

Lista Extra - Compiladores

1. (ALex*) O que é um Analisador Léxico?

Resposta: O analisador léxico, também conhecido como scanner, é a primeira fase do processo de compilação. Ele tem a função crucial de ler o código fonte e transformá-lo em uma sequência de tokens.

- Ele analisa o código fonte, caractere por caractere, procurando por padrões que definem as unidades básicas da linguagem, os tokens.
- Um token é um elemento significativo da linguagem, como um identificador, palavra-chave, operador ou literal.

2. (ASin*) O que é um Analisador Sintático?

Resposta: O analisador sintático, também conhecido como parser, é a segunda etapa crucial do processo de compilação. Ele recebe a sequência de tokens gerada pelo analisador léxico e verifica se a estrutura do código fonte está correta, seguindo as regras gramaticais da linguagem de programação.

- Ele recebe os tokens do analisador léxico e os organiza em uma estrutura sólida, seguindo as regras de construção (gramática) da linguagem.
- Se a estrutura não for válida, o analisador sintático sinaliza um erro, como uma parede sem teto ou uma porta sem maçaneta.

3. (ASem) O que é um Analisador Semântico?

Resposta: O analisador semântico é a terceira fase crucial do processo de compilação, e entra em cena após o analisador sintático ter verificado a estrutura gramatical do código. Sua missão principal é analisar o significado do código fonte, garantindo que ele faça sentido e siga as regras da linguagem de programação.

- Ele recebe a árvore sintática, construída pelo analisador sintático, e a analisa profundamente, buscando erros de significado e inconsistências.
- Ele não se importa com a gramática, mas sim com o significado das expressões, declarações e tipos de dados.

4. (GLC*) Uma determinada Gramática Livre de Contexto possui as seguintes regras de transformação:

$S \rightarrow aSa$

$S \rightarrow \epsilon$

Qual o padrão de entrada que ele reconhece?

Dica: Construa a Árvore de Derivação para facilitar a compreensão do padrão.

Resposta: Essa gramática reconhece números pares de a's.

5. (GLC*) Uma determinada Gramática Livre de Contexto possui as seguintes regras de transformação:

$S \rightarrow aSa$

$S \rightarrow bSb$

$S \rightarrow \varepsilon$

Qual o padrão de entrada que ele reconhece?

Dica: Construa a Árvore de Derivação para facilitar a compreensão do padrão.

Resposta: Essa gramática reconhece palavras que são palíndromos.

6. (GLC) Uma determinada Gramática Livre de Contexto possui as seguintes regras de transformação:

$S \rightarrow 0 \mid 2 \mid 4 \mid 6 \mid 8 \mid DS$

$D \rightarrow 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$

Qual o padrão de entrada que ele reconhece?

Dica: Construa a Árvore de Derivação para facilitar a compreensão do padrão.

Resposta: Essa gramática reconhece números pares.

7. (BNF*) Considere a seguinte Gramática em BNF:

$\langle \text{expressão} \rangle ::= \langle \text{fator} \rangle \mid \langle \text{expressão} \rangle '+' \langle \text{fator} \rangle \mid \langle \text{expressão} \rangle '-' \langle \text{fator} \rangle$

$\langle \text{fator} \rangle ::= \langle \text{número} \rangle \mid '(' \langle \text{expressão} \rangle ')'$

$\langle \text{número} \rangle ::= '0' \mid '1' \mid '2' \mid '3' \mid '4' \mid '5' \mid '6' \mid '7' \mid '8' \mid '9'$

Qual o padrão de entrada que ele reconhece?

Resposta: Essa gramática reconhece expressões aritméticas simples que envolvam números naturais, suportando operações de adição, subtração e parênteses.

8. (BNF) Na gramática acima, quais regras adicionais seriam necessárias para incorporar as operações de multiplicação e divisão?

Resposta: O conjunto de regras incorporando as duas operações matemáticas adicionais seria essa:

$\langle \text{expressão} \rangle ::= \langle \text{termo} \rangle \mid \langle \text{expressão} \rangle '+' \langle \text{termo} \rangle \mid \langle \text{expressão} \rangle '-' \langle \text{termo} \rangle$

$\langle \text{termo} \rangle ::= \langle \text{fator} \rangle \mid \langle \text{termo} \rangle '*' \langle \text{fator} \rangle \mid \langle \text{termo} \rangle '/' \langle \text{fator} \rangle$

`<fator> ::= <número> | '(' <expressão> ')'`

`<número> ::= '0' | '1' | '2' | '3' | '4' | '5' | '6' | '7' | '8' | '9'`

Em caso de dúvidas me procurar pessoalmente, enviar mensagem por Slack ou enviar um e-mail para: rafael.matsuyama@prof.inteli.edu.br

Prof. Rafael Matsuyama