ACH2043 INTRODUÇÃO À TEORIA DA COMPUTAÇÃO

Aula 5 Hierarquia de Chomski, Gramáticas Regulares e Linguagens não regulares

Profa. Ariane Machado Lima
ariane.machado@usp.br

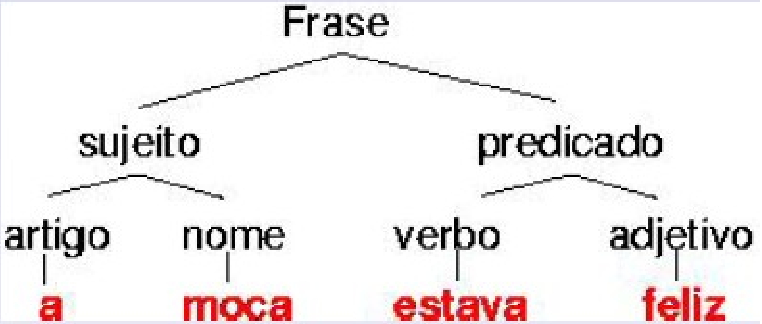
Lista 1

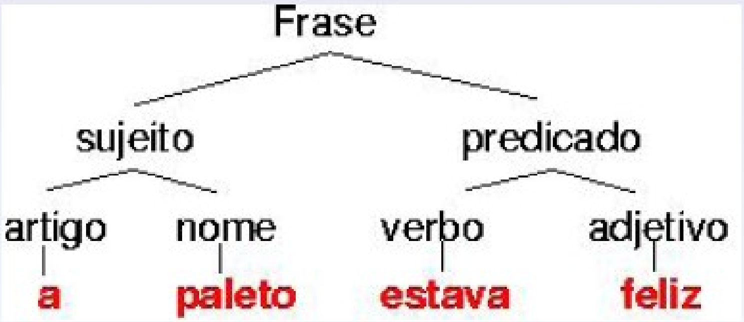
Entrega dia 27/08

Exercícios: 1.4f,g, 1.5c,g, 1.6e,g,h, 1.7e, 1.12, 1.14, 1.16, 1.21, 1.22, 1.24d,f, 1.27, 1.31

Gramáticas

Frase	→	sujeito	predicado
sujeito	→	artigo	nome
artigo	→	a	
artigo	→	o	
nome	→	paletó	
nome	→	moça	
nome	→	dia	
predicado	→	verbo	adjetivo
verbo	→	é	
verbo	→	estava	
adjectivo	→	feliz	
adjectivo	→	azul	



