=Q5+
K1
1550 IF Q5>K1 THEN Q5=Q5-
K1
1560 P(X,6)=Q5 :: Q4=INT(
Q5/60):: Q5=INT(Q5-Q
4*60+.5):: IF Q5=60
THEN Q5=0 :: Q4=Q4+1
1570 IF Q4=24 THEN Q4=0
1580 RETURN
1590 SU=A5*60+A4 :: PS=SU
+360 :: MS=SU-360 ::
IF PS>K1 THEN PS=PS
-K1
1600 IF MS<0 THEN MS=MS+K
1
1610 IF MS>PS THEN 1640
1620 IF P(X,6)<PS AND P(X
,6)>MS THEN 1670
1630 QQ$=" " :: RETURN

```

```

1640 IF P(X,6)<K1 AND P(X
,6)>MS THEN 1670
1650 IF P(X,6)<PS THEN 16
70
1660 GOTO 1630
1670 QQ$="*" :: RETURN
1680 DISPLAY AT(Q+3,1):K$
;"--";TEM$;Y;TAB(20
);STR$(ABS(LL));LL$;
:: RETURN
1690 A3$=STR$(A3):: IF A3
<10 THEN A3$=" "&A3$
1700 A4$=STR$(A4):: IF A4
<10 THEN A4$="0"&A4$
1710 A3$=A3$&" "&A4$ :: A
5$=STR$(A5):: IF A5<
10 THEN A5$=" "&A5$
1720 A5$=A5$&" "&A4$ :: Q
B=7-LEN(A3$):: Q9=7-
LEN(A5$):: RETURN
1730 LY=0 :: IF Y/4=INT(Y
/4)THEN LY=1
1740 IF Y/100=INT(Y/100)A
ND Y/400=INT(Y/400)A
ND Y/1000=INT(Y/1000
)AND Y/4000=INT(Y/40
00)THEN LY=0
1750 RETURN
1760 Y9=Y+1 :: IF Y9/4=IN
T(Y9/4)THEN ZY=1
1770 IF Y9/100=INT(Y9/100
)AND Y9/400<>INT(Y9/
400)AND Y9/1000=INT(
Y9/1000)AND Y9/4000=
INT(Y/4000)THEN ZY=0
1780 Y1=Y-1977 :: Y1=Y1*3
65+INT(Y1/4)+D1 :: I
F Y<2000 THEN 1800
1790 Y1=Y1-INT((Y-2001)/1
00)+INT((Y-2001)/400
)-INT((Y-1)/4000)
1800 RETURN
1810 CALL KEY(0,KK,SS)::
IF SS=0 THEN 1810
1820 I$=CHR$(KK):: IF I$=
"D" THEN Q=4 :: GOTO
840
1830 IF (I$="S" OR I$="T"
)AND S1=1 THEN 320
1840 IF I$="P" THEN 1150
1850 IF I$="L" AND S1=1 T
HEN 2460
1860 GOTO 1810
1870 P5=1.376344086 :: K5
=A2*4
1880 K5=ABS(K5-1233.73)*9
0/K1 :: K5=K5*D9 ::
K5=SIN(K5)*.32258122
4 :: P5=P5+K5 :: RET
URN
1890 IF CC<=1 THEN CC=CC+
84
1900 CD$=SEG$(CC$,CC+3,34
)
1910 IF SEG$(CD$,2,1)<>CH
R$(128)AND SEG$(CD$,
3,1)=CHR$(128)THEN C
D$=SEG$(CD$,1,32)::
GOTO 1940
1920 IF SEG$(CD$,33,1)<>C
HR$(128)AND SEG$(CD$,
32,1)=CHR$(128)THEN
CD$=SEG$(CD$,3,32)::
GOTO 1940
1930 CD$=SEG$(CD$,2,32)
1940 RETURN
1950 DATA 365.26,29.53059
,59.818184,42.719626
,262.364294,52.91676
3
1960 DATA 134.69697,218.7
9464,87.97,224.7,686
.98
1970 DATA 4332.79813,1075

```

