YESTERDAY'S NEWS

VOLUME 5 NUMBER 6 Established 2016 JUNE 2020

30 Years Ago...

Historical Information taken from Bill Gaskills TIMELINE

JUNE 1990:

MICROpendium publishes V7N5 consisting of 40 pages.

Jane LaFlamme announces that LaFlamme and Wrigley Wholesale, a Gloucester, Canada firm, have ceased operations due to lacK of sufficient sales volume.

Rumors circulate around the TI Community that interest in the GEnie TI Roundtable is waning.

Asgard Software releases The Animator by Brad Snyder (U01-disK) MSRP $14.95.

Barry Traver releases CONEY GAMES, a disK of games from the first two volumes of The Genial Traveler disKazine.

Keith Bergman, KBCC, releases YALP, Yet Another Lotto Program.

Bill GasKill begins a series of TI-Base tutorials in MICROpendium. YN

INSIDE INFORMATION

12 O'CLOCK PUZZLE

Wesley R. Richardson
Northcoast 99ers, Cleveland, OH

12-O'CLOCK is an Extended BASIC program for puzzle lovers. The objective of 12-O'CLOCK is to orient all 18 clocks to the same 12 O'CLOCK vertical position using the rotate buttons 1+5, 2+6, 3+7, 4+8 and the A+E, B+F, C+G, D+H gears. Rotation is clockwise for the clocK adjacent to the number. Gears are engaged, 'X' for the letter, and dis-engaged 'O'. Pressing the gear letter will change it to 'O'. Number pairs and letter pairs given above work in opposite directions.

Moves are counted for each rotation button used, but not incremented for sequential use of the same button. Although at first, it will seem impossible to solve this puzzle, it is possible to solve.

A green/red indicator is used when the program will/will not accept Key presses. You may hold a Key down for repeated operation.

The program uses all 28 possible SPRITES and all available characters from 32 to 143. The clocK positions are maintained in the C(18) array, and the gear positions in the D(4) array. Two functions are defined, E(Z) and F(Z), since these modular 12 routines are used repeatedly in the calculations.

The program size of 50 sectors causes Extended BASIC to save it in INT/FIX 254 format rather than PROGRAM image. This means that 12-O'CLOCK cannot be run from an autoloading menu, but must be loaded using "OLD DSK1.12-O'CLOCK" and then the command RUN. YN

TI CLASSROOM

TIPS FROM THE TIGERCUB

NUMBER 13

By Jim Peterson

I'm told that someone actually found a practical use for my number scrambling rutine, so here is an expanded version. It will scramble any sequence beginning with 1 and ending with any number less than 256 or any number greater than 256 which is evenly divisible by any number less than 256 and greater than 1, within the limits of computer memory. In Extended Basic with Memory Expansion, the limit is about 10,700; if you reformat it to Basic and run it bare bones, you might get close to 13,000.

```
100 CALL CLEAR :: OPEN #1:"P
IO",OUTPUT
110 INPUT "HIGHEST NUMBER? "
:HN :: IF HN<256 THEN TN=HN
:: XX=1 :: GOTO 150
120 FOR TN=255 TO 2 STEP -1
:: IF HN/TN=INT(HN/TN)THEN 1
40
130 NEXT TN :: PRINT HN;"IS
NOT DIVISIBLE BY":"ANYTHING
LESS THAN 256  - ":"CANNOT U
SE" :: GOTO 110
140 XX=HN/TN
150 DIM M$(50)
160 CALL CLEAR :: FOR J=1 TO
 TN :: M$(1)=M$(1)&CHR$(J)::
 NEXT J :: FOR J=1 TO XX ::
M$(J)=M$(1):: NEXT J :: FOR
J=1 TO HN :: TT=1+INT((J-1)/
255)
170 RANDOMIZE :: X=INT(XX*RN
D+1):: IF LEN(M$(X))=0 THEN
170
180 Y=INT(LEN(M$(X))*RND+1)
190 PRINT #1:ASC(SEG$(M$(X),
Y,1))+TN*(X-1);
200 M$(X)=SEG$(M$(X),1,Y-1)&
SEG$(M$(X),Y+1,LEN(M$(X))):
 NEXT J
```

