

A linguagem BAL

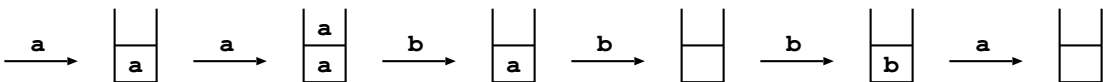
Considere a linguagem

BAL = palavras que possuem a mesma quantidade de a's e b's

A ideia natural aqui consiste em utilizar a pilha para lembrar a diferença entre as quantidades de a's e b's que foram lidos até o momento.

Para uma palavra de BAL essa diferença é igual a zero no final da computação, o que corresponde à pilha vazia, e portanto a palavra é aceita.

Por exemplo, nós teríamos a seguinte sequência de configurações da pilha para a computação sobre a palavra aabbba



Ou seja, quando um a é lido na palavra de entrada e o topo da pilha também é um a, o símbolo é empilhado; caso o topo da pilha seja um b, ele é removido.

A situação é análoga no caso da leitura de um b.

E quando a pilha está vazia, o símbolo lido é simplesmente colocado lá.

Mas, esse último caso introduz uma nova dificuldade:

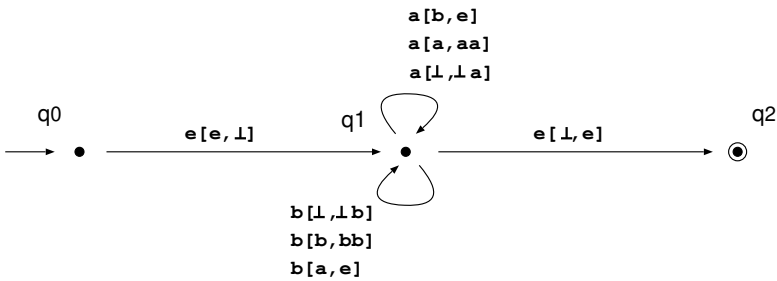
- Como é que o autômato pode saber que a pilha está vazia?

Uma solução simples consiste em utilizar um símbolo marcador $\perp$, que permanece na base da pilha durante toda a computação.

Quando $\perp$ é o topo da pilha isso é interpretado como se a pilha estivesse vazia.

Para fazer a ideia funcionar, é preciso colocar o símbolo $\perp$ na pilha no primeiro passo da computação e removê-lo no último passo.

Utilizando essa ideia, é fácil construir um autômato de pilha que reconhece a linguagem BAL:



Note que nos momentos em que se detecta que a pilha está vazia, o símbolo $\perp$ é efetivamente removido do topo da pilha.

Por esse motivo é preciso colocá-lo novamente de volta, como é feito nas transições

$q_1 \xrightarrow{a[\perp, \perp a]} q_1$ $q_1 \xrightarrow{b[\perp, \perp b]} q_1$

Finalmente, quando se chega ao final da palavra de entrada e o símbolo $\perp$ está no topo da pilha, o autômato pode realizar a transição

$q_1 \xrightarrow{e[\perp, e]} q_2$

para esvaziar a pilha e passar para o estado final.

Solução não-determinística:

