

Compiladores

...

Laboratório: Implementação do Analisador Léxico

Objetivos

Realizar a implementação do Analisador Léxico

Nesta atividade, os procedimentos de reconhecimento e retorno dos Tokens deve ser implementado

Assim, no final dessa implementação, vocês deverão ter um programa que, recebe um arquivo de texto contendo código fonte (da linguagem que estão definindo), faz o reconhecimento dos lexemas válidos contidos nesse arquivo e retorna os Tokens (que serão utilizados na fase subsequente, a de Análise Sintática)

Componentes dessa implementação - O programa principal

Um programa principal, que será o Compilador

Esse programa irá receber uma cadeia de strings de algum arquivo e chamar o Analisador Léxico (tipicamente chamado de Lexer também em implementações de compiladores e interpretadores)

Componentes dessa implementação - O Analisador Léxico

O analisador léxico, por sua vez, deverá ser capaz de fazer o reconhecimento de cada cadeia presente na cadeia de strings que ele recebeu

Assim, o analisador deve possuir um autômato finito ou conjunto de expressões regulares que fará esse reconhecimento

Ao reconhecer cada cadeia, o analisador deve retornar o Token que representa aquela cadeia

Retorno do Token

O token retornado deve possuir um formato apropriado, que seja capaz de guardar seu tipo e seu valor

Isso será muito útil da fase de análise sintática

A sugestão é que seja utilizada uma classe Token, como visto em aula, que possui como atributos o tipo e valor do token

Exemplo: Tipo = Identificador, Valor = “var1”

Exemplo: Tipo = Operador, Valor = “=”

Exemplo: Tipo = Numérico, Valor = “10”

Detalhes

O analisador léxico vai prover o input para o analisador sintático

Assim, o analisador sintático irá realizar chamadas ao analisador léxico para obter um fluxo de tokens para ser verificado sintaticamente

Nesse sentido, é uma decisão do grupo:

- Ao se ler uma linha de lexemas, todos os tokens daquela linha são retornados ao analisador sintático (na ordem em que foram reconhecidos)

- Ao se ler uma linha, um token é retornado ao analisador sintático, o analisador sintático verifica se este token é válido (segundo as regras sintáticas) e após isso solicita o próximo token ao analisador léxico

Dependendo da estratégia escolhida, o analisador deve possuir um método:

`getTokens()`: retorna todos os tokens presentes em uma linha de código

`getNextToken()`: retorna um token por vez, a partir do último token reconhecido

Observações

Erros devem ser tratados

Se um lexema não pôde ser reconhecido, retorne um erro

Nesse sentido, você pode personalizar seu compilador para exibir algo útil em relação ao erro, como a linha em que o erro ocorreu e qual foi o lexema que não pôde ser reconhecido

Entrega

Duas semanas a partir da data da aula

Os mesmos grupos

Uma documentação que possua:

- O nome de todos os integrantes do grupo!**

- O nome da linguagem que será compilada

- Os tokens definidos

- As expressões regulares para cada token

- Instruções para utilização do compilador até então

Inclua também casos de teste ,como um arquivo com código da sua linguagem que será executado pelo compilador