

The National



Exchange

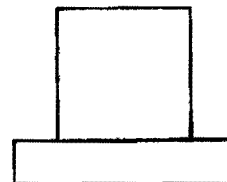
Volume 2 Number 8

FORWARD 100!

April 1984

Breaking Up Is Hard To Do

by
Cathy Frank



Who among us has not seen "spaghetti" programming in Logo? Sometimes it seems as if students are engaged in a contest of sorts to see who can write the longest continuous procedure!

I would like to concur with Sharon Burrowes' observations of Logo hacker habits in the January 1984 NLX. In particular, the tendency to write long procedures is distressing, especially when no thought is given to breaking them into parts. I feel that the concept of breaking problems up into smaller steps is one of the more important concepts that Logo can teach.

Although this concept may be obvious to us as teachers, perhaps we are not taking the time to introduce it to our students carefully enough.

I have been teaching Logo to 4th, 5th, and 6th graders for two years, and am still frustrated with their inability to grasp this concept sooner. The younger the student, the more difficult it seems to be.

As a result, I have been experimenting with various techniques as I first introduce children to Logo. Some of them are proving successful, and I would like to share them with you.

SEE THE PIECES

In order to write modular procedures, one has to see a picture in modules. That is, one must be able to see both the whole picture and its constituent parts. For example, in the picture at the top of the next column, can you see both the tophat and the rectangle and square?

I draw similar pictures on the board and ask the students to identify the objects and then list all the separate shapes they can find in them. Once they get the idea, it becomes almost like a contest to see who can list the most shapes! Followup worksheet drawings are eagerly pounced upon. Students quickly progress to the point where they begin designing pictures for each other.

MAKE YOUR OWN

Next, I give the students groups of geometric shapes (circles, triangles, rectangles, squares, etc.) cut out of heavy construction paper, and challenge them to assemble composite shapes. Usually, I will start out with a "vocabulary" of one or two shapes of various sizes, and introduce more shapes later.

My students assemble the shapes in many different combinations, creating beautiful and surprising results. Often, they will want to paste the shapes together on white paper, to preserve their creations.

LOGO PIECES

Next, I set up class projects, in which students write procedures to draw the outlines of the various shapes used in their work. Since I emphasize this concept before I introduce the students to variables, we usually put an extra letter in the title as a clue to the size. For example, SBOX might be a small square, while BBOX would be a big one.

Students place all shape project procedures in the computer and save them on disks for everyone. Now they are ready to start writing procedures which use these shapes to produce pictures on the screen.

Breaking Up continued

This technique is appealing to me because it lets me control the complexity of the projects by the number of different shapes I give the children. Although a little of the spontaneity may have been taken out of the programming aspect of the project, it is still present in the original design. The enthusiasm and excitement of the children mounts as they begin to understand the power and ease of modular programming.

Another technique I have used with older students is to suggest simple projects made up of the original shapes. These might include a wagon, a flower, a person, a house, etc. Once these have been completed and saved, the students then use them to create a more complex picture of a yard or landscape.

MODULAR SQUAD

The ultimate goal in all of this is to help students discover the effectiveness and ease of treating a complex problem in a modular fashion. After all, even the most complex picture is composed of different parts.

Once my students first experience the "bottom up" assembly of a complex picture from simple shapes, they seem to be able to grasp the "top down" concept more readily.

They don't turn away from challenging projects. Instead, they eagerly search for ways to divide them into smaller parts, and away they go. Hi yo Logo!

Cathy Frank, a parent volunteer Logo teacher for 4th, 5th, and 6th grade children in Burlington, VT, is preparing for full time work as an educational computer consultant.

Logo Teachers Wanted

Computer Summers, Inc., one of the first summer computer camp systems to focus on Logo, will operate day camps in Los Angeles, Houston, Tampa, and in Westchester County, New York, this summer, in addition to resident camps in North Carolina, Colorado, Massachusetts, and upstate New York. If you would be interested in teaching Logo at such a camp for a reasonable salary, write David Sachs, Computer Summers, Inc., PO Box 436, South Salem, NY, 10590, or call (914) 763-8829.

From The Editor

One of the more popular events in the Olympic Games is the high jump. It is always spectacular to see an athlete leap over a bar suspended far overhead. For years, high jumpers used a technique called the western roll, in which the athletes "rolled" over the bar face down.

In 1968, a young man named Dick Fosbury won the gold medal for the high jump in the Olympics in Mexico City. He astounded the world by soaring to a new Olympic record of 7 feet 4.25 inches with a revolutionary style, now called the "Fosbury Flop."

This young man ran parallel to the bar and suddenly leaped up at it backwards, sailing over it upside down! He had sought and found a better way.

One of Papert's criticisms of computer use in education is linked to resistance to change. He speaks of the so-called QWERTY phenomenon, based on the historical development of the typewriter (and computer) keyboard layout. Long after the reason for its design had vanished, the keyboard has remained unchanged. Several more efficient designs have been suggested, but, as of now, none have caught on. We've always done it this way.

Perhaps it is time to take a look around our Logo circles to see if similar signs of stagnation are setting in. Is this happening with our Logo teaching? Is it happening with Logo itself?

Many have found certain teaching methods and techniques which seem to be effective. Are we still trying to refine them? Do we continue to search for still other ways?

Let us not think that we have found the best way yet. Rather, let us always enjoy the search for it.

After all, if we ask our students to take risks in their Logo work, we should be willing to take risks in our Logo teaching, shouldn't we?

FD 100!



TIPPS for TEACHERS

by
Steve Tipps

Unsquare Square: The Story of Scrunch

A Logo student sits down at a computer to draw a square. But it turns out not square, but rectangular! Circles are ovals! What computer madness is this?

This squeeze of shapes results from the monitor being set to a different ratio of vertical picture elements (pixels) to horizontal pixels than the computer expects. Normally, Logo expects for the monitor to be set at about 8 steps vertical for 10 steps horizontal and automatically adjusts the commands to this ratio.

Having nonsquare squares and noncircular circles is disconcerting, but easily fixable. If the square or circle is flatter than desired, try resetting the ratio to a different value. For example, to set it to a ratio of 1 step vertical to 1 horizontal, type

```
.ASPECT 1      (MIT versions)
SETSCRUNCH 1   (Apple LCSI version)
.SETSCRUNCH 1  (IBM LCSI version)
.SETSCR 1      (Atari LCSI version)
```

The monitor may also have a switch on the back which can be reset to correct the ratio. But the command which corrects monitor deflection can also stimulate new turtle explorations.

PROPORTIONAL MICROWORLDS

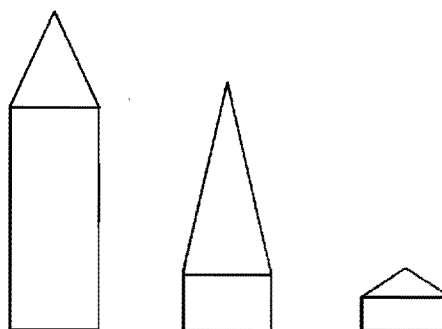
The scrunch commands provide experience with ratio and fractions. Except for Atari Logo, they can be set at levels from hundredths to hundreds, although .1 to 6 seems to be the most interesting range for initial explorations.