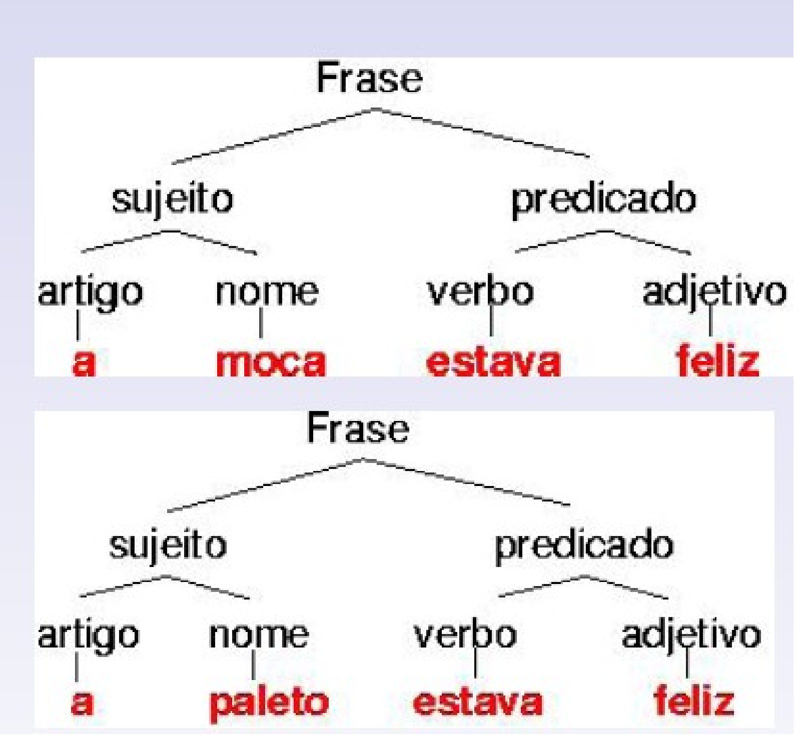


Gramáticas

conjunto de produções

símbolo inicial

Frase	→	sujeito	predicado
sujeito	→	artigo	nome
artigo	→	a	
artigo	→	o	
nome	→	paletó	
nome	→	moça	
nome	→	dia	
predicado	→	verbo	adjetivo
verbo	→	é	
verbo	→	estava	
adjectivo	→	feliz	
adjectivo	→	azul	



símbolos não-terminais

símbolos terminais

Gramáticas

- Definição: uma gramática G é uma quádrupla (V, Σ, S, P), onde
  - V é o conjunto de símbolos não-terminais (variáveis)
  - Σ é o conjunto de símbolos terminais
  - S é o símbolo inicial
  - P é o conjunto de produções da forma

$$(\Sigma \cup V)^* \, V \, (\Sigma \cup V)^* \rightarrow (\Sigma \cup V)^*$$

Gramáticas

- Uma **forma sentencial** de uma gramática G é qualquer cadeia obtida pela aplicação recorrente das seguintes regras:
  - S (símbolo inicial de G) é uma forma sentencial
  - Seja $\alpha\rho\beta$ uma forma sentencial de G e $\rho \rightarrow \gamma$ uma produção de G. Então $\alpha\gamma\beta$ é também uma forma sentencial de G .

$(\alpha,\beta,\gamma \in (\Sigma \cup V)^*$ e $\rho \in (\Sigma \cup V)^* V (\Sigma \cup V)^*)$

Gramáticas

- Uma **forma sentencial** de uma gramática G é qualquer cadeia obtida pela aplicação recorrente das seguintes regras:
  - S (símbolo inicial de G) é uma forma sentencial
  - Seja $\alpha\rho\beta$ uma forma sentencial de G e $\rho \rightarrow \gamma$ uma produção de G. Então $\alpha\gamma\beta$ é também uma forma sentencial de G .

  $(\alpha,\beta,\gamma \in (\Sigma \cup V)^* \text{ e } \rho \in (\Sigma \cup V)^* V (\Sigma \cup V)^*)$

-

Gramáticas

- Uma **forma sentencial** de uma gramática G é qualquer cadeia obtida pela aplicação recorrente das seguintes regras:
  - S (símbolo inicial de G) é uma forma sentencial
  - Seja $\alpha\rho\beta$ uma forma sentencial de G e $\rho \rightarrow \gamma$ uma produção de G. Então $\alpha\gamma\beta$ é também uma forma sentencial de G .

  $(\alpha,\beta,\gamma \in (\Sigma \cup V)^*$ e $\rho \in (\Sigma \cup V)^* V (\Sigma \cup V)^*)$
- Derivação direta:
  - $\alpha\rho\beta \Rightarrow \alpha\gamma\beta$

Gramáticas

- Derivação: aplicação de zero ou mais derivações diretas
  - $\alpha \Rightarrow^* \mu$
  - isto é, $\alpha \Rightarrow \beta \Rightarrow \ldots \Rightarrow \mu$
- Uma cadeia w ($w \in \Sigma^*$) é uma sentença de G se $S \Rightarrow^* w$
- Linguagem gerada por G:
  - $L(G) = \{ w \mid S \Rightarrow^* w \}$

Gramáticas - Exemplos

- $G = (V, \Sigma, S, P)$, onde
  - $V = \{S, A\}$
  - $\Sigma = \{0,1,2,3\}$
  - $S = S$
  - $P = \{$

$S \rightarrow 0S33$

$S \rightarrow A$

$A \rightarrow 12$

$A \rightarrow \varepsilon$

$\}$

Gramáticas - Exemplos

- $G = (V, \Sigma, S, P)$, onde
  - $V = \{S, A\}$
  - $\Sigma = \{0,1,2,3\}$
  - $S = S$
  - $P = \{$

