Departamento de Ciência de Computadores — FCUP

# Modelos de Computação (CC1004) — 2014/15

2º Teste (06.06.2015)

**Resolução de questões selecionadas e observações**

**1.** Seja $L$ a linguagem de alfabeto $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$ que é gerada pela gramática $G$ dada por:

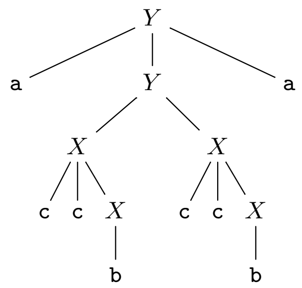
$$G = (\{X, Y\},\ \Sigma,\ \{Y \to \mathtt{a}Y\mathtt{a},\ Y \to XX,\ Y \to \mathtt{b},\ X \to \mathtt{cc}X,\ X \to \mathtt{b},\ X \to \mathtt{b}X\},\ Y)$$

**a)** Prove que $\mathtt{accbccba} \in \mathcal{L}(G)$, apresentando uma derivação e a árvore de derivação correspondente.

***Resposta:*** Uma derivação de $\mathtt{accbccba}$ a partir do símbolo inicial da gramática dada:

$$Y \Rightarrow \mathtt{a}Y\mathtt{a} \Rightarrow \mathtt{a}XX\mathtt{a} \Rightarrow \mathtt{acc}XX\mathtt{a} \Rightarrow \mathtt{accb}X\mathtt{a} \Rightarrow \mathtt{accbcc}X\mathtt{a} \Rightarrow \mathtt{accbccba}$$

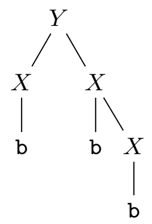
A árvore de derivação correspondente:

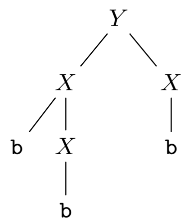


**b)** Prove que a gramática $G$ é ambígua.

***Resposta:***

Por exemplo, $\mathtt{bbb} \in \mathcal{L}(G)$ tem duas árvores de derivação distintas.





**c)** Indique uma expressão regular que descreva a linguagem de $\Sigma^\star$ que se pode gerar a partir da variável $X$.

***Resposta:*** $(\mathtt{cc} + \mathtt{b})^\star\mathtt{b}$

**d)** Indique a forma genérica das palavras de $L$. Explique sucintamente como chegou a essa conclusão, recorrendo a $\Rightarrow_G^\star$, $\Rightarrow_G$, e $\Rightarrow_G^n$, com $n \in \mathbb{N}$.

***Resposta:*** As palavras podem ser de duas formas: $\mathtt{a}^n\mathtt{ba}^n$ ou $\mathtt{a}^n w_1 w_2 \mathtt{a}^n$, com $n \in \mathbb{N}$ e $w_1, w_2 \in \mathcal{L}((\mathtt{cc}+\mathtt{b})^\star\mathtt{b})$.

O valor de $n$ é igual ao número de vezes que a regra $Y \to \mathtt{a}Y\mathtt{a}$ é usada na derivação da palavra. Em qualquer derivação, se tal regra for aplicada, só o pode ser nos primeiros passos porque, se se usar qualquer uma das outras $Y$-produções, a variável $Y$ não ocorrerá mais na derivação. Quando se aplica a regra $Y \to \mathtt{b}$, no passo $n+1$ (que é o primeiro se $n = 0$), obtém-se uma palavra da forma $\mathtt{a}^n\mathtt{ba}^n$ pois $Y \Rightarrow^n \mathtt{a}^n Y \mathtt{a}^n \Rightarrow \mathtt{a}^n\mathtt{ba}^n$. Mas, se em vez de $Y \to \mathtt{b}$ se aplicar $Y \to XX$ e depois $X$-produções para substituir as duas variáveis $X$ introduzidas, obtém-se uma palavra da forma $\mathtt{a}^n w_1 w_2 \mathtt{a}^n$, sendo $w_1$ e $w_2$ as palavras derivadas do primeiro e do segundo $X$, respetivamente, pois $Y \Rightarrow^n \mathtt{a}^n Y \mathtt{a}^n \Rightarrow \mathtt{a}^n XX \mathtt{a}^n \Rightarrow^\star \mathtt{a}^n w_1 X \mathtt{a}^n \Rightarrow^\star \mathtt{a}^n w_1 w_2 \mathtt{a}^n$.

