



LINGUAGENS FORMAIS E AUTÓMATOS / COMPILADORES

AUTÓMATOS DE PILHA (AP)

Artur Pereira / Miguel Oliveira e Silva <{artur, mos}@ua.pt>

DETI, Universidade de Aveiro

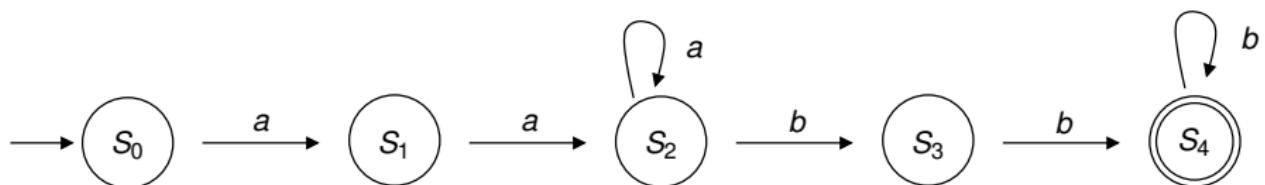
AUTÓMATO DE PILHA: INTRODUÇÃO

- Considere as linguagens seguintes, definidas sobre o alfabeto $A = \{a, b, c\}$, e tente projetar autômatos finitos que as reconheçam.

$$L_1 = \{a^n b^m : n, m \geq 2\}$$

$$L_2 = \{a^n b^m : n, m \geq 2 \wedge n = m\}$$

- Solução para L_1



- Não há solução para L_2 , porque não é uma linguagem regular

AUTÓMATO DE PILHA: INTRODUÇÃO

- O mesmo acontece com as linguagens seguintes, também definidas sobre o alfabeto $A = \{a, b, c\}$, também porque não são regulares

$$L_1 = \{\omega \in A^* : \#(a, \omega) = \#(b, \omega)\}$$

$$L_2 = \{\omega \in L_1 : |\omega| \geq 3 \wedge \omega_1 = a \wedge \omega_2 = b\}$$

$$L_3 = \{a^n b^m : n \geq m + 3\}$$

$$L_4 = \{\omega c \omega^R : \omega \in (\{a, b\})^*\}$$

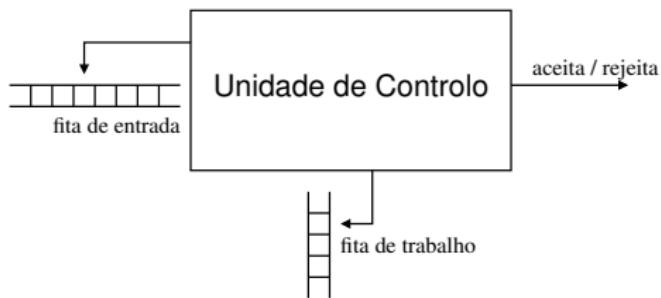
$$L_5 = \{\omega \omega^R : \omega \in (\{a, b, c\})^*\}$$

$$L_6 = \{\omega \in A^* : \#(ab, \omega) = \#(ba, \omega)\}$$

- Logo, não podem ser reconhecidas por um autómato finito.
- Mas a inclusão de uma fita de trabalho (pilha) no mecanismo reconhecedor permite fazê-lo.

AUTÓMATO DE PILHA: DEFINIÇÃO

- Um **autómato de pilha** é um mecanismo reconhecedor das palavras de uma linguagem, baseado na noção de estado e na de transição entre estados, usando uma fita de trabalho (pilha)



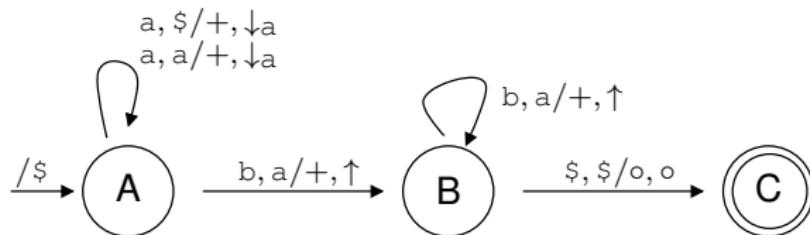
- número finito de estados
- fita de entrada: só de leitura, acesso sequencial
- fita de trabalho: funcionamento em pilha

AUTÔMATO DE PILHA: DEFINIÇÃO

- As transições dependem:
 - do estado corrente
 - do próximo símbolo da palavra de entrada
 - do símbolo no topo da pilha
- O resultado de uma transição é:
 - o próximo estado
 - a operação sobre a entrada
 - a operação sobre a pilha
- Operações sobre a entrada:
 - consumir o símbolo na entrada, pondo a descoberto o seguinte
 - não fazer nada
- Operações possíveis sobre a pilha:
 - inserção de um elemento de um alfabeto próprio (*push*)
 - remoção do elemento no topo (*pop*)
 - uma combinação dos anteriores
- Símbolo especial \$
 - é acrescentado ao fim da entrada, para detetar o fim da entrada
 - é colocado no fundo da pilha, para indicar pilha vazia