The effect of setting the ratio is to change the number of vertical moves to 1 horizontal move. The CIRCLE drawn at times as tall as wide; the SQUARE at SETSCRUNCH .2 is really a rectangle which is 1/5 as long on one side as it is on the other.

One way to illustrate this off the computer is to draw a shape on a piece of paper with wide lines. Fold the paper into narrow strips like a paper fan. The shape on the paper can now be scrunched up or expanded by pushing or pulling the edges of the paper like an accordion. Try this fan folding with different shapes and pictures to see what scrunch will do.

NEW SCRUNCH FOR OLD FRIENDS

Some interesting and unexpected things happen when old familiar drawings are done with different aspects. The classic Logo house becomes either a tall thin abode or a squat narrow ranch-style one. Rotating a regular figure about HOME creates interesting petals and overlaps.



Equilateral and isosceles triangles can be used for beginning explorations.

```
SETSCRUNCH 2 TRIANGLE 30
SETSCRUNCH 3 REPEAT 4 [TRIANGLE 25 RT 90]
SETSCRUNCH .5 REPEAT 9 [ FD 10 TRIANGLE
20 FD 10 RT 40 ]
```



Rotating the triangle changes how each side is drawn because the ratio changes the effective angle as well as the length. This quirk can be used to create triangles anywhere on the screen. Changing the scrunch for triangles is useful because the horizontal base can be fixed while vertical sides change.

Since the horizontal dimension is always the standard, the base of the triangle can be set with a variable and the height of the triangle set with the scrunch. A family of triangles can be made up with SETSCRUNCH or set the scrunch back to .8 before leaving the procedure so that the next shape is not likewise distorted.

Tips continued

```

TO TRIANGLE.HT3 :SIZE (Apple LCSI version)
  RT 30
  SETSCRUNCH 3
  TRIANGLE :SIZE
  SETSCRUNCH .8
  LT 30
END

```

```

TO TRIANGLE.HTHALF :SIZE (MIT version)
  RT 30
  .ASPECT .5
  REPEAT 3 [FD :SIZE RT 120]
  .ASPECT .8
  LT 30
END

```

Scrunch can even be set through an input variable.

```

TO SCRUNCH.SQUARE :HEIGHT
  SETSCRUNCH :HEIGHT
  SQUARE 50
  SETSCRUNCH .8
END

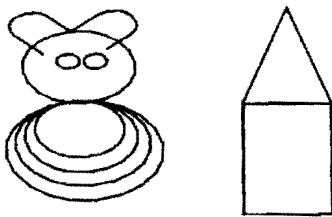
```

```

TO SCRUNCH.SQUARE :HEIGHT
  .ASPECT :HEIGHT
  SQUARE 50
  .ASPECT .8
END

```

Using scrunched squares and triangles, different combinations of houses can be built for three bears made of scrunched circles.

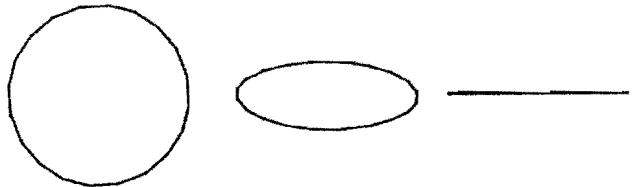


SCRUNCH, SQUEEZE, AND FLIP

Scrunch can be used to squeeze pictures or expand them. This can be handy in particular situations when you want to slightly alter or distort a shape.

One such use for scrunch was found by two students in our Introduction to Instructional Computing class at UVA. Cindy Rutledge and Jane Blanchard wanted to draw a melting snowman. They had tried redrawing and repositioning the snowman in a recursive procedure, but it kept wrapping off the bottom of the screen. Even after that problem was solved, they were unhappy with the results.

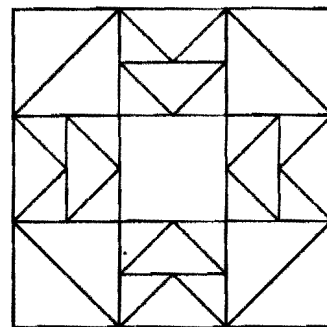
Scrunch was a solution to their problem. The snowman got flatter and flatter until it "melted" into a flat line.



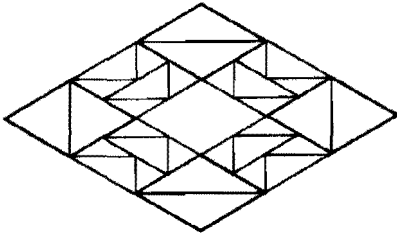
They then used scrunch in another procedure to draw a balloon which sailed into the sky. As the balloon went up, it got smaller and flatter. The perspective was tremendous. They were able to "animate" the balloon with several READPICTs using MIT Logo.

Scrunch can also be used to change the shape of quilt block designs. Working out quilt blocks with Logo is a great project in the first place. (See "Logo Quilting Party," by Tom Lough, NLX December 1982.) Scrunch allows you to change square blocks into diamond shaped blocks without calculating all the angles.

You can begin with a simple design based on squares and diagonals. After this is worked out to your satisfaction, simply turn the block on end, change the scrunch and draw it as a diamond. Becky Poplin found that a scrunch of 1.4 made a good diamond shape from quilt blocks. Here are samples of the DOUBLE-T block which she transformed into a DOUBLE-T DIAMOND.



Tipps continued



A great resource for ideas on changing square blocks to diamonds is Diamond Patchwork, by Jeffrey Gutcheon, \$9.95, from Alchemy Press, 510 Broadway, New York, NY 10012.

Another scrunch exploration which can be done with most Logo versions (one notable exception is the IBM LCSI version) is to use negative numbers as input. In a recursive routine with scrunch, this produces an effect like a mirror, resulting in a symmetrical design as the scrunch falls through 0 to the negative numbers.

```
TO SCRUNCH.DIAMOND :HEIGHT (MIT version)
  .ASPECT :HEIGHT
  LT 45
  SQUARE 50
  RT 45
  SCRUNCH.DIAMOND :HEIGHT - .2
END
```

This procedure needs a test to stop, of course.

TRUTH and SCRUNCH

While scrunch is great fun and should provide a new way of exploring turtle drawing, it has one major problem — truthfulness. The computer thinks that the angles of the square are still 90 and the sides are all the same, even if the picture is a diamond. What should a teacher should do about this dilemma?

The sophistication of the students suggests different approaches. Younger students explore the shapes as they are changed without any great concern. The distortions are accepted as part of the intrinsic geometry of the turtle's world.

When descriptive geometry is introduced, distortion of the angles needs consideration. Use protractors and draw equilateral triangles on grid paper with scales of 3 to 1 or distorted.

Scrunch plays a trick on the turtle which makes it think it is still drawing equilateral triangles. If you add a PRINT HEADING to an equilateral triangle

```
REPEAT 3 [ FD :SIZE PRINT HEADING RT 120]
```

the headings remain constant regardless of the scrunch. The discrepancy between the turtle's angles and the paper angles raise new questions about the rules of the turtle world. When is a square not a square? The answer is a clue to another aspect of the turtle world.

Support from Enter Computer, Inc., 6867 Nancy Ridge Drive, San Diego, CA 92121, in producing the SWEET P plotter drawings is gratefully acknowledged.

