UNIVERSIDADE FEDERAL DO PARÁ
INSTITUTO DE CIÊNCIAS EXATAS E NATURAIS
FACULDADE DE COMPUTAÇÃO
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

Disciplina: LINGUAGENS FORMAIS, AUTÔMATOS E COMPUTABILIDADE

Prof. Jefferson Morais
Email: jmorais@ufpa.br

Autômatos com Pilha

Conceito

Os **autômatos de pilha** constituem a segunda instância, em uma escala de complexidade crescente, do modelo genérico de reconhecedor introduzido anteriormente, prestando-se ao reconhecimento de linguagens livres de contexto;

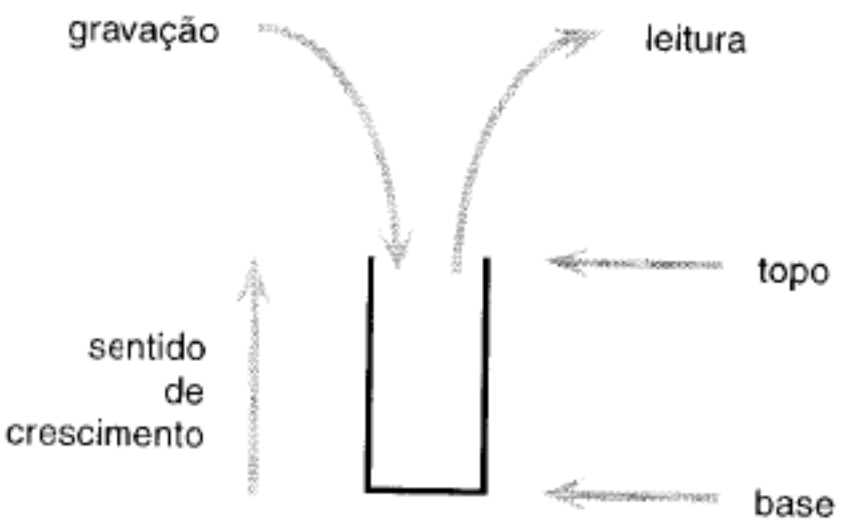
Diferentemente dos autômatos finitos, que não se utilizam da memória auxiliar prevista no modelo genérico, os autômatos de pilha têm o seu poder de reconhecimento estendido, quando comparado ao dos autômatos finitos, justamente pela disponibilidade e pela utilização de uma memória auxiliar organizada na forma de uma pilha.

Autômato com Pilha

- LLC pode ser associado a um formalismo tipo autômato
- AP é similar ao AF, mas inclui uma pilha como memória auxiliar, mais o não-determinismo.
- A pilha é independente da fita de entrada e não possui limite de tamanho ("infinita")
  - O último símbolo que entra é o primeiro que sai

Autômato com Pilha

- A base da pilha é fixa e define seu início
- O topo é variável e define a posição do último símbolo gravado



Definição

Formalmente, um autômato de pilha pode ser definido como uma sétupla M:

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

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 $Q \times (\Sigma \cup \{\varepsilon\}) \times \Gamma \rightarrow 2^{Q \times \Gamma^*}$;
- 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, $F \subseteq Q$.

Dos sete elementos que compõem este sistema formal, apenas três merecem explicação adicional — δ, Γ e Z_0 —, uma vez que os demais estão em correspondência direta com o que já foi estudado para o caso dos autômatos finitos.

