Goldenberg, P. (2020). Reflections on "Exploring Language With Logo."
Contemporary Issues in Technology and Teacher Education, 20(3), 574-576.

Reflections on “Exploring Language With Logo”

Paul Goldenberg
Education Development Center

Read related papers:
Introductory Article: Integrating Language Arts and Computational Thinking: A Reflection on the Importance of Gossip
Gossip and Other Life Sentences: A Simple Grammar
Exploring Language: The Journey from Logo to Snap!

Sometime in 1984, Wally Feurzeig suggested that we write a book together on Logo. I liked the idea and proposed that it be the first “Logo” book that had nothing to do with turtles. I also wanted it to be about experimentation and modeling as a way of learning and proposed linguistics as a topic. While Logo became so closely associated with the graphics that its turtle made easy for children to explore, the name Logo, chosen by Wally, is from the Greek *λόγος* (logos)meaning *word* or *reason*(ing), and that’s where my personal interest lay.

Our original preferred title for the book was *Exploring Language*. Why name the tool? After all, nobody would title a hi-tech-free exploration-based introduction to linguistics *Exploring Language With Pencil and Paper*. But MIT Press was betting on interest generated by the fervor-of-the-day around programming and by interest in a particular philosophy of education that Logo world represented. The resulting book — enormous fun to write — launched what became a series of such explorations (of mathematics, art, and computer science) published by MIT Press.

The book was written at a time when people who were attracted by any of those ideas — language, exploration-based education, programming, something to do with personally owned computers (which were, at that time, still very new) — may well not have programmed in any language. The first chapter, therefore, had to begin with very basic programming structures. After all, to explore *with Logo*, the reader had to know how. But a book about language had the advantage of letting us treat *computer* language (in this case, Logo) as fair game along with English (and, later in the book, *many* other human languages), so even in the introductory pages we could remain on topic.

While the book started with a very simple grammar of short, individual sentences — introductory both to modeling language and to programming — much of the fun lay in where that led: models of bigger structures (poems, stories) with their syntax, semantics, and style, and models of smaller structures within a word (spelling, form, and sound). A few years back, Cindy Carter (who is acknowledged in the book but contributed so much that she really should have been listed as co-author with Wally and me) ran a Computer Science elective in which sixth-graders, using Snap*!*, wrote and then iteratively improved a program to give the correct spelling of any English noun (including all the special cases like *fly* and all the irregular cases, like *child* or *fish* or *ox*). Then, using the programming knowledge they'd gained, they happily showed their Spanish teacher a program they created to take any Spanish verb (including exceptions) in its infinitive form and a pronoun and output the correct conjugation. They were on their way to writing gossip in Spanish!

For me, in writing the book, the greatest joy was in discovering some of the rules that govern English spelling well enough to make up new plausible English words, like *throist* (but not *prowb* even though it is pronounceable, and certainly not *shrimk*). Some rules of English phonology can be found in texts I've found, but finding surprises beyond those rules was a thrill. In general, that's the beauty of learning through experiment and exploration: It's possible to discover things that may not (or may) be true advances for the academic field but are genuinely your own.

The chapter reproduced here is an adaptation, not a copy of the first chapter in the original publication. That is mostly because the world has changed since the book was written. At that time, Logo was unusual in providing powerful features with a simple syntax designed for beginners and was essentially the only accessible option for playing with linguistics.

The two key requirements for exploring linguistics are (a) the ability to create blocks of code that can output results to other blocks of code — confusingly, the cultures surrounding each computer language use different terms (functions, methods, and reporters) for such blocks of code; and (b) the easy ability to create and manipulate nested lists of words. That still leaves out many languages, but these days, more options exist than then: some dialects/extensions of Logo (e.g., NetLogo, StarLogo), descendants of Logo (e.g., Snap*!*), and some unrelated languages (e.g., Python or JavaScript).

If you are comfortable in any such language, you can read the Logo activities and program in the language of your choice. Therefore, for this revision, we omitted technical details that are Logo-specific (or instructions on how to save a file to a disk!) that we had to include in 1987 — the world has moved on — but we still preserve the "technical" computer language comments that made connections with natural language.

Enjoy exploring.