Steve Tipps is a professor in the University of Virginia's School of Education, and presents Logo workshops for school systems throughout the United States.

Logo for the Sony

by Hillel Weintraub

Logo versions are popping up in Japan. One of the more interesting is the 8-bit version of DR Logo supplied with the Sony SMC-777 computer. When the systems disk is booted, the user can choose either BASIC or Logo. As far as I know, this is the first Logo version supplied with a computer by a manufacturer.

There are several nice features in this Logo system. If your procedure has a bug, typing ED will take you to the editor, with the cursor at the bug. The graphics and text pages overlap, allowing for full mixtures. The color of the text can be changed and flashed. You are given a choice of 4000 (four thousand) background colors!

There are no LAST and BUTLAST commands, and the graphics are fairly slow, about half the speed of comparable Apple versions.

The Sony SMC-777 computer with its particular DR Logo version is planned for release in the US in late 1984.

Hillel Weintraub is a teacher at Doshisha International High School in Kyoto-fu, Japan, and is the president of the Society for Microcomputing in Life and Education (SMILE).

Apple Sprite Logo

A review by
Griff Wigley

For well over a year, Logo Computer Systems Inc. (LCSI) has demonstrated versions of Apple Sprite Logo at various conferences, teasing the TI Logo diehards among us who yearned for a Logo that combined the best of both the Apple and TI versions. At last, it has been released. Despite some limitations, Apple Sprite Logo easily surpasses the Apple / TI combination.

Apple Sprite Logo is both a hardware and a software package. That is, it comes with both a plug in card and a language disk. The card goes into slot 2 of the Apple computer, and the monitor cable attaches directly to it. Once the card is in place, you can boot the disk to begin.

HERE COME THE SPRITES

Apple Sprite Logo comes with 30 sprites (dynamic turtles). They can take on whatever shape and color you like, and be given speed and heading that are retained while the computer attends to other matters.

In addition, they can draw, detect collisions, and leave copies of their shapes on the screen. The latter delightful feature is invoked by the command STAMP. One of my favorite STAMP projects was to write a procedure which printed a path of sprite shapes across the screen. It made a great addition to my version of Beachworld for my pre-schoolers. (See Byte Magazine, August 1982, "Designing Computer Based Microworlds," by R. Lawler.)

With the predicate TOUCHINGP, you can write procedures which determine whether a sprite is touching another, or any one of a list of others. Need I elaborate on the possibilities here?

SHADES OF TI LOGO

Since at least one of the chips on the board is made by TI, there are some direct similarities to TI Logo. For example, Apple Sprite Logo has the same maximum speed of 127, and the same "5 sprites in a row" problem. Other similarities include:

FREEZE and THAW -- halts and releases all moving sprites.

SETXVEL and SETYVEL -- sets x- and y-components of the velocity. (Bouncing sprites are a great activity for these commands.)

EACH -- causes separate sprites to follow a list of instructions.

Colors -- sprites and backgrounds can be one of 16 different colors.

BETTER THAN EVER

Sprite enhancements over TI Logo are:

ASK -- allows for momentary talking directly to one or more sprites without affecting others who are currently listening.

HIDE and SHOW -- same as HT and ST.

More shapes are available -- up to 56 different shapes can be defined at any one time.

MOVE -- moves a sprite at a certain speed for a certain period of time.

SLIDE -- moves a sprite forward a certain number of steps in a certain period of time.

COLOR.OVER -- detects the color under a sprite.

SNAP -- replaces sprite's shape with the pattern on which its 16 x 16 grid is resting.

OTHER GOODIES

Apple Sprite Logo has some impressive abilities for background graphics:

LOADPICT and SAVEPICT -- same as MIT Logo for saving and loading graphics.

CHANGE.COLOR -- allows for changing ink and text color.

FILL -- fills an area with the color of the sprite's pen.

SHADE -- fills an area with copies of the shape of the sprite inside that area, in the color of its pen.

IN CHARACTER

Although sprites get the publicity, I feel the character graphics really enhance the Sprite Logo scene. The character manipulation of TI Logo has been improved upon in several ways.

Character shapes can be defined, edited, and saved just like sprite shapes. For example, you can redefine the "4" to look like a "5", so that PRINT 2+2 will give 5 as an apparent result!

Sprite Logo continued

There are fewer restrictions for coloring the characters, too. Apple Sprite Logo has a command called SETTC (short for Set Text Color) which changes the foreground and background color of each character. There are no character sets to worry about. If you want the alphabet to be different combinations of colors on the screen at the same time - no problem! Another feature I like is Sprite Logo's ability to use either upper or lower case letters with an Apple IIe.

All these graphics capabilities allow for fancy layering. The screen color is the bottom layer. Characters and graphics from STAMP, FILL, SHADE, and pen drawings share the middle. Sprites occupy the top. Additionally, sprites themselves are layered, with lower numbered sprites having visual priority, permitting a passing behind / in front effect.

READING MATTER

The Apple Sprite Logo documentation is thorough. Two introductory sections, one on turtle graphics and one on dynamic turtle graphics, contain many sample projects. The reference manual contains explanations and examples of every feature of the language, including all the word and list processing features of Apple Logo.

BUT ... BUT

Alas, Apple Sprite Logo is not perfect. My biggest wish is that it had music and speech. Now, that's not asking too much, is it? Time to dream about MacIntosh Logo...

Most of the minor problems are evidently due to hardware limitations. The one that bothered me most was the lack of a cursor indicator when scrolling text on the screen. If you type CATALOG, POTS, POALL, etc., and toggle the scrolling with CTRL-W, it is difficult to tell the new text from the old.

CTRL-S, -T, and -L are included no more. You must use the full commands, or, as I did, define your own short forms.

The turtle doesn't visually rotate as often as the Apple and TI Logo turtle. For example, if the turtle is given a series of eight RT 5 commands, you will see it pivot only three times. If you type SETHEADING 82, it appears to point the same as SETHEADING 90. This is not a major drawback, but may cause some initial confusion for beginners.

A sprite can draw while in motion with the SETSPEED command, but, as the speed increases, the line becomes increasingly dotted. With FD and BK commands, the line is solid all the way, as usual.

The negative y-coordinate boundary changes if you switch between SPLITScreen and FULLSCREEN.

When first calling on a sprite, you must set its shape to the turtle before it will accept another shape. Maybe that's why I should be calling them turtles instead of sprites!

I've had two crashes with the version of Apple Sprite Logo I tested. "!!!LOGO SYSTEM BUG!!!" was the provided error message, as the screen gradually went bananas. Rebooting the disk was the only way to restore things to normal. The version I have is at least nine months old, and newer versions may not have this problem.

As you can see, I have to pick nits to find fault with Sprite Logo. I can live with these problems, considering the additional Logo power available.

TO GET YOUR COPY

LCSI has elected to distribute Apple Sprite Logo, instead of using the network of Apple dealers. Each Sprite Logo set (includes card and disk) costs \$299. Discounts are available for purchases of five or more units. Send your order or request for more information to Norman Baron, Logo Computer Systems Inc., 220 Fifth Avenue, Suite 1604, New York, NY 10001, or call (212) 684-0710.

