The Rabbit Programming Language

Computer programming is often found to be a rather esoteric, labor-intensive endeavor. It tends to be both very difficult to learn, and, once learned, take large amounts of work and huge numbers of lines of code to accomplish simple tasks. A new field of programming languages—functional programming languages—hope to solve the second problem by providing the programmer with powerful tools to perform complex tasks in simple ways, using less code. These abstractions, as they are referred to in computer science, are unique to functional programming and are found to be incredibly powerful tools for writing simpler, shorter, more concise, and more modular code. Functional programming languages are also easier to debug, think about, and optimize. Unfortunately, they are currently too specialized and fail to solve the first problem of learning difficulty. The field of functional programming languages currently lacks a scripting language, an easy-to-use, broadly applicable language built for the average, everyday programmer. Instead, existing functional programming languages focus on mathematical purity and execution speed over general applicability and ease-of-use. This paper presents the new Rabbit programming language, written by the author, as the solution to these problems. Rabbit is a functional scripting language, built to combine the power of functional programming with the ease-of-use and generality of scripting languages. It is thought that this will make Rabbit useful for a variety of purposes, including the creation of Domain-Specific Languages to solve problems in robotics, engineering, mathematics, and applied science.