Símbolo inicial da pilha

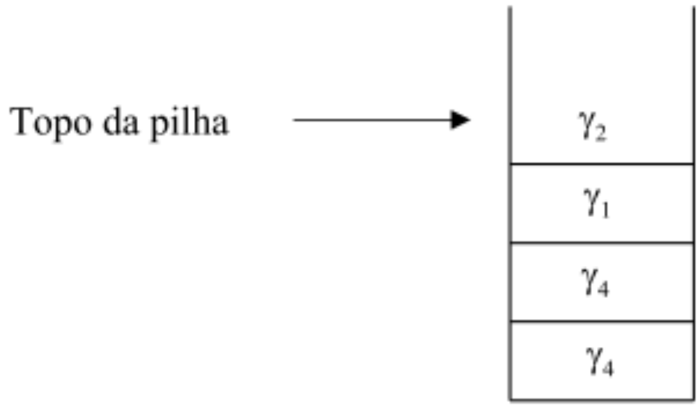
Note-se, inicialmente, a presença de um alfabeto de pilha Γ e de um símbolo inicial de pilha Z_0. O alfabeto de pilha especifica os símbolos que podem ser armazenados na pilha. Por convenção, um desses símbolos, denotado Z_0, representa o conteúdo inicial da pilha toda vez que o autômato de pilha principia o reconhecimento de uma nova sentença. Ao longo de sua operação, elementos de Γ são acrescentados e/ou removidos da pilha, e seu conteúdo total pode ser interpretado, em um dado instante, como sendo um elemento de Γ^*. Por convenção, cadeias de Γ que representam o conteúdo da pilha em um determinado instante são interpretadas considerando-se os símbolos mais à esquerda da cadeia no topo da pilha, e os símbolos mais à direita da cadeia no fundo da pilha.

Função de transição

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

Exemplo

Considere-se $\Gamma = \{\gamma_1, \gamma_2, \gamma_3, \gamma_4\}$. A cadeia $\gamma_2\gamma_1\gamma_4\gamma_4$, por exemplo, representa o conteúdo da pilha conforme ilustrado na Figura 5. Note-se que γ_2 se encontra no topo da pilha e que γ_4 se encontra no fundo da mesma:



Stack x pushdown

Os dispositivos que fazem uso de pilha costumam definir a forma de operação da mesma como uma das seguintes possibilidades:

- **Pilha "Stack"**: além das operações de empilhamento e desempilhamento de elementos no topo da pilha ("push" e "pop"), permite que os demais elementos da mesma sejam endereçados diretamente, somente para consulta;
- **Pilha "Pushdown"**: permite o acesso apenas ao elemento armazenado no topo da pilha, através das operações de empilhamento e desempilhamento ("push" e "pop"). Não permite o endereçamento dos demais elementos da pilha.

A definição de autômato de pilha neste texto considera que a sua memória auxiliar seja uma pilha "pushdown".

Configuração e configuração inicial

A **configuração de um autômato de pilha** é definida pelo seu estado corrente, pela parte da cadeia de entrada ainda não analisada e pelo conteúdo da pilha. A **configuração inicial de um autômato de pilha** é aquela em que o autômato se encontra no estado inicial q_0, o cursor se encontra posicionado sob a célula mais à esquerda na fita de entrada e o conteúdo da pilha é Z_0. Algebricamente, a configuração de um autômato de pilha pode ser representada como uma tripla $(q, \alpha, \gamma) \in Q \times \Sigma^* \times \Gamma^*$.

A configuração inicial para o reconhecimento de uma cadeia w é representada como (q_0, w, Z_0).

Movimentação

- As possibilidades de movimentação de um autômato de pilha em uma dada configuração são determinadas a partir de três informações: o seu estado corrente, o próximo símbolo presente na cadeia de entrada e o símbolo armazenado no topo da pilha;
- Observe-se, pela definição da função δ, a possibilidade de movimentações em vazio, sem consumo de símbolos da fita de entrada, e também a possibilidade de serem especificadas transições não-determinísticas;
- Note-se também a obrigatoriedade, imposta por essa formulação, de se consultar o símbolo presente no topo da pilha em toda e qualquer transição efetuada pelo autômato.

Movimentação

- Após a aplicação de uma transição, o cursor de leitura sofre um deslocamento de uma posição para a direita, e o símbolo presente no topo da pilha é removido, sendo substituído pela cadeia de símbolos especificada no lado direito da transição;
- No caso de transições em vazio, em que não há consulta de símbolo na fita de entrada, a posição do cursor permanece inalterada após a sua aplicação.

Movimentação

Desejando-se efetuar alguma transição de forma independente do conteúdo do topo da pilha, torna-se necessário especificar para cada elemento do alfabeto de pilha uma transição que efetue o mesmo tratamento do símbolo de entrada, removendo e reinserindo o mesmo símbolo da pilha, simulando assim uma transição independente do conteúdo da pilha.

