Expressões regulares

IBM1088 Linguagens Formais e Teoria da Computação

Evandro Eduardo Seron Ruiz
evandro@usp.br

Universidade de São Paulo

Frase do dia

A vida é uma luta inteira em que cada passo é contestado. Logo, é uma alma covarde aquela que, assim que as nuvens negras se condensam ou apenas se mostram no horizonte, abate-se, perde o ânimo e põe-se a lamentar.

Arthur Schopenhauer[a]

[a] Filósofo alemão (1788 — 1860).

Prólogo

- Toda LR pode ser descrita por uma Expressão Regular, ER
- ER é um formalismo denotacional
- ER explica como gerar palavras da LR
- ER é ideal para comunicação homem $\times$ máquina

Objetivos

- Introduzir expressões regulares (ER) como uma linguagem formal
- Mostrar a equivalência entre ER e AF
- Caracterizar as limitações de AF

Conteúdo

1. Definição de ER
2. Exemplos de ER
3. Equivalência entre ER e AF
4. Limitações dos AF
5. Lema do Bombeamento
6. Exemplos

Expressões regulares

- Toda linguagem regular pode ser descrita por uma expressão denonimada expressão regular

Expressões regulares

- Toda linguagem regular pode ser descrita por uma expressão denonimada expressão regular
- ER são muito usadas em utilitários dos sistemas operacionais *Unix-like* e em linguagens de programação

Expressões regulares

- Toda linguagem regular pode ser descrita por uma expressão denonimada expressão regular
- ER são muito usadas em utilitários dos sistemas operacionais *Unix-like* e em linguagens de programação
- Uma ER é definida a partir de conjuntos e as operações de concatenação, união e da operação estrela

Expressões regulares

- Toda linguagem regular pode ser descrita por uma expressão denonimada expressão regular
- ER são muito usadas em utilitários dos sistemas operacionais *Unix-like* e em linguagens de programação
- Uma ER é definida a partir de conjuntos e as operações de concatenação, união e da operação estrela
- O valor de uma expressão regular é uma linguagem

Exemplos

- $\text{ER}_1 = (a \cup b)a^*$
- $\text{ER}_1 = L_1$, uma linguagem t.q. w começam com 'a', ou começam com 'b', seguidas por um número qualquer ($n \geq 0$) de 'a'.

+ Exemplos de ER

- Σ^*, para Σ um alfabeto qualquer
  - É a linguagem constituída de todas as cadeias de caracteres sobre Σ
  - Sendo $\Sigma = \{a, b\}, \Sigma^* = (a \cup b)^* + \epsilon$

+ Exemplos de ER

- Σ^*, para Σ um alfabeto qualquer
  - É a linguagem constituída de todas as cadeias de caracteres sobre Σ
  - Sendo $\Sigma = \{a, b\}, \Sigma^* = (a \cup b)^* + \epsilon$
- Σ^*1
  - É a ER que representa a linguagem na qual todas as cadeias sobre Σ que terminam com o caracter '1'

Vejamos qual é a definição de uma ER...

Definição de ER

Sejam R_1 e R_2 duas linguagens. Sejam também as ER r_1 e r_2 que denotam as linguagens R_1 e R_2.
Dizemos que r é uma *expressão regular* (ER) se r for:

1. $a, \forall a \in \Sigma$, ou seja, a linguagem que contém exclusivamente a palavra constituída pelo símbolo a;
2. ϵ, ou seja, a cadeia vazia;
3. $\varnothing$, ou seja, a linguagem vazia. E, indutivamente, considerando r_1 e r_2, expressões regulares, temos:
4. $r = (r_1 \cup r_2)$, a união, também denotada por $r_1 + r_2$;
5. $r = (r_1 \circ r_2)$, a concatenação, também denotada por $r_1 r_2$;
6. $r = (r_1^*)$, a operação estrela, ou concatenação sucessiva.

