

A linguagem REG

Essa é a linguagem das expressões regulares

$$(ab \cup ba)^*, \quad b(a \cup ba)^*(ab)^*, \quad (a(aaa^*b)^*)^*, \quad \dots$$

E a nossa tarefa é construir um autômato de pilha que verifica se a palavra de entrada é uma expressão regular válida ou não.

Mas, isso é muito parecido com a tarefa de reconhecer expressões aritméticas.

A primeira diferença é que ao invés da letra **X** (que representa um número), nós vamos utilizar a letra **E** para representar uma expressão regular.

Mais precisamente, o **a** e **b** são expressões regulares

$$a, b \longrightarrow E$$

E se a gente acrescenta um **a** ou **b** a uma expressão regular, ela ainda é uma expressão regular

$$Ea, Eb \longrightarrow E$$

Daí, a operação de união ($\cup$) funciona como as operações aritméticas

$$(E \cup E) \longrightarrow E$$

Finalmente, a operação estrela ($\star$) só pode aparecer depois de uma expressão regular (**E**) — (*i.e., ela não pode ser o 1o símbolo, e não pode vir depois de abre-parênteses ou a união ($\cup$)*)

Mas, é fácil verificar isso.

Quer dizer, no momento em que a gente encontra a operação estrela ($\star$), basta verificar se o topo da pilha é uma expressão regular (**E**)

$$E\star \longrightarrow E$$

Agora, é fácil construir o autômato de pilha