Autômato determinístico

Para cada tripla (q, σ, γ) pertencente ao domínio da função δ, o elemento $\delta(q, \sigma, \gamma)$ pode conter zero, um ou mais elementos de $Q \times \Gamma^*$. Considere-se inicialmente o caso em que $\sigma \neq \varepsilon$. Havendo zero elementos em $\delta(q, \sigma, \gamma)$, isso indica que não há possibilidade de movimentação a partir da configuração considerada. Havendo um único elemento, isso significa que há apenas uma possibilidade de movimentação e, portanto, a transição é determinística. Quando todas as transições de um autômato de pilha são determinísticas, diz-se que o mesmo é **determinístico**.

Autômato não determinístico

Nos casos em que há mais de um elemento em $\delta(q, \sigma, \gamma)$, isso significa que há mais de uma opção de movimentação a partir desta configuração, e então a transição é dita não-determinística. Havendo pelo menos uma transição não-determinística, diz-se que o autômato de pilha é **não-determinístico**.

Autômato não determinístico

Autômatos de pilha não-determinísticos e com transições em vazio apresentam um comportamento dinâmico determinístico quando as duas seguintes condições forem simultaneamente verificadas:

1. $\forall q \in Q,\ \gamma \in \Gamma$, se $|(\delta, q, \varepsilon, \gamma)| \geqslant 1$, então $|\delta(q, \sigma, \gamma)| = 0,\ \forall \sigma \in \Sigma$;
2. $\forall q \in Q,\ \gamma \in \Gamma,\ \sigma \in \Sigma \cup \{\varepsilon\}, |\delta(q, \sigma, \gamma))| \leqslant 1.$

Determinismo x não-determinismo

Diferentemente do que foi visto para o caso dos autômatos finitos, no que se refere à equivalência dos modelos determinístico e não-determinístico quanto à classe de linguagens que são capazes de reconhecer, os autômatos de pilha não apresentam correspondente equivalência;

Conforme será visto mais adiante, os autômatos de pilha determinísticos são capazes de reconhecer apenas um subconjunto das linguagens livres de contexto;

Por esse motivo, exceto por ressalvas em sentido contrário, os autômatos de pilha mencionados daqui em diante serão não-determinísticos.

Movimentação

A movimentação de um autômato de pilha de uma configuração para a configuração seguinte é denotada pelo símbolo "$\vdash$", que representa a relação:

$$\vdash : Q \times \Sigma^* \times \Gamma^* \rightarrow Q \times \Sigma^* \times \Gamma^*$$

Estando o autômato de pilha em uma configuração $(q_i, \sigma\alpha, \phi\gamma)$, com $q_i \in Q$, $\sigma \in \Sigma$, $\alpha \in \Sigma^*$, $\phi \in \Gamma$ e $\gamma \in \Gamma^*$, sua movimentação a partir dessa configuração para a seguinte, como decorrência da aplicação de uma transição $\delta(q_i, \sigma, \phi) = (q_j, \eta)$, com $\eta \in \Gamma^*$, é representada por:

$$(q_i, \sigma\alpha, \phi\gamma) \vdash (q_j, \alpha, \eta\gamma)$$

Movimentação

A aplicação de transições em vazio, em que não há consumo de símbolo, é representada de maneira similar através de:

$$(q_i, \alpha, \phi\gamma) \vdash (q_j, \alpha, \eta\gamma)$$

Da mesma forma que no caso dos autômatos finitos, a aplicação de zero ou mais transições entre duas configurações quaisquer é representada através do símbolo $\vdash^*$, e a aplicação de no mínimo uma transição, por intermédio do símbolo $\vdash^+$.

Configuração final

A **configuração final de um autômato de pilha** costuma ser caracterizada de duas maneiras distintas, porém equivalentes. Na primeira delas, exige-se o esgotamento da cadeia de entrada e também que o autômato atinja um estado final. Nesta caracterização, o conteúdo final da pilha é irrelevante. Na segunda caracterização, exige-se o esgotamento da cadeia de entrada e também que a pilha tenha sido completamente esvaziada, não importando que o estado atingido seja final ou não-final.
Dependendo do tipo de definição adotada para caracterizar a configuração final de um autômato de pilha, é possível definir, também de duas maneiras distintas, a linguagem aceita pelo dispositivo.