Veja os detalhes a seguir:

Em detalhes a última parte

Dizemos que R é uma *expressão regular* (ER) se R for:

1. Sejam r_1 e r_2 ER que denotam as linguagens R_1 e R_2 respectivamente
2. $(r_1 + r_2)$, a união, é uma ER e denota a linguagem $(R_1 \cup R_2)$;
3. $(r_1 r_2)$, a concatenação, é uma ER e denotada a linguagem $R_1 R_2 = \{uv \mid u \in R_1, v \in R_2\}$;
4. (r_1^*), a operação estrela, ou concatenação sucessiva, denota a linguagem R_1.

Convenção

Quando omitimos os parênteses nas ER, respeitamos as convenções de precedência de operadores. Vejam as convenções na Tabela abaixo.

Precedência	**Operação**
1	Estrela
2	Concatenação
3	União

Tabela : Convenções de precedência de operadores nas ER.

Exemplos de ER

Veja alguns exemplos de ER relacionados às linguagens geradas por estas expressões. Considere $\Sigma = \{a, b\}$.

ER, (R)	**Linguagem gerada**, $L(R)$
aa	Somente a palavra aa
ba^*	Palavras que começam com 'b' e são seguidas por zero ou mais 'a's
a^*ba^*	Palavras que contém um *único* 'b'
$\Sigma^*b\Sigma^*$	Palavras que tem, *pelo menos*, um símbolo 'b'
$\Sigma^*aba\Sigma^*$	Palavras que contém a subcadeia 'aba'
$b^*(ab^+)^*$	Palavras em que todo 'a' é seguido por pelo menos um 'b'
$(\Sigma\Sigma)^*$	Palavras de comprimento par

+ Exemplos de ER

Na Tabela abaixo mostramos outros exemplos de ER relacionando com as linguagens geradas por estas expressões. Aqui também $\Sigma = \{a, b\}$.

Expressão Regular, (R)	**Linguagem gerada**, $L(R)$
$ab + ba$	$L(R) = \{ab, ba\}$
$(a + b)^*$	Todas as palavras sobre Σ
$(a + b)^*aa(a + b)^*$	Todas as palavras que contém a subcadeia 'aa'
$(a + b)^*(aa + bb)$	Todas as palavras que terminam com 'aa' ou 'bb'

Vamos relembrar concatenação de conjuntos

- Sejam os conjuntos A e B
- $AB = A \circ B = \{ab \,|\, a \in A, b \in B\}$

Vamos relembrar concatenação de conjuntos

- Sejam os conjuntos A e B
- $AB = A \circ B = \{ab \,|\, a \in A, b \in B\}$
- mas...
- Se $B = \varnothing, \nexists b$. Assim $\nexists ab$ tal que $a \in A, b \in B$, assim
- $A \circ \varnothing = \varnothing$

Ou seja, a concatenação do $\varnothing$ é sempre $\varnothing$.

Outros exemplos de ER

Na Tabela abaixo mostramos outros exemplos de ER relacionando com as linguagens geradas por estas expressões. Aqui também considere o alfabeto $\Sigma = \{a, b\}$.

ER, (R)	**Linguagem gerada**, $L(R)$
$(a \cup \epsilon)b^*$	$= \{w \mid w = ab^* \cup b^*\}$
$(a \cup \epsilon)(b \cup \epsilon)$	$= \{\epsilon, a, b, ab\}$
$b^*\varnothing$	$= \varnothing$
$\varnothing^*$	$= \{\epsilon\}$, (ou seja, juntar qualquer número de cadeias, pode ser pode ser juntar 0 cadeias, o que gera a cadeia vazia)

Ainda. . . união e concatenação

Se tomarmos R como uma ER, devemos saber que:

- $R \cup \varnothing = R$, ou seja, adicionar a linguagem vazia a qualquer R não modifica R; e

Ainda. . . união e concatenação

Se tomarmos R como uma ER, devemos saber que:

- $R \cup \varnothing = R$, ou seja, adicionar a linguagem vazia a qualquer R não modifica R; e
- $R \circ \epsilon = R$, ou seja, a concatenação de uma expressão regular com uma cadeia vazia é a própria expressão regular.