WHO AND WHY

Apple Sprite Logo would be ideally suited for home and computer camp use. Although most schools would probably add peripherals such as color monitors and printers to their Logo setups before anything else, I would advise those who can afford it to put Sprite Logo at the top of their priority lists for hardware and software purchases. (It's both, remember.) It will enrich your Logo environment ten-fold!

Griff Wigley is a regular columnist for The National Logo Exchange, and frequently contributes additional articles.

Boston Logo Users' Group News diSessa and the Dynaturtle

reported by
Nola Sheffer

At the January meeting of the Boston Logo Users' Group, MIT Physics Professor Andy diSessa reported on his work with students and the dynaturtle.

"Kids interact with a dynaturtle in a different way than they do with the regular Logo turtle," he began. With a geometry turtle, you can understand what is going on fairly quickly, even though what you are doing may have sophisticated implications.

In contrast, when working with a dynaturtle, designed to model the laws of physics as they apply to a frictionless zero gravity environment, it is difficult to understand what is happening. Our intuition is not a good guide.

For example, if asked how to force an object to travel in a circle, you might reply that you push it in a circular path with a tangential force. However, in reality, you must keep exerting an inward force.

FALSE INTUITION

These false intuitive notions of physics last a long time, and it is difficult to develop sophisticated ones. However, good intuition is important to being a good physicist; it is not enough to be able to manipulate formulae. diSessa believes that a physics microworld might help develop this much needed sophisticated intuition.

He explains that microworlds come in all intellectual domains. He uses the metaphor of an intellectual playground where children can interact in an unrestrained way. This environment allows for:

- (a) pleasurable repetition of skills with which the student is comfortable,
- (b) personal goals and an individual style in reaching those goals,
- (c) a sense of power (reward) for feeling competent and in control, and
- (d) a rich social setting.

In his work with kids and dynaturtles, diSessa found that the students all tended to go through a similar sequence of strategies. For example, when trying to use the turtle to hit a target, students tended to:

- (1) try to aim straight at the target and kick, and
- (2) move the turtle even with the target, turn 90 degrees, and kick.

Both strategies are unsuccessful.

(Note: In such a situation, the typical location of the turtle is at the bottom of the screen, with the target located up and perhaps to the right 45 degrees. The user can turn the turtle through increments of 30 degrees right or left, and apply a kick or impulse in the direction the turtle is facing.)

One successful strategy hit upon by many was that of the "antikick." This cancellation idea has been used by students at all levels, from upper elementary to college. In fact, diSessa found that first year college students tended to go through the same strategies as elementary kids. This may be a comfort to you the next time you are struggling through one of the dynaturtle games!

These strategies can be used, diSessa feels, as a means of explaining laws of physics, and helping students develop an accurate reliable intuition, rather than the immature one with which most of us operate.

I hope that the work Professor diSessa has done with the dynaturtle can be developed and expanded into a form which can be readily used by high school and upper elementary school teachers. It seems too rich a resource to lie undeveloped!

Thank you, Professor diSessa, for sharing your physics microworld.

Nola Sheffer is Co-Director of the Logo Users' Group of the Boston Computer Society.

Here are the current Logo prices from Gold Disk. Krell Logo \$74.95, Terrapin Logo \$119.00. Call 1-800-368-2260.

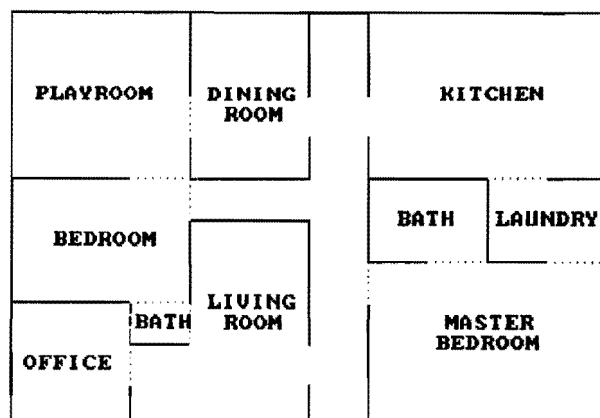
MICROWORLDS

by
Glen Bull

ETV: Educational TurtleVentures Part I

Previous columns have dealt with using Logo to write adventure stories. Now, let's explore how to use Logo to design an adventure game!

Shown below is the floor plan of a turtle's home ... often a place of adventure.



The rooms of this turtle's home are conveniently labeled, but that is not always the case. During the course of an adventure, it is sometimes necessary to ask questions, such as:

TELL WHERE?
DINING ROOM

TELL WHAT?
CANDLE KEY MIRROR

These commands help determine the room in which the turtle is located, and what is in the room. Along with procedures which will enable the turtle to CARRY and DROP objects, these will form the main part of our adventure game design. Let's begin with an exploration of how to develop TELL WHERE?.

WHERE IS THE TURTLE?

There are a number of ways to determine the room the turtle is in. The method I will develop has three advantages.

(1) It requires only a few short procedures.

(2) It can locate the turtle quickly.

(3) It can easily be adapted to rooms of any size or shape, as long as the corners of the room are square.

It has the disadvantage of being somewhat wasteful of memory. Up to 100 variables are used to follow the turtle from room to room.

Think of the screen as a grid of blocks, numbered 0 through 99.

0	1	2	3	4	5	6	7	8	9
10	11	12	13	14	15	16	17	18	19
20	21	22	23	24	25	26	27	28	29
30	31	32	33	34	35	36	37	38	39
40	41	42	43	44	45	46	47	48	49
50	51	52	53	54	55	56	57	58	59
60	61	62	63	64	65	66	67	68	69
70	71	72	73	74	75	76	77	78	79
80	81	82	83	84	85	86	87	88	89
90	91	92	93	94	95	96	97	98	99

Each room is defined by the blocks which fill the room. For example, the following blocks make up the office in the lower left hand corner of the turtle's home:

70 71
80 81
90 91

Thus, if we can find the block the turtle is on, we can tell the room quickly.

Logo has coordinate commands which tell how many steps the turtle is from the center of the screen. If the turtle is 70 steps to the right of the center of the screen,

PRINT XCOR

will give this result:

70

The screen I am using at this time is 320 turtle steps wide. (The number of turtle steps on your screen may be different depending on the brand of computer you are using.) Suppose that I want to divide the screen into ten blocks across, as in the above diagram. Each block will be 32 turtle steps wide.

Microworlds continued

BLOCKING THINGS OUT

320 (screen width) / 10 (number of blocks) =

32 (width of each block)

Suppose the turtle is in the top row, on the seventh block from the left side of the screen. Can I identify this block by dividing the turtle's position by the width of a block?

PRINT XCOR / 32

When I tried this, I was surprised to get the answer:

2.1875

SHIFTING TURTLES

The turtle was trying to tell me that it was on block 2, not block 7. A moment's thought supplied the answer. I was thinking of the screen in this way:

1 2 3 4 318 319 320

The turtle was thinking of the screen in this way:

-159 -158 -157 ... 0 ... 158 159 160

In my mind, I thought of the turtle's position as running from left to right. In reality, the turtle's x-coordinates are positive to the right of center, and negative to the left of center. How can the turtle's position be changed to all positive numbers? Add 159.

PRINT (XCOR + 159) / 32
7.15625

Now the turtle correctly told me that it was on the seventh block across the screen. The value added to XCOR to make all the numbers positive is called "offset". Therefore the number 159 could be called the horizontal offset.