    $S \rightarrow 0S33$

    $S \rightarrow A$

    $A \rightarrow 12$

    $A \rightarrow \varepsilon$

  $\}$

- Ex de formas sentenciais:

S, 0S33, 00S3333, 00A3333

- 0S33 => 00S3333
- 0S33 =>* 00A3333
- 0S33 =>* 0S33
- Ex de sentenças:

00123333, 12, ε

-

Gramáticas - Exemplos

- G = (V, Σ, S, P), onde
  - V = {S, A}
  - Σ = {0,1,2,3}
  - S = S
  - P = {

    $S \rightarrow 0S33$

    $S \rightarrow A$

    $A \rightarrow 12$

    $A \rightarrow \varepsilon$

  }

- Ex de formas sentenciais:

S, 0S33, 00S3333, 00A3333

- 0S33 => 00S3333
- 0S33 =>* 00A3333
- 0S33 =>* 0S33
- Ex de sentenças:

00123333, 12, ε

- L(G) =

Gramáticas - Exemplos

- G = (V, Σ, S, P), onde
  - V = {S, A}
  - Σ = {0,1,2,3}
  - S = S
  - P = {

    S → 0S33

    S → A

    A → 12

    A → ε

  }

- Ex de formas sentenciais:

S, 0S33, 00S3333, 00A3333

- 0S33 => 00S3333
- 0S33 =>* 00A3333
- 0S33 =>* 0S33
- Ex de sentenças:

00123333, 12, ε

- L(G) = $\{0^m 1^n 2^n 3^{2m} \mid m \geq 0 \text{ e } n = 0 \text{ ou } n = 1\}$

Gramáticas - Simplificação

- G = (V, Σ, S, P), onde
  - V = {S, A}
  - Σ = {0,1,2,3}
  - S = S
  - P = {

    $S \rightarrow 0S33$

    $S \rightarrow A$

    $A \rightarrow 12$

    $A \rightarrow \varepsilon$

  }

- G = (V, Σ, S, P), onde
  - V = {S, A}
  - Σ = {0,1,2,3}
  - S = S
  - P = {

    $S \rightarrow 0S33 \mid A$

    $A \rightarrow 12 \mid \varepsilon$

  }

Gramáticas

- Gramáticas são dispositivos generativos (geram cadeias)
- Dada uma cadeia w, reconhecer se w Є L(G) é um processo chamado análise sintática
- Dependendo do formato das produção, a análise sintática pode ser mais ou menos complexa

Gramáticas lineares

$\alpha \rightarrow \beta$

- Gramática linear à esquerda:
  - $\alpha \in V$
  - $\beta \in \Sigma$, $\beta \in V$, $\beta \in V\Sigma$, $\beta = \varepsilon$
- Gramática linear à direita:
  - $\alpha \in V$
  - $\beta \in \Sigma$, $\beta \in V$, $\beta \in \Sigma V$, $\beta = \varepsilon$

Gramáticas lineares

- Gramática linear à direita:
  - $\alpha \in V$
  - $\beta \in \Sigma$, $\beta \in V$, $\beta \in \Sigma V$, $\beta = \varepsilon$
- Gramáticas lineares à direita lembram alguma coisa?
- Ex: $G = (V, \Sigma, S, P)$, onde
  - $V = \{S, A\}$
  - $\Sigma = \{0,1,2\}$
  - $S = S$
  - $P = \{ S \rightarrow 0S, S \rightarrow A, S \rightarrow \varepsilon, A \rightarrow 1A, A \rightarrow 2\}$

Gramáticas lineares à direita => autômatos finitos

Gramáticas lineares à direita => autômatos finitos

$G = (V, \Sigma, S, P)$, $M = (Q, \Sigma, q_0, \delta, F)$

$Q = V \cup \{Z\}$, Z não pertence a Q

$q_0 = S$

$F = \{Z\}$

$\delta = \ldots$ (vou construir) $\delta \leftarrow \emptyset$

para cada produção em P

se $X \rightarrow aY$, então $\delta \leftarrow \delta \cup \{ (X,a) = Y \}$

se $X \rightarrow Y$, então $\delta \leftarrow \delta \cup \{ (X,\varepsilon) = Y \}$

se $X \rightarrow a$, então $\delta \leftarrow \delta \cup \{ (X,a) = Z \}$

se $X \rightarrow \varepsilon$, então $\delta \leftarrow \delta \cup \{ (X,\varepsilon) = Z \}$