Fique atento também para o seguinte:

- $(R \cup \epsilon)$ pode ser diferente de R.
  Por exemplo, se em $\Sigma = \{a, b\}$ e $R = a$, temos que $L(R) = \{a\}$ mas $L(R \cup \epsilon) = \{a, \epsilon\}$; e, como sabemos

Fique atento também para o seguinte:

- $(R \cup \epsilon)$ pode ser diferente de R.
  Por exemplo, se em $\Sigma = \{a, b\}$ e $R = a$, temos que $L(R) = \{a\}$ mas $L(R \cup \epsilon) = \{a, \epsilon\}$; e, como sabemos
- $R \circ \varnothing = \varnothing$, para qualquer expressão regular.
  Ou seja, a concatenação de uma expressão regular com uma linguagem vazia gera uma linguagem vazia.
  Relembre a definição da operação de concatenação.

Teorema entre ER e $L(R)$

Teorema

Se R é uma expressão regular então a linguagem gerada por R, $L(R)$, é uma linguagem regular

Equivalência ER–AF

Teorema

Uma linguagem é regular se e somente se alguma expressão regular a descreve

Equivalência ER–AF

Teorema

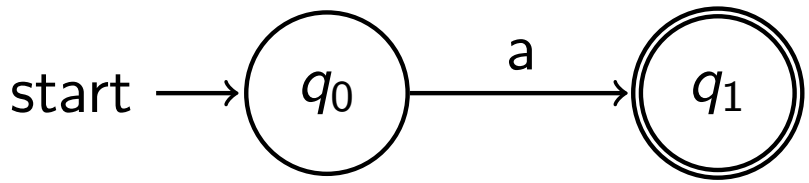
Uma linguagem é regular se e somente se alguma expressão regular a descreve

Lema

Se uma linguagem é descrita por uma expressão regular, então ela é regular

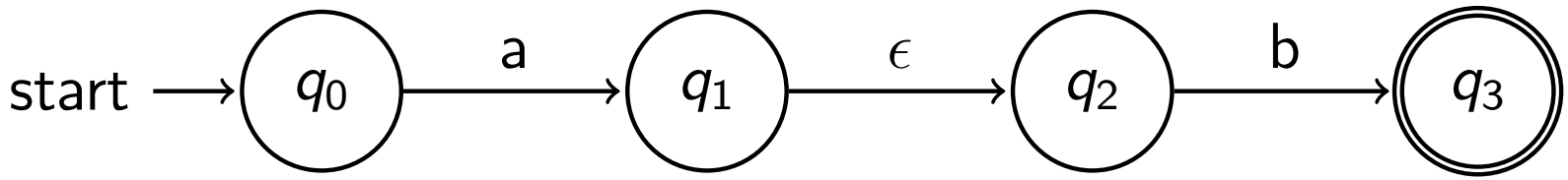
Exemplos: 1

A figura abaixo representando um autômato é equivalente a ER a.



