Autômatos de Pilha - Parte 3

Victor Carvalho
Thais Pires
Lara Santana
Lucas Felipe

Curso de Engenharia da Computação – Disciplina: Linguagens Formais
Escola de Engenharia Elétrica, Mecânica e Computação
Universidade Federal de Goiás

10 de novembro de 2017

Sumário

Funcionamento dos Autômatos de Pilha por Estado Final

O autômato de pilha por estado final (AP_{ef}) é o autômato que, após ler uma cadeia válida, atinge um estado de aceitação pertencente ao grupo de estados finais. Este modo utiliza a memória interna (estados) do autômato.

Funcionamento dos Autômatos de Pilha por Estado Final

Definição Formal (Relembrando)

Um autômato de pilha M é formalmente definido por uma 6-tupla:

$$M = (\mathcal{Q},\ \Sigma,\ \Gamma,\ \Delta,\ q_0,\ F)$$

Onde:

- $\mathcal{Q}$ é um conjunto finito de estados;
- Σ é um conjunto finito de símbolos, denominado alfabeto de entrada;
- Γ é um conjunto finito de símbolos, denominado alfabeto da pilha;
- Δ é a relação de transição;
- q_0 é o estado inicial;
- F é o conjunto de estados finais (ou de aceitação).

Funcionamento dos Autômatos de Pilha por Estado Final

Definição - Linguagem

Linguagem $\mathcal{L}$ reconhecida pelo AP_{ef}:

$$\mathcal{L}(M) = \{\omega \in \Sigma^* \mid (q_0, \omega, \mathcal{Z}) \vdash_M^* (f, \lambda, \gamma)\ com\ f \in F\ e\ \gamma \in \Gamma^*\}$$

Funcionamento dos Autômatos de Pilha por Estado Final

Exemplo

Tomamos o AP_{ef} M descrito por

$$M = (\mathcal{Q},\ \Sigma,\ \Gamma,\ \delta,\ p,\ ,Z,\ F),$$

Onde

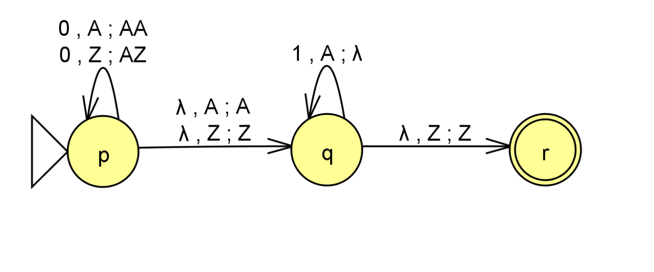
- $Q = \{p, q, r\}$
- $\Sigma = \{0, 1\}$
- $\Gamma = \{A, Z\}$
- $F = \{r\}$

Funcionamento dos Autômatos de Pilha por Estado Final

Exemplo - Continuação

δ consiste nas seis instruções seguintes:
$(p, 0, Z, p, AZ)$, $(p, 0, A, p, AA)$, (p, λ, Z, q, Z), (p, λ, A, q, A), $(q, 1, A, q, \lambda)$, (q, λ, Z, r, Z).

Representação do autômato M:



Funcionamento dos Autômatos de Pilha por Estado Final

Exemplo - Continuação

O autômato M descrito reconhece a linguagem $\{0^n1^n \mid n \geq 0\}$

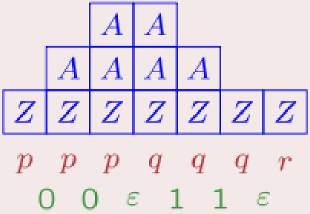
Tendo 0011 como cadeia de entrada ω, há várias computações possíveis:

1. $(p, 0011, Z) \vdash (q, 0011, Z) \vdash (r, 0011, Z)$. O estado final é o de aceitação, mas a entrada não é aceita, pois a cadeia não foi lida completamente;
2. $(p, 0011, Z) \vdash (p, 011, AZ) \vdash (q, 011, AZ)$. Não há mais estados possíveis;
3. $(p, 0011, Z) \vdash (p, 011, AZ) \vdash (p, 11, AAZ) \vdash (q, 11, AAZ) \vdash (q, 1, AZ) \vdash (q, \lambda, Z) \vdash (r, \lambda, Z)$. Computação de aceitação: termina em um estado de aceitação e a cadeia é lida totalmente.

Funcionamento dos Autômatos de Pilha por Estado Final

Exemplo - Continuação

$\lambda = \epsilon$



$$
\begin{array}{llrrl}
(p, & 0011, & Z) & \vdash \\
(p, & 011, & AZ) & \vdash \\
(p, & 11, & AAZ) & \vdash \\
(q, & 11, & AAZ) & \vdash \\
(q, & 1, & AZ) & \vdash \\
(q, & \varepsilon, & Z) & \vdash \\
(r, & \varepsilon, & Z) &
\end{array}
$$

Sumário

Funcionamento dos Autômatos de Pilha por Pilha Vazia

O autômato de pilha por pilha vazia (AP_{pv}) é o autômato que, após ler uma cadeia válida, esvazia sua pilha. Nesse caso ele usa sua memória externa (pilha) e o conjunto de estados finais é irrelevante.