Autômatos finitos => Gramáticas lineares à direita

Autômatos finitos => Gramáticas lineares à direita

$M = (Q, \Sigma, q_0, \delta, F)$, $G = (V, \Sigma, S, P)$

$V = Q$

$S = q_0$

$P = \ldots$ (vou construir) $P \leftarrow \emptyset$

para cada transição de δ

Se $\delta(X,a) = Y$ então $P \leftarrow \{X \rightarrow aY\}$

Se $\delta(X,\varepsilon) = Y$ então $P \leftarrow \{X \rightarrow Y\}$

para cada estado X de F

$P \leftarrow \{X \rightarrow \varepsilon\}$

Gramáticas lineares

- Gramáticas lineares à direita são equivalentes a autômatos finitos (geram a mesma linguagem)
- Duas gramáticas são equivalentes se elas geram a mesma linguagem
- Gramáticas lineares à direita e lineares à esquerda são equivalentes
- Gramáticas lineares geram linguagens regulares
- Uma gramática é regular se ela for linear à esquerda ou linear à direita

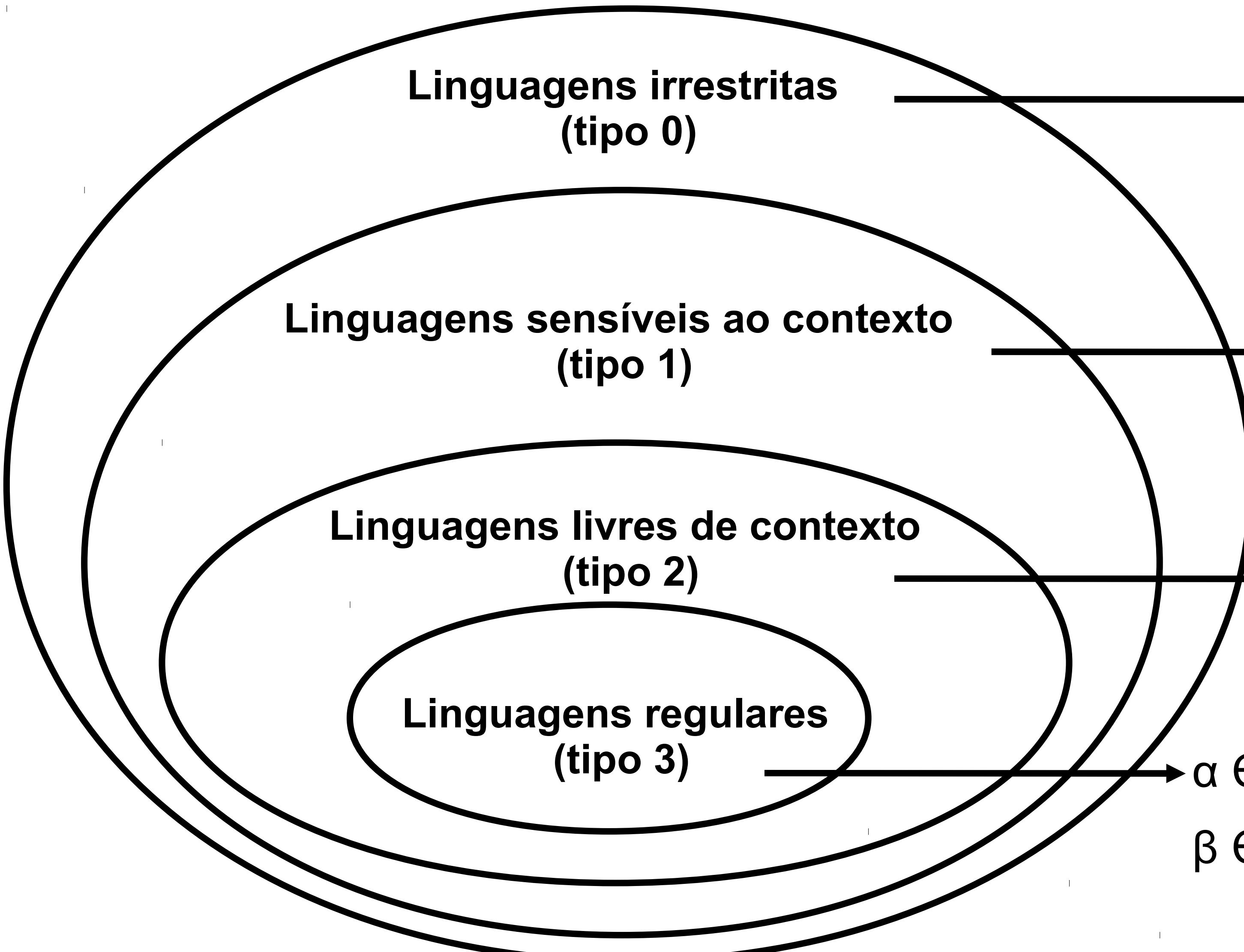
Gramáticas

- Gramáticas são dispositivos generativos (geram cadeias)
- Dada uma cadeia w, reconhecer se w Є L(G) é um processo chamado análise sintática
- Dependendo do formato das produção, a análise sintática pode ser mais ou menos complexa

Hierarquia de Chomsky

