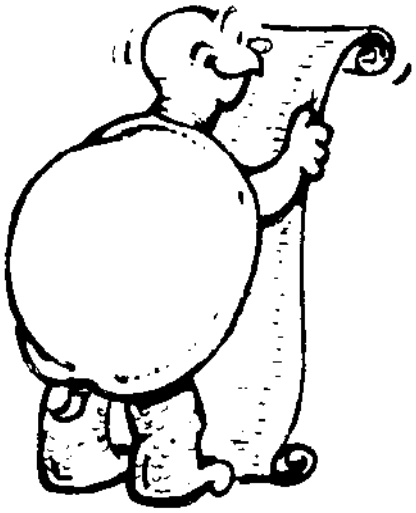
Chapter 4. Writing Procedures

Have you been getting all wrapped up in typing all those commands over and over and over again?

It's the only way you can tell the turtle what to do, right?

Maybe not! Think about this for a minute.

Did you have to tell the turtle how to move? Of course not! She already knew that. FORWARD, BACK, LEFT, RIGHT are all Logo primitives, remember? Primitives are the instructions, or procedures, the turtle already knows.



Now think about this. What if there were a way to teach the turtle how to draw squares, rectangles, triangles, flowers, snowflakes, and other things — and she wouldn't forget?

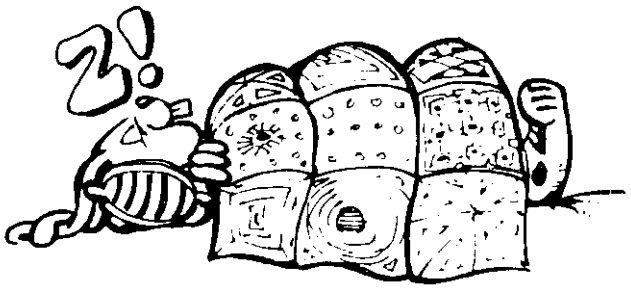
That's just what this chapter is all about. You can teach the turtle to do all sorts of things — lots more than just drawing shapes. However, since drawing shapes is what we've been doing so far, let's start with shape procedures. Then we can use them to do lots more things.

What is a Procedure

First of all, just what is a procedure?

Look at it this way. What procedure do you follow when you wake up in the morning?

You get out of bed. Morf has a lot of trouble with that sometimes.



You get cleaned up, put on your clothes, eat your breakfast, brush your teeth, go to school — or something like that.

Procedures are what you do. They are the steps you take to make something happen.

Logo procedures are things you teach Logo to do. They include all the steps Logo must take to make something happen.

Writing Your Own Procedures

When you write a procedure to draw a shape, you teach Ernestine, the turtle, how ***to*** draw it. This is why you always start a procedure with the word, TO. You want the turtle or Logo TO do something. What would call your procedure for getting up in the morning?

You could call it

TO GET.UP

Then you could add procedures for all the things you do when you get up.

```
TO GET.UP
GET.OUT.OF.BED
WASH.YOUR.FACE
GET.DRESSED
EAT.BREAKFAST
BRUSH.YOUR.TEETH
END
```

Naming Procedures

What's with all the periods between the words?

That's one of the rules about naming procedures. You can name a procedure anything you want. You can call it SUPERCALIFRAGILISTICEXPIALIDOCIOUS if you want, or maybe just GET, or even G.

Name your procedure whatever you want it to be, just as long as the name

- has no spaces. TO GET is fine. TO GET UP is not.

- is not just a number. TO SEVEN is OK. TO 7 is not. TO 7A or TO A7 is OK. TO ; is OK. A procedure name must have a least one character that is not a number.

- is not a symbol that Logo uses such as: (:) \ # " | [or] You can use all the others. TO GET.UP or TO GET_UP are OK.

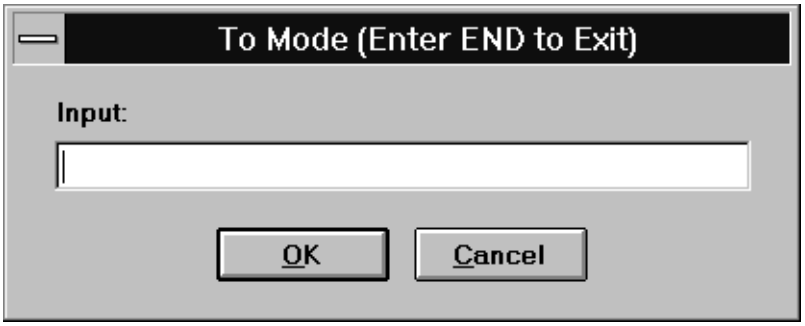
To get started writing procedures, why not teach the turtle how to make the shapes you made in Chapter 3?

Defining a Corner

The simplest shape you made was the corner: FORWARD so many steps and then turn RIGHT or LEFT 90. So let's teach the turtle

TO CORNER

When you type TO CORNER in the MSW Logo Input Box and press Enter, a little dialog box pops up on the screen.



If you type a command on the same line as TO — for example:

TO CORNER FD 100

you will get a message telling you that [] doesn't like FD as an input. That's just another way of saying that you can't have blank spaces in the name of a procedure.

You'll hear more about Inputs when we talk about variables. For now, just remember that TO and the name you give the procedure have to go on a line all by themselves.

Using the To Mode Dialog

Type the first line of what you want the turtle to do in that little box right below the word **Input:** Left-click on **OK** or press **Enter** after each line that you type. If you don't like what you typed, you can delete a word or two. Or left-click on **Cancel**. That erases the whole procedure so you can start all over again.

When you have typed the last command of your procedure, type the word **END** on a line by itself so Logo will know this is the end of your procedure.

"Hey, I want to be able to see my procedure as I type them," Morf moaned. "Can't I do that?"

Yes, you can. You can use the Editor window, which we talk about shortly. All you need to do is go to the Input Box and type

EDIT "CORNER

MSW Logo and some other Logo packages let you type ED "CORNER for short.

The Editor window is displayed showing

```
TO CORNER
END
```

in the upper left corner. Start typing your procedure in between the first line and END. When you have finished

typing, left-click on **File/Exit** to define your procedure. Then you can check it out.

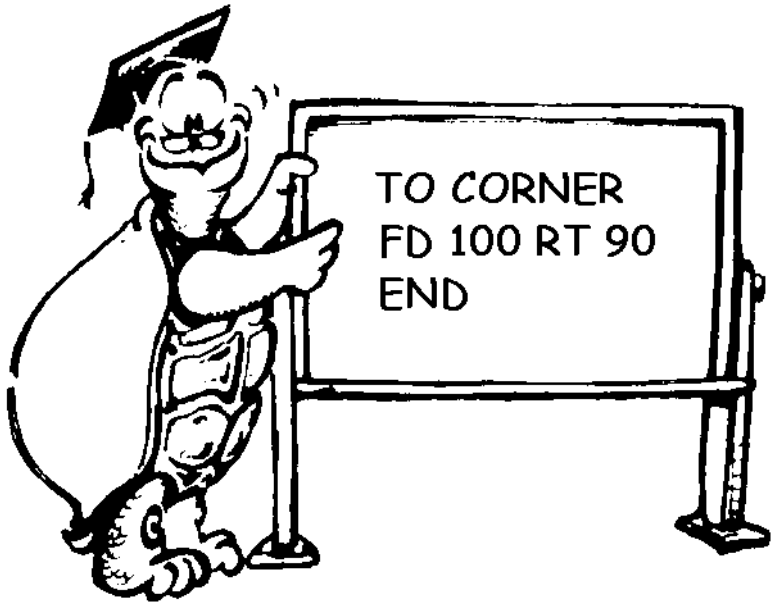
More on using the Editor in just a little bit.

