

Compilador BAGCompile

Breno Klywer Olegário de Moura, Afonso Simão de Gois Neto

December 9, 2024

Abstract

Este trabalho descreve o desenvolvimento de um analisador léxico simples para a linguagem OWL2 (Web Ontology Language) na sintaxe Manchester. O compilador foi implementado em C++, com a primeira etapa focada na análise léxica. Para isso, utilizou-se a ferramenta Flex (Fast Lexical Analyzer Generator), que gera um analisador a partir de um arquivo de especificações baseadas em expressões regulares. O objetivo principal dessa etapa foi a identificação e a divisão do código fonte em tokens representativos da linguagem, facilitando os estágios subsequentes do processo de compilação.

Contents

1	Introdução	2
2	Como Usar a Unidade I	2
2.1	Clonando o Repositório	2
2.2	Executando o Analisador Léxico	2
2.2.1	No Windows	2
2.2.2	No Linux	3
2.3	Observações Importantes	3
3	Implementação da Análise Léxica	3
4	Resultados e Conclusão	3

1 Introdução

O trabalho visa criar um compilador para a linguagem OWL2 na sintaxe Manchester. A primeira etapa consiste no desenvolvimento do analisador léxico, responsável por dividir o código em tokens que representam os componentes da linguagem. O projeto utiliza o Flex, uma ferramenta poderosa e amplamente empregada para análise léxica.

O código-fonte do projeto está disponível no repositório do GitHub: <https://github.com/Brenokly/Compiladores.git>.

2 Como Usar a Unidade I

Esta seção apresenta o passo a passo necessário para utilizar o analisador léxico desenvolvido. O processo inclui a clonagem do repositório e a execução do programa em ambientes Windows ou Linux.

2.1 Clonando o Repositório

Siga os passos abaixo para obter o código-fonte do projeto:

1. Abra o terminal ou prompt de comando.
2. Execute o seguinte comando para clonar o repositório:

```
git clone https://github.com/Brenokly/Compiladores.git
```

Após a clonagem, o projeto estará disponível na pasta **Compiladores**. Abra sua IDE, de preferência VS code, na pasta **Compiladores** que surgiu após a clonagem do repositório.

2.2 Executando o Analisador Léxico

A execução do analisador varia conforme o sistema operacional utilizado. Siga as instruções apropriadas para o seu ambiente:

2.2.1 No Windows

1. Navegue até a pasta do analisador léxico via terminal da IDE:

```
cd Unidade_1_AnalisadorLexico
```

2. Execute o programa com um arquivo de teste como argumento:

```
Lexico.exe Teste1
```

3. O programa exibirá os resultados no terminal, mostrando informações sobre as classes, propriedades e indivíduos encontrados no arquivo de teste.

2.2.2 No Linux

1. Navegue até a pasta do analisador léxico pelo terminal da IDE:

```
cd Compiladores/Unidade_1_AnalisadorLexico
```

2. Execute o seguinte comando:

```
./Lexico.exe Teste1
```

3. Os resultados serão exibidos no terminal, detalhando os tokens encontrados no arquivo de teste.

2.3 Observações Importantes

- Não é necessário compilar o código, pois o arquivo executável **Lexico.exe** já está incluído no repositório.
- Certifique-se de que o arquivo de teste, como **Teste1**, esteja presente na pasta para evitar erros.

3 Implementação da Análise Léxica

O analisador léxico foi desenvolvido para processar a linguagem OWL2 na sintaxe Manchester, utilizando o **Flex**. Abaixo, algumas informações detalhadas sobre a implementação:

- **Flex (Fast Lexical Analyzer Generator):** Ferramenta que gera automaticamente analisadores léxicos a partir de expressões regulares definidas em um arquivo de entrada. Utilizou-se essa abordagem para identificar padrões da linguagem, como palavras-chave, identificadores e símbolos especiais.
- **Identificação de Tokens:** Foram definidos os principais elementos da linguagem OWL2, como **Classes**, **Object Properties**, **Data Properties** e **Individuals**. Cada token é classificado e armazenado, permitindo que os estágios futuros do compilador possam processá-los de forma estruturada.
- **Testes:** Foram criados arquivos de teste, como o **Teste1**, que contém exemplos de entradas válidas para verificar a funcionalidade do analisador.

4 Resultados e Conclusão

A execução do analisador léxico mostrou-se eficiente na identificação e categorização dos tokens na sintaxe Manchester. Esta primeira etapa é fundamental para o desenvolvimento das fases subsequentes do compilador, como análise sintática e semântica.