```

9.7195,30686.5884
1980 DATA "MERCURY",.3871
,"VENUS",.7233,"MARS
",1.5237,"JUPITER",5
.2028
1990 DATA "SATURN",9.5308
,"URANUS",19.182
2000 DATA "SA","SC","LI",
,"VI","LE","CA","GE",
,"TA","AR","PI","AQ",
,"CP"
2010 DATA "NEW","WAXING C
RESCENT","1ST QUARTE
R","WAXING GIBBOUS",
"FULL"
2020 DATA "WANING GIBBOUS
","3RD QUARTER","WAN
ING CRESCENT"
2030 DATA 1770,1719,1620,
1500,1418,1365,1335,
1310,1290,1275,1260
2040 DATA 1238,1220,1200,
1178,1115,915,720,66
0,640,625,610
2050 DISPLAY AT(20,3):"(R
)E-INPUT OR (C)ONTIN
UE"
2060 CALL KEY(0,KK,SS)::
IF SS=0 THEN 2060
2070 Z$=CHR$(KK):: RETURN
2080 M2=M1/M9 :: IF M1<1
OR M1>28.5 THEN M3=1
2090 IF M1>=1 AND M1<6.9
THEN M3=2
2100 IF M1<=8 AND M1>=6.9
THEN M3=3
2110 IF M1>8 AND M1<14.2
THEN M3=4
2120 IF M1>=14.2 AND M1<=
15.2 THEN M3=5
2130 IF M1>15.2 AND M1<21
.6 THEN M3=6
2140 IF M1>=21.6 AND M1<=
22.6 THEN M3=7
2150 IF M1>22.6 AND M1<=2
8.5 THEN M3=8
2160 RETURN
2170 B$="" :: IF Y<>1985
AND Y<>1986 THEN 670
2180 IF (Y=1985 AND D1<30
5)OR(Y=1986 AND D1>1
49)THEN 670
2190 HD=D1+365 :: IF HD>5
16 THEN HD=HD-365
2200 H1=(HD-295)/10 :: HD
=INT(H1):: H1=H1-HD
2210 T4=HC(HD)-HC(HD+1)::
T4=HC(HD)-H1*T4 ::
IF T4>K1 THEN T4=T4-
K1
2220 GOSUB 710 :: IF Y9=9
99 THEN 670
2230 GOSUB 800 :: IF T4>1
15 AND T4<1200 THEN
U9=U9+40
2240 IF T4>1290 THEN U9=U
9-40
2250 IF T4>615 AND T4<=11
15 THEN U9=U9+80
2260 U(7)=U9 :: B$="JHALL
EY'S COMET" :: GOTO
630
2270 B$="HALLEY'S COMET"
2280 GOTO 630
2290 DISPLAY AT(Q,1):00$
:: FOR I=1 TO 250 ::
NEXT I :: CALL HCHA
R(Q,3,32,14):: RETUR
N
2300 CALL CHAR(64,"384444
4438000000",128,RPT$
("0",16))
2310 FOR I=0 TO 3 :: READ
SS :: CALL CHARPAT(

```