Writing the Corner Procedure

Whether you are working in the Mode dialog box or in the Editor window, here's how to write the Corner procedure. Type

FD _____ RT 90

Fill in any number you want. You can change it later.



When writing procedures, you can have more than one action command on a line if you want. You can have either

FD 100 RT 90

or

FD 100
RT 90

Now type END on a separate line. After you press **Enter**, Logo then sends you a message that says

corner defined

This only happens if you're working in the Mode dialog box. The editor sends you a different message, as you will see.

OK! You've defined your first procedure for the turtle. It's like a new command, so give it a try. Type

CORNER

How 'bout that!

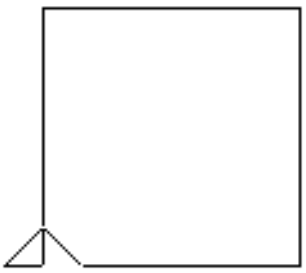
Now, can you use your new procedure to make a square?

Writing a Square Procedure

Sure you can. Go ahead and type **CORNER** again and press **Enter**. What happened? What would happened if you did that again? And again? Well, why not just type

REPEAT 4 [CORNER]

Now we're getting some where. So let's write a procedure for a square.



TO SQUARE
REPEAT 4 [CORNER]
END

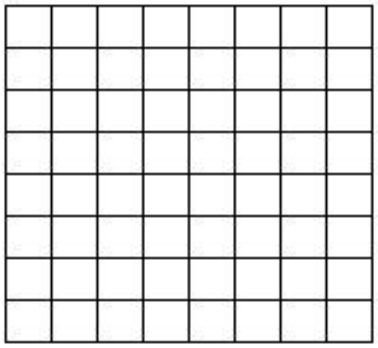
Now type SQUARE to run your new procedure.

Hey, we're on a roll!

Using Your Square

Now let’s put that square to good use. Where are some of the places that you see squares in action? How about a checkerboard?

That shouldn’t be too hard. You’re going to need eight columns of eight squares each. All you have to do is just think through each step the turtle has to take.



First things first. Erase what you’ve done so far by typing ERALL — ERASE ALL for some other Logo packages. Everything is erased from your Logo workspace.

SPECIAL NOTE: Your workspace is the active part of the computer’s memory where your procedures and things stay while your are working on them, and while Logo is up and running.

You’ll see later on that a lot more goes on in the computer’s memory than just holding procedures.

The first step is to draw a column of small squares. Start your checkerboard with squares that have a side length of 20.

If you make the side much bigger, the checkerboard won't fit on the screen.

```
TO SQUARE
REPEAT 4 [FD 20 RT 90]
END
```

When you run the SQUARE procedure, the turtle ends up at the lower left corner of the square. So you have to move up the side and then draw the next square. Try this:

```
SQUARE FD 20
```

Where's the turtle? Does it look OK? What would happen if you tried the same commands again? And again?

Seems to work doesn't it. What about this?

```
REPEAT 8 [SQUARE FD 20]
```

Works for me. However, how are you going to draw the second and other columns?

You could pick the pen up, go HOME, move to the right, and then draw another column. However, that seems awfully complicated.

How about this? Turn right at the top of the first column, move over, and then draw the second column moving from top to bottom.

Would this work?

```
REPEAT 8 [FD 20 RT 90]
RT 90 FD 20 RT 90
```

Think about it — before you try it on the computer. Use your pencil turtle to draw it if you need to?

```
TO COLUMNS
REPEAT 8 [SQUARE FD 20]
RT 90 FD 40 RT 90
REPEAT 8 [SQUARE FD 20]
LT 180
END

TO CHECKERBOARD
REPEAT 4 [COLUMNS]
END
```

Works for me! You'll learn how to fill in the squares in the next chapter.

Defining a Triangle

Do you remember the commands you wrote for a triangle with equal sides?

```
REPEAT 3 [FD 100 RT 120]
```

Now we can make that a procedure, too! Since TRIANGLE is a long title for such a short procedure, why not call it TRI for short.

```
TO TRI
REPEAT 3 [FD 100 RT 120]
END
```

Building Houses

What do you think would happen if we put the triangle on top of the square? You remember the SQUARE procedure, don't you?

```
TO SQUARE
REPEAT 4 [CORNER]
END

TO CORNER
FD 100 RT 90
END
```

Try it. Don't worry about getting it right the first time. Play around and see what you can come up with. Morf came up with a shape that looks like a house.

So he wrote a procedure.

```
TO HOUSE
SQUARE
FD 100 RT 30
TRI
END
```

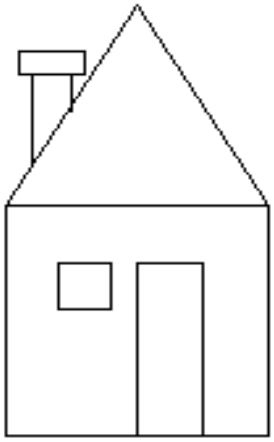
Isn't it easier using procedures as commands?

Not only do you save yourself a lot of typing, you make things easier for other people to understand. They can look at your set of procedures and know right away what it's supposed to do.

How would you ever know what the NYC procedure (later in this chapter) is supposed to do if you didn't see the picture?

Since you're getting so good at this, why not do some things on your own?

Why not add a door
to your house?

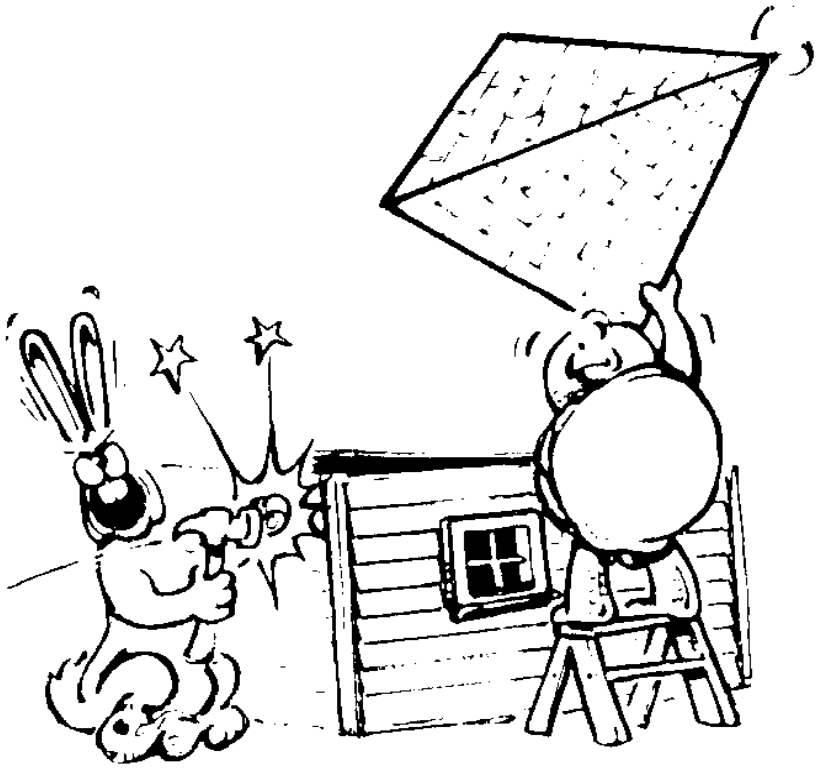


Add a window.