- Hierarquia das linguagens em classes de acordo com a sua complexidade relativa (Noam Chomsky, 1956)
- Cada classe de linguagem pode ser gerada por um tipo de gramática (formato das produções)
- Cada tipo de gramática tem uma complexidade de análise sintática diferente
- Na prática: dada uma linguagem, saber qual o dispositivo mais eficiente para análise sintática

Hierarquia de Chomsky $\alpha \rightarrow \beta$



$\alpha \in (V \cup \Sigma)^* V (V \cup \Sigma)^*$
$\beta \in (V \cup \Sigma)^*$

$\alpha \in (V \cup \Sigma)^* V (V \cup \Sigma)^*$
$\beta \in (V \cup \Sigma)^*$
$|\alpha| <= |\beta|$

$\alpha \in V$

$\beta \in (V \cup \Sigma)^*$

$\alpha \in V$

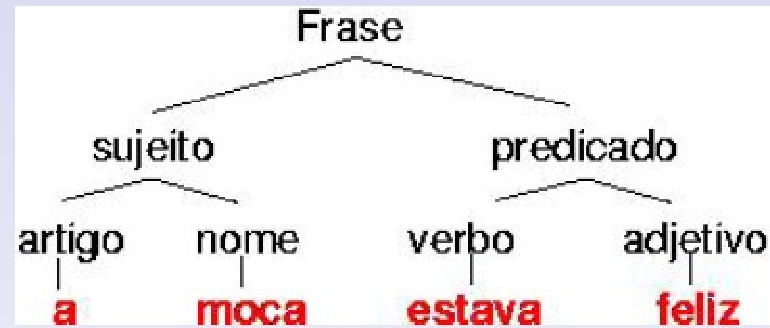
$\beta \in \Sigma_\varepsilon$, $\beta \in V$, $\beta \in$ ($V\Sigma$ ou ΣV)

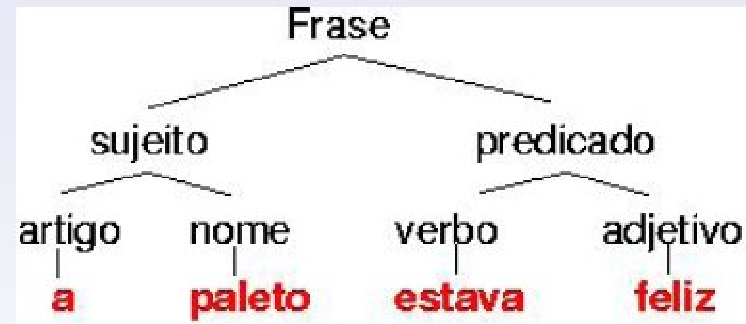
Hierarquia de Chomsky

Tipo	Classe de linguagens	Modelo de gramática	Modelo de reconhecedor
0	Irrestritas ou Recursivamente enumeráveis	Irrestrita	Máquina de Turing
1	Sensíveis ao contexto	Sensível ao contexto	Máquina de Turing com fita limitada
2	Livres de contexto	Livre de contexto	Autômato a pilha
3	Regulares	Regular (linear à direita ou à esquerda)	Autômato finito