Here's a little routine you can use to jazz up your title screen or text.

```
100 CALL CLEAR
110 DATA "THIS IS A DEMONSTR
ATION","OF THE","TIGERCUB SO
FTWARE","TWO-WAY PRINT ROUTI
NE"
112 FOR T=1 TO 4
113 READ M$
120 IF LEN(M$)/2=INT(LEN(M$)
/2)THEN 135
130 M$=M$&" "
131 GOTO 140
135 M$=M$&"  "
140 L=LEN(M$)
150 C=16-L/2
160 FOR J=L/2 TO 1 STEP -1
170 CALL HCHAR(10+T*2,C+J,AS
C(SEG$(M$,J,1)))
180 CALL HCHAR(10+T*2,16+L/2
-J,ASC(SEG$(M$,L-J,1)))
190 NEXT J
200 NEXT T
```

Did you ever go through your checkbook 5 times in order to add up your gas bill, then your electric bill, etc.? With this little handy-dandy, you can do it all in one pass.

```
100 CALL CLEAR
110 REM -  ADDER-UPPER by Ji
m Peterson
120 A$="ABCDEFGHIJKLMNOPQRST
UVWXYZ"
130 DIM C$(26),T(26)
140 PRINT "        ADDER-UPP
ER": : :
150 PRINT "WITH THIS PROGRAM
 YOU CAN GO THROUGH YOUR CHE
CKBOOK, OR ANYTHING ELSE, AN
D ADD UP   AMOUNTS IN SEVERA
L CATE-"
160 PRINT "GORIES ALL AT ONE
 TIME.": :
170 PRINT " FIRST, LIST THE
CATEGORIES":"YOU WANT TO ADD
 UP.":" TYPE 'END' WHEN FINI
SHED.": :
180 PRINT " NEXT, ENTER THE
CATEGORY":"CODE AND AMOUNT F
OR EACH":"BILL."
190 PRINT : :"WHEN YOU HAVE
ENTERED ALL":"THE BILLS, TYP
E =": :
200 N=N+1
210 PRINT "CATEGORY #";N
220 INPUT "           ":C$(N
)
230 IF C$(N)="END" THEN 340
240 W$=SEG$(C$(N),1,1)
250 IF POS(A$,W$,1)<>0 THEN
290
260 PRINT :"CODE LETTER ";W$
;" ALREADY  USED - PICK A CO
DE LETTER."
270 INPUT W$
280 GOTO 250
290 X=POS(A$,W$,1)
300 A$=SEG$(A$,1,X-1)&SEG$(A
$,X+1,LEN(A$))
310 X$=X$&W$
320 PRINT :"CODE LETTER FOR
";C$(N);" WILL BE ";W$: :
330 GOTO 200
340 C$(N)=""
350 N=N-1
360 X$=X$&"="
370 IF FLAG=1 THEN 420
380 FLAG=1
390 PRINT : :"READY TO START
 - ": : :
400 PRINT "WHEN FINISHED, TY
PE =": :
410 INPUT "DO YOU WANT TO VE
RIFY EACH  INPUT? ":V$
420 PRINT :"CODE (";X$;")"
430 INPUT Q$
440 IF Q$="=" THEN 600
450 IF POS(X$,Q$,1)<>0 THEN
510
460 PRINT "THAT IS NOT ONE O
F THE CODES": :
470 INPUT "IS IT A NEW CATEG
ORY?(Y/N) ":Q$
480 IF SEG$(Q$,1,1)<>"Y" THE
N 420
490 X$=SEG$(X$,1,LEN(X$)-1)
500 GOTO 200
510 Y=POS(X$,Q$,1)
520 INPUT "AMOUNT ?":A
530 IF SEG$(V$,1,1)="N" THEN
 580
540 PRINT :C$(Y);A: :
550 INPUT "CORRECT? (Y/N)":L
$
560 IF SEG$(L$,1,1)="Y" THEN
 580
570 IF SEG$(L$,1,1)="N" THEN
 420 ELSE 550
580 T(Y)=T(Y)+A
590 GOTO 420
600 FOR J=1 TO N
610 PRINT :C$(J);T(J)
620 TT=TT+T(J)
630 NEXT J
640 PRINT :"GRAND TOTAL OF A
LL IS";TT
650 END
```