Add a chimney.

How about a TV antenna?

Just don’t add the sore fingers!

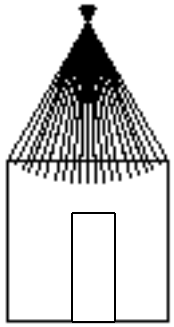


Other Houses

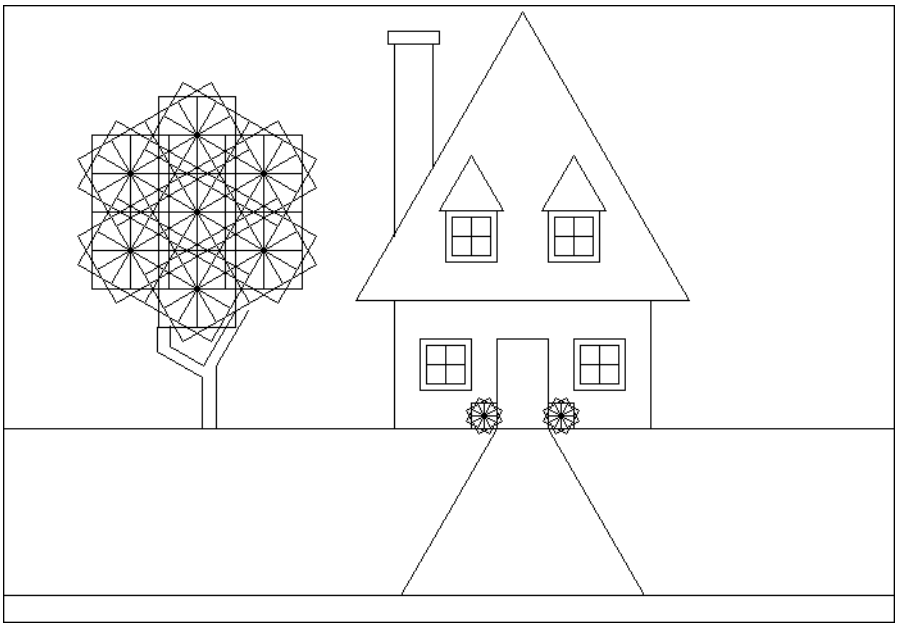
How about trying a few other things with your HOUSE procedure?

Tell you what, how about a native hut? You can use a square block, a smaller rectangle, and some straws or sticks for the roof.

```
TO HUT
REPEAT 4 [FD 60 RT 90]
FD 60 RT 30 FD 60 RT 120
REPEAT 20 [BK 6 FD 66 BK 60 RT 3]
END
```



Maybe this procedure will give you some other ideas. Maybe you can draw the entire yard.



Why not write one procedure to draw several houses. Or, why not try a 2-story house — or something other than a house? Morf has some ideas for you in his next Rabbit Trail.

Rabbit Trail 13. Exploring With Blocks

You started out working with squares, triangles, and rectangles. Now let's put them to work. If you want to draw something else using these shapes, go right ahead!

One of the things that will help you get some ideas is to use blocks of different shapes. Sometimes seeing real things on the table top or on the floor can give you some good ideas to work with on the computer. Also, working with real shapes gives you a better idea of how things fit together.

Get some large sheets of paper. Then place the blocks on the paper to make a figure that you like. Now, carefully trace each shape with your pencil. You'll end up with a picture of your figure on the paper.

Get yourself a ruler and measure each side of each block. Write each measurement by each side.

Unless you're using very small blocks, the drawing you just made is much larger than the screen. Therefore, you're going to have to "scale" your drawing to fit the screen.

Scaling Your Drawings

What does it mean to scale a drawing?

Take a look at a map. Down at the bottom, there is usually a box that describes the map. This section also gives the scale. For example, one inch could equal ten miles. That means that one inch on the paper map equals ten miles of real distance.

You can do the same thing. You can draw your block figure on the screen using a scale such as one inch equals ten turtle steps. As an example, let's say you have a square block that's 3 inches on each side. You can write a procedure that says

```
TO SQUARE
REPEAT 4 [FD 3 * 10 RT 90]
END
```

FD 3 * 10 means go FORWARD 3 times 10, or FD 30.

Do this with your other shapes, using the dimensions that you measured instead of FD 3.

Now try these new procedures on the screen. You can draw a small version of your block figures as a Logo procedure.

But what if *10 is too small? You'll find a way to change that in the *Editing Your Procedures* section later in this chapter.

Houses, Squares, Wheels, and Things

One young lady made a wheel out of the house procedure. How do you suppose she did that?

Well, let's take a look at the house procedure.

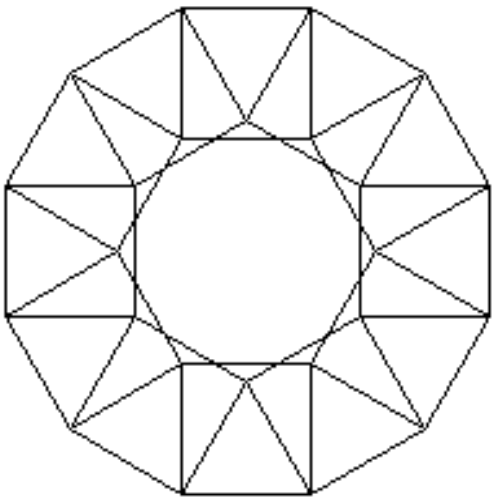
```
TO HOUSE
SQUARE
FD 100 RT 30
TRI
END
```

Before you run the HOUSE procedure, hide the turtle. Do you remember how to do that? If not, turn back to Chapter 2 and check it out.

After you hide the turtle, type HOUSE. Then type HOUSE again.

What happened?

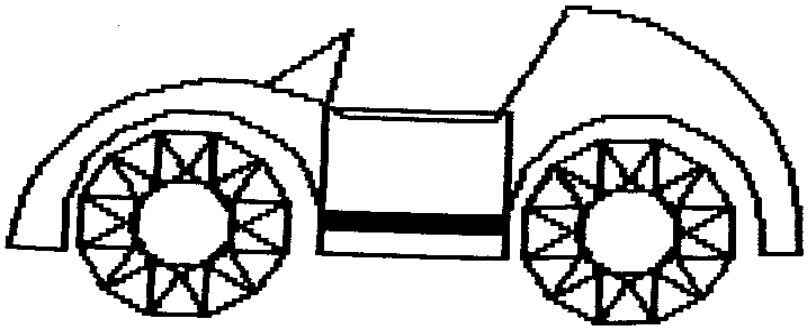
Try it again. What happened this time?



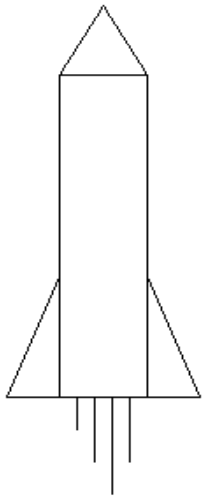
What number do you add to the procedure on the next page to make our friend's wheel?

```
TO WHEEL
HT
REPEAT ____ [HOUSE]
END
```

Here's a car our young friend designed. We'll talk more about circles and curves later on. In the meantime, what can you dream up?



