A linguagem BAL_3

Considere a seguinte linguagem formada por palavras construídas com o alfabeto $\{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$.

BAL_3 = palavras que contém a mesma quantidade de a's, b's e c's

Intuitivamente, não é possível verificar que os 3 tipos de símbolos aparecem na mesma quantidade utilizando uma pilha como memória. Logo, BAL_3 não deve ser LC.

Para demonstrar esse fato, nós assumimos que m é um número arbitrariamente grande, e consideramos a palavra

$$\mathtt{w} = \mathtt{a}^m\mathtt{b}^m\mathtt{c}^m$$

A seguir, nós consideramos todas as formas como a palavra w pode ser decomposta em 5 segmentos

$$\mathtt{w} = \mathtt{x}\,\mathtt{y}\,\mathtt{z}\,\mathtt{u}\,\mathtt{v}$$

Existe um caso simples:

caso 1: y e u são blocos de símbolos do mesmo tipo

Nesse caso, y e u estão ambos contidos em blocos que constituem a palavra w.

Por exemplo,

Logo, ao bombear (ou remover) os segmentos y,u a palavra deixa de ter a mesma quantidade de a's, b's e c's, e portanto não pertence mais à linguagem BAL_3. Ou seja, o bombeamento falha nesse caso.

Mas, o próximo caso apresenta problemas.

caso 2: y (ou u) possuem mais de um tipo de símbolo

Nesse caso, os segmentos podem estar localizados como indicado no esquema abaixo

Ao bombear os segmentos y,u a palavra deixa de ter o padrão

$$\mathtt{a}\ldots\ldots\mathtt{ab}\ldots\ldots\mathtt{bc}\ldots\ldots\mathtt{c}$$

mas isso não é relevante para a linguagem BAL_3.

E, o que é pior, caso os segmentos y e u tenham a forma

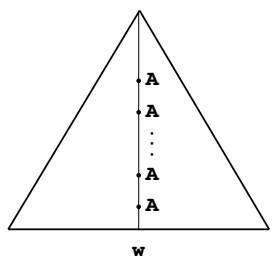
$$\mathtt{y} = \mathtt{a}^k\mathtt{b}^k \qquad\qquad \mathtt{u} = \mathtt{c}^k$$

o bombeamento ou a remoção de y,u deixa a palavra com a mesma quantidade de a's, b's e c's. Ou seja, nós não conseguimos obter o resultado desejado dessa vez.

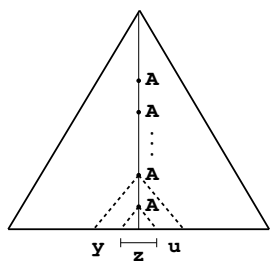
A solução consiste em utilizar uma versão um pouco mais poderosa do lema do bombeamento.

Considere uma árvore de derivação realmente muito grande, com altura bem maior do que o número de símbolos não-terminais da gramática G.

Nesse caso, os caminhos mais longos da árvore devem conter, não apenas duas, mas muitas ocorrências do mesmo símbolo A.



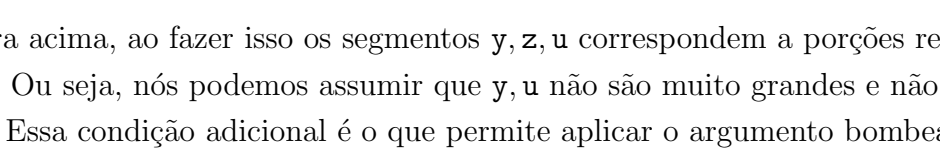
A observação chave é que o argumento do bombeamento apresentado na Seção **??** pode ser aplicado a qualquer par de símbolos A. Em particular, nós podemos escolher o par que fica mais embaixo no caminho.



Como indicado na figura acima, ao fazer isso os segmentos $\mathtt{y}, \mathtt{z}, \mathtt{u}$ correspondem a porções relativamente pequenas da palavra w. Ou seja, nós podemos assumir que $\mathtt{y}, \mathtt{u}$ não são muito grandes e não ficam muito distantes um do outro. Essa condição adicional é o que permite aplicar o argumento bombeamento em situações como o caso 2 acima.

caso 2: (revisitado)

Utilizando a condição de que os segmentos y e u devem estar próximos, eles só podem conter parte de dois blocos da palavra w. Por exemplo,



Isto é, sempre existe um tipo de símbolo que não aparece nos segmentos $\mathtt{y}, \mathtt{u}$. Logo, ao bombear ou remover os segmentos $\mathtt{y}, \mathtt{u}$ a palavra deixa de ter a mesma quantidade de a's, b's e c's, e portanto não está mais em BAL_3. Ou seja, o bombeamento falha também nesse caso.

Como o bombeamento falha em todos os casos, nós podemos concluir que a linguagem BAL_3 não é LC.

Abaixo, nós temos a versão formal do lema do bombeamento que foi utilizada nesse exemplo.

Teorema 2: (Lema do bombeamento - versão forte)

Para toda gramática livre de contexto G existem números n, m tais que toda palavra de comprimento maior do que n pode ser decomposta como

$$w = x\,y\,z\,u\,v$$

de modo que

i) $yu \neq \mathtt{e}$

ii) $|yzu| \leq m$

iii) a palavra $w_k = xy^kzu^kv$ pode ser gerada pela gramática G, para todo $k \geq 0$.

$\diamond$