And, did you ever wish that you could make numbers smaller, so that you could squeeze more of them onto a chart or graph? The problem is that resolution is so poor, at least on my TV screen, but maybe you'll find a use for this.

```
100 REM - NUMBER SCRUNCHER -
 programmed by Jim Peterson
110 CALL SCREEN(5)
120 FOR S=2 TO 14
130 CALL COLOR(S,15,1)
140 NEXT S
150 CALL CLEAR
160 RANDOMIZE
170 DATA 75557,22222,25127,6
1216,55571,74616,74757,71222
,75257,75711
180 FOR J=0 TO 9
190 READ C$
200 CH$(J)="00"&C$
210 NEXT J
220 CH=91
230 INPUT "NUMBER? ":RX
240 N$=STR$(RX)
250 IF LEN(N$)/2=INT(LEN(N$)
/2)THEN 270
260 N$="0"&N$
270 FOR J=1 TO LEN(N$)STEP 2
280 P1=VAL(SEG$(N$,J,1))
290 P2=VAL(SEG$(N$,J+1,1))
300 FOR T=1 TO 7
310 Z$=Z$&SEG$(CH$(P1),T,1)&
```

```
SEG$(CH$(P2),T,1)
320 NEXT T
330 CALL CHAR(CH,Z$)
340 Z$=""
350 P$=P$&CHR$(CH)
360 CH=CH+1
370 NEXT J
380 PRINT N$;" ";P$
390 P$=""
400 N$=""eterson
410 GOTO 230

Almost OUT OF MEMORY.
Happy hacKin' Jim Peterson
```

FLY SNUFFER HCM VOL. 4 NO. 1 by Judy Sanoian

Anyone old enough to remember the game Pong Knows that fancy graphics and complex scenarios are not necessary for an entertaining video game. Pong, a precursor of our modern cursor entertainment, featured a moving dot which players batted backK and forth between simple vertical lines. Fly Snuffer, an offering from Futura Software, is a game formed in the Pong mold, although its graphics are not nearly so primitive. Your main goal in Fly Snuffer is to exterminate a horde of flies with a deadly aerosol spray. It is really just another variation on the hit-dot-with-stick theme, and yet, liKe Pong, it is surprisingly engrossing.

The opening screen of Fly Snuffer displays a room with two windows in the center. One by one, the flies enter from all sides of the screen. Using your joystick, you stalK the metamorphosed maggots, pressing your fire button to release the toxic gas. Kill ten flies and you win a round. But flies are not the only insects to invade your abode. There are also large orange cocKroaches that scurry across the floor, and menacing bumble bees. To Kill the roach, you must smash him with your can. The bee succumbs to your poisonous spray, but don't be siow in Killing him. He is one of the long-awaited Killer bees who have finally arrived from South America. If you don't vaporize him, you will fall prey to his deadly stinger.

The flies behave much liKe their real-life counterparts. Some are fast, some are slow, some buzz around the window trying to get out. Even though the little pests are quite harmless, you cannot taKe a live-and-let-live attitude. If you sit around marveling at this tiny member of God's creation, you will lose the game. you must gas ten of them before the time runs out. And don't go around spraying indiscriminately. Even if you don't care about our deteriorating ozone layer, Keep in mind that you lose the game if you run out of insect spray.

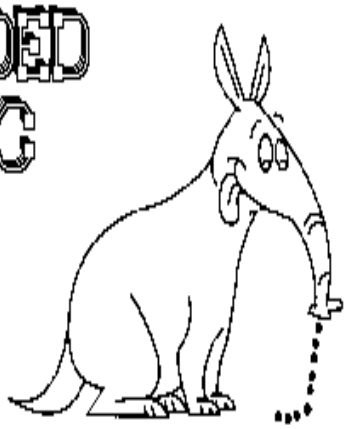
Supersonic Cyber-flies

Fly Snuffer's graphics and animation are quite nicely done. The flies are fat, blacK and glossy with flapping wings. The cocKroaches and bees also sport appropriate features -moving legs and menacing stingers- and the spray can emits a realistic puff of DDT. Fly Snuffer's sound effects are also well done: The flit of the flies wings, the pfft of the aerosol can, and the crunch of the squashed roach are all quite realistic. At first I was disappointed that there was no buzzing fly noise, but on second thought realized that a constant cyber-fly buzz would be as annoying as the real thing.