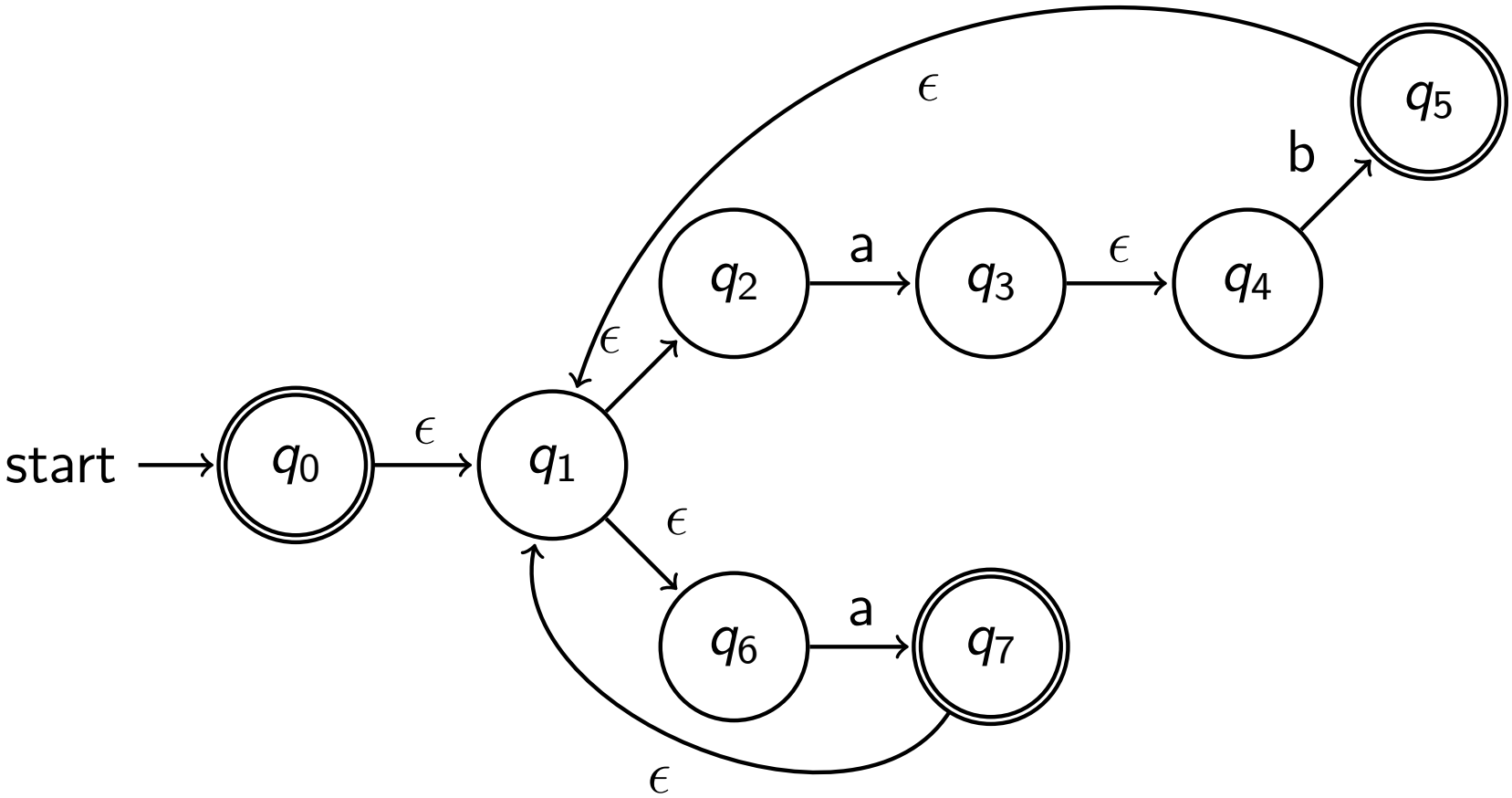
Exemplos: 2

A figura abaixo representando um autômato é equivalente a ER *ab*.



Exemplos: 3

A figura abaixo representando um autômato é equivalente a ER $(ab \cup a)^*$.



Limitações dos AF

- $L = \{a^n b^n \mid n \geq 0\}$, por mais que possa se assemelhar a uma linguagem regular, L não é regular

Limitações dos AF

- $L = \{a^n b^n \mid n \geq 0\}$, por mais que possa se assemelhar a uma linguagem regular, L não é regular
- O número de símbolos a e b em L é ilimitado e igual

Limitações dos AF

- $L = \{a^n b^n \mid n \geq 0\}$, por mais que possa se assemelhar a uma linguagem regular, L não é regular
- O número de símbolos a e b em L é ilimitado e igual
- Um autômato deveria computar a quantidade ilimitada n para os dois símbolos, o que é impossível para um autômato finito

O Lema do Bombeamento para Linguagens Regulares é um método para provar como linguagens como L_1 não são regulares. Vejamos...

