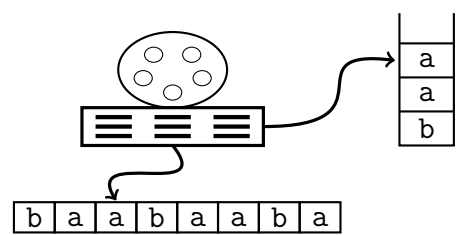


O autômato de pilha

A Figura abaixo mostra um autômato de pilha



Agora, a cada passo de computação, o autômato pode ler 2 símbolos

- o próximo símbolo da palavra de entrada
- o símbolo no topo da pilha

E em cada passo de computação, o autômato pode

- mudar o seu estado atual
- escrever alguma coisa no topo da pilha

Como usual, o controle dessas operações é feito por meio de parâmetros associados às transições

$$q_i \xrightarrow{a[b,a]} q_j$$

Quer dizer,

- o primeiro **a** é o símbolo que é lido na palavra de entrada
- de dentro dos colchetes, nós vemos
 - . o símbolo que é lido (e removido) no topo da pilha
 - . e o símbolo que é colocado no topo da pilha ao final da transição

Em todos esses lugares, nós podemos colocar o símbolo **e** da palavra vazia

$$q_i \xrightarrow{e[b,a]} q_j \qquad q_i \xrightarrow{a[e,a]} q_j \qquad q_i \xrightarrow{a[b,e]} q_j$$

E frequentemente é conveniente ler e/ou escrever múltiplos símbolos na pilha em um único passo de computação

$$q_i \xrightarrow{a[bb,aaa]} q_j$$

Os autômatos de pilha são máquinas *não-determinísticas* por definição.

Isso significa que

→ *podem existir múltiplas transições disponíveis em um certo passo de computação*

E como usual, nós vamos assumir que o autômato de pilha *aceita* a palavra de entrada se existe um caminho de computação que lê a palavra de entrada inteira e alcança um estado final.

Finalmente, nós vamos assumir que o autômato precisa terminar a computação com a pilha vazia para que a palavra de entrada seja aceita.