The end of my first game of Fly Snuffer left me with a cramped right hand - always the sign of an absorbing joystick game. But even though it was exciting, it was also frustrating. Maybe I am exceptionally slow on the spray button, but I could not - try as I might - get past level three (of six levels). The flies on levels four and beyond are simply too fast to catch. Also the roach is too quicK to smash unless you happen to be right in its vicinity when it appears. Perhaps staying low on the screen (where the roach crawls) should be part of your game strategy. All I Know is that once I got to level three, I found myself darting in all directions, futilely spraying at the speeding flies while the roaches came and went as they pleased. And after 30 seconds in a room full of dive-bombing flies, I felt more inclined to wave the white flag than maKe use of the BlacK one.

The bottom line in any game review (or at least the bottom paragraph) is whether or not the game is any fun. In the case of Fly Snuffer, I must say it is. Playing to win is challenging and yet possible, up to a point. and perhaps with practice you can even reach the highest level The game loads easily and performs well if you follow its excellent documentation. (The instructions are among the most clear and complete I have ever seen,) Fly Snuffer is neither a visual masterpiece nor an exciting mental challenge. LiKe Pong, it's a good game to play when you want to give your cerebral side a rest. So if simple joystick action is your cup of tea, I'd recommend you give Fly Snuffer a try. YN

EXTENDED BASIC



AARDVARK

Programmed by
Patrick Pelletier

99'er
June 1983
Vol. 2, No. 8

An ant's work is never done. What with tunneling all morning, and gathering food all afternoon - then tucking the juicy morsels away to be eaten later - it's enough to exhaust anyone. And when night comes to your African grassland home do you think you can finally relax and have a bite to eat? No! Just as you are ready to mosey up to the storage cell to enjoy your food, a large blue figure casts its shadow over your anthill. Cautiously, you begin to make your way through the labyrinthine tunnels toward your food. Although you cannot see the aardvark lurking at your hill's entrance, your ant sensors can detect the pounding of his huge tail. You can almost feel it as he strikes it on the ground each time he extends his long sticky tongue in search of you and your fellow ants. Quick! Hide under that mound of sand over there! Do it now or he'll give you an aardvarkish lick and have you for dinner! Well, you got by him that time. Just a few more passages and you'll have your own meal. Can you out-turn and out-twist your pursuer's devious extensile tongue? Oh no! There's that tongue again, poking around the corner and sliding right over your head ... a close call. Now to go through that last passageway and finally ... food! Ha! You've made it! But what about the rest of the hungry ants waiting their turn? The aardvark is smart and fast ...

The name of the game
Eat before being eaten - that is your goal in this game of survival. Using a joystick, you guide six hungry ants to the food they've set aside. The ravenous aardvark, guided by the computer, will eat the six ants if they don't move quickly and carefully enough. Each time an ant moves, it has about 4 seconds before the eager aardvark comes after him, extending his long tongue down into the maze. If the ant successfully outwits the pursuing anteater and gets to the food, he gobbles up the food to a reward tone. If, on the other hand, the aardvark succeeds in making a meal of the ant, he celebrates with graphically obvious delight as he munches his meal.

This Extended Basic version of hide-and-seek is fun for even the youngest joystick jockey. Its rules and goals are simple yet the random movements of the aardvark's menacing tongue make it challenging for older players as well. With six ants to move through the anthill, family members can play together with each player maneuvering one of the ants toward the food, working as a team to outwit the aardvark.

The score and more
The game ends either when the aardvark is full (having eaten all the ants), or when the ants have eaten all their food. The screen will display the scores for the aardvark and the ants separately, so you can determine which species has survived and which has been eaten up (ants) or starved to death (aardvark).

Aardvark has many endearing features. The screen is entertaining to look at whether you are playing or on the sidelines patiently waiting your turn. The graphics were programmed artistically, with a captivating sense of humor.

This game could also serve many purposes in a classroom. A teacher might use it as a simple reward for work well done, or as a vocabulary enriching tool in a foreign language class. No, you won't learn to speak like an aardvark, but you may learn to order ants in a French restaurant! This excellent program, sent to us by a *twelve-year-old* Montreal programmer, displays the instructions and scores in both English and French! - A bilingual aardvark in pursuit of bilingual ants. C'est si bon!

YN

CONSOLE BASIC

GUITAR TUNER

99'er - June 1983
Vol. 2, No. 8
By James R. Noel

Merge BXB to run program in Extended Basic

I was one of those hesitant newcomers to computing, someone who never really liked math and tried his best to stay away from anything involving more than a very few numbers. But about seven months ago my wife and I finally agreed that I should learn something about these machines, and the TI-99/4A moved in with us. My job involved operating a computer monitoring system, and I wanted to know more than just how to press the right button at the right time. So I began pressing buttons and learning TI Basic.