Conceito do Lema do Bombeamento

- Usado para demonstrar que um conjunto não é aceito por uma máquina de estado finito
- Que o conjunto existe fora do espaço dos AF

Conceito do Lema do Bombeamento

- Usado para demonstrar que um conjunto não é aceito por uma máquina de estado finito
- Que o conjunto existe fora do espaço dos AF
- Os conjuntos dentro dos limites dos AF têm certas propriedades
- Se o conjunto não tem estas propriedades não pode ser originário de um AF

Uso do Lema do Bombeamento

O uso do lema ocorre de forma "contrária" (reversa) ao que o lema diz:

Lema $\exists$ propriedade do conjunto regular $\rightarrow$ propriedades do bombeamento

Demo $\nexists$ propriedade do bombeamento $\rightarrow \nexists$ propriedade do conjunto regular

Lema do Bombeamento

Lema do Bombeamento

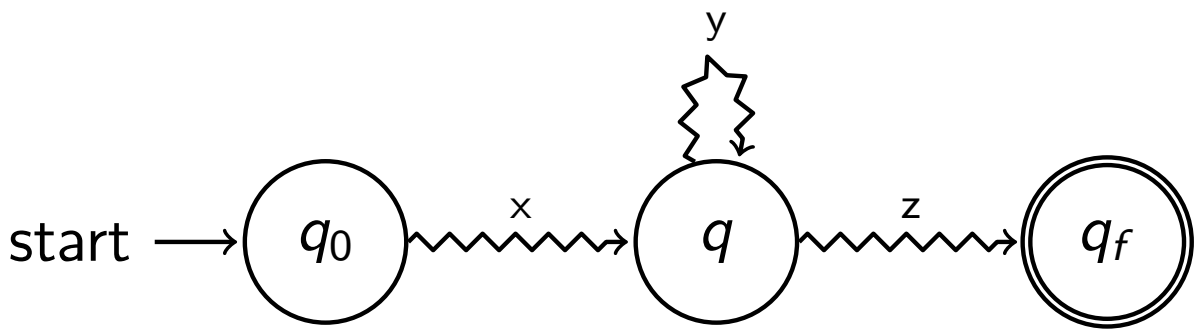
Se L é uma linguagem regular, então existe um número p, chamado comprimento do bombeamento, tal que, se para qualquer palavra $w \in L$, com $|w| \geq p$, então w pode ser dividida em três subcadeias, $w = xyz$, satisfazendo as seguintes condições:

1. $|xy| \leq p$;
2. $|y| \geq 1$, e
3. $w = xy^i z \in L$, $\forall i \geq 0$.

Provado em 1959 por Dana Scott e Michael Rabin, ambos receberam o ACM Turing Award em 1976. Ambos foram orientados por Alonzo Church.

Explicando o lema

- Se uma linguagem L é regular, então ela aceita um AFD com p estados
- Se o AFD reconhece w, t.q $|w| \geq p$, é porque o AFD assume um estado mais de uma vez, ou seja, tem um ciclo num estado q



Lema do bombeamento: releitura

Este lema diz que todas as cadeias de uma linguagem podem ser 'bombeadas' se elas são longas quanto p, ou seja, qualquer uma destas cadeias de comprimento p tem uma parte que pode ser repetida um número qualquer de vezes, sendo que a cadeia resultante ainda permanece na mesma linguagem.

Lema do Bombeamento

Se L é uma linguagem regular, então existe um número p, chamado comprimento do bombeamento, tal que, se para qualquer palavra $w \in L$, com $|w| \geq p$, então w pode ser dividida em três subcadeias, $w = xyz$, satisfazendo as seguintes condições:

1. $|xy| \leq p$;
2. $|y| \geq 1$, e
3. $w = xy^i z \in L$, $\forall i \geq 0$.

Ainda sobre o Lema

Sabemos que y^i é a repetição, por concatenação de y, i vezes. Se $i = 0, y^i = y^0 = \epsilon$. Note também que $|y|$ é o comprimento da cadeia y e que, $|y| \geq 1$, ou seja, $y \neq \epsilon$. Repare que, pela primeira condição, o comprimento da concatenação xy deve ser, no máximo, p, ou seja, $|xy| \leq p$.