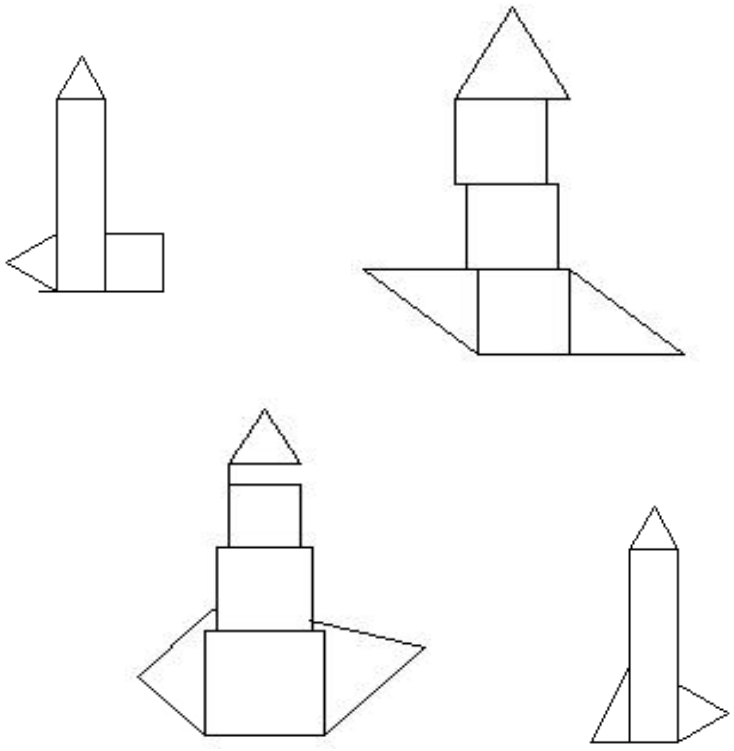
Write down all the things you can think of that use squares, triangles, rectangles — maybe even wheels.



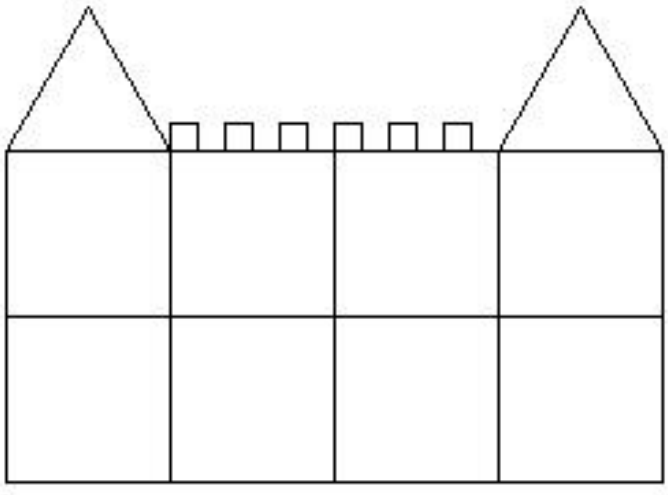
How about a rocket ship?

Maybe the Space Shuttle?

Take a look at the next page. Here are some NASA rejects.



If you prefer to stay on Earth, how about a castle?



To see how these work together, look at the Castle procedure on the CD. It's called CASTLE.LGO.

Bet you can make a much better looking castle. Why not give it a try? Go ahead, explore a bit. You can discover all sorts of things that way.

Reading Procedures

Another way to trace procedures is to read them the way that Logo does. For example, let's take a look at one of Morf's one-liners.

```
REPEAT 12 [REPEAT 4 [FD 100 RT 90] RT 30]
```

Is this right?

Logo starts from the left and reads the first word, the command REPEAT. To run correctly, REPEAT needs a number to tell it how many times to repeat, followed by a list of instructions to repeat.

So, Logo reads to the right. Yes, there's the number 12. So the next step is to look for a list that will tell REPEAT what it is going to repeat twelve times.

The brackets — those are the things that look like square parentheses [ ] — tell you that the things inside the brackets are a Logo list. In Logo, lists can be groups of words, numbers, letters, or even other lists. You'll see lots of lists in this book.

And that's just what you find after REPEAT 12, another list.

There's that REPEAT command again. And, yes, it is followed by the number 4 and a list. The list tells the turtle to

do what's inside the brackets: go forward 100 steps and turn 90-degrees to the right.

That's OK. This list is followed by the command RT 30. So it seems that there's a perfectly good list for the first REPEAT command. This is what Logo repeats 12 times.

[REPEAT 4 [FD 100 RT 90] RT 30]

Now let's try something. Type the REPEAT command shown below and press **Enter**.

REPEAT 12 [REPEAT 4 [FD 100 RT 90 RT 30]

What happened? What's that tell you about brackets?

For every left bracket, there has to be a right bracket.

Sometimes, you'll use parentheses. You'll see much more of them later in this book. Like brackets, for every left parenthesis, there has to be a right parenthesis.

Time Out For PRINT

Here's an experiment to try using parentheses and brackets — quotation marks, too.

You know how Morf loves experiments. Let's experiment with a new command while looking at brackets, parentheses, and quotes — the PRINT command.

Type this:

PRINT "HELLO, "LOGY!

What happened?

Now try this:

(PRINT "HELLO, "LOGY!)

What happened this time? Do you see what the parentheses did?

Now try this one:

SHOW "HELLO,\ LOGY!

What's the difference between the command above and this one?

SHOW "|HELLO, LOGY!|

One command uses the backslash and the other uses the big vertical lines. But what they show looks the same, doesn't it.

PRINT and SHOW are a lot alike as you'll see. Use either one, for now. Can you find the difference between the two commands?

What does that backslash do? It creates a space, right? See what happens with this one:

PRINT "HELLO,\ LOGY.\ \ MY\ NAME\ IS\ MORF.

OK, let's get back to brackets. Try this one:

PRINT [HELLO, LOGY!]

No quotation marks this time. Why not? What happens if you include parentheses or quotation marks inside the brackets?

```
PRINT ["HELLO, "LOGY!]
```

What does this tell you about quotation marks? Think about that. We'll talk about quotes, words, lists, brackets, and parentheses lots more.

But, before we go, what else can you do with the PRINT command? How about this!

```
REPEAT 12 [REPEAT 4 [FD 100 RT 90] RT 30
  PRINT "WOW]
```

Or, how about adding a title to your procedures.

```
TO STARFLAKE
HT REPEAT 6 [SNOW RT 60]
PR [A STARFLAKE BY ERNESTINE.]
END
```

You guessed it, I bet. PR is a shortcut for PRINT. But we're getting off track. Brackets and parentheses are very important but we need to get back to what we were doing.

Snowflakes

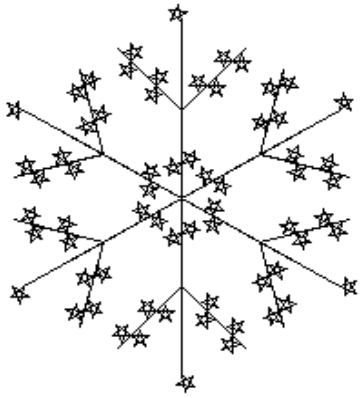
Here's a fun project to help you get practice writing procedures.