Gramáticas estocásticas

Frase	→	sujeito	predicado	[1.0]
sujeito	→	artigo	nome	[1.0]
artigo	→	a		[0.3]
artigo	→	o		[0.7]
nome	→	paletó		[0.2]
nome	→	moça		[0.3]
nome	→	dia		[0.5]
predicado	→	verbo	adjetivo	[1.0]
verbo	→	é		[0.7]
verbo	→	estava		[0.3]
adjectivo	→	feliz		[0.6]
adjectivo	→	azul		[0.4]





P(x, t) = produto das probabilidades das produções usadas na derivação de x

$P(x) = \sum_i P(x, t_i)$

Gramáticas Estocásticas

- Definição: uma gramática estocática G é uma quíntupla (V, Σ, S, P, ρ), onde
  - V é o conjunto de símbolos não-terminais (variáveis)
  - Σ é o conjunto de símbolos terminais
  - S é o símbolo inicial
  - P é o conjunto de produções da forma

    $(\Sigma \cup V)^* \, V \, (\Sigma \cup V)^* \rightarrow (\Sigma \cup V)^*$
  - ρ é o conjunto de distribuições de probabilidades sobre as produções de mesmo lado esquerdo

    $\sum_i \rho(\alpha \rightarrow \beta_i) = 1$

1.4 – Linguagens não-regulares

Linguagens não-regulares

- A linguagem $B = \{0^n1^n \mid n >= 0\}$ é regular?
- Como provar que uma linguagem não pode ser reconhecida por um autômato finito?