Lema do Bombeamento

Se L é uma linguagem regular, então existe um número p, chamado comprimento do bombeamento, tal que, se para qualquer palavra $w \in L$, com $|w| \geq p$, então w pode ser dividida em três subcadeias, $w = xyz$, satisfazendo as seguintes condições:

1. $|xy| \leq p$;
2. $|y| \geq 1$, e
3. $w = xy^iz \in L$, $\forall i \geq 0$.

Para que serve o Lema?

Como usar?

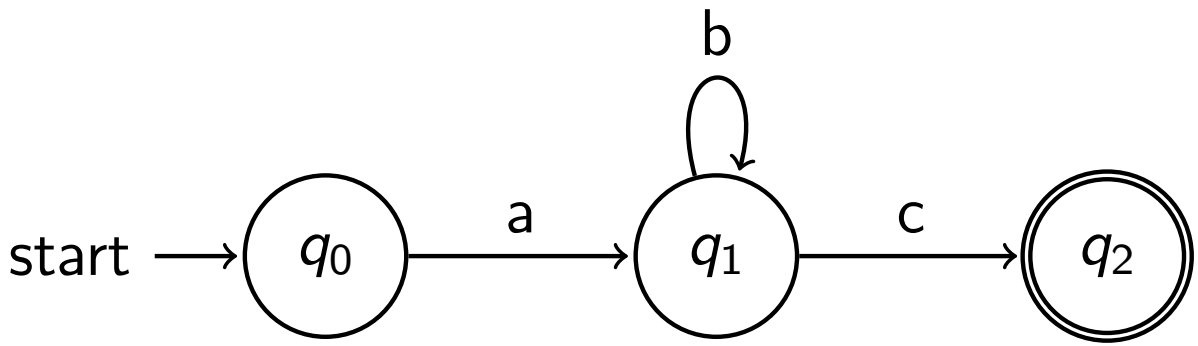
Por exemplo, podemos usar este lema para provar a não-regularidade de uma linguagem.

Exemplo 1

Seja a linguagem $L_1 = \{ab^n c \mid n \geq 0\}$.
Elabore um AFD que reconheça L_1

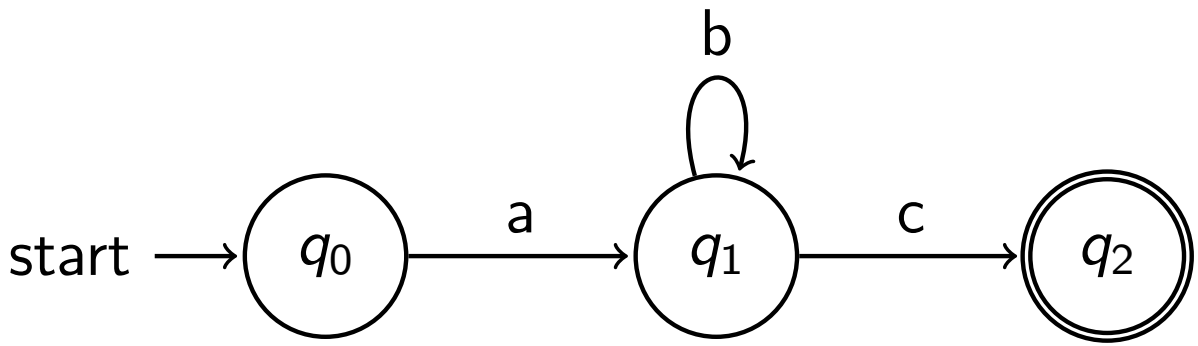
Exemplo 1

Seja a linguagem $L_1 = \{ab^n c \mid n \geq 0\}$.
Elabore um AFD que reconheça L_1



Exemplo 1

Seja a linguagem $L_1 = \{ab^n c \mid n \geq 0\}$.
Elabore um AFD que reconheça L_1



Vamos imaginar, por absurdo, que L_1 não é uma linguagem regular.
Assim, não seria possível aplicar o Lema do Bombeamento.