Know what a snowflake is? It's an ice crystal. They say that no two snowflakes are alike. But they all have something

in common. They are all crystals that have six sides or six points.

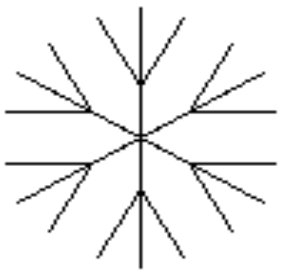
Why not get some friends together to see who can draw the fanciest snowflake. There are some sample Snowflake procedures on the CD than came with this book. Here's a look at them.

```
TO STARFLAKE
HT REPEAT 6 [SNOW RT 60]
END

TO SNOW
SNF FD 30 RT 45 SNF SNF
FD 10 BK 50 LT 90 SNF
SNF FD 10 BK 50 RT 45
FD 50 LT 45 S RT 45 BK 100
END

TO SNF
FD 20 RT 45 S LT 180 S RT 135
END
TO S
REPEAT 5 [FD 10 RT 144]
END
```



Here's another one called SPI

```
TO SPIKEFLAKE
SNOWFLAKE
PR [BY MEGAN HANKINS]
```



```
PR [2ND GRADE]
END

TO SNOWFLAKE
HT REPEAT 6 [SPIKE RT 60]
END

TO SPIKE
FD 50 BK 30 RT 30 FD 30 BK 30 LT 60
FD 30 BK 30 RT 30 BK 20
END
```

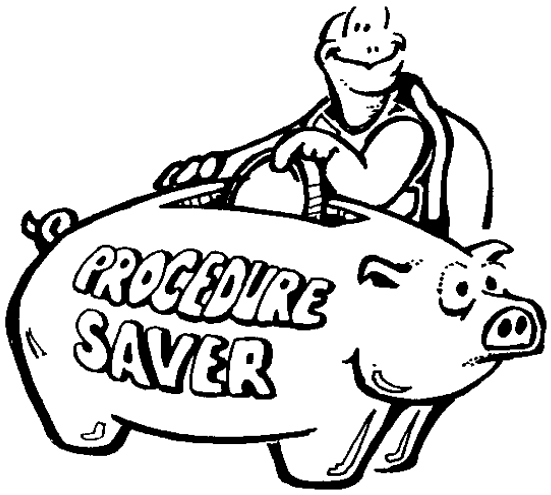
Michelle wrote procedures for a Snowflake show. It's on your CD as FLAKES.LGO. Does it give you any ideas?

Maybe this will get you started.

1. Draw a snowflake.
2. Pick up the pen and move the turtle to another part of the screen.
3. Draw another snowflake and then move again.
4. Do this several times and the screen will look like a snow storm.

But remember! Since no two snowflakes are alike, each of yours should be different. You didn't think this was going to be easy, did you?

Saving Procedures



Saving your procedures is easy. Just type

SAVE "HOUSE.LGO

or whatever you want to call it.

Now all the procedures in your workspace are saved under that name. Let's see how this works.

Type ERALL (or ERASE ALL) so that all your procedures are erased from your Logo workspace.

Now what happens when you type **HOUSE**. Logo comes back and says

I don't know how to house.

Procedures are saved in your MSW Logo directory. You can, of course, save procedures to whatever directory you want. Remember, you have to use double backslashes like this:

SAVE "C:\\<*pathname*>\\HOUSE.LGO

The backslash is a special character in Logo. The easiest way to explain it is to say that Logo reads the first backslash as a special character and gets ready to print whatever the next character is, even if it's a space. As you will see later on, it's a way to include spaces in Logo words.

If this is too much typing, you can always use File/Save or File/Save As... to save your procedures to whatever name you want.

By the way, if you're using another version of Logo, check in the **\projects\chpt4** directory of the CD that came with this book for more information on saving, loading, printing, and editing your procedures.

Printing Procedures

You already learned about printing pictures. There are actually several ways to print procedures. Which way you select depends on your printer.

Here are two procedures that work with dot matrix printers and those printers set to text mode. They may work on a laser or inkjet printer depending on the printer settings.

```
TO DUMP
DRIBBLE "LPT1
POALL
NODRIBBLE
END
```

```
TO DUMP
OPENWRITE "LPT1
SETWRITE "LPT1
POALL
SETWRITE []
CLOSE
END
```

DRIBBLE and the OPENWRITE/SETWRITE combination send what ever follows to the file name or device to type after the command. In this case, you type the name of the printer port to send the procedure to be printed. There are some examples of these commands in the \projects\Chpt4 directory on the CD that came with this book. You can also read more about these commands in the Help file.

POALL is one of the Print Out commands discussed later in this chapter. It prints all you have in your workspace.

You can also use the **Edit** or **Notepad** programs that come with Microsoft Windows to print your Logo procedures. Morf likes this idea because it doesn't matter what's in his workspace. He can print whatever he wants.

To print your procedure from Windows 95:

1. Click on **Start** and open the **Accessories** folder.
2. Select **Notepad**.
3. Select **File/Open**.
4. Open the directory where your Logo procedures are stored.
5. Select the procedure to print.
6. After it is displayed in Notepad, you can print it.

Instructions for Windows 3.1 are similar. Minimize MSW Logo temporarily. Open the Accessories Group and select Notepad. Select and print your procedure as described above.

Awfully simple? Or simply awful?

Loading Procedures

If you want to see your house, all you have to do is load it back into the workspace, right? Well, don't just sit there, go ahead and do it!

Type LOAD "HOUSE.LGO
or
LOAD "C:\\<*pathname*>\\HOUSE.LGO

where <*pathname*> is the directory in which your Logo files are stored.)

You can also use your mouse. Open the **File** menu and left-click on **Load**. A menu appears from which you can pick the procedure you want to load.

Editing Your Procedures

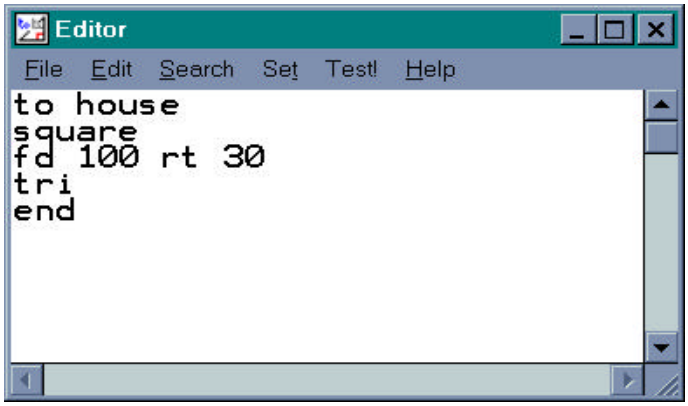
Sometimes your procedures don't do exactly what you want them to do. You have to change them. In Logo, we call this "editing." Guess how you do that?

Give yourself a Gold Star and double-dip ice cream cone if you said go to the "Editor."

To edit the House procedure, type

EDIT "HOUSE or ED "HOUSE for short. PC logo, as an example, uses EDIT HOUSE.

+



Presto! The Editor window appears. This is where you can make any changes you want to make to your procedure.

The Editor Window

There are six menus in the Editor window. Here's a brief description of each. This is a good time to explore these choices to see just what each does.