This program started out as an experiment. I was wondering if my limited programming knowledge was sufficient to let me write a useful program. GOSUB statements baffled me, so

I tried to incorporate them into the program to see how they work. I thought graphics added a lot of interest to the programs I'd seen, so I figured I'd fit some in. Music has always interested me; I've played the guitar since my teens. So it all added up to a program for tuning the guitar. With a few minor changes, this program can be modified for tuning other string instruments as well. In fact, if you are a more advanced programmer, you can probably adapt these humble beginnings into a program to tune your piano. Once your program is created with the pitches for the notes from middle C to the C above it stored in memory, you can use your own sense of relative pitch to tune the other octaves. Somebody out there could probably even suggest some way to put relative pitch right into the program. For me, however, creating a program to tune my guitar was accomplishment enough.

I designed the guitar graphics to spruce up the title page. A character definition program from a friend was particularly helpful here. A hint from an earlier issue of 99'er showed me how to combine FOR-NEXT and PRINT statements to scroll the title.

The menu is pretty straightforward. It lists the six guitar strings by number and allows you to choose the one you want to hear. After the choice is entered a graphic of the guitar neck appears. You then hear the string tone repeated 20 times (the string you are hearing is highlighted to set it off from the remaining strings). This gives you time to adjust the tension of your string to the computer tone. Do this six times and PRESTO! - you've tuned your guitar.

Graphic harmony

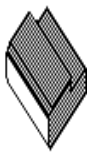
I worked up a guitar neck graphic and used a GOSUB statement to provide the graphic each time you enter a new string number. The TI manual listed the string frequencies, so that was easy. Another new statement, ON X GOTO, got me from the menu to the particular string on the guitar neck graphic. Color helped me highlight the desired string. This string appeared white on the portable black and white set that I use for programming. Things moved along well until I hooked it all up to our color set to "watch it fly."

It flew like a rock. It seemed that I had defined some guitar string characters with numbers that fell within the sets for the regular letters. This meant that when the program sent you back to the menu, several of the letters were blue while the rest appeared black. Don't get me wrong - I have nothing against blue, but it looks better if all your letters are the same color. Substituting numbers within different sets and adding a FOR-NEXT statement/CALL COLOR combination to return all the letters to black solved this problem.

The same problem appeared with the guitar string colors. Each string had to be defined in a different set so that only one would appear highlighted. A color TV made me feel more "color creative," so I changed the screen color to cyan. Finally , two FOR-NEXT statements gave me the sound of the string with a delay and 20 repetitions.

This isn't a program with a lot of flash. It doesn't dry the dishes while you wash them. In fact, someone will undoubtedly change a few things here and there to clean it up. But you see, that's all right. I wrote the program to prove to myself that I could do it. Remember, I'm the guy who hates anything involving numbers. The TI-99/4A is a remarkable machine that will provide hours of entertainment to anyone willing to invest the time to get to know it. Now, if I can just save enough money, maybe I'll learn to program in Extended Basic next! YN

Programmed by
Eric Lafortune



EXCALIBUR

Requirements

- TI-99/4A - Mini Memory module - Joystick.

Loading the program

Select TI BASIC, load the program and RUN it. It will install Excalibur in the Mini Memory module and automatically return to the master title screen. The game and its high scores will remain available as long as the module's contents aren't erased.

Running the program

Select 3.Mini Memory, 2.Run and press <enter>. The title screen with the high scores should appear now. Press the fire-button of joystick #1 or #2 to play.. The object of the game is to get the Knight through all 9 rooms as fast as possible. In each room he has to be guided from the entrance to the exit, which is marked by a small arrow. The Knight is controlled using the joystick:

- left/right: walk
- down: climb down
- up: small jump/climb up
- fire: normal jump
- up+fire: high jump

Each screen must be completed before the bonus time reaches 0. If the Knight touches any of the appearing creatures he will die. They can be destroyed by hitting them with the sword, but this doesn't earn you any points.

Note that you only have a single life to complete your mission. If you get through the last screen you will get a bonus of 1000 points and you will start again at the first screen. The game speed will increase as you advance through the game.

If you breaK a high score you can enter your name, up to 6 characters long and using letters only. Any other Key will act as backspace Key.

