Lógica da Computação - 2019/2

Aula 08/T04 - 28/Aug/2019

Raul Ikeda - rauligs@insper.edu.br

Objetivos

1. Linguagens Livres de Contexto.
2. Autômato de Pilha.

Formalização

Como visto na aula anterior, a linguagem do compilador agora não é mais regular. Dentro da Hierarquia de Chomsky, ela passa a ser considerada em um grupo mais abrangente denominado Linguagens Livres de Contexto. Formalmente uma CFL (Context Free Language) é composta por uma gramática semelhante a utilizada na regular. Relembrando:

$$G = (V, \Sigma, P, A)$$

Onde:

- V é o vocabulário da linguagem, onde $V = N \cup \Sigma$ e N é o conjunto de símbolos não terminais.
- Σ é o alfabeto da linguagem
- P é o conjunto das regras de produção
- A é o símbolo não terminal inicial da gramática

E uma gramática livre de contexto contém regras de produção da seguinte forma: $\alpha \to \beta$, sendo $\alpha \in N$ e $\beta \in V^*$

Exemplo: $L = \{(^n a)^n | n \geq 0\}$

1. Esboçar um autômato para a linguagem acima

Aplicações práticas

- Linguagens de programação
- Linguagens de marcação (HTML, XML, etc)
- JSON
- etc

E as Linguagens Naturais?

- Chomsky, 1957: "English is not a regular language". As for context-free languages, "I do not know whether or not English is itself literally outside the range of such analyses". https://en.wikipedia.org/wiki/Syntactic_Structures
- Swiss-German example: https://www.eecs.harvard.edu/shieber/Biblio/Papers/shieber85.pdf

Formas de representar uma CFL

- Linguagem
- Gramática
- Diagrama Sintático
- BNF (EBNF, ABNF)
- Autômato (???)

PDA - Pushdown Automata

Para realizar o reconhecimento de uma CFL é necessário utilizar um Autômato de Pilha. Consiste em um autômato similar ao instrumento usado para as linguagens regulares, porém com uma pilha de suporte. A pilha é uma estrutura de dados que funciona como uma memória auxiliar, permitindo expandir a capacidade de processamento de um autômato finito.

Formalmente, PDA é uma sétupla:

$$M = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$$

Onde:

- Q é um conjunto finito de estados
- Σ é um alfabeto finito e não vazio de entrada
- Γ é um alfabeto finito e não vazio de Pilha
- δ é uma função de transição $Q \times \Sigma \times \Gamma \rightarrow Q \times \Gamma^*$
- q_0 é o estado inicial de M, $q_0 \in Q$
- Z_0 é o símbolo inicial da pilha, $Z_0 \in \Gamma$
- F é o conjunto de estados finais de M, $F \subseteq Q$

Basicamente, o autômato trabalha com funções de transição que mapeiam o estado atual, o símbolo lido na entrada e o símbolo **removido** do topo da pilha (*pop*). Essa função mapeia em um novo estado e uma inserção na pilha (*push*)

Exemplo usando a linguagem da questão 1:

2. O autômato de pilha acima é determinístico ou não determinístico?

Diferentes funções de transição...

- PDA determinístico sem transições em vazio: $\delta : Q \times \Sigma \times \Gamma \rightarrow Q \times \Gamma^*$
- PDA determinístico com transições em vazio: $\delta : Q \times (\Sigma \cup \lambda) \times \Gamma \rightarrow Q \times \Gamma^*$
- PDA não determinístico sem transições em vazio: $\delta : Q \times \Sigma \times \Gamma \rightarrow 2^{Q \times \Gamma^*}$
- PDA não determinístico com transições em vazio: $\delta : Q \times (\Sigma \cup \lambda) \times \Gamma \rightarrow 2^{Q \times \Gamma^*}$

3. Mas afinal, como o autômato valida uma cadeia?

`>> Ver Teoremas 4.11 e 4.12 Marcus et al.`

4. Um PDA consegue validar uma Linguagem Regular?

`>> Ver Teoremas 4.15 e 4.16 Marcus et al.`

5. Exercício: Construir um autômato de pilha para validar:

a. $L = \{a^n b^n | n \geq 1\}$
b. $L = \{wcw^R | w \in \{a, b\}^*\}$
c. $L = \{ww^R | w \in \{a, b\}^*\}$

`>> Sugestão: usar o aplicativo java JFLAP para simular.`

E a Gramática Livre de Contexto afinal?

Teorema: "Seja G uma gramática livre de contexto. Então é possível definir um autômato de pilha não determinístico M, com critério de aceitação baseado em pilha vazia de modo que $V(M) = L(G)$". Ver Marcus et al. Pag. 346

Algoritmo:

1. $Q \leftarrow \{q\}$
2. $\Gamma \leftarrow V$
3. $F \leftarrow \emptyset$
4. Funções de transição:

- $\delta \leftarrow \emptyset$
- $\delta(q, \lambda, A) = \{(q, \gamma) | A \rightarrow \gamma \in P\}, \forall A \in N, \gamma \in V^*$
- $\delta(q, \sigma, \sigma) = \{(q, \lambda)\}, \forall \sigma \in \Sigma$

Exemplo: $P : E \rightarrow \lambda | (E)$

Teorema: "Seja M um autômato de pilha com critério de aceitação de pilha vazia. Então é possível definir uma gramática livre de contexto G, de modo que $L(G) = V(M)$". Ver Marcus et al. Pag. 353

Lista de Exercícios

Ramos et al Cap. 4.17: Exercícios 1, 4, 7, 9, 11 e 13.