File: This menu offers you just one choice. You can select Exit to close the Editor window.

Edit: The Edit menu offers you six selections that do the same thing that they do in other windows programs.

> **Undo**: If you edit a line and then change your mind, Undo changes things back to the way they were.
>
> **Cut**: This lets you cut things from one place so you can paste them somewhere else.
>
> **Copy**: This lets you copy things from one place and paste them somewhere else.
>
> **Paste**: Things you cut or copy stay on the clipboard until you paste them where you want them.
>
> **Delete**: Use Delete to erase things from your procedures.
>
> **Clear All**: This clears everything from the Editor.

Search: This menu gives you three standard Windows choices: Find, Replace, and Next.

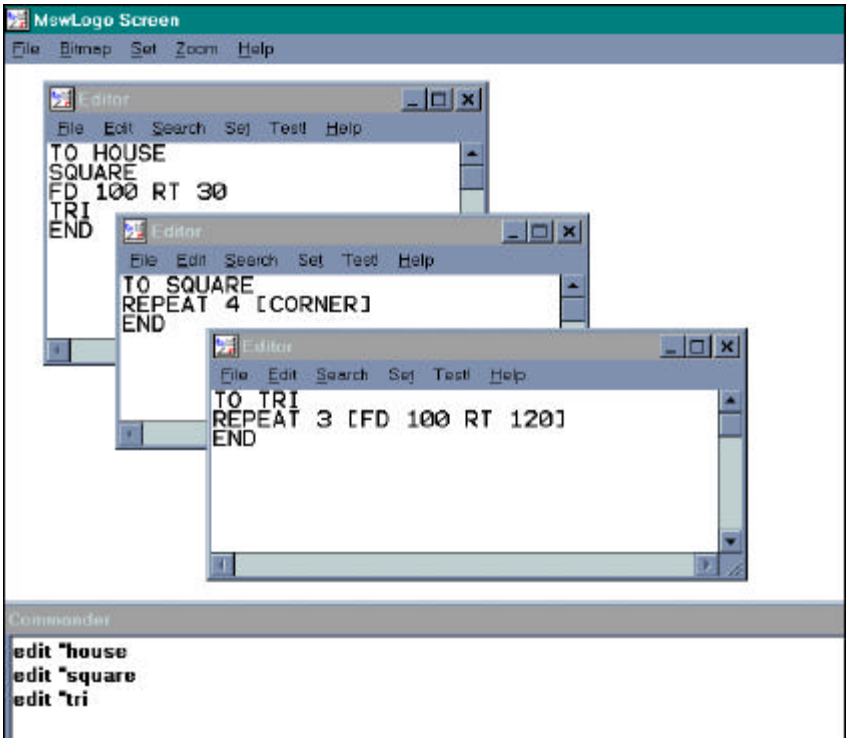
> **Find:** This helps you find a word or phrase in your procedures.
>
> **Replace**: This allows you to replace words or phrases. For example, in the Rabbit Trail on blocks, if you want to change the * 10 to * 20 or something else. Try it.
>
> **Next**: This is used with Find. Left-click on Next or press the F3 key to find the next use of a word or phrase.

Test: This lets you test a portion of your procedure to see if it does what you want it to do. More about this later.

Why not try some of the Editor menu options on the CASTLE procedures?

Opening More Than One Editor

MSW Logo lets you open as many Editor windows as you want. Take a look.



When you have lots of windows to chose from like this, it's easy to copy procedures from one window to another. In this way, Morf can keep his House procedure. You can make a copy of it to build a new two-story house.

Now Let's Edit

"You know, Logy. I think I like that idea of making a two-story house. But do I have to look at your cousin all the time? Why not hide the turtle?"

"Well, OK, let's do it," Logy answered. "First, we'll hide the turtle. Then you can try the easy way to build your two-story house. Later we'll get tricky and create your new

procedures where the only thing we'll type is the name. Watch and see!"

The first thing to do is erase everything and then load the HOUSE procedures back into your workspace.

The Easy Way

Now you're ready to try the easy way of changing a house into a two-story building.

Type **EDALL,** **E**DIT ALL in some versions. You can also open the File menu, left-click on Edit, and select All.

The Editor window opens displaying all the House procedures.

Now you're ready to add a procedure to make a two story apartment. Type the following in the Editor window.

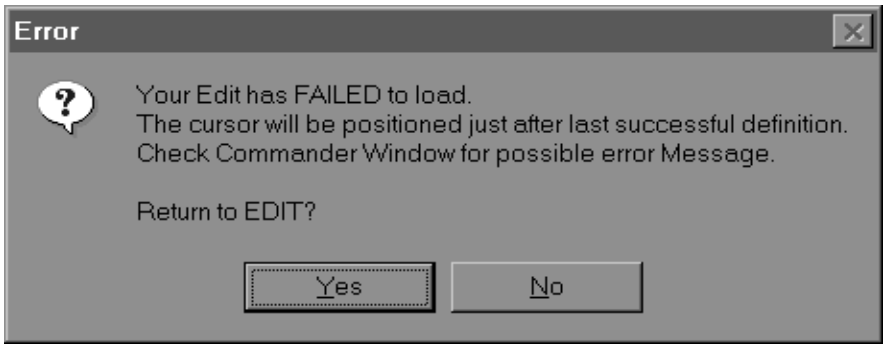
```
TO APT
SQUARE
FD 100 HOUSE
END
```

To close the MSW Logo Editor window, select **File/Exit**. If you have changed the file — you edited something — a box is displayed telling you just that. It says

You have the choice of selecting Yes, No, or Cancel. Left-click on **Yes**. The edited file is saved in your workspace. If you select No, your editing is not saved and you go back to the Commander window. To go back to the Editor, left-click on Cancel.

Editing Errors

If there is an error in your procedure, you get the message shown below.



Select **Yes** to go back to the Editor where the procedure displays the definition causing the problem. Once you have corrected the problem, select **File/Exit** from the Editor window to get back to the Commander window.

To save this as a new procedures, left-click on **File/Save as...** Name the procedure **APT** or whatever you want and select **OK** to save it.

What would happen if you clicked on Save? Your new APT procedure would be saved over your HOUSE procedure. So be careful when saving procedures. You don't want to loose valuable work.

Let's Get Tricky

Erase your workspace. You remember how to do that, don't you? Sure you do. The command is ERALL. Then load the HOUSE procedures again.

