

Blocos de tamanho par

Agora imagine que nós queremos um autômato que realiza a seguinte tarefa

→ *Verificar se todos os blocos de a’s na palavra de entrada possuem tamanho par*

Quer dizer, um *bloco* é uma sequência de símbolos consecutivos iguais.

Daí que, nós sempre podemos decompor a palavra de entrada em blocos

a a a b a a b b b a a b ⇒ a a a | b | a a | b b b | a a | b

E a tarefa consiste em verificar se todos os blocos de a’s da palavra possuem tamanho par.

No exemplo acima, a resposta é NÃO.

Certo.

Mas como a gente constrói um autômato que verifica isso ?

Bom, a ideia é que o autômato vai “contando” os a’s do bloco — (*usando os estados q₀ e q₁*)

O bloco chega ao fim quando a gente lê um b.

E daí a gente vai saber se ele tem tamanho par ou ímpar.

Quando o bloco tem tamanho par, a gente apenas espera o próximo bloco aparecer — (*fazendo a leitura dos b’s*)

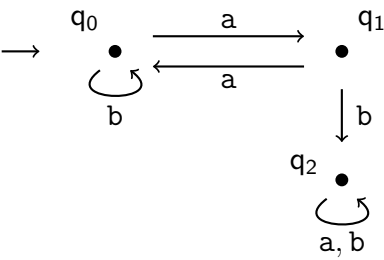
Mas, o que acontece quando o bloco tem tamanho ímpar?

Bom, nesse caso a gente já sabe que a resposta é NÃO.

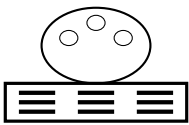
Mas essa resposta só é dada no final da leitura.

Daí que, a ideia é mudar para um estado q₂ daonde a gente não sai nunca mais — (*porque a gente já sabe que a resposta é NÃO*)

A coisa fica assim



Esse é um autômato com 3 luzinhas — (*ou 3 estados*)



Daí que, no final da leitura ele pode dar 3 respostas: q₀, q₁ ou q₂.

Mas, o que isso significa?

Bom, a gente já sabe que o q₂ significa NÃO.

E não é difícil ver que o q₀ significa SIM.

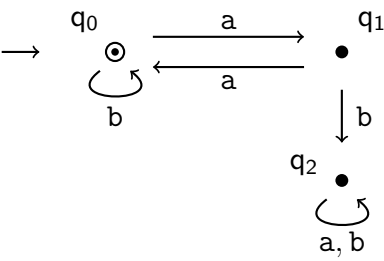
Mas e o q₁, o que significa?

Bom, se a computação acaba em q₁, isso significa que

- no meio da leitura, não foi encontrado nenhum bloco ímpar de a’s
- mas a computação está terminando após a leitura de uma quantidade ímpar de a’s

Daí que, q₁ também deve dar resposta NÃO.

A gente indica a resposta de cada estado no diagrama da seguinte maneira



quer dizer, os estados com resposta SIM são destacados com um círculo.