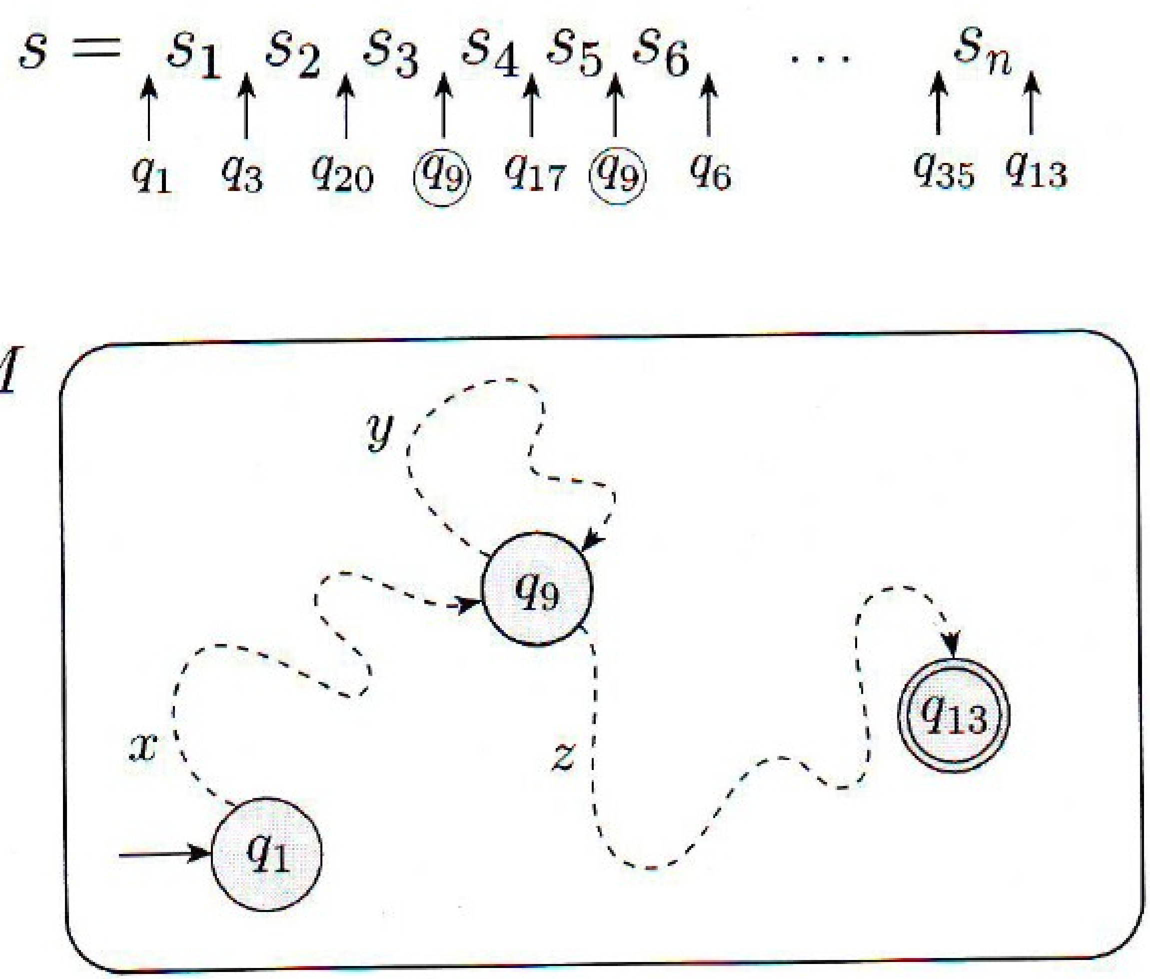
TEOREMA 1.70

Lema do bombeamento Se A é uma linguagem regular, então existe um número p (o comprimento de bombeamento) tal que, se s é qualquer cadeia de A de comprimento no mínimo p, então s pode ser dividida em três partes, $s = xyz$, satisfazendo as seguintes condições:

1. para cada $i \geq 0$, $xy^iz \in A$,
2. $|y| > 0$, e
3. $|xy| \leq p$.

Ideia da prova

usamos p = número de estados do AFD



Prova

PROVA Seja $M = (Q, \Sigma, \delta, q_1, F)$ um AFD que reconhece A e p o número de estados de M.

Seja $s = s_1 s_2 \cdots s_n$ uma cadeia em A de comprimento n, onde $n \geq p$. Seja $r_1, \ldots, r_{n+1}$ a seqüência de estados nos quais M passa enquanto processa s, de forma que $r_{i+1} = \delta(r_i, s_i)$ para $1 \leq i \leq n$. Essa seqüência tem comprimento $n + 1$, que é pelo menos $p + 1$. Entre os primeiros $p + 1$ elementos da seqüência, dois devem ser o mesmo estado, pelo princípio da casa de pombos. Chamamos o primeiro desses de r_j e o segundo de r_l. Como r_l ocorre entre as primeiras $p + 1$ posições da seqüência começando em r_1, temos que $l \leq p + 1$. Agora, seja $x = s_1 \cdots s_{j-1}$, $y = s_j \cdots s_{l-1}$ e $z = s_l \cdots s_n$.

Como x leva M de r_1 para r_j, y leva M de r_j para r_j e z leva M de r_j para r_{n+1}, que é um estado de aceitação, M deve aceitar xy^iz para $i \geq 0$. Sabemos que $j \neq l$, e portanto $|y| > 0$; e $l \leq p + 1$, e logo $|xy| \leq p$. Dessa forma, satisfizemos todas as condições do lema do bombeamento.

EXEMPLO 1.73

Seja B a linguagem $\{0^n1^n | n \geq 0\}$. Usamos o lema do bombeamento para provar que B não é regular. A prova é por contradição.

Suponha, ao contrário, que B seja regular. Seja p o comprimento de bombeamento dado pelo lema do bombeamento. Escolha s como a cadeia 0^p1^p. Como s é um membro de B e tem comprimento maior que p, o lema do bombeamento garante que s pode ser dividida em três partes, $s = xyz$, onde para qualquer $i \geq 0$ a cadeia xy^iz está em B. Consideramos três casos para mostrar que esse resultado é impossível.

