

A estrutura recursiva de PARBAL

Considere a seguinte palavra de PARBAL

$$()((()))(())()$$

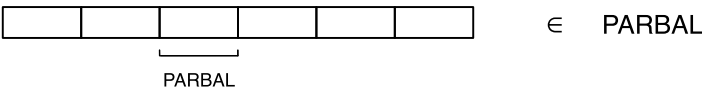
A primeira observação é que ela pode ser decomposta em 3 partes

$$() \mid ((())) \mid (())()$$

onde cada parte também é uma palavra de PARBAL.

Essa observação mostra que existe um padrão repetitivo na linguagem PARBAL

→ *Nós podemos formar palavras de PARBAL fazendo concatenações arbitrárias de palavras de PARBAL*



A seguir, nós observamos que a primeira parte da decomposição acima é a palavra mais simples da linguagem PARBAL: $()$ — (*com a exceção da palavra vazia, claro*)

Por outro lado, a segunda parte da decomposição

$$((()))$$

é uma palavra complexa que não pode ser decomposta como uma concatenação de palavras mais simples de PARBAL.

O que está acontecendo aqui?

Bom, o que está acontecendo é que existe um outro tipo de padrão nas linguagens reconhecidas por autômatos de pilha: os *padrões recursivos*.

Mas, o que é isso?

Examinando novamente a segunda parte da decomposição

$$((()))$$

nós vemos claramente que há alguma forma de repetição ali — (*uma repetição controlada*)

Quer dizer, nós podemos descrever a situação da seguinte maneira

- primeiro aparecem 3 símbolos abre parênteses,
e depois aparecem 3 símbolos fecha parênteses

Mas, nós também podemos decompor essa palavra em 3 palavras mais simples de PARBAL,

da seguinte maneira

$$(((()))) \implies \begin{matrix} (& &) \\ (& &) \\ (& &) \end{matrix}$$

Quer dizer, nós podemos imaginar que existem duas maneiras de combinar palavras simples de PARBAL para formar palavras mais complexas

$$\begin{matrix} (&) \\ (&) \\ (&) \end{matrix} \implies (&) & (&) & (&)$$
$$\begin{matrix} (&) \\ (&) \\ (&) \end{matrix} \implies (& (&) &) &)$$

A primeira forma corresponde a um *padrão repetitivo*, e a segunda corresponde a um *padrão recursivo*.

É interessante observar que a última parte da decomposição é formada por uma combinação dos dois tipos de padrão

$$((())())$$