Critério de aceitação “estado final”

A linguagem aceita por um autômato de pilha M, com base no **critério de estado final**, denotada $L(M)$, é definida como:

$$L(M) = \{w \in \Sigma^* \mid (q_0, w, Z_0) \vdash^* (q, \varepsilon, \gamma), q \in F, \gamma \in \Gamma^*\}$$

Critério de aceitação “pilha vazia”

Analogamente, a linguagem aceita por um autômato de pilha M, com base no **critério de pilha vazia**, denotada $V(M)$, é definida como:

$$V(M) = \{w \in \Sigma^* \mid (q_0, w, Z_0) \vdash^* (q, \varepsilon, \varepsilon), q \in Q\}$$

Notação

Com o intuito de facilitar a leitura do texto no restante deste capítulo, as transições de um autômato de pilha serão denotadas como:

$$(q_i, \sigma, X) \rightarrow (q_j, \gamma)$$

indicando, com isso, o par ordenado $((q_i, \sigma, X), (q_j, \gamma))$ pertencente à função δ, ou, simplesmente, $\delta(q_i, \sigma, X) = (q_j, \gamma)$.

Exemplo

- Exemplo 1
  - Seja $M1$ o autômato de pilha determinístico abaixo discriminado, cujo critério de aceitação de sentenças é baseado no esvaziamento da pilha.

$$
\begin{aligned}
Q &= \{q_0, q_1\} \\
\Sigma &= \{a, b, c\} \\
\Gamma &= \{Z_0, C\} \\
\delta &= \{(q_0, a, Z_0) \to \{(q_0, CCZ_0)\}, \\
&\quad (q_0, a, C) \to \{(q_0, CCC)\}, \\
&\quad (q_0, b, Z_0) \to \{(q_1, Z_0)\}, \\
&\quad (q_0, b, C) \to \{(q_1, C)\}, \\
&\quad (q_1, c, C) \to \{(q_1, \varepsilon)\}, \\
&\quad (q_1, \varepsilon, Z_0) \to \{(q_1, \varepsilon)\}\} \\
F &= \emptyset
\end{aligned}
$$

Exemplo

- Exemplo 1
  - Seja M_1 o autômato de pilha determinístico abaixo discriminado, cujo critério de aceitação de sentenças é baseado no esvaziamento da pilha.
  - $V_1(M_1) = \{a^i b c^{2i} \mid i \geq 0\}$

$$
\begin{array}{rcl}
Q & = & \{q_0, q_1\} \\
\Sigma & = & \{a, b, c\} \\
\Gamma & = & \{Z_0, C\} \\
\delta & = & \{(q_0, a, Z_0) \to \{(q_0, CCZ_0)\}, \\
 & & (q_0, a, C) \to \{(q_0, CCC)\}, \\
 & & (q_0, b, Z_0) \to \{(q_1, Z_0)\}, \\
 & & (q_0, b, C) \to \{(q_1, C)\}, \\
 & & (q_1, c, C) \to \{(q_1, \varepsilon)\}, \\
 & & (q_1, \varepsilon, Z_0) \to \{(q_1, \varepsilon)\}\} \\
F & = & \emptyset
\end{array}
$$

Exemplo 1

- Considere algumas sentenças desta linguagem e a correspondente sequência de movimentos executada pelo autômato durante o seu reconhecimento:

Sentença: b

Movimentos: $(q_0, b, Z_0) \vdash (q_1, \varepsilon, Z_0) \vdash (q_1, \varepsilon, \varepsilon)$

Sentença: $abcc$

Movimentos: $(q_0, abcc, Z_0) \vdash (q_0, bcc, CCZ_0) \vdash (q_1, cc, CCZ_0) \vdash (q_1, c, CZ_0) \vdash (q_1, \varepsilon, Z_0) \vdash (q_1, \varepsilon, \varepsilon)$

Sentença: $aabcccc$

Movimentos: $(q_0, aabcccc, Z_0) \vdash (q_0, abcccc, CCZ_0) \vdash (q_0, bcccc, CCCCZ_0) \vdash (q_1, cccc, CCCCZ_0) \vdash (q_1, ccc, CCCZ_0) \vdash (q_1, cc, CCZ_0) \vdash (q_1, c, CZ_0) \vdash (q_1, \varepsilon, Z_0) \vdash (q_1, \varepsilon, \varepsilon)$

Exemplo 1

- Tome-se agora como exemplo duas sentenças que não pertencem à linguagem definida por este autômato, e as correspondentes sequências de configurações :