```

SS,00$):: CALL CHAR(
I+96,00$):: NEXT I
2320 DATA 45,42,41,40
2330 FOR I=0 TO 9 :: READ
QQ$ :: CALL CHAR(10
0+I,QQ$):: NEXT I ::
CALL COLOR(13,2,9)
2340 DATA 0010387C7C38100
0,3C7E66663C187E10,0
3063C6666663800
2350 DATA 003C427E7E423C0
0,033E6E76667CC080,0
066666624180000,0A15
2A352A74F8E0
2360 DATA 0000000000000000
0,003C7E7E7E3C00,FFC
381818181C3FF
2370 RETURN
2380 GOSUB 800 :: PK=1423
-Y9+U9+LB
2390 IF LL<0 THEN PK=2247
+80*XX-PK
2400 RETURN
2410 LL$="0N" :: IF LL<0
THEN LL$="0S"
2420 L1=ABS(LL):: IF ABS(
LL)<24 THEN L1=40
2430 LC=INT((L1-40)/7+.5)
:: LB=LC*40 :: D1=VA
L(SEG$(D$,M*3-2,3))+
D
2440 IF ABS(LL)<24 THEN L
B=40*INT(ABS(LL)/7+.
5)
2450 RETURN
2460 GOSUB 2510 :: DISPLA
Y AT(6,1):"LATITUDE
CHANGE" :: DISPLAY A
T(7,1):RPT$("- ",16)
2470 DISPLAY AT(9,1):"INP
UT NEW LATITUDE:" ::
ACCEPT AT(9,21):LL
2480 IF ABS(LL)>90 THEN 2
470
2490 GOSUB 2050 :: IF Z$=
"R" THEN 2410
2500 GOSUB 2410 :: I$="S"
:: GOTO 1830
2510 CALL CLEAR :: DISPLA
Y AT(2,6):"*** SKYS
CAPE ***" :: Q=1 ::
GOSUB 1680 :: RETUR
N

```

This Publication
is available in Microform.

University Microfilms
International

Please send additional information

for _____

Name _____

Institution _____

Street _____

City _____

State _____ Zip _____

300 North Zeeb Road, Dept. P.R., Ann Arbor, Mi. 48106



The States Of The Union

I have previously written programs to identify the states and their capitals in certain regions ("Southern States," *COMPUTE!*, August 1984; "Western States" and "New England States," *Programmer's Reference Guide to the TI-99/4A*, *COMPUTE!* Books). Now, after numerous requests, I'll demonstrate a program for the middle Atlantic states.

The central portion of the United States can be outlined on a map as I did with the Southern States and Western States programs. However, some of the middle Atlantic states such as Delaware, Maryland, and New Jersey are too small to outline adequately on the TI-99/4A. The color-blinking method used in the New England States program won't work, either, because more color sets are needed than are available. Therefore, I've used a slightly different method for the region which includes Delaware, Maryland, New Jersey, New York, Pennsylvania, Virginia, and West Virginia. The states are drawn in high-resolution graphics with some color sets used for more than one state.

When a question mark appears on a state, the student types in the state's name. If the response is correct, the program asks for the state capital.

Be careful typing the DATA statements—make sure the commas are placed correctly. Lines 290-380 contain definitions for graphics characters. Lines 820-840 contain sets of row, column, and character numbers for drawing the graphics.

If you wish to save typing effort, you can obtain a copy of this program by sending a blank cassette or disk, a stamped, self-addressed mailer, and \$3 to:

C. Regena
P.O. Box 1502
Cedar City, UT 84720

Please be sure to specify the title ("Middle Atlantic States") and that you need the TI version.

Middle Atlantic States

```

100 REM NORTH ATLANTIC
110 CALL CLEAR
120 PRINT " *****
*****": " *";TA
B(25); "*"
130 PRINT " * IDENTIFY T
HE STATES *": " *";TA
B(25); "*"
140 PRINT " *****
*****"
150 PRINT ":(3 SPACES)NO
RTH ATLANTIC STATES"
160 PRINT "!!"TYPE THE NA
ME OF THE STATE THEN
PRESS <ENTER>."
170 PRINT "IF THE STATE
IS CORRECT,
(4 SPACES)TYPE THE CA
PITAL CITY."
180 PRINT "NAMES MUST BE
SPELLED(7 SPACES)COR
RECTLY TO BE ACCEPTED
."
190 PRINT "PRESS FCTN 8
REDO TO ERASE."
200 CALL CHAR(33, "3F3F3F3
F3F3F3F3")
210 CALL CHAR(35, "FFFFFF
FFFFFF")
220 CALL CHAR(43, "F8F8F8F
8F8F8F8F")
230 CALL CHAR(45, "F0F0F0F
0F0F0F0F")
240 CALL CHAR(47, "FFFFFF
FFFFFF")
250 FOR C=96 TO 159
260 READ C#
270 CALL CHAR(C,C#)
280 NEXT C
290 DATA FFFFEFEFCFCFCF8
,F0F0F0F0F0F0FEFF,FFF
FFEFCFCFCFCF8F8,3F3F3F3
F3F3F3F3,0
300 DATA 0,,,00000103073F
FFFF,03030707676FFFFF
310 DATA 0000010103030F0F
,0F0F0F0F0F0F0703,C0C
00000000E0E0,F8F8F8F
8F0E0C08,FFFFFFFEFCFC
F8F
320 DATA F0E0602,80F8FEFF
FFFFFFF,F7E0C0C0000,
F0E0C08,FFFFFFF2F
0C
330 DATA FFFFFFFFCFCF8F8

```