Conditions

This program may be distributed freely. It is provided "as is" and comes with absolutely no warranty.

More information

The program has been written using the Mini Memory module, the line-by-line assembler and a cassette recorder. Although not as elaborate as many other games it fits entirely inside the 4K module RAM, which is a feat on its own. If you have any comments, please contact the author at the address below. Have fun!

Eric Lafortune

GRIDDER

Requirements

- TI-99/4A - Mini Memory module - JoysticK.

Loading the program

Select TI BASIC, load the program and RUN it. It will install Gridder in the Mini Memory module and automatically return to the master title screen. The game and its high scores will remain available as long as the module's contents aren't erased.

Running the program

Select 3.Mini Memory, 2.Run and press <enter>. The title screen with the high scores should appear now. You can select the level by moving joystick #1 or #2 up or down (1=very slow, 9=very fast), and a screen by moving it left or right (A=fairly simple, X=most difficult). Then press the fire-button to start playing.

You control the white face that is constantly moving across the grid with the joystick. The object is to color all the lines of the grid by moving over them. If a square is surrounded by colored lines it will turn blue and you receive points in accordance with the level you're playing on. All the squares have to be colored within the given bonus time.

To make life more difficult you are chased by two other faces. You loose one of your five lives every time one of them catches you. By pressing <fire> briefly you can place a star as a temporary obstacle. This will cost you some points however. From time to time one face will leave behind an apple. Eating it will provide you with an extra life. If you complete a screen the bonus time will be added to your score.

If you breaK a high score you can enter your name, up to 6 characters long and using letters only. Any other Key will act as backspace Key.

Conditions

This program may be distributed freely. It is provided "as is" and comes with absolutely no warranty.

More information

The program has been written using the Mini Memory module, the line-by-line assembler and a cassette recorder. Although not as elaborate as many other games it fits entirely inside the 4K module RAM, which is a feat on its own. If you have any comments, please contact the author at the address below. Have fun!

Eric Lafortune YN

THE LOST TREASURE OF THE AZTEC

REPORT CARD	
PERFORMANCE	A
EASE OF USE	A
DOCUMENTATION	A
VALUE	A
FINAL GRADE	A

MICROPENDIUM June 1984
Volume 1, Number 5
By Christopher Bobbitt

The adventure program Lost Treasure of the Aztec is a text-only game designed in the great adventuring tradition of the Scott Adams and Infocom adventures. Built around a rather tried and true theme in adventuring, the search for treasure, this program is flawlessly executed. Its complexity will stupefy even the most expert adventurers for hours on end. The sheer number of commands and objects in this adventure put most other BASIC adventures to shame. It was written by Bob Ulrich.

Performance: To those not familiar with adventures, the adventure is a game where the player must solve a riddle-like problem which has many interacting elements. These adventures often take the form of a quest for something or the exploration of some place out of the ordinary. A good adventure is much like a good mystery novel, the more complex and descriptive, the better. This adventure is very complex, and contains a fair amount of description.

The scenario for this adventure is very intriguing. According to the instructions, the whole adventure begins in 1521. At this time the Spanish general Cortez was marching toward the Aztec capital of Tenochtitlan with his army of thousands. Monteczuma, fearing that the Spanish would take all the gold objects decorating Tenochtitlan, sent out an order which stated that all the treasures were to be hidden from the advancing Spaniards. After much searching the ideal hiding place was found, the treasure was then hidden, and a stone map showing the treasure's location was placed in the city. Consequently, when the Spanish finally entered Tenochtitlan, after slaughtering the less advanced Aztec army, they didn't find neady the amount of treasure that they expected. Cortez searched the city and couuntryside in vain for the treasure. Many good men were lost in the search parties that never returned. In frustration, Cortez withdrew from the Aztec Empire with most of the Aztec people dead or enslaved. The treasure remained lost among the mountains and valleys of the area that was the Aztec Empire.

The player begins the game at a point more than 450 years after the tragedy. As the adventure unfolds, the player finds that he or she is on a deserted ocean beach. The ship that was to have brought the player to the land of the Aztecs has capsized in a storm. The player escapes from the wreck with a book of matches as his sole possession. This is the situation when the adventure begins. The game description ends with this warning: "Many adventurers have gone before you, but none have ever found the treasure. Some have never returned." More truth than poetry.