AUTÓMATO DE PILHA: EXEMPLO

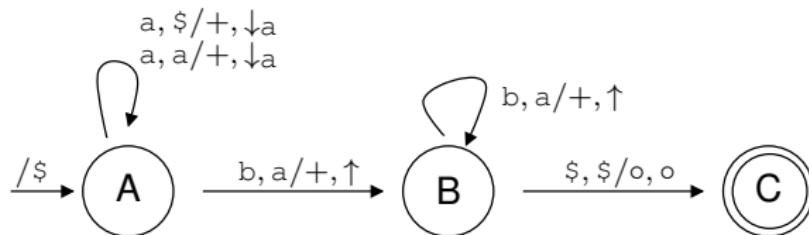
O grafo seguinte representa um autómato de pilha



- as transições têm a forma $x, z/w, \sigma$
 - x representa o próximo símbolo à entrada
 - z representa o símbolo no topo da pilha
 - w representa a ação sobre a entrada: \circ não faz nada; $+$ consome o símbolo, passando para o seguinte
 - σ representa a ação sobre a pilha: \circ não altera; \uparrow retira um símbolo (*pop*); $\downarrow\gamma$, insere γ na pilha (*push*), sendo γ uma sequência de símbolos

AUTÓMATO DE PILHA: EXEMPLO

O grafo seguinte representa um autómato de pilha



Q A palavra aaabbbb pertence à linguagem reconhecida pelo autómato?

R Sim, porque

$$\begin{aligned} (A, \$, aaabbbb\$) &\vdash (A, a\$, aabbbb\$) \vdash (A, aa\$, abbb\$) \\ &\vdash (A, aaa\$, bbb\$) \vdash (B, aa\$, bb\$) \vdash (B, a\$, b\$) \\ &\vdash (B, \$, \$) \vdash (C, \$, \$) \end{aligned}$$

Q Qual é a linguagem reconhecida por este autómato?

AUTÓMATO DE PILHA: DEFINIÇÃO

- D** Um **autómato de pilha** (AP) é um tuplo $M = (A, Z, Q, q_0, \delta, C)$, em que:
- A é o alfabeto de entrada;
 - Z é o alfabeto da pilha;
 - Q é um conjunto finito não vazio de estados;
 - $q_0 \in Q$ é o estado inicial;
 - $\delta \subset Q \times A_{\$} \times Z_{\$} \times Q \times \{\circ, +\} \times \{\circ, \downarrow_{\gamma}, \uparrow\}$, com $A_{\$} = A \cup \{\$\}$, $Z_{\$} = Z \cup \{\$\}$ e $\gamma \in Z^*$, é a relação que determina a transição entre estados; e
 - C é a condição de aceitação

AUTÓMATO DE PILHA: DEFINIÇÃO

D Um **autómato de pilha** (AP) é um tuplo $M = (A, Z, Q, q_0, \delta, C)$, em que:

- ...
- $\delta \subset Q \times A_{\$} \times Z_{\$} \times Q \times \{\circ, +\} \times \{\circ, \downarrow_{\gamma}, \uparrow\}$,
- ...
- em $\{\circ, +\}$
 - significa não avançar na entrada
 - + significa avançar para o próximo símbolo
- em $\{\circ, \downarrow_{\gamma}, \uparrow\}$
 - significa não alterar a pilha
 - \downarrow_{γ} significa inserir os símbolos γ na pilha (*push*)
 - \uparrow significa retirar o símbolo no topo (*pop*)

AUTÓMATO DE PILHA: DEFINIÇÃO

D Um **autómato de pilha** (AP) é um tuplo $M = (A, Z, Q, q_0, \delta, C)$, em que:

- ...
- C é a condição de aceitação
- ...
- Consideram-se três modos de aceitação:
 - por estado(s) de aceitação
 - qualquer que seja o estado da pilha
 - por pilha vazia
 - qualquer que seja o estado final
 - por estado(s) de aceitação com pilha vazia

AUTÓMATO DE PILHA: CONFIGURAÇÃO

D Em relação a um AP $M = (A, Z, Q, q_0, \delta, C)$, uma configuração é um triplo (q, σ, v) , onde

- $q \in Q$ é o estado num dado instante,
- $\sigma \in Z^*$ é a palavra presente na pilha, nesse instante
- $v \in A_s^*$ é a palavra na entrada ainda por consumir, nesse instante

- Uma configuração representa o estado global do autómato a cada passo no processo de reconhecimento de uma palavra dada
- Dada uma palavra u , a configuração inicial é dada por $(q_0, \$, u\$)$
- Dadas duas configurações c_1 e c_2 , escreve-se $c_1 \vdash c_2$ se existir uma transição em δ que o permita fazer
- Dadas duas configurações c_1 e c_2 , escreve-se $c_1 \vdash^* c_2$ se existir uma sequência de 0 ou mais transições que permitam transitar de c_1 para c_2

