



A+

A-

## Unidade 3

### Seção 3

Acesse este conteúdo  
pelo smartphone



O que é isso?  
Clique no código e saiba  
mais.

# Linguagens Formais e Autômatos





A+

A-



# Webaula 3

Autômatos com Pilha



## Autômatos com pilha

Nesta webaula vamos abordar os autômatos com pilha (AP) que é um tipo de autômato que **reconhece** as **linguagens livres de contexto** (CLP).

Considere o alfabeto  $\Sigma = \{a, b\}$  e  $L_1$  a linguagem sobre  $\Sigma$  definida como  $L_1 = \{a^n b^n \mid n \geq 1\}$ .



A função **nova\_pilha(c)** retorna uma pilha com um único símbolo empilhado, aquele passado como o parâmetro c.

A função **pilha.topo()** não só retorna o elemento que está no topo da pilha, como também o desempilha. Já a função **pilha.empilha(w)** empilha a cadeia de caracteres **w** (possivelmente vazia). Observe a imagem a seguir, que será referência desta seção.

### Reconhecedor para a linguagem L1

```
estado = 0
pilha = nova_pilha('Z')
Enquanto (1) { /* Loop Infinito até sair no retorne */
    entrada = leCaractere()
    t = pilha.topo()
    se (estado==0 && entrada=='a' &&(topo=='a' || topo=='Z'))
    {
        estado=0
        pilha.empilha("a")
    } senão se (estado==0 && entrada=='b' && topo=='a') {
        estado=1
        pilha.empilha("") /* Empilha a cadeia vazia */
    } senão se (estado==1 && entrada=='b' && topo=='a') {
        estado=1
        pilha.empilha("") /* Empilha a cadeia vazia */
    } senão se (estado==1 && entrada==EOF && topo=='Z') {
        Retorne 1 /* 'Z' foi desempilhado, a pilha esta vazia
        */
    } senão {
        Retorne 0
    }
}
```

Fonte: elaborada pelo autor.

A ideia do algoritmo anterior é empilhar o símbolo 'a' enquanto a entrada for este símbolo. Por isso o termo 'pilha'.

Quando entrar o primeiro 'b', começamos a desempilhar o símbolo 'a', além disso, mudamos de estado, do 0 para o 1, assim o algoritmo não voltará a empilhar o símbolo 'a', caso este volte a ocorrer na entrada, o programa terminará entrando no **"Retorne 0"**.

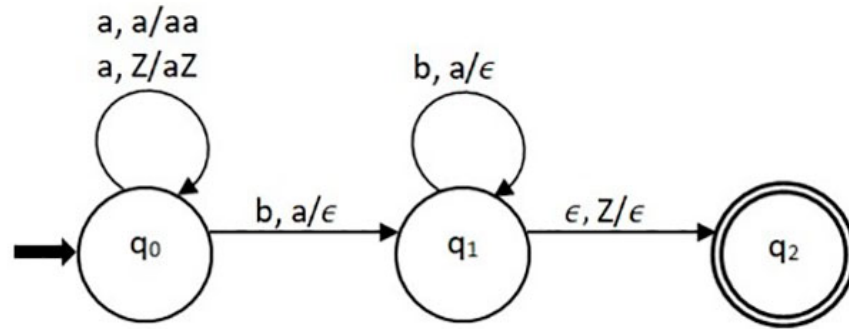
Observe que cada teste de condição depende de três variáveis:

Estado

Entrada

Topo da pilha

Autômato com pilha para a linguagem L1



Fonte: elaborada pelo autor.

No autômato da figura, a pilha é inicializada com o símbolo  $Z$  e a transição entre os estados  $q_0$  e  $q_1$  tem o texto  $b, a/\epsilon$ , correspondente ao segundo **if** do programa da figura anterior, isto é, estando no estado  $q_0$ , ao ler um  $b$  da entrada e tendo um símbolo  $a$  no topo da pilha, irá desempilhar este símbolo, empilhar  $\epsilon$ , e ir para o estado  $q_1$ .

Da mesma forma, a transição do estado  $q_0$  para ele próprio, corresponde ao primeiro **if** do programa, estando no estado  $q_0$ , ao ler um  $a$  da entrada e tendo um símbolo  $Z$  no topo da pilha, irá desempilhar este símbolo, empilhar  $aZ$  (o mesmo que empilhar primeiro  $Z$  e depois  $a$ ), e permanecer no estado  $q_0$ . O efeito prático neste caso é empilhar um  $a$  acima do  $Z$ .

No programa da primeira figura, não havia o estado  $q_2$ , no quarto **if** do programa havia o comando “**Retorne 1**”. Na segunda figura, essa condição de aceitação é indicada fazendo com que o autômato, sem consumir nenhum elemento da entrada e com o símbolo  $Z$  no topo da pilha, passe do estado  $q_1$  para o estado final  $q_2$ . A aceitação se dá se o autômato parar em um estado final após ler toda a entrada.

Vimos até agora como, a partir de uma GLC, construir um AP equivalente.

Também é possível fazer o processo inverso: dado um AP, construir uma GLC equivalente.

Ainda existem os autômatos com pilha ascendentes (APA), cuja versão determinística é usada na análise sintática ascendente.

Pesquise mais sobre o assunto.





## Vídeo de encerramento



## Você já conhece o Saber?

Aqui você tem na palma da sua mão a **biblioteca digital** para sua **formação profissional**.

Estude no celular, tablet ou PC em qualquer hora e lugar sem pagar mais nada por isso.

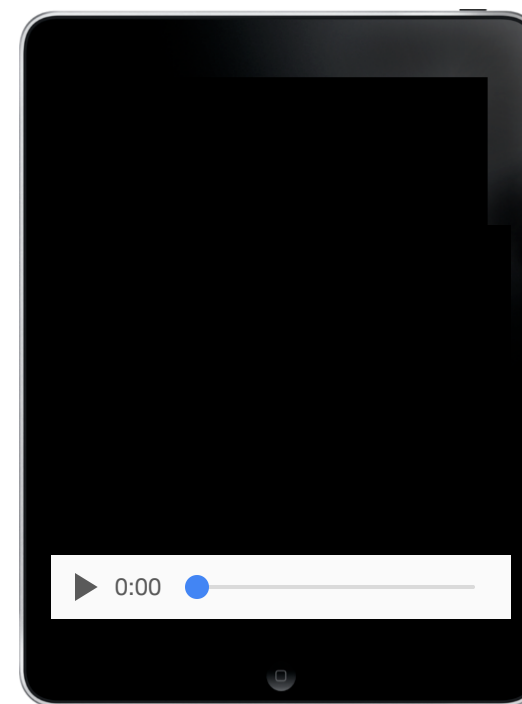
Mais de 475 livros com interatividade, vídeos, animações e jogos para você.



Android:  
<https://goo.gl/yAL2Mv>



iPhone e iPad - IOS:  
<https://goo.gl/OFWqcq>





Bons estudos!