Sentença: *abccc*

Movimentos:
$(q_0, abccc, Z_0) \vdash (q_0, bccc, CCZ_0) \vdash (q_1, ccc, CCZ_0) \vdash (q_1, cc, CZ_0) \vdash (q_1, c, Z_0)$

Sentença: *aabccc*

Movimentos: $(q_0, aabccc, Z_0) \vdash (q_0, abccc, CCZ_0) \vdash (q_0, bccc, CCCCZ_0) \vdash (q_1, ccc, CCCCZ_0) \vdash (q_1, cc, CCCZ_0) \vdash (q_1, c, CCZ_0) \vdash (q_1, \varepsilon, CZ_0)$

Digrama de estados

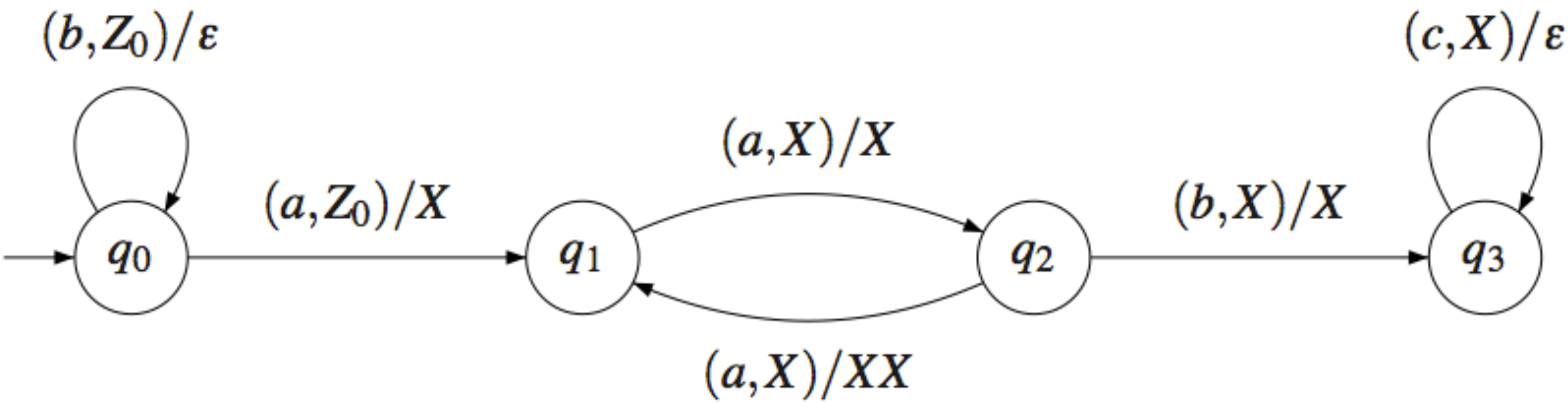
- Diferentemente dos Diagramas de Estado definidos para os autômatos finitos, os arcos entre dois estados *p* e *q* são rotulados com cadeias da forma

$$(\sigma, Z)/\gamma,\ \text{com } \sigma \in \Sigma, Z \in \Gamma \text{ e } \gamma \in \Gamma^*$$

para cada produção $\delta(p, \sigma, Z) = (q, \gamma)$

Exemplo 2

- Autômato M_2 com critério de aceitação baseado no esvaziamento da pilha, reconhece a linguagem $V_2(M_2) = \{a^{2i}bc^i \mid i \geq 0\}$