```

,F0F0F0F0C,FFFFFFFFF
F8,FF3F3F3F3F3F1F,030
3030303030303
340 DATA FFFFFFFCF8F0C,FF
FF,,,,,FFFFFFFFF3F1
F1F,0F070301,FF7FFFFF
FF3F1F0F,FFFEFE,,,,F
FFFFFFFFFFFFFFF
350 DATA DFDF9F9F9F9F9F9F
,9F9F9F9F9F8FCFC7,,,,
,FFFFFFFFFFFFFFF,00
0000000030F3F,000000
01071FFFFF
360 DATA 00000098FCFFFFFFF
,F8FCFEFFFFFFF,FFF
FFFFFFEFEC7E,C0E0E0F0
C00080C,0000000C0E0F
0F
370 DATA FFFFE7E1E0E,FCF
CFCFCFCFCFCFC,0103070
F1F3F7FFF,0103071F3FF
FFFF,000000000000FFF
F
380 DATA 0000000003F3F1F
,1F0F0F0F0F1F3FFF,010
303070F0F0F0F
390 CALL COLOR(1,7,1)
400 CALL COLOR(9,7,12)
410 CALL COLOR(10,12,1)
420 CALL COLOR(11,12,3)
430 CALL COLOR(12,14,12)
440 CALL COLOR(13,14,3)
450 CALL COLOR(14,14,1)
460 CALL COLOR(15,3,1)
470 CALL COLOR(16,3,1)
480 RESTORE 490
490 DATA NEW YORK,ALBANY,
7,25,PENNSYLVANIA,HAR
RISBURG,10,21,NEW JER
SEY,TRENTON,11,26
500 DATA WEST VIRGINIA,CH
ARLESTON,15,17,MARYLA
ND,ANNAPOLIS,13,23
510 DATA DELAWARE,DOVER,1
4,25,VIRGINIA,RICHMON
D,17,21
520 FOR L=1 TO 7
530 READ S*(L),CAP*(L),X(
L),Y(L)
540 NEXT L
550 PRINT "PRESS <ENTER>
TO START.";
560 CALL KEY(0,K,S)
570 IF K<>13 THEN 560
580 CALL CLEAR
590 CALL SCREEN(0)
600 CALL COLOR(2,16,1)
610 PRINT TAB(16);"!####
##"
620 PRINT TAB(16);"!####
#'p"
630 PRINT TAB(16);"!####
#ad1"

```

```

640 PRINT TAB(16);"c####
#bdm"
650 PRINT TAB(15);"hdxyz
(3 SPACES)+n"
660 PRINT TAB(14);"iddddq
(3 SPACES)-o"
670 PRINT TAB(13);"jdddsr
(4 SPACES)/"
680 PRINT TAB(13);"kddt"
690 PRINT TAB(14);"wvu":
:
:
:
:
700 CALL VCHAR(3,27,153,3
)
710 CALL VCHAR(6,27,144,4
)
720 CALL VCHAR(3,26,144,7
)
730 CALL VCHAR(4,25,144,5
)
740 CALL VCHAR(5,24,144,4
)
750 CALL HCHAR(7,21,144,3
)
760 CALL HCHAR(8,20,144,4
)
770 RESTORE 820
780 FOR I=1 TO 28
790 READ R,C,G
800 CALL HCHAR(R,C,G)
810 NEXT I
820 DATA 3,25,154,4,24,15
4,6,23,155,6,22,156,6
,21,156,6,20,157,7,20
,158,8,19,159,10,27,1
52,10,28,146
830 DATA 10,29,147,13,22,
128,13,23,136,13,24,1
36,14,22,129,14,23,13
6,14,24,137,15,23,138
,15,24,138
840 DATA 16,25,131,16,24,
151,16,23,144,17,24,1
50,17,23,149,18,23,14
8,18,24,147,18,14,146
,17,15,145
850 CALL VCHAR(15,22,144,
4)
860 CALL VCHAR(15,21,144,
4)
870 CALL VCHAR(16,20,144,
3)
880 CALL VCHAR(16,19,144,
3)
890 CALL HCHAR(18,15,144,
4)
900 FOR C=1 TO 7
910 T=0
920 RANDOMIZE
930 R=INT(7*RND)+1
940 IF S$(R)=" THEN 930
950 CALL GCHAR(X(R),Y(R),
G)
960 CALL HCHAR(20,1,100,1
60)
970 FOR L=1 TO 7
980 CALL HCHAR(21,2+L,ASC
(SEG$("STATE ?",L,1))
)
990 CALL HCHAR(X(R),Y(R),
32)
1000 CALL HCHAR(X(R),Y(R)
,63)
1010 NEXT L
1020 CALL HCHAR(21,11,100
,15)
1030 S1$=""
1040 CALL SOUND(150,1397,
2)

```