The turtle told me more than I wanted to know, since I didn't care about the fractional part of the number. Asking for the integer of the result took care of this problem.

PRINT INT (XCOR + 159) / 32
7

Now let's write a procedure to tell us what block the turtle is on. A setup procedure defines the variables which will be needed. Use the number of turtle steps for the height and width of the screen in your particular version of Logo in SCREEN.HEIGHT and SCREEN.WIDTH. The number of blocks in each row and column can be any number you wish. I will use 10 here, for a total of 100 blocks. The horizontal offset is used to change the blocks from this:

-4 -3 -2 -1 0 +1 +2 +3 +4 +5

to this:

0 1 2 3 4 5 6 7 8 9

TO SETUP

```
MAKE "SCREEN.WIDTH 320
MAKE "SCREEN.HEIGHT 250
MAKE "NO.OF.ROWS 10
MAKE "NO.OF.COLS 10
MAKE "BLOCK.WIDTH :SCREEN.WIDTH /
:NO.OF.COLS
MAKE "BLOCK.HEIGHT :SCREEN.HEIGHT /
:NO.OF.ROWS
MAKE "HORIZONTAL.OFFSET (:SCREEN.WIDTH
/ 2) - 1
MAKE "VERTICAL.OFFSET (:SCREEN.HEIGHT
/ 2) - 1
END
```

After the SETUP procedure is run, we can use the following procedure to obtain information about the column and row of the block the turtle is on.

TO GRID.LOCATION

```
LOCAL "X LOCAL "Y
MAKE "X INT (XCOR + :HORIZONTAL.OFFSET)
/ :BLOCK.WIDTH
MAKE "Y INT (YCOR + :VERTICAL.OFFSET)
/ :BLOCK.HEIGHT
MAKE "Y (:NO.OF.ROWS - 1) - :Y
OUTPUT LIST :X :Y
END
```

For example, if the turtle is on block 3 of the top row,

PRINT GRID.LOCATION
3 0

The top row of blocks is row 0. Remember that the numbering starts with 0, so that the fourth block over would be labeled 3:

More Microworlds

Block in Row
1st 2nd 3rd 4th

Column on Grid: 0 1 2 3

The only Logo command in this procedure which may be unfamiliar is OUTPUT. The command OUTPUT deserves a column or two of its own, but I have never felt confident enough to write one. (Any volunteers out there?)

Briefly, OUTPUT allows a procedure to act as a function. A function may be thought of as a procedure which accepts an input and returns a value. In this case, the input is the turtle's coordinates, and the output is the column and row information about the block the turtle is sitting on.

What does the next to last line (MAKE "Y (NO.OF.COLS -1)-:Y) of the GRID.LOCATION procedure do? Try removing this line and see what happens.

BLOCKING THINGS IN

The blocks in the sample grid are individually numbered. For example, block 5 in row 3 is labeled 35. If the turtle is placed on that block,

PRINT GRID.LOCATION

produces the result 5 3, and not 53, or 35. Although it might simplify things to number the grid differently, let's use what we might remember of list processing to combine these two numbers into the block number.

WORD AND LOCATION

The procedure GRID.LOCATION outputs a list of two numbers: the column and the row the turtle is sitting on. The list commands FIRST, LAST, and WORD can be used to combine these two numbers into a single number.

The Logo command WORD is used to glue two words together:

```
PRINT WORD "FIRE "TRUCK
FIRETRUCK
```

It can also be used to glue numbers together.

```
PRINT WORD 5 3
53
```

The command WORD can be combined with the commands FIRST and LAST to glue the first

and last members of a list together in reverse order.

```
PRINT FIRST [5 3]
5
PRINT LAST [5 3]
3
PRINT WORD LAST [5 3] FIRST [5 3]
35
```

(Actually the definition of a "word" in Logo is slightly different from the English definition of a word, and can include numbers. A Logo "word" is any item with a quotation in front of it. In most informal Logo versions, the quotations are optional in front of numbers.)

Now we can create a procedure which tells the exact block the turtle is sitting on.

TURTLE TREASURES

In the meantime, try burying treasure under some of the blocks by typing a description of the treasure after the name of the block.

```
MAKE 37 [DIAMONDS]
MAKE 14 [GOLD]
```

(Note: With some MIT Logo versions, a quote is needed before the number. For example, MAKE "37 [DIAMONDS].)

To find out if a block with a particular name (for instance, block 37) has treasure buried under it, type:

```
PRINT NAMEP 37
```

(Note: Some MIT versions require a quote before the number, as in PRINT THING? "37.)

```
TO BLOCK?
  LOCAL "XY
  MAKE "XY GRID.LOCATION
  OUTPUT WORD LAST :XY FIRST :XY
END
```

Note that this procedure will work only if there are no more than 10 rows or 10 columns of blocks.

```
PRINT BLOCK?
35
```

In next month's column the command WHERE? will be developed. WHERE? tells the name of the room the turtle is in by referring to a list of blocks associated with the room.

Microworlds concluded

Logo will print TRUE if there is a buried treasure under the block, and FALSE if there is not. If there is treasure present, you can identify it by typing:

```
PRINT THING 37
```

In this case, Logo will respond with:

```
DIAMONDS
```

Here is a procedure to tell if there is treasure on the block where the turtle is located.

```
TO TREASURE?  
  TEST NAMEP BLOCK?  
  IFTRUE OUTPUT THING BLOCK?  
  IFFALSE OUTPUT [THERE IS NO TREASURE  
HERE.]  
END
```

After burying treasures under some of the squares, use the procedure by typing:

```
PRINT TREASURE?
```

If the turtle is sitting on a block with treasure under it (such as block 14), it will reply:

```
GOLD
```

Young children can use the following procedure to hide treasure at any turtle location they choose, without having to worry about the block number.

```
TO HIDE :TREASURE  
  MAKE BLOCK? :TREASURE  
END
```

For example, while the turtle is in the office, HIDE [MONEY] would hide MONEY beneath whatever block the turtle is on.

To make a treasure hunt more interesting, clues could be placed under the blocks also.

```
MAKE 41 [NORTHEAST TO SHINY YELLOW]
```

If the turtle is on block 41 when you type

```
PRINT TREASURE?
```

Logo will respond with

```
NORTHEAST TO SHINY YELLOW
```

This may help you find the gold at block 14.

AN INVITATION

There are many other applications which might make use of the BLOCK? and TREASURE? procedures we have developed. For example, you might attach crosses and circles to the blocks on a 3 by 3 tic tac toe grid. I substituted the coordinates from a touch tablet (the Koala Pad) in place of the turtle's screen coordinates in the GRID.LOCATION procedure and associated turtle commands with each region on a 5 by 5 block Koala Pad grid to create an Instant Touch Logo.

What other uses can you think of for BLOCK? and TREASURE? We will offer a prize for the best suggestion. Send your ideas to Glen Bull, NLX Microworlds, PO Box 5341, Charlottesville, VA 22905, before May 15, 1984.

Note for MIT Logo Users

The procedures developed in the column for SETUP, GRID.LOCATION, BLOCK?, TREASURE?, and HIDE are provided in the LCS version of Logo for Apple and IBM PC. If you are using Terrapin or Krell Logo for the Apple, or Terrapin Logo for the Commodore, there are slight differences for some procedures.

