



# Universidade de Aveiro

Departamento de Electrónica, Telecomunicações e Informática

## Linguagens Formais e Autómatos

Exame de recurso

(Ano Lectivo de 2013/14)

8 de Julho de 2014

NOTA: O exame tem 13 questões. As 3 mais bem classificadas serão cotadas a 2,0 valores cada; as restantes serão cotadas a 1,4 valores cada.

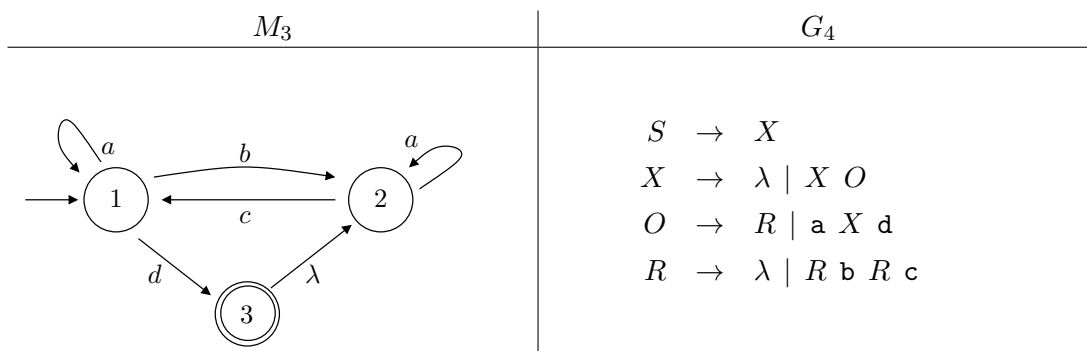
1. Considere, sobre o alfabeto  $T = \{a, b, c, d\}$ , as linguagens  $L_1$ ,  $L_2$ ,  $L_3$ , e  $L_4$  definidas da seguinte forma:

$$L_1 = \{a^n(bc)^k d^n : n \geq 0 \wedge k > 0\}$$

$$L_2 = \{w \in T^* : w \text{ é gerada pela expressão regular } e_2 = a^*(b|c)^*(cd)^*\}$$

$$L_3 = \{w \in T^* : w \text{ é reconhecida pelo autómato } M_3\}$$

$$L_4 = \{w \in T^* : w \text{ é gerada pela gramática } G_4\}$$



- Mostre que  $abcd \in L_3 \cap L_4$ .
- Mostre que  $L_1 \subset L_4$ .
- Determine uma gramática regular que represente a linguagem  $L_2$ .
- Determine uma expressão regular que represente a linguagem  $L = \{\omega_1 \omega_2 : \omega_1 \in L_3 \wedge \omega_2 \in L_2\}$ .  
Apresente o raciocínio e/ou os passos intermédios usados para chegar à sua resposta.
- Construa um autómato finito que reconheça a linguagem  $\overline{L_3}$  (complementar de  $L_3$ ).  
Apresente o raciocínio e/ou os passos intermédios usados para chegar à sua resposta.
- Projecte uma gramática independente do contexto que represente a linguagem  $L_1$ .
- $d \in \text{predict}(X \rightarrow OX)$ . Explique porquê.
- A gramática  $G_4$  possui recursividade à esquerda. Obtenha uma gramática equivalente sem esse tipo de recursividade.
- O teorema da repetição ou da bombagem (*pumping lemma*) diz que se  $L$  é uma linguagem regular, existe um número  $p > 0$  tal que se  $u$  é uma palavra qualquer de  $L$  com  $|u| \geq p$ , então pode-se escrever  $u = xyz$ , satisfazendo as condições:  $|y| > 0$ ;  $|xy| \leq p$ ; e  $xy^i z \in L$ , para qualquer  $i \geq 0$ .  
Mostre, usando o teorema da repetição, que  $L_1$  é uma linguagem não regular.

2. Sobre o alfabeto  $T_5 = \{c, t, s, i, =, e\}$  considere a gramática  $G_5$  dada a seguir e seja  $L_5$  a linguagem por ela descrita.

$$\begin{aligned} S &\rightarrow D \\ D &\rightarrow t L_1 \mid c t L_2 \\ L_1 &\rightarrow i \mid L_1 s i \\ L_2 &\rightarrow V \mid L_2 s V \\ V &\rightarrow i \mid i = e \end{aligned}$$

Considere ainda a coleção canônica de conjuntos de itens, usada na construção de um reconhecedor (*parser*) ascendente, parcialmente apresentada a seguir e onde a função  $\delta(Z_i, a)$  representa a transição de estado.

$$\begin{aligned} Z_0 &= \{ S \rightarrow \bullet D, D \rightarrow \bullet t L_1, D \rightarrow \bullet c t L_2 \} \\ Z_1 &= \delta(Z_0, D) = \{ S \rightarrow D \bullet \} \\ Z_2 &= \delta(Z_0, t) = \{ D \rightarrow t \bullet L_1, L_1 \rightarrow \bullet i, L_1 \rightarrow \bullet L_1 s i \} \\ Z_3 &= \delta(Z_0, c) = \{ D \rightarrow c \bullet t L_2 \} \\ Z_4 &= \delta(Z_2, L_1) = \{ D \rightarrow t L_1 \bullet, L_1 \rightarrow L_1 \bullet s i \} \\ Z_5 &= \delta(Z_2, i) = \{ L_1 \rightarrow i \bullet \} \\ Z_6 &= \delta(Z_3, t) = \{ \dots \} \\ Z_7 &= \delta(Z_4, s) = \{ \dots \} \\ &\dots \end{aligned}$$

- Trace as árvores de derivação das palavras “ $c t i = e$ ” e “ $t i s i = e$ ”.
- Preencha as linhas da tabela de reconhecimento (*parsing*) para um reconhecedor ascendente relativamente aos estados  $Z_0$  a  $Z_5$ .
- Determine o valor dos estados (conjuntos)  $Z_6$  e  $Z_7$  e de mais 3 à sua escolha.
- A gramática  $G_5$  representa uma abstração de uma declaração de constantes ou de variáveis. Sabendo que:
  - o símbolo terminal **c** indica que se trata de uma declaração de constantes;
  - o símbolo terminal **t** representa o tipo e possui um atributo chamado **type** que representa o tipo específico que lhe está associado.
  - o símbolo terminal **i** representa uma variável ou constante e tem um atributo chamado **name** que representa o nome da variável ou constante que lhe está associado.
  - o símbolo terminal **e** representa a inicialização e tem um atributo chamado **value** que representa o valor da constante ou valor inicial da variável a que está associado.
  - se dispõe de uma função de manipulação de uma tabela de símbolos para inserções de novas entradas (variáveis ou constantes), com a assinatura **addsym(cv, nm, tp, vl)**, onde
    - nm** representa o nome da variável;
    - tp** representa o tipo específico;
    - vl** representa o valor a atribuir à variável/constante;
    - cv** é um valor booleano, que indica se se trata da inserção de uma contante ou de uma variável.

Na invocação, o argumento **vl** pode ser omitido para assumir um valor por defeito. Construa uma gramática de atributos que permita invocar a função **addsym** de forma adequada por cada constante ou variável declarada.