TwigScript Project Proposal

Camden Voigt | cvoigt@pdx.edu | https://github.com/camdenvoigt/TwigScript

Subject Area: Programming Language Creation

Description

I am really interested in programming languages, how they work, and how they are made and so I want to learn more by making my own. Rust seems like a great language to do this. In the previous programming language courses I have taken I have gotten to make small "Toy" languages, but not something fully fleshed out.

With this project I want to make an interpreted programming language from scratch called TwigScript. I plan to create the lexer, parser, and interpreter from scratch. So that I can learn in detail about all the different aspects. If I find myself getting stuck I can also switch to generating a lexer and parser using other crates, such as (lalrpop, or pest).

For the grammar I don't want to spend a lot of time making my own as that's not really what I want to get out of the project. So I plan on using this grammar from EMCAscript (https://tc39.es/ecma262/multipage/ecmascript-language-lexical-grammar.html#sec-ecmascript-language-lexical-grammar) as the basis for my project. I don't necessarily plan on following all the rules exactly and certainly won't follow all of the semantics perfectly, but it gives a good place to start and refer back to if I am stuck.

Project Goals

- Create a lexer for TwigScript
- Create a parser for TwigScript
- Create a interpreter for TwigScript
- Program Features available by end of project
    - Literals
        - Numerics
        - Strings
        - Objects
        - Arrays
    - Control Flow
        - If
        - Loops
        - Pattern Matching (This isn't in EMCA)
    - Functions
    - Classes

- Write some basic example programs that work in TwigScript

Concerns

I have two main concerns. First, is scope. Programming Languages need a lot of features to really feel usable. I am worried that I won't have enough fully implemented by the end of the term for the language to feel full enough. Second, I am using a grammar that was designed for specific semantics. Since I may not follow these exactly, I am worried the language won't really make sense fully.