Aplicando o Lema

Vamos tomar os seguintes exemplos:

- Seja $w = abc$. Temos assim $|w| = 3$. Rescrevendo w como $w = xy^iz \in L_1, \forall i \geq 0$ de modo que $|xy| \leq 3$ e $|y| \geq 1$ podemos escolher $x = a, y = b, z = c$. Deste modo, todas as cadeias abaixo pertencem a L_1. Em outras palavras:

$$w = \left\{ \begin{array}{r} xz = ac \\ xyyz = abbc \\ xyyyz = abbbc \end{array} \right. \in L_1$$

Portanto L_1 é uma LR.
Repare que no primeiro caso $w = xz = xy^iz$, para $i = 0$

Outra maneira de ver:

- Seja agora $w = abbbc$, ou seja, $|w| = 5$. Podemos reescrever w como $w = xy^iz \in L_1, \forall i \geq 0$ de modo que $|xy| \leq 5$ e $|y| \geq 1$. Aqui podemos fazer três escolhas de subdivisão da cadeia s:

$$w = \begin{cases} \underbrace{a}_{x}\underbrace{b}_{y}\underbrace{bbc}_{z}. & \text{Tal como } (a)(b)^*(bbc), \\ \underbrace{a}_{x}\underbrace{bb}_{y}\underbrace{bc}_{z}. & \text{Tal como } (a)(bb)^*(bc), \\ \underbrace{ab}_{x}\underbrace{b}_{y}\underbrace{c}_{z}. & \text{Tal como } (ab)(b)^*(c), \end{cases} \text{ todas } \in L_1.$$

No entanto...

Nem todas as divisões possíveis de cadeias w produzem cadeias $\in L_1$.
Vejam estas possibilidades abaixo:

$$w = \begin{cases} \underbrace{\epsilon}_{x}\underbrace{a}_{y}\underbrace{bbbc}_{z}. \\ \underbrace{\epsilon}_{x}\underbrace{ab}_{y}\underbrace{bbc}_{z}. \\ \underbrace{\epsilon}_{x}\underbrace{abb}_{y}\underbrace{bc}_{z}. \end{cases}$$

Nestes casos $w = xy^iz \notin L_1$.
Entretanto, existem divisões possíveis de L_1 de modo a satisfazer o Lema do Bombeamento.

Exemplo 2

Seja a linguagem $L_2 = \{ww \mid w \in \{a, b\}^*\}$.
Prove que L_2 não é uma LR.

Exemplo 2

Seja a linguagem $L_2 = \{ww \mid w \in \{a, b\}^*\}$.
Prove que L_2 não é uma LR.
Vamos tentar "bombear" L_2.
Seja $w = a^n b$, então $ww = a^n b a^n b \in L_2$. Assim, $|w| > 2n$.
Precisamos fatorar w para encontrar $|xy| \leq 2n$.

$$ww = xyz = \underbrace{a^r}_{x} \underbrace{a^s}_{y} \underbrace{a^t b a^n b}_{z}$$

tal que: $r + s + t = n$, ou seja, conseguimos $|xy| < n$.
Vamos continuar...

Exemplo 2: continuação

Tomemos um expoente para xy^iz, i=2:

$$\begin{aligned} xy^2z &= a^r a^{2s} a^t b a^n b \\ &= a^r a^s a^s a^t b a^n b \\ &= \underbrace{a^r a^s a^t}_{a_n} a^s b a^n b \\ &= a^n a^s b a^n b \\ &= \underbrace{a^{n+s} b}_{p_1} \underbrace{a^n b}_{p_2} \end{aligned}$$

Como $p_1 \neq p_2$ não é uma palavra repetida, portanto $\notin L_2$ e não é uma LR usando esta "fatoração".
Talvez outras fatorações possa levar a outra conclusão.

Para fazer em casa!

Apliquem o Lema do Bombeamento para estes casos:

- $L_2 = \{a^n b^n \mid n \geq 0\}$
- $L_3 = \{w \mid w \text{ tem número igual de } a \text{ e } b\}$
- $L_4 = \{a^i b^j \mid i > j\}$

Caros,

Ainda voltaremos a este tema de bombeamento num outro encontro!