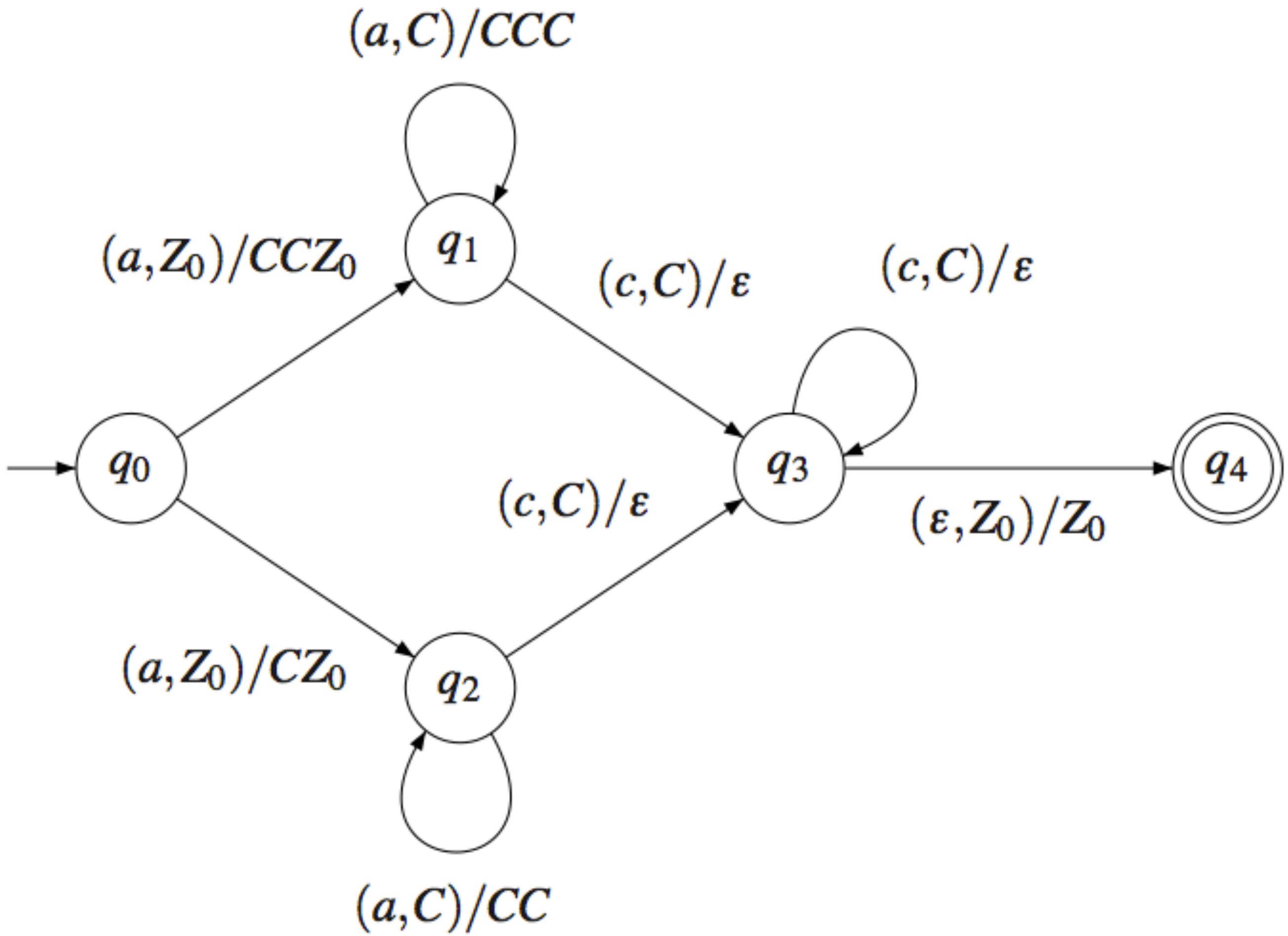


Exemplo 3

Considere-se o autômato de pilha não-determinístico M_3 a seguir apresentado, cujo critério de aceitação de sentenças é baseado em estado final.

$$
\begin{aligned}
Q &= \{q_0, q_1, q_2, q_3, q_4\} \\
\Sigma &= \{a, c\} \\
\Gamma &= \{Z_0, C\} \\
\delta &= \{(q_0, a, Z_0) \rightarrow \{(q_1, CCZ_0), (q_2, CZ_0)\}, \\
&\quad (q_1, a, C) \rightarrow \{(q_1, CCC)\}, \\
&\quad (q_1, c, C) \rightarrow \{(q_3, \varepsilon)\}, \\
&\quad (q_2, a, C) \rightarrow \{(q_2, CC)\}, \\
&\quad (q_2, c, C) \rightarrow \{(q_3, \varepsilon)\}, \\
&\quad (q_3, c, C) \rightarrow \{(q_3, \varepsilon)\}, \\
&\quad (q_3, \varepsilon, Z_0) \rightarrow \{(q_4, Z_0)\}\} \\
F &= \{q_4\}
\end{aligned}
$$