AUTÓMATO DE PILHA: LINGUAGEM RECONHECIDA

D Diz-se que um AP $M = (A, Z, Q, q_0, \delta, C)$ **aceita** a palavra $u \in A^*$ se existir uma sequência de configurações c_0, c_1, \dots, c_n que verifica as seguintes condições:

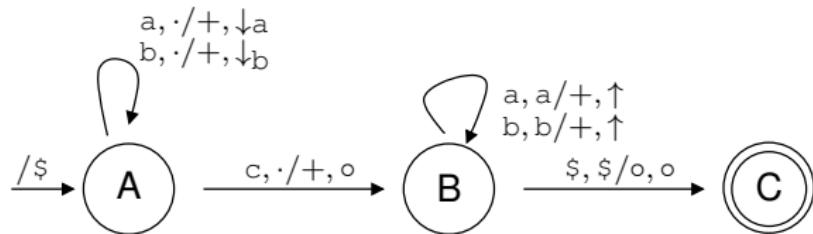
- $c_0 = (q_0, \$, u\$)$
- $\forall i=1, \dots, n \quad c_{i-1} \vdash c_i$
- $c_n = (q_f, \sigma, \$)$, com c_n satisfazendo a condição de aceitação

D A **linguagem reconhecida** pelo AP $M = (A, Z, Q, q_0, \delta, C)$, denota-se $L(M)$ e é definida por

$$\begin{aligned} L(M) &= \{ u \in A^* : M \text{ aceita } u \} \\ &= \{ u \in A^* : (q_0, \$, u\$) \vdash^* (q_f, \sigma_f, \$) \\ &\quad \wedge (q_f, \sigma_f, \$) \in C \} \end{aligned}$$

- Se C é pilha vazia, então $\sigma_f = \$$
- Se C é por estado de aceitação, então $q_f \in Q_{accept}$

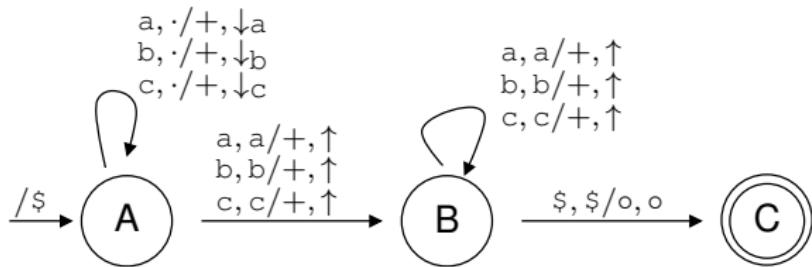
AUTÓMATOS DE PILHA: EXEMPLO (2)



Q Qual é a linguagem reconhecida por este autómato?

R $\{\omega_1 \subset \omega_2^R : \omega_1, \omega_2 \in \{a, b\}^*\}$

AUTÓMATOS DE PILHA: EXEMPLO (3)



Q Qual é a linguagem reconhecida por este autómato?

R $\{\omega_1 \omega_2^R : \omega_1, \omega_2 \in \{a, b, c\}^*\}$

AUTÓMATO DE PILHA DETERMINISTA

- D** Um **autómato de pilha determinista** (APD) é um tuplo $M = (A, Z, Q, q_0, \delta, C)$, em que:
- A é o alfabeto de entrada;
 - Z é o alfabeto da pilha;
 - Q é um conjunto finito não vazio de estados;
 - $q_0 \in Q$ é o estado inicial;
 - $\delta : Q \times A_{\$} \times Z_{\$} \rightarrow Q \times \{\circ, +\} \times \{\circ, \downarrow_\gamma, \uparrow\}$,
com $A_{\$} = A \cup \{\$\}$, $Z_{\$} = Z \cup \{\$\}$ e $\gamma \in Z_{\$}$, é a função de transição entre estados; e
 - C é a condição de aceitação
-
- A diferença em relação a um AD está na relação δ que agora é uma função
 - Logo, um APD é um AP (APND)

NÃO EQUIVALÊNCIA ENTRE APD E APND

- Os autómatos de pilha deterministas e não deterministas não são equivalentes
- Se ignorarmos o facto de uma pilha não poder crescer indefinidamente, os deterministas não implementáveis

AUTÓMATOS DE PILHA: PROJETO

Q Projete AP que, sobre o alfabeto $A = \{a, b, c\}$, reconheçam as linguagens seguintes

$$L_1 = \{\omega \in A^* : \#(a, \omega) = \#(b, \omega)\}$$

$$L_2 = \{\omega \in L_1 : |\omega| \geq 3 \wedge \omega_1 = a \wedge \omega_2 = b\}$$

$$L_3 = \{a^n b^m : n \geq m + 3\}$$

$$L_4 = \{\omega \in A^* : \#(a, \omega) = 2 * \#(b, \omega)\}$$

$$L_5 = \{\omega \in A^* : \#(ab, \omega) = \#(ba, \omega)\}$$

$$L_6 = \{\omega c \omega : \omega \in (\{a, b\})^*\}$$

- Sempre que possível obtenha uma solução determinista

R ???