

# Contents

Preface . . . . .	ix
The intellectual content of computer programming. Computer science apprenticeship. Why Logo? Hardware and software requirements. Words of wisdom.	
Acknowledgments . . . . .	xv
1 Exploration . . . . .	1
Getting acquainted with Logo... in two senses. Another greeting. Fooling around. A slightly longer conversation. A sneaky greeting. A quiz program. About chapter 2. No exercises.	
2 Procedures . . . . .	10
Procedures and instructions. Technical terms. Evaluation. Error messages. Commands and operations. Words and lists. How to describe a procedure. Manipulating words and lists. Writing your own procedures. Editing procedures. Order of evaluation. Special forms of evaluation.	
3 Variables . . . . .	33
User procedures with inputs. What kind of container? An abbreviation. More procedures. An aside on variable naming. Don't call it X. Writing new operations. Scope of variables. The little person metaphor. Changing the value of a variable. Global and local variables. Indirect assignment.	
4 Predicates . . . . .	54
True or false. Defining your own predicates. Conditional evaluation. Three-input IF. Conditional evaluation another way. About those brackets. Logical connectives. IF as an operation. Stopping a procedure. Improving the quiz program. Reporting success to a superprocedure.	

5.1 Recursion: Combining Method . . . . .	69
Starting small. Generalizing the pattern. What went wrong? The stop rule. Local variables. More examples. Other stop rules.	
5.2 Recursion: Little People Method . . . . .	87
5.3 Recursion: Tracing Method . . . . .	94
Tracing procedure entry and exit. Level and sequence. Displaying the instructions. Instruction stepping. Summary. Improving the tracing tools.	
5.4 Recursion: Leap of Faith Method . . . . .	106
Recursive patterns. The tower of Hanoi. More complicated patterns. Procedure templates.	
6 Turtle Geometry . . . . .	121
A review, or a brief introduction. Local vs. global descriptions. The turtle's state. Symmetry. Fractals. Further reading.	
7 Recursive Operations . . . . .	142
A simple substitution cipher. More templates. Numerical operations. Pig Latin. A word about tail recursion.	
8 Planning . . . . .	161
Structured programming. Critique of structured programming. A sample project: counting poker hands. An initialization procedure. Planning and debugging. Classifying poker hands. Embellishments. Putting the project in a context.	
9 Debugging . . . . .	182
Using error messages. Invalid data. Other tricky error messages. Incorrect results. Tracing and stepping. Pausing. Final words of wisdom.	
10 Iteration, Control Structures, and Extensibility . . . . .	199
Recursion as iteration. Numeric iteration. Logo: an extensible language. No perfect control structures. Iteration over a list. Mapping. Mapping as a metaphor. List reduction. Filters. Mapping over trees. Iteration and tail recursion. Multiple-input mapping. Multiple inputs to other iteration tools. The evaluation environment bug.	

11 Nonlocal Exit . . . . .	228
Quiz program revisited. Nonlocal exit and modularity. Stopping an iteration tool. Catching errors. Ending it all.	
12 Program as Data . . . . .	237
TEXT and DEFINE. Automated definition. A single-keystroke program generator. Procedure cross-reference listings.	
13 Demons . . . . .	247
Multiple dynamic turtles. Collisions. Stopping an animation. The exploratory style. Bouncing off the walls. Bang bang, shoot shoot. The clock demon. Demons without graphics.	
14 Property Lists . . . . .	262
Naming properties. Writing property list procedures in Logo. Property lists aren't variables. How language designers earn their pay. Fast replacement. Defaults. Using property lists as arrays. An example: family trees.	
15 Data Files . . . . .	273
Reader and writer. End of file. Case sensitivity. Dribble files. A text formatter. Page geometry. The program. Improving the formatter. Exhortation.	
Appendix A Versions of Logo . . . . .	288
Appendix B Hints for Teachers . . . . .	304
General Index . . . . .	311
Index of Defined Procedures . . . . .	317