TO SETUP - no change

```
TO GRID.LOCATION  
  MAKE "X INTEGER (XCOR + :HORIZONTAL.OFFSET)  
  / :BLOCK.WIDTH  
  MAKE "Y INTEGER (YCOR + :VERTICAL.OFFSET)  
  / :BLOCK.HEIGHT  
  MAKE "Y (:NO.OF.COLS - 1) - :Y  
  OUTPUT LIST :X :Y  
END
```

```
TO BLOCK?  
  MAKE "X.Y GRID.LOCATION  
  OUTPUT WORD LAST :X.Y FIRST :X.Y  
END
```

```
TO TREASURE?  
  TEST THING? BLOCK?  
  IFTRUE OUTPUT THING BLOCK?  
  IFFALSE OUTPUT [THERE IS NO TREASURE HERE.]  
END
```

TO HIDE - no change

Glen Bull is a professor in the University of Virginia's School of Education, and teaches Logo courses at both the graduate and undergraduate level.

NLXionary

A lectionary of selected Logo readings

compiled by Griff Wigley

Welcome to the NLXionary! Each month, we will present a lectionary of articles on Logo from other publications for your reading pleasure.

RECENT

PCjr Magazine, April 1984, "The Turtle as Tutor," by James Chumbly.

>An overly enthusiastic report of Logo's impact on the author's 9 year old daughter, within the framework of a brief introduction to IBM Logo. Though it seems innocent enough (a newcomer's excitement), articles such as this may build unrealistic expectations in parents for their children's first exposure to Logo.

PCjr Magazine, April 1984, "The Advanced Course," by Winn L. Rosch.

>This fellow knows Logo and writes with wit to boot. History, philosophy, and applications for Logo all are given interesting treatment here. His "Logo is better" arguments are concise and unusual. Inserts include an example of a neat Logo business graphics program, and a disagreement with his editors on LOGO vs. Logo. Don't be misled by the ridiculous pictures of adults dressed up in turtle suits which grace the cover, the two Logo articles, and the editor's column. They undermine Rosch, who argues convincingly for taking Logo seriously.

Compute!, November 1983, "Bucky and the Turtle," by David Thornburg.

>More stimulating thoughts on turtle geometry. Thornburg connects Buckminster Fuller's futuristic Synergetics philosophy - a "geometry of thinking" - with the mathematical formalism of turtle geometry.

Teaching, Learning, Computing, November 1983, "Papert Reflects on Logo, Past and Future," by Barbara Kellam-Scott.

>A well-written account of an interview with Papert. His recent thinking matches that set forth in "Mindstorms," but he's more pragmatic about the extent to which schools can accommodate total self-directed learning environments.

TurtleTalk, Winter 1984, "The Future of Logo," by Edward and Mark Friedland.

>A more accurate title would be "The Thinking Behind the Features of Our New Logo." The authors describe SOCRATES LOGO (caps are theirs), currently being developed by Krell Software. Multiple touch sensitive turtles, voice input, robot output, redefinable error messages, and a tutorial package are all promised. Their rationale about how current versions of Logo don't go far enough in achieving Papert's purposes is quite interesting, yet the promotional tones of the article overrides and obscures their message. Nonetheless, it sounds like a version to get excited about. It is apparently scheduled for release late this year.

ANCIENT GEM

Byte Magazine, August 1982, "Why Logo," by Brian Harvey.

>Track this article down if you have been involved with computers and Logo for a while now, and are yearning for a deeper understanding of Logo, both as a computer language and a philosophy. This piece is tough sledding at times; yet, every time I re-read it, I learn something new. Harvey thoroughly covers the Logo waterfront in this article, and virtually none of it is out of date. He is one of a rare breed who can talk about subjects over your head in an interesting way, thus teasing you back again and again.

MENTIONABLES

Educational Technology, January 1984, Ed Tech Product Reviews, "Print Materials on Learning Logo for Children and their Teachers," by Douglas Clements.

>A thorough four page review of three Logo products for the classroom: "Logo: A Problem Solving Approach," a set of activity cards by Joan Davis and Joan Schwenker, "1,2,3, My Computer and Me," a book by Donna Bearden, and Dan Watt's "Learning with Logo." You will find no surprises here.

The Christian Science Monitor, January 27, 1984, "Learning a Computer Language May Be Easier Than You Think," by Lois Colt.

Griff Wigley is co-owner of Family Computing, Inc., a facilitator in the Faribault (MN) Public Schools, and a school board member of Prairie Creek Community School.

ParaboLogo

by
Tom Lough

From the beginning, I was intrigued with Logo's ability to draw complex geometric shapes without using complex algebraic formulae. The circle is a classic example, of course.

```
REPEAT 360 [FD 1 RT 1]
```

This expression is both simple and elegant. It is easily understood in terms of how we move ourselves. If we go repeatedly forward 1 step and turn right a little, we will walk in a circle.

I have often wondered, "What about a parabola?" Is there a simple and elegant way to draw parabolic paths without using an algebraic relationship?

Once again, the key is in how we would move ourselves. But who moves in a path that looks like a parabola? We do, when we jump from the bed of a truck or a porch. Our path through the air is a combination of horizontal movement at a nearly constant speed and downward movement accelerated by gravity.

How can this be put together in a Logo procedure? One way is to do it in parts.

The part with the movement at constant speed can be modeled by the following procedure.

```
TO CONSTANT :SPEED
  FD :SPEED
  CONSTANT :SPEED
END
```

Experimenting with commands such as CONSTANT 10, CONSTANT 1, and CONSTANT .3 will result in turtle movement across the screen at constant speeds. The endless recursion can be stopped with CTRL-G or CTRL BREAK for most Logo versions.

The part of the movement with acceleration is a little more difficult. Acceleration increases the distance traveled in each increment of time. Constant acceleration, such as that of earth's gravity, means that the extra distance traveled in each increment of time is the same.

For example, consider this pattern of dots marking the position of a body after equal intervals of time. The body here is moving from left to right, with a constant acceleration.

.

The distance between the dots increases as the body moves from left to right. If you will measure the distances between adjacent dots, you will find that the difference between them is a constant amount.

Thus, a sequence of commands like

```
FD 3 FD 5 FD 7 FD 9 FD 11 FD 13 FD 15
```

would result in a brief accelerated motion of the turtle.

Again, a recursive procedure is an ideal way to simplify the concept.

```
TO ACCELERATE :DISTANCE :EXTRA
  FD :DISTANCE
  ACCELERATE (:DISTANCE + :EXTRA) :EXTRA
END
```

An enhancement I suggest to my physics students is for the turtle to mark its successive positions with dots. With LCSi versions, the command DOT POS works nicely. With MIT versions, it is necessary to define a DOT procedure. A popular one is

```
TO DOT
  PD FD 1 BK 1 PU
END
```

Thus, the ACCELERATE procedure becomes

```
TO ACCELERATE :DISTANCE :EXTRA
  DOT
  FD :DISTANCE
  ACCELERATE (:DISTANCE + :EXTRA) :EXTRA
END
```

**** Note:** If you have Apple (LCSi) Logo, use PD DOT POS PU instead of DOT in the above procedure. For IBM (LCSi) Logo, use DOT POS.