The first thing we're going to do is copy the HOUSE procedure to a new Editor window. To open that new window, type

```
EDIT "APT
```

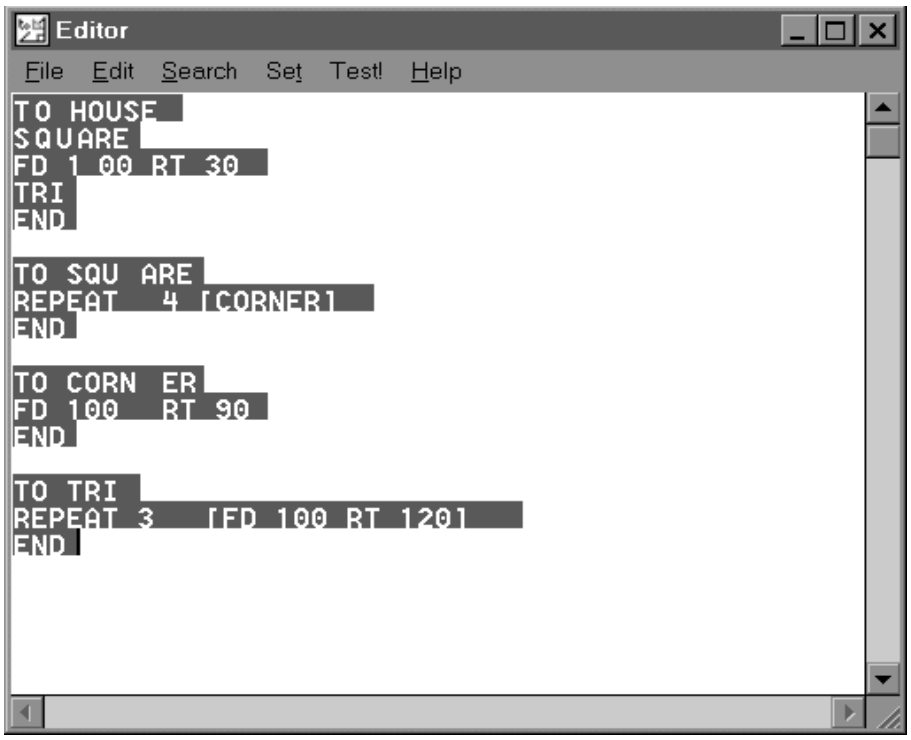
A new Editor window opens displaying two lines:

```
to apt
end
```

Now, left-click your mouse in the Editor Title Bar. Hold the button down and drag this Editor window out of the way so you can see the window with the HOUSE procedures in it.

1. Put the cursor to the left of the words TO HOUSE.
2. Drag the mouse over all the HOUSE procedures.

The text you highlighted changes color.



3. Left-click on the **Edit** menu and select **Copy**.
4. Now go over to the APT procedure. Put the cursor below the APT procedure. Left-click on the **Edit** Menu and select **Paste**.

Presto! You now have an APT procedure.

Copying and Pasting

Left-click to the left of the word **SQUARE** in the HOUSE procedure and drag the cursor over these commands:

SQUARE
FD 100

Select **Edit/Copy**. Now go to the APT procedure in the other Editor window and paste those commands into the APT procedure. Put the cursor to the left of the word **END**. Open the **Edit** menu and select **Paste**.

Now go back to the HOUSE procedure. Copy the word **HOUSE** from the procedure name and paste it to right after **FD 100** in the APT procedure. Make sure that END is on a line by itself.

Your procedure should look like this:

```
TO APT
SQUARE
FD 100 HOUSE
END
```

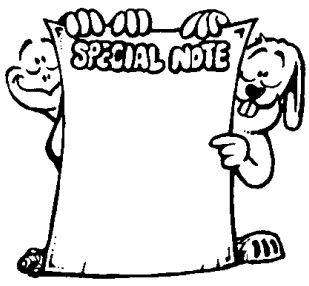
Yes, this is a bit complicated. However, you did save yourself some typing. More importantly — and this is the point of the whole thing — you saw how you can copy all or parts of procedures from one Editor window to another.

Pretty cool! Open the **File** menu and select **Exit** to close the House Editor window and then we can get to more editing.

Now, put the cursor in front of the word SQUARE in the first line of the APT procedure.

Type CS HT and leave a space. The first line of your procedure should now look like this:

```
CS HT SQUARE
```



SPECIAL NOTE: It's always a good idea to clear the screen and bring the turtle HOME before you begin any procedure. In this case, you know your house will be

on the ground and not tilted up in the air somewhere. Hide the turtle if you want.

What's In Your Workspace

One more thing — and this is sort of important. Sometimes when you're writing procedures or editing, you want to see what's in your workspace. To do this, you can use the Print Out command, PO for short.

Do you remember all the procedures you wrote to make the house?

```
HOUSE
SQUARE
TRI
CORNER
```

Maybe you wrote some others for a chimney, a door, or a window. To list all the procedures you have stored in your workspace, type

```
POTS    (That's short for Print Out TitleS.)
```

Presto! There are the titles of all the procedures in your workspace.

To list all the procedures you have stored in your workspace, type

```
POPS    (That's short for Printout ProcedureS.)
```

To see just the titles of procedures, you can type POTS or try

PROCEDURES

Presto! There are the names of all the procedures in your workspace. This is handy when you are working on lots of different procedures.

To see everything in your workspace — procedures, property lists, variable names, and other stuff you haven't read about yet — use the command POALL.

PO will, of course, list other things that you'll get into later — things like variable names and constants. To get a preview of these commands, select **Help/Index**. Select **Help Topics** and then type **PO** in the box that is displayed.

PO and all the other Print Out commands are listed. You can look at any or all of these that you want.

Tracing Procedures

Everybody makes mistakes. We all know that. We talked about correcting typing mistakes before. But what about mistakes when a set of procedure doesn't do what you want them to do. Sometimes it can be tough finding these mistakes.

That's where "tracing" comes in. This lets you see just how your procedures work. MSW Logo offers you a TRACE command and a TRACE button. You can use either one. To turn tracing on, press the Trace button or type

TRACE <*procedure(s) to be traced*>

The list below shows you what happens when you trace the HOUSE procedure. You can trace a single procedure.

```
trace “house
house
 (house)
  house stops
```

Or you can trace a list of procedures.

```
trace [house square corner tri]
house
(house)
 (SQUARE)
 (CORNER)
  CORNER stops
 (CORNER)
  CORNER stops
 (CORNER)
  CORNER stops
 (CORNER)
  CORNER stops
 SQUARE stops
 (TRI)
 TRI stops
house stops
```

Do you see how it works? Some versions of Logo simply turn tracing on. Then everything you run is traced whether you want to see it or not. In MSW Logo, only the procedures you list are traced.

To turn TRACE off, type UNTRACE or press the UNTRACE button.

Stepping Through Your Procedures

"But what if I want to see how each line in a procedure works?"

That's when you need to use the STEP command. It is similar to the TRACE command except there is no STEP button. When run a stepped procedure, each instruction line in the procedure is displayed in a box like you see below.

Logo waits for you to say OK before moving to the next line. You might say that STEP is a detailed TRACE. To STEP through a procedure, type

STEP <procedure list>

To turn STEP off, type UNSTEP.

Why not try STEP on the TESSEL.LGO and DALLAS.LGO procedures? This will give you a good comparison of the two procedures.

Organizing Your Procedures

"Enough of this boring stuff. It's time to move on to some fun," Morf said, banging away at the computer.

"But wait a minute, Morf! There's lots more to writing procedures than just this. So far, we've been talking about short, easy procedures. What are you going to do when you want to make a really complicated procedure? How are you going to know what's supposed to happen if you don't organize your what you're doing?"

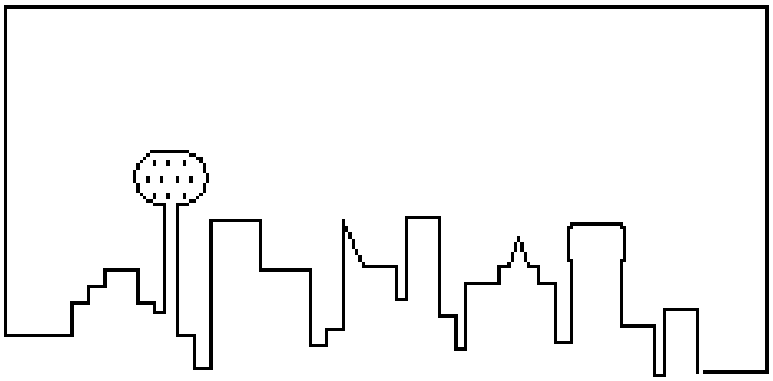
"You mean like this humongous procedure of New York City? Now this took a lot of organizing!"

