

Apreendi a estruturar uma gramática usando o JavaCC e a compreender melhor o funcionamento de compiladores, principalmente as fases de análise léxica e sintática. Foi interessante observar como os tokens são definidos e como a ordem das regras pode afetar diretamente o reconhecimento de comandos.

As maiores dificuldades estiveram na implementação e ajuste das expressões, principalmente na inclusão dos operadores aritméticos (+, -, \*, /), dentro das regras de `Expression()`, pois inicialmente a linguagem só reconhecia expressões lógicas simples. Também foi necessário criar uma estrutura com precedência correta dos operadores e adicionar suporte a chamadas de função dentro de expressões, o que exigiu atenção com ambiguidade nos tokens como `<ID>` e chamadas com `()`.

Outro ponto trabalhoso foi garantir que todos os comandos da linguagem (como `for`, `while`, `foreach`, `break`, `continue`, `if`, `else`, `return`, etc.) fossem testados em conjunto, sem causar conflitos na gramática. Porém deu para montar um interpretador sintático robusto e funcional, cobrindo todos os requisitos propostos.