Experimentation with ACCELERATE and various combinations of input numbers will produce a pattern of dots similar to those above.

Now let's put the horizontal and vertical motions of our jump together in an attempt to trace a parabolic path.

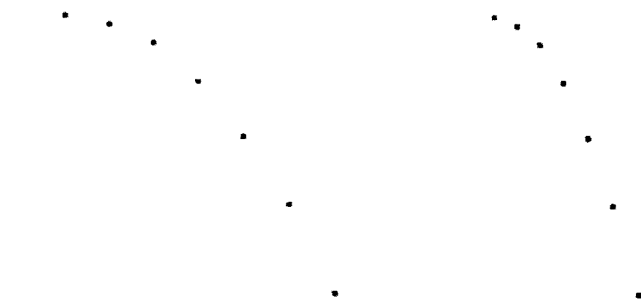
ParaboLogo continued

```
TO JUMP :HOR :VERT :GRAVITY
  DOT (see previous DOT note)
  FD :HOR RT 90
  FD :VERT LT 90
  JUMP :HOR (:VERT + :GRAVITY) :GRAVITY
END
```

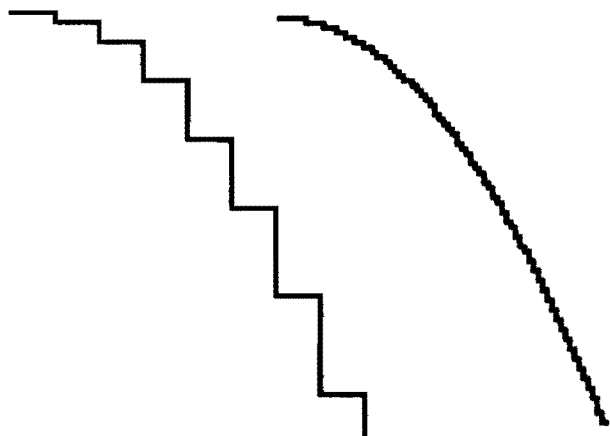
Note that turns of RT 90 and LT 90 are used to change from horizontal motion to vertical motion and back again.

```
TO SETUP
  HOME CS
  PU FD 100 RT 90 BK 100
END
```

Type SETUP to position the turtle in the upper left part of the screen. Then select three numbers for JUMP and begin experimenting. JUMP 10 2 4 is a suggested start.



After some exploration, eliminate the DOT command line from your procedure, and try to find the numbers which will draw a smooth parabolic shape.



Don't get the idea that this is the only way to draw parabolas. If you already have read "Tipps for Teachers" in this issue, you may have wondered if parabolas could be produced using the scrunch commands. Why not give them a try?

FD 100!

Tom Lough teaches undergraduate physics at Piedmont Virginia Community College near Charlottesville, VA, and is the editor of The National Logo Exchange.

Logo Books from Prentice-Hall

Prentice-Hall of Australia has published two Logo books which will be extremely helpful to Logo teachers and students.

Learning Logo on the Apple II, by Anne McDougall, Tony Adams, and Pauline Adams, contains over 200 pages of Logo instructions, examples, and projects for both MIT (Terrapin and Krell) and LCSi (Apple) Logo versions. Unlike many other Logo books for beginners, one third of the text is devoted to the use of Logo beyond turtle graphics. Several helpful appendices, including one with a side by side listing of the commands of the MIT and LCSi Logo versions, round out this excellent book. It is available in the United States now for \$19.95. French and Dutch translations are underway, as is an edition for the TRS-80 Color Logo by Radio Shack.

Let's Talk Apple Turtle, by Liddy Neville and Carolyn Downing, is a 74 page activity book for children. It contains a rich mixture of both on and off computer projects, each selected to help children discover concepts underlying particular aspects of Logo work. Coloring, drawing, and paper folding are used to full advantage to amplify corresponding Logo activities. The teachers' edition is especially well designed. This book will be available in the United States shortly, with a list price anticipated to be under \$10. The teachers' edition will have a list price most likely under \$25. An edition for Commodore Logo will be available soon.

For more information, write Prentice-Hall, Inc. Englewood Cliffs, NJ 07632 USA, or Prentice-Hall of Australia, 209 Glenhuntly Road, Elsternwick, Victoria 3185 Australia.

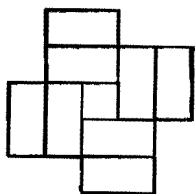
NLXual Challenges

with Robs Muir

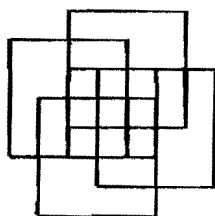
I make a point of offering a new Logo challenge to my students on a regular basis. Every week, I either bring in a hard copy printout of an interesting Logo design, or I sketch a problem on the chalkboard (sometimes "low tech" is faster and easier!). Many of my fourth and fifth grade students live for these problems, and can't wait to see if they will be among the first to crack the next one.

One Sunday evening, a few months ago, I realized that I didn't have a new challenge to bring into class on Monday. To compound the problem, I didn't have an inkling of what I would do - a sure sign of a relaxing week-end well spent! After cranking up the Apple and booting Logo, I found myself staring at a blank fullscreen editor. Since all procedures begin with TO and end with END, I knew that all I had to do was fill in the middle. Serendipity and hacking are old friends; they both came to visit that night.

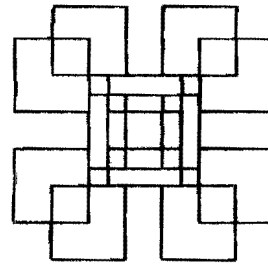
This month's NLXual Challenge is the outgrowth of that short hacking session. Simply stated, write the procedures necessary to recreate these murderous mysteries.



MURDER.MYSTERY1



MURDER.MYSTERY2



MURDER.MYSTERY3

"Mathematician George Polya has argued that general methods for solving problems should be taught ... For example, Polya recommends that whenever we approach a problem we should run through a mental checklist of heuristic questions such as: Can this problem be subdivided into simpler problems? Can this problem be related to a problem I already know how to solve? Turtle geometry lends itself to this exercise." (S. Papert, 1980)

As with all of the NLXual Challenges, if you want to receive a possible solution, send a copy of your solution (or your attempt) with a letter size self addressed stamped envelope to NLXual Challenges, Attn: Robs Muir, 1688 Denver Avenue, Claremont, CA 91711. We will compile contributions and send you the most elegant solution.

If you would like to submit a challenge or puzzle for this column, send your ideas to the above address.

Robs Muir is a teacher in the Claremont CA Unified School District and Claremont Graduate School. He presents Logo workshops in the southern California area.

The National Logo Exchange, copyright 1984 by Posy Publications, a part of The Posy Collection, all rights reserved. Published monthly, September through May, \$25 (US) per year, mailed FIRST CLASS from Charlottesville, Va. \$5 (US) additional per year for addresses outside the USA, Canada, and Mexico. The opinions expressed by the authors are not necessarily those of The National Logo Exchange. Permission is granted for libraries and others registered with the Copyright Clearance Center (CCC) to photocopy articles herein for the flat fee of \$2 per copy of each article. Payment should be sent directly to CCC, 21 Congress Street, Salem, MA 01970. Address other correspondence to: The National Logo Exchange, Box 5341, Charlottesville, VA 22905. ISSN 0734-1717

Editor.....Tom Lough

NLXTRA

Reinhold Wappler wrote a comprehensive evaluation of PC Logo, IBM Logo, and DR Logo for the January issue of the YPLA's Turtle News. Since then, he has added Waterloo Logo, and will share copies of his complete 14-page report with interested NLX readers. For a copy, send a long self addressed envelope to: Logo for the IBM Report, c/o NLX, PO Box 5341, Charlottesville, VA 22905. Please affix 54c in US postage, or include postal coupons.