"How do you like that picture? Pretty cool, huh!"

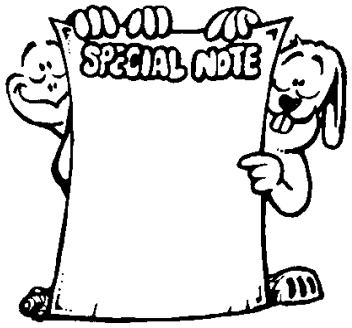
"Morf, I looked at that procedure (NYC.LGO on the CD that came with this book). It must be at least four pages long. How could you ever hope to edit something like that?"

When you're writing procedures, you do much better when you divide your procedures into chunks that are easy to work with, and easy to understand.

Take a look at this skyline of Dallas.



You can see how our friend, Larry, organized his procedures by looking at DALLAS.LGO on the CD.



SPECIAL NOTE: The lights on the Reunion Tower could have been drawn the hard way of lifting the pen after the tower was drawn, going forward, drawing a dot, lifting the pen, and so on. SETPOS or SETXY make it much easier. If you want to peek at Chapter 7 for a preview of how they work, go right ahead.

Do you see the difference between the NYC and the DALLAS procedures? NYC may be the biggest single procedure you will ever see. DALLAS makes a lot more sense. Each building is a separate procedure. If you ever want to change the drawing, you can simply change one procedure. If you ever wanted to change the NYC procedure, where would you start?

These two skylines actually came from a contest that Logy and Morf conducted to see who could draw the most creative skyline for their city? They got some really creative procedures.

Why not give it a try? You can use pictures from magazines, or photographs from the travel section of your newspaper. Go for it! Get creative!

Something else to think about. Do these very different procedures give you any ideas about superprocedures and subprocedures? Let's look at some other examples.

Tessellations and Optical Illusions

Let's tessellate!

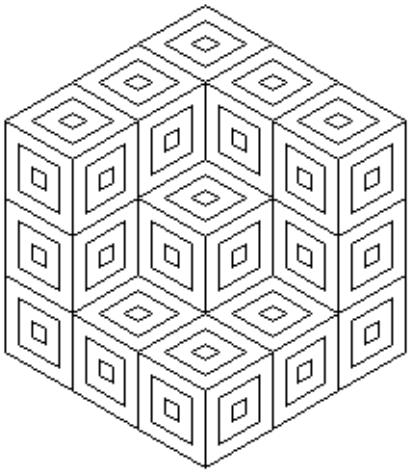
Tessellations can be lots of fun, real brain teasers. They can also be great examples of superprocedures and subprocedures. But before we think about super and subprocedures, what's a tessellation?

You see tessellations all over the place. They're the patterns you see in wallpaper, in tile floors, in blankets — even in clothes. Strictly speaking, tessellations are patterns made by repeating one or more shapes over and over again in what seems like an endless pattern.

There are lots of books on tessellations. You can find them in your library and then try them on the computer.

Some tessellations are optical illusions. That means they look like something they are not. For example, if you look at the picture below, it looks like it is a stack of blocks.

You can also find lots of [illegible] can be lots of fun on the com[illegible]



Here's a long procedure that uses diamond shapes to make what looks like a stack of blocks. It's long, but it makes interesting picture. (It's on the CD as TESSEL.LGO.)

To see what it's all about, type TESSELLATE. Pretty neat, huh?

Compare TESSELLATE with NYC. TESSELLATE almost uses too many procedures compared to the one humongous thing used to draw the NYC skyline.

Here's a challenge. Can you simplify TESSELLATE by combining two or more procedures?

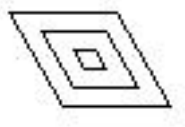
What the difference between TESSLLATE and DALLAS?

These show you two different programming styles. You'll see many more different programming styles before you're done.

Rabbit Trail 14 th Diamonds

TESSELLATE uses 27 diamond shapes to create the optical illusion. To see more of what you can do with these shapes,

why not cut out 27 diamonds like the one shown below and then piece them together in different patterns?



When making your diamond designs, try stacking diamonds on top of each other, like the TESSELLATE procedure does. Or maybe you can discover an entirely new TESSELLATION all on your own.

Most importantly, have fun exploring.

Superprocedures and Subprocedures

The DALLAS and TESSELLATE procedures are good examples of different types of superprocedures. But what's a superprocedure?

Superprocedures don't do anything by themselves. But they do make a lot happen. That's because they call tell the subprocedures when it's time for them to do their jobs.

Take a look at the DALLAS procedures.

This superprocedure tells the other subprocedures when to do their jobs, when to execute their commands. In this case, each subprocedure in DALLAS is run, one after the other.

```
TO DALLAS
HOTEL
REUNION
```

```
BUILDING
BANK
BUILDING2
SPIRE
BUILDING3
BUILDING4
LIGHTS
BORDER
END
```

TESSELLATE uses a bunch of superprocedures to create that complicated pattern. Think what it would be like if you had to type in every move the turtle made to draw that optical illusion? That would sure be a lot of typing. It might even be worse than the NYC procedure!

More importantly, how would you ever organize your thoughts?

Let's take a closer look at TESSELLATE. Maybe there's more to learn from it. Start with a basic diamond.

```
TO DIAMOND
REPEAT 2 [FD 8 RT 60 FD 8 RT 120]
END
```

This draws the smallest diamond. Then you MOVE outside of that diamond to draw the next one.

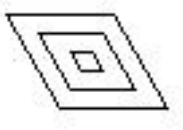
```
TO MOVE
PU RT 60 BK 8 LT 60 BK 8 PD
END
```

DIAMOND1 has sides that are three times longer than the first one. How do you know? Why three times longer? Why not two? Or four?

Why not try FD 16 and FD 32 to see what happens? Does it look right to you?

```
TO DIAMOND1
REPEAT 2 [FD 24 RT 60 FD 24 RT 120]
END
```

Now look at DIAMOND2. Why do you think it uses FD 40? Why not some other number?



The answer is that the sequence of FD 8, FD 8 * 3, and FD 8 * 5 gives you the nice even appearance of the diamonds, each nested inside another.

Now, see what you can dream up. If you need a little help getting started, see what kind of vines you can grow.

```
TO VINE
REPEAT 12 [HALFVINE OTHERHALF.VINE FD 20]
END

TO HALFVINE
FD 6 RT 30 REPEAT 6 [FD 4 RT 20]
RT 60 REPEAT 6 [FD 4 RT 20] RT 30
END

TO OTHERHALF.VINE
FD 6 LT 30 REPEAT 6 [FD 4 LT 20]
LT 60 REPEAT 6 [FD 4 LT 20]
```

END

This vine grows in a circle. Can you make it grow up a wall? What would you have to change?

Here's another thing to think about. Did you ever see a black and white vine? How about adding some pizazz to it?

That's coming up in the next chapter.


MY JOURNAL
OR HOW I DID IT!