Exemplo 3



$(a,C)/CCC$
$(a,Z_0)/CCZ_0$
q_1
$(c,C)/\varepsilon$
$(c,C)/\varepsilon$
q_0
q_3
q_4
$(c,C)/\varepsilon$
$(\varepsilon,Z_0)/Z_0$
$(a,Z_0)/CZ_0$
q_2
$(a,C)/CC$

Exemplo 3

A operação deste autômato é exemplificada a seguir através do reconhecimento das seguintes sentenças:

1. Sentença: *aacc*

   Movimentos: $(q_0, aacc, Z_0) \vdash (q_2, acc, CZ_0) \vdash (q_2, cc, CCZ_0) \vdash (q_3, c, CZ_0) \vdash (q_3, \varepsilon, Z_0) \vdash (q_4, \varepsilon, Z_0)$

2. Sentença: *aacccc*

   Movimentos: $(q_0, aacccc, Z_0) \vdash (q_1, acccc, CCZ_0) \vdash (q_1, cccc, CCCCZ_0) \vdash (q_3, ccc, CCCZ_0) \vdash (q_3, cc, CCZ_0) \vdash (q_3, c, CZ_0) \vdash (q_3, \varepsilon, Z_0) \vdash (q_4, \varepsilon, Z_0)$

Note-se que o reconhecimento, em ambos os casos, pôde ser bem-sucedido já na primeira seqüência de movimentos, uma vez que a escolha da transição a ser aplicada na configuração inicial foi corretamente "adivinhada" nas duas situações.

Exemplo 3

No entanto, a escolha da transição a ser aplicada na configuração inicial do primeiro caso poderia ter sido diferente da apresentada, e neste caso ocorreria o seguinte:

1. Sentença: *aacc*

   Movimentos: $(q_0, aacc, Z_0) \vdash (q1, acc, CCZ_0) \vdash (q_1, cc, CCCCZ_0) \vdash (q_3, c, CCCZ_0) \vdash (q_3, \varepsilon, CCZ_0)$

Exemplo 3

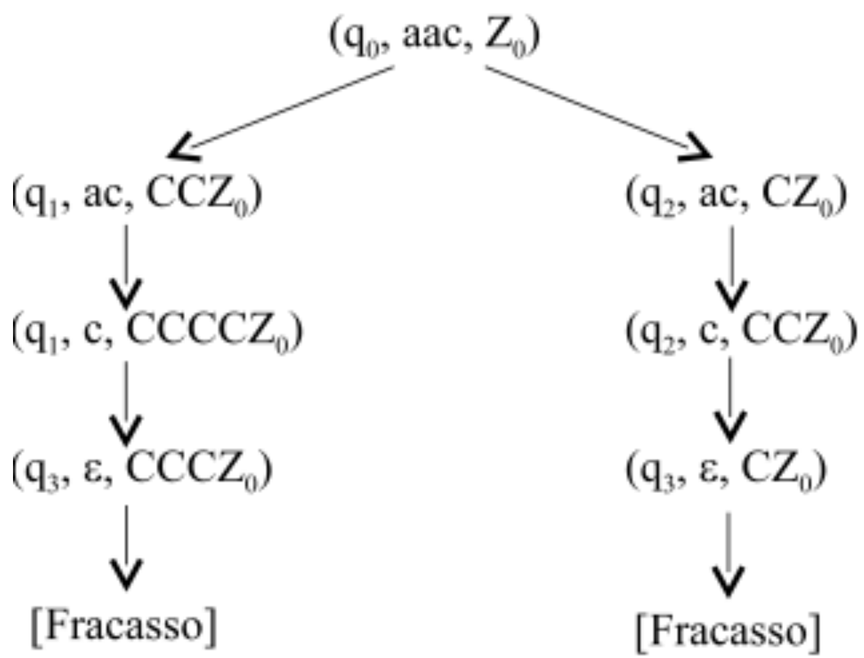
$$(q_0, aacc, Z_0)$$

$$(q_0, aacc, Z_0) \to (q_1, acc, CCZ_0) \to (q_1, cc, CCCCZ_0) \to (q_3, c, CCCZ_0) \to (q_3, \varepsilon, CCZ_0) \to \text{[Fracasso]}$$

$$(q_0, aacc, Z_0) \to (q_2, acc, CZ_0) \to (q_2, cc, CCZ_0) \to (q_3, c, CZ_0) \to (q_3, \varepsilon, Z_0) \to (q_3, \varepsilon, Z_0) \to \text{[Sucesso]}$$