Friends of LISP, Logo, and Kids (FOLLK) has a new address. Write to them at PO Box 22094, San Francisco, CA 94122. Their Logo Bulletin Board number is 415-333-5663. FOLLK membership is available for \$25 per year, and includes a subscription to their quarterly newsletter, FOLLK Lore.

The Society for Microcomputing in Life and Education (SMILE) is a non-profit organization dedicated to Japanese Educational Computing. The use of Logo is a special interest of the group. A bimonthly CAI newsletter is published. International membership is available for \$18 (4500 yen). Write to Hillel Weintraub, SMILE, Doshisha International High School, Tatara, Tanabe-cho, Tsuzuki-gun, Kyoto-fu 610-03 Japan.

Dr. Darrell Marks announces a Logo Course for beginners, June 18 - 22, at Northwest Nazarene College, Nampa, ID. For more information, write to Dr. Darrell Marks, Division of Mathematics and Natural Science, Northwest Nazarene College, Nampa, ID 83651.

Dave Mintz of CompuTech Curriculum will teach a summer Logo class for educators in Los Angeles, CA, June 25 - July 20. For more information, write Dave Mintz, CompuTech Curriculum, 1116-A Eighth Street, Suite 33, Manhattan Beach, CA 90266, or call (213) 318-6480.

Richard G. Peddicord has written a concise 50 page booklet called Understanding Logo. It is available through B. Dalton book stores for \$2.95. For more information, write Alfred Publishing Co., Inc., PO Box 5964, Sherman Oaks, CA 91413.

The Advisory Unit for Computer Based Education, Chiltern Region, has released "Interim Report No. 2 of The Chiltern Logo Project," by Richard Noss. The 154 page report contains a history of the first year of the research project, what children learned, student examples, and preliminary conclusions. For further information, interested researchers may write: Richard Noss, Advisory Unit for Computer Based Education, Endymion Road, Hatfield, Hertfordshire, AL10 8AU, United Kingdom.

Lynne Mass and her Logo students (see "Students Write the Book on Logo," NLX December 1983) received a nice writeup in the March issue of Scholastic's Teaching and Computers magazine.

In the same T&C issue, Eleanor Zimmerman tells how her students practice Logo commands with a classroom parade. Students take turns being the grand marshall, calling out movement commands for classmates.

Evelyn Dale has an article in the January 1984 issue of School Microcomputing Bulletin. "Teach Thinking with Logo" outlines attributes of Logo which make it an excellent thinking tool.

Diane Hollister and Joan Peart offer Logo workshops to school systems throughout California, in addition to consultation services. A special interest is setting up Logo activities for parents and students. Both Diane and Joan are credentialed teachers, and have had extensive classroom and research experience. For more information, write to them at CompuLEARN, PO Box 1019, Glendora, CA 91790.

Norman and Joyce Baron operate THE BARONS, Educational Consulting, Inc., in Tuxedo, NY, and offer computer consulting services to schools in New York, New Jersey, and Connecticut. They also teach inservice Logo workshops at schools throughout the northeast. Norman is a Logo instructor for Bank Street College, Long Island University, Rutgers University, and TERC. For more information, write to them at 160 Spice Bush Lane, Tuxedo, NY 10987.

At a recent meeting of the Chicago Local Logo Exchange, members indicated their interests included (1) how to teach Logo at various levels, (2) how to teach Logo to teachers, (3) which materials to read, and (4) how to become better Logo programmers. They plan programs around the special topics of Logo theory, Logo's future, and list processing. Members have also established contact with other computer education organizations in the Chicago area. For more information, contact Barbara Randolph, 1455 East 56th Street, Chicago, IL 60637.

Here's a tip for LCSI Logo users. Have you ever wanted to bring more than one procedure to the editor at a time? Here is one way to do it. Let's say you have three procedures, called SQUARE, TRI, and TRI.ONE in your workspace.

Define a list containing the procedure names. For example, MAKE "ALL [SQUARE TRI TRI.ONE]. Then, type EDIT :ALL. You will find yourself in the Logo editor with all three of your procedures.

Looking for a Logo Computer Camp for this summer? For a directory of over 100 computer camp listings, send a check for \$1 (made out to Verbatim) along with your name and address to Camps 'n' Computers, Suite 228, 4966 El Camino Real, Los Altos, CA 94022.

The Computer Camp Book is a 232 page volume which contains both instructions to set up your own computer camp and what to look for in choosing a computer camp. A listing of camps by state is given. The book emphasizes the use of Logo, and includes many helpful references. It costs \$12.95 + \$2.00 for postage. Order from Yellow Springs Computer Camp, 8327 Sheridan Lane, Eden Prairie, MN 55344. For more information, call Sharon Sandusky at (612) 937-2066.

CP & You, a computer learning center in Toledo, OH, announces Logo activities for the summer. Programs for both children and adults are scheduled. For more information, write to Dr. Howard Moskowitz, CP & You, 5403 Elmer Drive, Toledo, OH 43615, or call (419) 535-0130.

IBM has just announced its latest small computer, a 30 pound portable model with a 9 amber screen, 256 K RAM, 2 disk drives, and the PC keyboard, to sell for \$2795. If the IBM PC Logo versions run on it, here is your portable Logo. Have turtle, will travel!

Terrapin is working on a Logo version for the IBM PC and PCjr, to be released sometime in the fall. Also nearing completion is a Terrapin Logo version for the Commodore 264 computer. This particular Logo version will be built into the computer, and will be activated by pressing the F2 key. It is rumored to be one of the fastest Logo versions ever.

Mary Cron of Microcomputer Resources announces complete curriculum kits for elementary school Logo classrooms. Included are discovery cards, duplication masters, facilitator's guide, posters, and more. The materials are designed to support learning Logo through guided discovery. For a free colorful brochure, send a self addressed stamped envelope to Microcomputer Resources, 2845 Temple Avenue, Long Beach, CA 90806.

In February, Punahou School in Honolulu was visited by a group of Japanese educators who wanted to study their nationally known Logo program. (See Electronic Learning, March 1983. Ed.) While there, they presented principal Duane Yee with a Hitachi personal computer which responds to both English and Japanese commands.

Educational Computing Network, PO Box 8236, Riverside, CA 92515, lists DR Logo for \$80, and Terrapin Logo for \$120.

Northeastern Software, 101 William Henry Drive, Monroe, CT 06468, lists Apple Logo at \$160.00 and Terrapin Logo for \$108.00. Call 1-800-382-2242. From Connecticut, call (203) 268-1850.

The March 1984 issue of Instructor contains an excellent article by Herbert Kohl, "The Suspension of Ego in Teaching." It gives several practical suggestions for observation and intervention strategies which could be adapted for use with Logo classes.