```

1050 FOR L=1 TO 15
1060 CALL KEY(0,K,S)
1070 IF S<1 THEN 1060
1080 IF K=13 THEN 1130
1090 IF K=6 THEN 1020
1100 CALL HCHAR(21,10+L,K
)
1110 S1$=S1$&CHR$(K)
1120 NEXT L
1130 CALL SOUND(100,880,2
)
1140 IF S$(R)=S1$ THEN 12
60
1150 CALL SOUND(100,330,2
)
1160 CALL SOUND(100,262,2
)
1170 T=T+1
1180 IF T<2 THEN 1020
1190 CALL HCHAR(21,11,100
,15)
1200 FOR L=1 TO LEN(S$(R)
)
1210 CALL HCHAR(21,10+L,A
SC(SEG$(S$(R),L,1)))
1220 NEXT L
1230 GOSUB 1600
1240 C=C-1
1250 GOTO 1570
1260 GOSUB 1670
1270 FOR L=1 TO 9
1280 CALL HCHAR(23,2+L,AS
C(SEG$("CAPITAL ?",L
,1)))
1290 NEXT L
1300 T=0
1310 CALL HCHAR(23,13,100
,15)
1320 S1$=""
1330 CALL SOUND(150,1397,
2)
1340 FOR L=1 TO 15
1350 CALL KEY(0,K,S)
1360 IF S<1 THEN 1350
1370 IF K=13 THEN 1430
1380 IF K=6 THEN 1310
1390 CALL HCHAR(23,12+L,K
)
1400 S1$=S1$&CHR$(K)
1410 NEXT L
1420 CALL SOUND(100,880,2
)
1430 IF CAP$(R)=S1$ THEN
1550
1440 CALL SOUND(100,330,2
)
1450 CALL SOUND(100,262,2
)
1460 T=T+1
1470 IF T<2 THEN 1310
1480 CALL HCHAR(23,12,100
,15)
1490 FOR L=1 TO LEN(CAP$(
R))
1500 CALL HCHAR(23,12+L,A
SC(SEG$(CAP$(R),L,1)
))
1510 NEXT L
1520 GOSUB 1600
1530 C=C-1
1540 GOTO 1570
1550 GOSUB 1670
1560 S$(R)=" "
1570 CALL HCHAR(X(R),Y(R)
,G)
1580 NEXT C
1590 GOTO 1720
1600 FOR L=1 TO 11

```