Funcionamento dos Autômatos de Pilha por Pilha Vazia

Definição - Linguagem

Linguagem $\mathcal{N}$ reconhecida pelo AP_{pv}:

$$\mathcal{N}(M) = \{\omega \in \Sigma^* \mid (q_0,\ \omega,\ \mathcal{Z}) \vdash_M^* (f,\ \epsilon,\ \epsilon)\ com\ q \in Q\}$$

Funcionamento dos Autômatos de Pilha por Pilha Vazia

Exemplo

Tomamos o AP_{pv} M descrito por

$$M = (\mathcal{Q},\ \Sigma,\ \Gamma,\ \delta,\ p,\ Z),$$

Onde

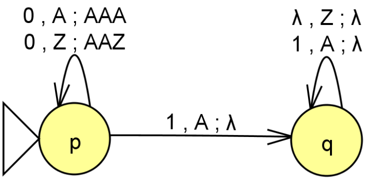
- $Q = \{p, q\}$
- $\Sigma = \{0, 1\}$
- $\Gamma = \{A, Z\}$

Funcionamento dos Autômatos de Pilha por Pilha Vazia

Exemplo - Continuação

δ consiste nas cinco instruções seguintes:
$(p, 0, Z, p, AAZ)$, $(p, 0, A, p, AAA)$, $(p, 1, A, q, \lambda)$,
$(q, 1, A, q, \lambda)$, $(q, \lambda, Z, q, \lambda)$.

Representação do autômato M:



Funcionamento dos Autômatos de Pilha por Pilha Vazia

Exemplo - Continuação

O autômato M descrito reconhece a linguagem $\{0^n 1^{2n} \mid n > 0\}$

Tendo 001111 como cadeia de entrada ω, há apenas uma computação:

- $(p, 001111, Z) \vdash (p, 01111, AAZ) \vdash (p, 1111, AAAAZ) \vdash (q, 111, AAAZ) \vdash (q, 11, AAZ) \vdash (q, 1, AZ) \vdash (q, \lambda, Z) \vdash (q, \lambda, \lambda)$. Computação de aceitação: termina quando a pilha está vazia e a cadeia foi totalmente lida.

Sumário

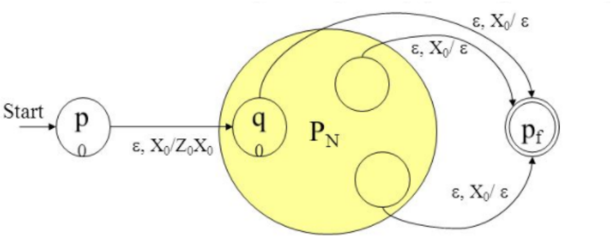
Equivalência entre AP_{ef} e AP_{pv}

Conversão de Pilha Vazia para Estado Final

Temos o $P_{pv} = (\mathcal{Q},\ \Sigma,\ \Gamma,\ \Delta,\ q_0,\ F)$, podemos converter em um AP_{ef} com as seguintes instruções:

1. Adicionamos o símbolo de entrada X_0 no AP_{ef}.
   X_0 funcionará como o símbolo de pilha vazia para o AP_{ef}.
   Portanto, se o AP_{ef} ler X_0, então ele saberá que o AP_{pv} irá esvaziar a pilha naquele simbolo de entrada;
2. Adicionamos um novo estado inicial p_0.
   O novo estado irá empilhar Z_0, símbolo de pilha vazia do AP_{pv}, na pilha do AP_{ef}. Além disso, p_0 irá mudar para o estado inicial q_0 do AP_{pv}. $\Delta(p_0, \epsilon, X_0) = (q_0, Z_0X_0)$;
3. Adicionamos um estado final p_f ao AP_{ef}.
   Todos os estados do AP_{pv} que esvaziarem a pilha, irão mudar para o novo estado final p_f.

Equivalência entre AP_{ef} e AP_{pv}



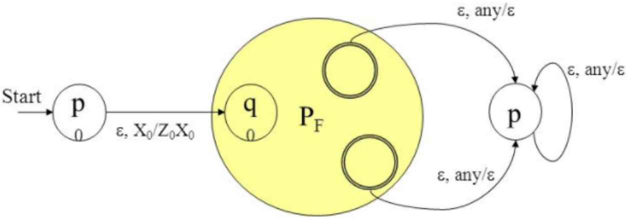
Equivalência entre AP_{ef} e AP_{pv}

Conversão de Estado Final para Pilha Vazia

Temos o $P_{ef} = (\mathcal{Q},\ \Sigma,\ \Gamma,\ \Delta,\ q_0,\ F)$, podemos converter em um AP_{pv} com as seguintes instruções:

1. Adicionamos um novo estado p para o AP_{ef};
2. Para todo estado final do AP_{ef}, acrescentamos transições vazias para o novo estado p, considerando qualquer simbolo do alfabeto do AP_{ef}.
   $\Delta_{final}(q, \epsilon, X) = \{(p, \epsilon)\},\ \forall X \in \Gamma\ sendo\ q \in F$;
3. Adicionamos um estado inicial p_0 para o AP_{pv}.
   Neste estado, adicionamos a transição
   $\Delta(p_0, \epsilon, X_0) = (q_0, Z_0X_0)$, sendo X_0 o marcador de pilha vazia para impedir que AP_{ef} esvazie sua pilha sem reconhecer a cadeia de entrada.