Ease of Use: The program is designed to work out of one disk drive. Throughout the game the disk drive constantly turns on and off as data are entered into the main program. The amount of data is immense; the program and the data take up two-thirds of the disk that they reside on. The gargantuan effort expended in writing this program probably isn't measured in programming hours, or even days, but weeks.

The game automatically loads itself when Extended BASIC is selected. The first program loaded contains the title screen and a good portion of the instructions and background material found in the instruction manual. After the instructions have been viewed, the program then gives the option of beginning the adventure over or continuing a previously saved game. Any person who plays adventure games realizes that a save game feature is a must. This is because most of the good adventures take days, or even weeks, to solve. The disk which contains the adventure has enough space left to store many game files, since each one takes only four sectors out of the more than 100 available. After the option desired has been chosen, the computer then goes on to load the game and the data. If you are loading a saved game, this process can take as long as two minutes. Starting the game from scratch doesn't take as long, only about a minute. When the adventure has loaded, it automatically runs itself and after a few seconds presents the adventurer's location, what is visible and a query for the player's command. At this point he must type in either a single command, such as N to go north, or a two-word command such as DIG HOLE or CLIMB TREE. The two-word commands are usually more descriptive of their function. Because it uses menus and prompts, this program can be easily used without even looking at the instruction manual. Anyone familiar with the Scott Adams adventures will find this program very similar in its style, the type of commands it uses and the way it presents data to the player.

Documentation: The documentation, though sparse, is completely adequate for this adventure. The eight-page manual gives the scenario of the adventure, the loading instructions and a description of the types of commands accepted. The manual also explains the types of errors which may be obtained from improperly loading saved game files, and those that may result from negligence by the user. The only thing missing from this manual which often appears in the instructions for other adventure games is a section containing hints on solving the adventure. With a game of this difficulty level, hints may be crucial to preserving one's sanity.

Value: The value of an adventure is hard to determine. Once an adventure has been solved it is useless to the adventure player. Unlike an arcade game which is slightly different every time, the quest in an adventure never changes. This is the main reason why adventures are so hard to solve in the first place. If the value of an adventure is a reflection of its difficulty, then this is a program with great value. In comparing its difficulty to other adventure games, I believe it is roughly as difficult as the Savage Island adventures in the Scott Adams Adventure Series. At a price of $16.95 this game is a great value for the money in any case. While comparable to more expensive assembly language adventure games, it is priced to compete with adventures in BASIC.

The author responds

The reviewer mentions that the time to load a saved game is two minutes and starting from scratch about one minute. In point of fact, the times are the same because they use files of the same format. The actual load time is about 1 3/4 minutes.

The question of hints in the manual is something I worried over for quite sometime. What I decided to do was to leave the hints out. I have been giving out hints to people who write and ask, provided they pay the postage. The reason I did this is that I didn't want to give away the store before a person could get to the candy counter.
Bob Ulrich
Mind Games Software

Yesterday's News Information

Yesterday's News is a labor of love offered as a source of pleasure & information for users of the TI-99/4A and Myarc 9640 computers.

TI-99/4A HARDWARE

TI99/4A COMPUTER
MODIFIED PEB
WHT SCSI AND SCSI2SD
MYARC DS0D FDC
MYARC 512K MEMORY
HORIZON 1.5 MEG HRD
TI RS232
CORCOMP TRIPLE TECH
1 360K 5.25 DRIVE
1 360K 3.50 DRIVE
1 720K 5.25 DRIVE
1 720K 3.50 DRIVE

TI-99/4A SOFTWARE

PAGEPRO 99
PAGEPRO COMPOSER
PAGEPRO FX
PAGEPRO HEADLINER
PAGEPRO GOFER
PAGEPRO FLIPPER
PAGEPRO ROTATION
PIXPRO
PICASSO PUBLISHER
BIG TYPE
TI ARTIST PLUS
GIF MANIA

PC HARDWARE

COMPAQ ARMADA 7800
COMPAQ ARMADASTATION
SAMSUNG SYNCMASTER

PC SOFTWARE

DEAD WINDOWS 98SE
FILECAP
PRN2PBNS
IRFANVIEW
ADOBE DISTILLER
ADOBE ACROBAT

Yesterday's News is composed entirely using a TI-99/4A computer system. It consists of 11 PagePro pages which are "printed" via RS232 to PC to be published as a PDF file.