```

1610 CALL HCHAR(24,20+L,A
SC(SEG$("PRESS ENTER
",L,1)))
1620 NEXT L
1630 CALL KEY(0,K,S)
1640 IF K<>13 THEN 1630
1650 CALL HCHAR(24,21,100
,11)
1660 RETURN
1670 CALL SOUND(100,262,2
)
1680 CALL SOUND(100,330,2
)
1690 CALL SOUND(100,392,2
)
1700 CALL SOUND(200,523,2
)
1710 RETURN
1720 CALL CLEAR
1730 PRINT "TRY AGAIN? Y
OR N"
1740 CALL KEY(0,K,S)
1750 IF K=89 THEN 480
1760 IF K<>78 THEN 1740
1770 CALL CLEAR
1780 END

```

©

the software, including a diagnostic checkout program, a warm reboot program, and a *MicroPainter* file display program.

Suggested retail is \$49.95 for the interface, software, and manual. There is a 30-day money back guarantee, a 90-day replacement warranty, and a lifetime \$19 repair/replacement policy.

Integrated Computer Equipment Company, 8507 Natural Bridge Road, St. Louis, MO 63121.

Circle Reader Service Number 225.

64, Apple Karate

Data East has converted its arcade action game, *Karate Champ*, to a new computer version for the Commodore 64 and Apple II series. The \$29.95 game features two-player and player-versus-computer modes as you guide your karate fighter through successive matches. Using the joystick, you can make your fighter lunge, kick, spin, somersault, reverse-punch, and block.

Data East USA, Inc., 470 Gianni Street, Santa Clara, CA 95054.

Circle Reader Service Number 226.

Pascal For 64 & 128

A complete Pascal development system for the Commodore 64 and 128 has been released by Abacus Software. *Super Pascal* includes an extensive compiler, a source file editor, an integrated assembler, and a comprehensive utility package for file and disk management.

Also included are a handbook with more than 200 pages and a systems disk. Suggested retail price is \$59.95.

Abacus Software, 2201 Kalamazoo SE, P.O. Box 7211, Grand Rapids, MI 49510.

Circle Reader Service Number 227.

New From Epson

Epson has developed several new printers for home users. Among these are the DX-10 (\$399), a daisywheel printer which prints at ten characters per second (cps); and the DX-20 (\$499), a daisywheel with a 1K byte print buffer, 20 characters-per-second (cps) print speed, and a Diablo All Purpose Interface (RS-232C, IEEE-488, and parallel).

Also new from Epson is the Spectrum LX-90, a dot-matrix printer with draft and near-letter-quality (NLQ) modes. It comes with a printer interface cartridge that makes it ready for use with the IBM PC, PCjr, or Apple IIc. Draft copy is printed at 100 cps; NLQ at 16 cps. The Spectrum LX-90 retails for \$389.

Comrex, a division of Epson, has released the CR-128 intelligent printer buffer. Features include 128K buffer

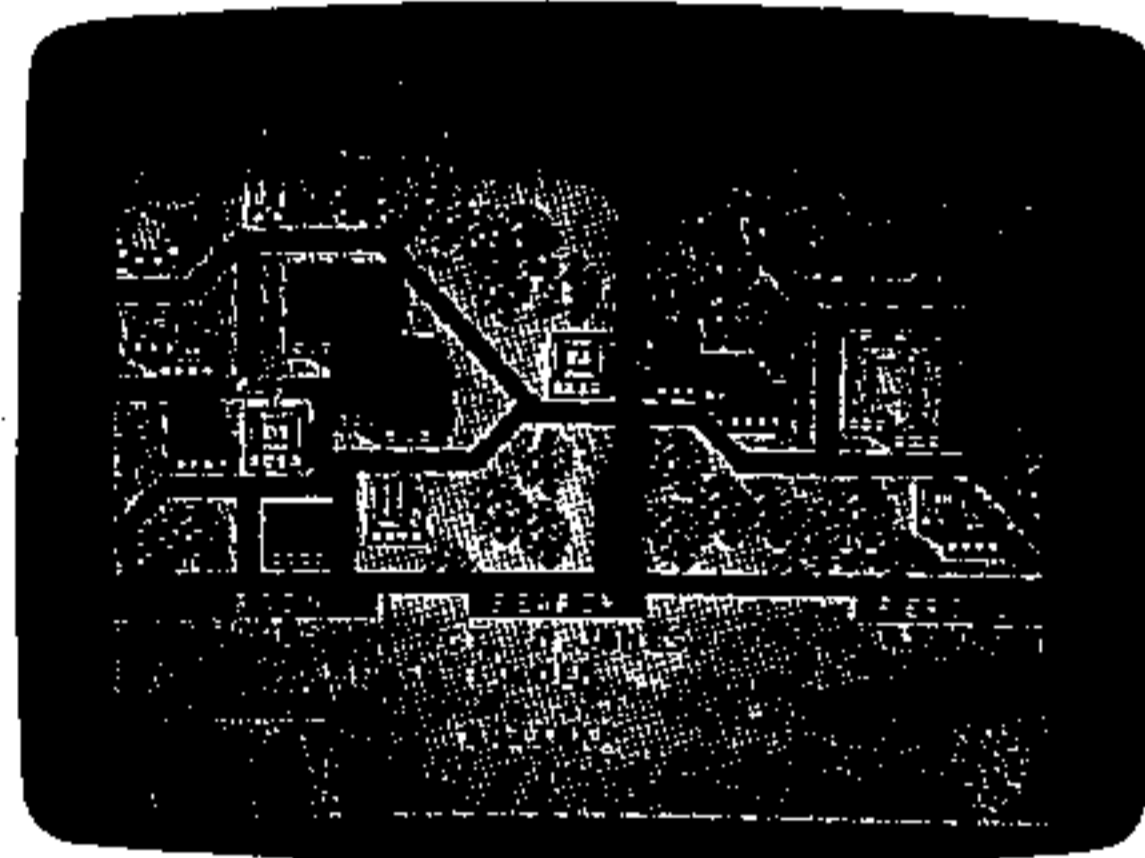
memory and built-in serial-to-serial, serial-to-parallel, parallel-to-parallel, and parallel-to-serial interfaces. Suggested retail is \$299. Another new Comrex product is the CR-700 series of bidirectional A-B switch boxes, which simplify connections to the computer and eliminate the need to swap cables when changing peripherals. The switch boxes retail for \$39.95.

Epson America, 2780 Lomita Blvd., Torrance, CA 90505.

Circle Reader Service Number 228.

World War II Combat Game