**e)** Usando ou o teorema de Myhill-Nerode ou o lema da repetição, e **1d)**, prove que $L$ não é regular.

***Resposta:*** (NB: Apenas uma das duas provas seguintes poderia ser apresentada.)

- *Prova pelo teorema de Myhill-Nerode:*
  $L$ não é regular porque o número de classes de equivalência da relação $R_L$ é infinito, pois, quaisquer que sejam $n$ e $m$, com $n \neq m$, as palavras $\mathtt{a}^n$ e $\mathtt{a}^m$ não são equivalentes segundo $R_L$, já que, para $z = \mathtt{ba}^n$, tem-se $\mathtt{a}^n z \in L$ e $\mathtt{a}^m z \notin L$. Portanto, o conjunto das classes de equivalência de $R_L$ inclui o conjunto infinito $\{\,[\mathtt{a}^n] \mid n \in \mathbb{N}\}$. Logo, é infinito.
- *Prova pelo lema da repetição para linguagens regulares:*
  $L$ não é regular porque não satisfaz a condição do lema da repetição referido. Dado $n > 0$, tomemos $z = \mathtt{a}^n\mathtt{ba}^n$. Então, $z \in L$ e $|z| = 2n+1 \geq n$, mas nenhum prefixo de $z$ com comprimento menor ou igual a $n$ tem uma subpalavra não vazia $v$ que possa ser retirada (nem repetida, mas basta saber que não pode ser retirada). De facto, se decompusermos $z = \mathtt{a}^n\mathtt{ba}^n$, como $uvw$, com $|uv| \leq n$, o prefixo $uv$ é formado por a's do primeiro bloco. Se cortarmos $v$ (i.e., tomarmos $i = 0$, na condição do lema), a palavra $uv^0w$ é $\mathtt{a}^{n-|v|}\mathtt{ba}^n$ e tem menos a's à esquerda do que à direita de b. Logo, $uv^0w \notin L$, qualquer que seja o prefixo $uv$ nas condições indicadas.

**f)** Converta $G$ à forma normal de Chomsky e aplique o algoritmo CYK para decidir se bbaba pertence a $L$. Apresente alguns dos passos intermédios (mais complexos) detalhadamente. Por análise do resultado final, diga ainda, justificando, quais das subpalavras próprias de bbaba pertencem a $L$.

***Resposta:*** A gramática $G$ não tem produções unitárias nem produções-$\varepsilon$. As regras $X \to \mathtt{b}$, $Y \to \mathtt{b}$ e $Y \to XX$ estão normalizadas. Assim, basta aplicar o método de normalização para transformar as regras $Y \to \mathtt{a}Y\mathtt{a}$, $X \to \mathtt{cc}X$ e $X \to \mathtt{b}X$, por exemplo assim:

$$\begin{array}{lll@{\qquad}lll@{\qquad}lll} Y & \to & AW & X & \to & CR & X & \to & BX \\ W & \to & YA & R & \to & CX & B & \to & \mathtt{b} \\ A & \to & \mathtt{a} & C & \to & \mathtt{c} & & & \end{array}$$

Logo, $G$ é equivalente à GIC na forma normal de Chomsky $G' = (\{X, Y, A, B, C, W, R\}, \Sigma, P', Y)$, para $P'$ dado por:

$$\begin{array}{lll} Y & \to & AW \mid XX \mid \mathtt{b} \\ X & \to & CR \mid BX \mid \mathtt{b} \\ W & \to & YA \\ R & \to & CX \\ A & \to & \mathtt{a} \\ B & \to & \mathtt{b} \\ C & \to & \mathtt{c} \end{array}$$

Análise da palavra bbaba pelo algoritmo de Cocke-Younger-Kasami (CYK):

| | | | | | |
|---|---|---|---|---|---|
| #5 | $x_1x_2x_3x_4x_5$ $\emptyset$ | | | | |
| #4 | $x_1x_2x_3x_4$ $\emptyset$ | $x_2x_3x_4x_5$ $\emptyset$ | | | |
| #3 | $x_1x_2x_3$ $\{W\}$ | $x_2x_3x_4$ $\emptyset$ | $x_3x_4x_5$ $\{Y\}$ | | |
| #2 | $x_1x_2$ $\{Y, X\}$ | $x_2x_3$ $\{W\}$ | $x_3x_4$ $\emptyset$ | $x_4x_5$ $\{W\}$ | |
| #1 | $x_1$ $\{B, X, Y\}$ | $x_2$ $\{B, X, Y\}$ | $x_3$ $\{A\}$ | $x_4$ $\{B, X, Y\}$ | $x_5$ $\{A\}$ |
| | b | b | a | b | a |

A palavra bbaba não pertence a $\mathcal{L}(G') = L$ pois no topo da tabela está $\emptyset$. Procurando as entradas que incluem a variável $Y$, vemos que as subpalavras de bbaba $= x_1x_2x_3x_4x_5$ que pertencem a $L$ são $x_1$, $x_2$, $x_4$, $x_1x_2$ e $x_3x_4x_5$, ou seja, b, bb, e aba.

A tabela foi preenchida por linhas, começando de baixo para cima.

Para $x_1x_2$, analisámos $\{B, X, Y\}\{B, X, Y\} = \{BB, BX, BY, XB, XX, XY, YB, YX, YY\}$. Apenas $BX$ e $XX$ são lado direito de regras (nomeadamente de $X \to BX$ e de $Y \to XX$). Logo, $x_1x_2$ pode ser gerado por $X$ ou $Y$, apenas.