Exemplo 3

- Cadeia aac que não pertence a linguagem



Transições em vazio

- Para finalizar, cumpre notar que, de acordo com a definição, os autômatos de pilha são capazes de efetuar movimentos independentemente da existência de símbolos na fita de entrada, através das chamadas transições em vazio, ao passo que o esvaziamento da pilha necessariamente impede qualquer possibilidade de movimentação futura. Ambos estes fatos são utilizados para demonstrar a equivalência dos critérios de aceitação por estado final e por pilha vazia

Equivalência dos critérios de aceitação

- A classe de linguagens aceita por autômatos de pilha não-determinísticos com critério de aceitação baseado em estado final é idêntica à classe de linguagens aceita por autômatos de pilha não-determinísticos com critério de aceitação baseado em pilha vazia. A importância desse resultado deve-se à liberdade de escolha que ele oferece quando se pretende demonstrar algum teorema relativo aos autômatos de pilha e às linguagens livres de contexto, podendo-se optar indistintamente entre um e outro critério de aceitação, sem prejuízo para a sua generalização.

Equivalência entre GLCs e APs

- Inicialmente, mostra-se que para qualquer gramática livre de contexto é possível definir um autômato de pilha não-determinístico que reconhece exatamente a mesma linguagem gerada pela gramática. A seguir, é apresentado o resultado inverso, ou seja, de que toda e qualquer linguagem aceita por um autômato de pilha não-determinístico pode ser gerada por uma gramática livre de contexto.

GLC => AP

Algoritmo 8.1 *"Obtenção de um autômato de pilha não-determinístico a partir de uma gramática livre de contexto qualquer."*

- *Entrada: uma gramática livre de contexto* $G = (V, \Sigma, P, S)$;
- *Saída: um autômato de pilha não-determinístico* $M = (Q, \Sigma, \Gamma, \delta, q, S, F)$ *com critério de aceitação de pilha vazia, tal que* $V(M) = L(G)$;
- *Método:*
  1. $Q \leftarrow \{q\}$;
  2. $\Gamma \leftarrow V$;
  3. $F \leftarrow \emptyset$;
  4. *Função de transição:*
     1. $\delta \leftarrow \emptyset$;
     2. $\delta(q, \varepsilon, A) = \{(q, \gamma) \mid A \rightarrow \gamma \in P\}, \forall A \in N, \gamma \in V^*$;
     3. $\delta(q, \sigma, \sigma) = \{(q, \varepsilon)\}, \forall \sigma \in \Sigma$.

GLC => AP

- Autômatos construídos segundo este critério operam através da repetida substituição dos símbolos não-terminais no topo da pilha, sem consumo de símbolos da fita de entrada, até que surja um símbolo terminal do topo da pilha;
- Nesta configuração, a sua remoção é condicionada à presença do mesmo símbolo na posição de leitura correntemente apontada pelo cursor da fita de entrada;
- Autômatos construídos segundo este critério também simulam a seqüência de derivações mais à esquerda que seria feita pela gramática correspondente na geração da mesma sentença.

Exemplo GLC => AP

Considere-se a gramática das expressões aritméticas:

$$\begin{array}{rcl} \{E & \rightarrow & T|T+E, \\ T & \rightarrow & F|F*T, \\ F & \rightarrow & (E)|a\} \end{array}$$

Exemplo GLC => AP

$$
\begin{array}{rcl}
\{(q,\varepsilon,E) & \to & \{(q,T),(q,T+E)\},\\
(q,\varepsilon,T) & \to & \{(q,F),(q,F*T)\},\\
(q,\varepsilon,F) & \to & \{(q,(E)),(q,a)\},\\
(q,a,a) & \to & \{(q,\varepsilon)\},\\
(q,(,() & \to & \{(q,\varepsilon)\},\\
(q,),)) & \to & \{(q,\varepsilon)\},\\
(q,+,+) & \to & \{(q,\varepsilon)\},\\
(q,*,*) & \to & \{(q,\varepsilon)\}\}
\end{array}
$$