

Universidade Federal de Pernambuco
Centro de Informática

Gabriel Henrique Daniel da Silva

Ferramenta voltada para o aprendizado do funcionamento de parsers

Orientador: Leopoldo Motta Teixeira

Recife, agosto de 2018.

Resumo

Um compilador pode ser definido como um programa que traduz programas recebidos como entrada em uma linguagem fonte para um equivalente em uma determinada linguagem alvo. Os analisadores sintáticos ou parsers são partes essenciais no desenvolvimento de um compilador. Apesar de existirem diversas ferramentas de geração automática de parsers como Antr, Yacc, Cup entre outras, existe uma carência de ferramentas voltadas para o ensino. Portanto buscamos apresentar uma ferramenta que auxilie na compreensão das técnicas utilizadas na criação de um analisador sintático.

Introdução

Um compilador pode ser definido como um programa que traduz programas recebidos como entrada em uma linguagem fonte para um equivalente em uma determinada linguagem alvo. [1] Tradicionalmente o processo de compilação pode ser dividido em duas fases: Frontend (Análise) e Backend (Síntese). [2]

No frontend, focamos em entender o programa na linguagem fonte, verificando se o mesmo é bem formado. Já no backend nos preocupamos em construir o programa na linguagem alvo. [2]

Ao observarmos os componentes dentro do frontend, percebemos que existem três fases de análise, sendo elas: léxica, sintática e semântica. [2]

A primeira delas, chamada análise léxica, tem como função a leitura de caracteres de entradas e produção de tokens que serão usados posteriormente pelo analisador sintático, também conhecido como parser. Por sua vez, o parser utiliza a cadeia de tokens gerados pelo analisador léxico e verifica se a mesma pode ser gerada pela gramática advinda da linguagem fonte. [1] Já a análise semântica busca detectar problemas além da sintaxe, como os relacionados ao contexto, como erros de tipo, variáveis não declaradas anteriormente etc.

No contexto de aprendizagem de uma disciplina como compiladores, existe uma certa dificuldade de associação entre teoria e prática no entendimento. Em virtude disso, surgiram diversas ferramentas que visam facilitar a aprendizagem entre elas o Verto, [4] que é focado em técnicas e ferramentas presentes no backend de um compilador.

Analisando a literatura, nota-se que apesar de existirem diversas ferramentas voltadas a geração automática de parsers como Antlr [5], Yacc [6], Cup [7] e vários outros, mas ainda sim, existe uma certa carência de ferramentas que facilitem o entendimento das técnicas e algoritmos que estão sendo utilizadas no funcionamento dos mesmos.

Portanto, buscamos apresentar uma ferramenta que auxilie na compreensão do funcionamento das técnicas utilizadas na criação de analisadores sintáticos.

Dessa forma, ao invés de focarmos em trabalhar com parsers voltados para produtividade e performance, iremos focar em uma abordagem mais didática, buscando mostrar o passo a passo do funcionamento e se preocupando com a ordem e a organização dos resultados, tentando deixar o mais claro possível como que os mesmos foram obtidos.

Objetivos

Este trabalho tem como objetivo desenvolver uma ferramenta que facilite o aprendizado de técnicas utilizadas na construção de compiladores, mais especificamente, na fase de análise sintática, situada no frontend de um compilador.

Com essa ferramenta, poderemos observar o passo a passo do que está acontecendo durante a execução do parser sobre uma determinada gramática, de forma a ajudar no entendimento por parte do estudante em vez de somente gerar a sua respectiva saída.

Referências Bibliográficas

- [1] AHO, Alfred V. **Compilers: principles, techniques and tools (for Anna University), 2/e**. Pearson Education India, 2003.
- [2] COOPER, Keith; TORCZON, Linda. **Engineering a compiler**. Elsevier, 2011.
- [3] FOLEISS, Juliano Henrique et al. Scc: Um compilador C como ferramenta de ensino de compiladores. In: **WEAC2009-Workshop Educação em Arquitetura de Computadores**. 2009. p. 15-22.
- [4] SCHEIDER, Carlos; PASSERINO, Liliana Maria; DE OLIVEIRA, Ricardo Ferreira. Compilador Educativo VERTO: ambiente para aprendizagem de compiladores. **RENOTE**, v. 3, n. 2, 2005.
- [5] PARR, Terence J.. ; QUONG, Russell W.. . ANTLR: A predicated-LL (k) parser generator. **Software: Practice and Experience**, v. 25, n. 7, p. 789-810, 1995.
- [6] DONNELLY, Charles; STALLMAN, Richard. Bison. The YACC-compatible parser generator. 2000.
- [7] HUDSON, Scott. Cup parser generator for java. [http://www. cs. princeton. edu/~ appel/modern/java/CUP](http://www.cs.princeton.edu/~appel/modern/java/CUP), 1999.

Possíveis Avaliadores

Os possíveis avaliadores do trabalho a ser desenvolvido são: Henrique Rebelo e Gustavo Carvalho.