1. A cadeia y contém apenas 0s. Neste caso, a cadeia $xyyz$ tem mais 0s que 1s e, portanto, não é um membro de B, violando a condição 1 do lema do bombeamento. Esse caso é uma contradição.
2. A cadeia y contém somente 1s. Esse caso também dá uma contradição.
3. A cadeia y contém ambos, 0s e 1s. Nesse caso, a cadeia $xyyz$ pode ter o mesmo número de 0s e 1s, mas eles estarão fora de ordem, com alguns 1s antes de 0s. Logo, ela não é um membro de B, o que é uma contradição.

EXEMPLO 1.74

Seja $C = \{w|\, w$ tem número igual de 0s e 1s$\}$. Usamos o lema do bombeamento para provar que C não é regular. A prova é por contradição.

$$0^p1^p$$

Se fizermos x e z serem a cadeia vazia e y ser a cadeia 0^p1^p, então xy^iz sempre terá um número igual de 0s e 1s e, portanto, está em C.

Logo, *parece* que s pode ser bombeada.

EXEMPLO 1.74

Seja $C = \{w| \, w$ tem número igual de 0s e 1s$\}$. Usamos o lema do bombeamento para provar que C não é regular. A prova é por contradição.

$$0^p 1^p$$

Se fizermos x e z serem a cadeia vazia e y ser a cadeia $0^p 1^p$, então $xy^i z$ sempre terá um número igual de 0s e 1s e, portanto, está em C.

Logo, *parece* que s pode ser bombeada.

Aqui a condição 3 no lema do bombeamento é útil.

Se $|xy| \leq p$, então y deve conter somente 0s; logo, $xyyz \notin C$.

Por conseguinte, s não pode ser bombeada.

EXEMPLO 1.74

Seja $C = \{w|\, w$ tem número igual de 0s e 1s$\}$. Usamos o lema do bombeamento para provar que C não é regular. A prova é por contradição.

$$0^p 1^p$$

Se fizermos x e z serem a cadeia vazia e y ser a cadeia $0^p 1^p$, então $xy^i z$ sempre terá um número igual de 0s e 1s e, portanto, está em C.

Logo, *parece* que s pode ser bombeada.

Aqui a condição 3 no lema do bombeamento é útil.

Se $|xy| \leq p$, então y deve conter somente 0s; logo, $xyyz \notin C$.

Por conseguinte, s não pode ser bombeada.

Cuidado: $s = (01)^p$ $\quad x = \varepsilon$, $y = 01$ e $z = (01)^{p-1}$

EXEMPLO 1.74

Seja $C = \{w \mid w$ tem número igual de 0s e 1s$\}$. Usamos o lema do bombeamento para provar que C não é regular. A prova é por contradição.

$$0^p1^p$$

Se fizermos x e z serem a cadeia vazia e y ser a cadeia 0^p1^p, então xy^iz sempre terá um número igual de 0s e 1s e, portanto, está em C.

Logo, *parece* que s pode ser bombeada.

Aqui a condição 3 no lema do bombeamento é útil.

Se $|xy| \leq p$, então y deve conter somente 0s; logo, $xyyz \notin C$.

Por conseguinte, s não pode ser bombeada.

Cuidado: $s = (01)^p$ $\quad x = \varepsilon$, $y = 01$ e $z = (01)^{p-1}$

Um método alternativo de provar que C é não-regular segue de nosso conhecimento que B é não-regular. Se C fosse regular, $C \cap 0^*1^*$ também seria regular. Os motivos são que a linguagem 0^*1^* é regular e que a classe das linguagens regulares é fechada sob interseção, resultado que provamos na nota de rodapé 3 (página 47). Mas $C \cap 0^*1^*$ é igual a B, e sabemos que B é não-regular do Exemplo 1.73.

EXEMPLO 1.77

$$E = \{0^i 1^j | i > j\}$$

$$s = 0^{p+1} 1^p.$$

Condição 3 => y contém somente zeros

xyyz ainda está em E

EXEMPLO 1.77

$$E = \{0^i 1^j | i > j\}$$

$$s = 0^{p+1} 1^p.$$

Condição 3 => y contém somente zeros

xyyz ainda está em E

Mas $xy^0z = xz$ não está

Não esqueça que você pode tentar bombear para baixo!