Under Fire, from Avalon Hill, combines the depth of a war game simulation with the colorful graphics of an arcade game. Authentic armies, weapons, and maps from World War II add to the game's realism. Different maps and scenarios are included on disk; players can also create their own.



A sample screen from Avalon Hill's *Under Fire* strategy game.

Under Fire is available for the Apple II series. A joystick is optional for the Apple IIc and IIe, but required for the II+. Suggested retail price is \$59.95.

Avalon Hill Game Company, 4517 Hartford Rd., Baltimore, MD 21214.

Circle Reader Service Number 229.

TI Disk Organizer

TI programs can be organized on a single disk with *Disk Data Base* from Asgard Software. The program lets you sort and print a catalog by either disk name or filename, to print it out unsorted, or to selectively print out all entries that contain a certain string. The catalog can also be broken up into blocks of 250 entries for easy management.

Data files can be converted from the Master Disk File to the *Disk Data Base* format. Also featured are numerous prompts and an online dictionary of terms. *Disk Data Base* requires Extended BASIC, a 32K memory expansion unit, and a disk drive and controller. A printer and second disk drive are recommended. Price: \$15.

Asgard Software, P.O. Box 10306, Rockville, MD 20850.

Circle Reader Service Number 230.

Bulletin Board Directory

A new directory of computer bulletin boards, called *Plumbline*, is now available from the publishers of *Plumb*, a newsletter about personal telecommunications. The directory lists over 1,000 bulletin boards available to the public. Each entry includes a brief description of the bulletin board, the type of computer it runs on, and its primary area of interest.

Plumbline is included with a subscription to *Plumb*, \$26.50; or can be purchased separately for \$8.

Plumb, P.O. Box 300, Harrods Creek, KY 40027

Circle Reader Service Number 231.

Pascal Tutorial For Apple

Wiley Software's new *Visible Pascal* uses graphics, word processing, and music to teach the Pascal programming language on Apple II computers. Programs are displayed while they're being created, at a speed controlled by the programmer. The system has more than 80 error messages for pointing out mistakes. Users can create "productions," with animated characters and a soundtrack. The package also includes 56 sample programs that are ready to run.

No prior computing knowledge is needed. *Visible Pascal* runs on Apple II-series computers with at least 64K RAM. A joystick is recommended.

Wiley Professional Software, 605 Third Ave., New York, NY 10158.

Circle Reader Service Number 232.

Boolean Games

Sunburst has introduced *High Wire Logic*, a game for teaching Boolean logic to youngsters in grades 5 through 12. Two sets of colored shapes appear on the screen: one on a high wire and another set that falls to the net below. Using the logical functions AND, OR, AND-AND, OR-OR, and EXCLUSIVE OR, students earn points by writing rules to fit the shapes on the high wire but not the shapes in the net.

High Wire Logic is available for Apple II computers with at least 48K RAM; retail price is \$59.

Sunburst Communications, Inc., 39 Washington Ave., Pleasantville, NY 10570.

Circle Reader Service Number 233.

Life/Time Manager

A new program from Psychometric