Equivalência entre AP_{ef} e AP_{pv}



Sumário

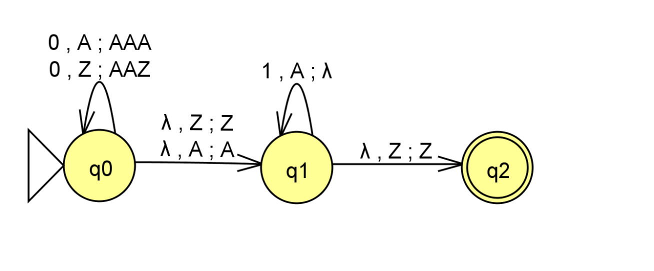
Questionário

Questão 01

Faça um autômato por estado final que reconheça a linguagem $\{0^n 1^{2n} \mid n \geq 0\}$.

Questionário

Questão 01 – Resposta Esperada



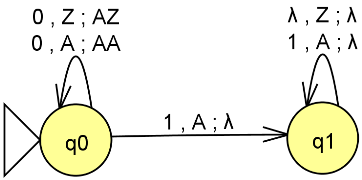
Questionário

Questão 02

Faça um autômato por pilha vazia que reconheça a linguagem $\{0^n 1^n \mid n > 0\}$.

Questionário

Questão 02 – Resposta Esperada



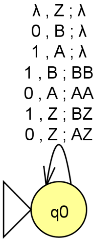
Questionário

Questão 03

Encontre um AP_{pv} M que reconheça o conjunto $\mathcal{L} = \{\omega \mid \omega \in \{0,1\}^* \ e \ \omega \text{ tem igual número de 0's e 1's}\}$, por pilha vazia.

Questionário

Questão 03 – Resposta Esperada



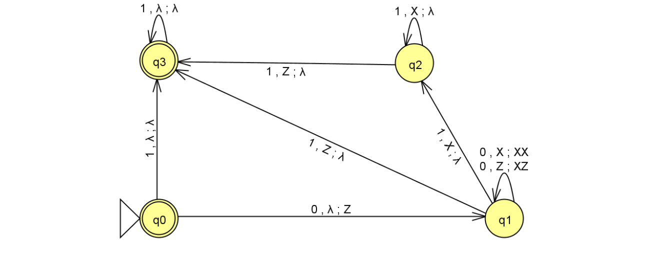
Questionário

Questão 04

Faça um autômato por estado final que reconheça a linguagem $\{0^m 1^n \mid n \geq m \geq 0\}$.

Questionário

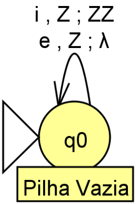
Questão 04 – Resposta Esperada



Questionário

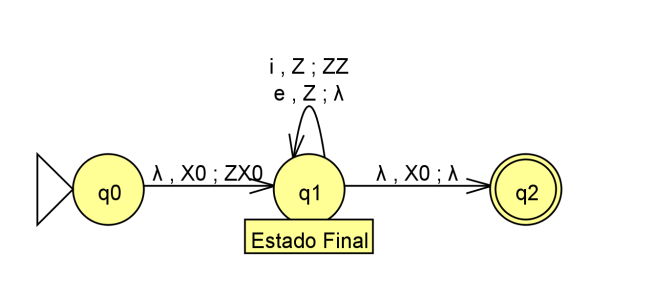
Questão 05

Converta o AP_{pv} que processa sequências de i e e, detectando sequências inválidas (sequências que têm mais es que is num prefixo), para um AP_{ef}.



Questionário

Questão 05 – Resposta Esperada



Sumário

Referências Bibliográficas

- Michael Sipser. Introdução à Teoria da Computação. Tradução brasileira de "Introduction to the Theory of Computation"(PWS Publishing Company, 2nd edition, 2005), por Ruy de Queiroz, revisão Newton Vieira, Cengarle Learning, 2007 ISBN 978-85-221-0499-4.
- ean-Michel Autebert, Jean Berstel, Luc Boasson, Context-Free Languages and Push-Down Automata, in: G. Rozenberg, A. Salomaa (eds.), Handbook of Formal Languages, Vol. 1, Springer-Verlag, 1997, 111-174.
- Linguagens Formais, J. L. Range - 1999