Para determinar o conjunto de variáveis que geram $x_1x_2x_3x_4$ em $G'$, analisámos as três decomposições possíveis, representadas abaixo, concluindo que nenhuma variável gerava essa palavra.

$$\begin{array}{l} x_1|x_2x_3x_4 = \{B, X, Y\}\emptyset = \emptyset \\ x_1x_2|x_3x_4 = \{Y, X\}\emptyset = \emptyset \\ x_1x_2x_3|x_4 = \{W\}\{B, X, Y\} = \{WB, WX, WY\} \end{array}$$

**g)** Averigue se existem GICs lineares à direita ou lineares à esquerda que sejam equivalentes a $G$. Justifique.

***Resposta:*** O enunciado diz que $L = \mathcal{L}(G)$ não é regular (c.f., **1e)**). Portanto, não existem GICs lineares à direita nem GICs lineares à esquerda equivalentes a $G$, pois provou-se nas aulas que as GICs desse tipo geram linguagens regulares.

**2.** Seja $r$ a expressão regular $((1 + \varepsilon)(((01) + 0)^\star))$ sobre $\Sigma = \{0, 1\}$.

**a)** Desenhe o diagrama de transição do AFND-$\varepsilon$ que resulta da aplicação do método de Thompson à expressão $r$, de acordo com a construção dada nas aulas. Por aplicação do método de conversão baseado em subconjuntos, converta esse AFND-$\varepsilon$ num AFD equivalente (considere apenas os estados acessíveis do estado inicial do AFD e preserve as designações de estados resultantes do método de conversão).

**b)** Por aplicação do método de Moore, minimize o AFD que obteve em **2a)**. Deve apresentar a sequência de passos intermédios e, se for útil, pode começar por renomear os estados do AFD de partida.

***Observações sobre 2a) e 2b):***

A **construção de Thompson dada nas aulas (em 2014/15)** consta dos Apontamentos (na secção 3.3.3, págs 59 e 61). Para o fecho de Kleene e para a concatenação as construções são as que constam como "Construções alternativas" (no fim da pág 61). No último esquema da pág 61 (fecho de Kleene), existe uma transição de $f_1$ para $i_1$ por $\varepsilon$, que pode aparecer distorcida (a definição de $A_\star$ clarifica-a). **Qualquer resolução que não siga exatamente estas construções está sempre errada.**

O **método de conversão do AFND-$\varepsilon$ para um AFD** resulta da prova da Proposição 14, pág 49. Em particular, o estado inicial do AFD equivalente é o $Fecho_\varepsilon(s_0)$, sendo $s_0$ o estado inicial do AFND-$\varepsilon$. Portanto, o estado inicial do AFD é um subconjunto de estados que inclui $s_0$ mas pode incluir outros estados, como, no Exemplo 52 (pág 50), em que é $\{s_0, s_1, s_3\}$. Os estados do AFD são representados como subconjuntos de estados do AFND-$\varepsilon$. De acordo com o enunciado de 2a), na resposta, essa designação terá de ser preservada.

Para recordar o **algoritmo de Moore** (para minimização de um AFD), consultar os Apontamentos (págs. 85-89). Foram também fornecidos alguns apontamentos complementares, e mais exemplos, na resolução da Folha Prática 7 (págs 5-10).

**3.** Justifique a veracidade ou falsidade de cada uma das afirmações seguintes, onde o alfabeto é $\Sigma = \{\mathtt{a}, \mathtt{b}\}$.

**a)** O autómato de pilha $(\{s\}, \Sigma, \{\mathtt{K}, \mathtt{B}\}, \delta, s, \mathtt{K}, \{\,\})$, com $\delta(s, \mathtt{b}, \mathtt{K}) = \{(s, \mathtt{BB}), (s, \varepsilon)\}$, $\delta(s, \varepsilon, \mathtt{K}) = \{(s, \varepsilon)\}$, $\delta(s, \mathtt{b}, \mathtt{B}) = \{(s, \mathtt{BB})\}$, $\delta(s, \mathtt{a}, \mathtt{B}) = \{(s, \varepsilon)\}$, e $\delta(s, \varepsilon, \mathtt{B}) = \delta(s, \mathtt{a}, \mathtt{K}) = \{\,\}$, aceita bbaaaa por pilha vazia.

***Resposta:*** A afirmação é falsa. A partir da configuração inicial $(s, \mathtt{bbaaaa}, \mathtt{K})$, há três alternativas possíveis que levam a que a pilha fique vazia. Contudo, em nenhum caso a configuração a que se chega é $(s, \varepsilon, \varepsilon)$, que seria a de aceitação por pilha vazia.

- $(s, \mathtt{bbaaaa}, \mathtt{K}) \vdash (s, \mathtt{bbaaaa}, \varepsilon)$, não pode prosseguir.
- $(s, \mathtt{bbaaaa}, \mathtt{K}) \vdash (s, \mathtt{baaaa}, \varepsilon)$, não pode prosseguir.
- $(s, \mathtt{bbaaaa}, \mathtt{K}) \vdash (s, \mathtt{baaaa}, \mathtt{BB}) \vdash (s, \mathtt{aaaa}, \mathtt{BBB}) \vdash (s, \mathtt{aaa}, \mathtt{BB}) \vdash (s, \mathtt{aa}, \mathtt{B}) \vdash (s, \mathtt{a}, \varepsilon)$, não pode prosseguir.

Embora a pilha fique vazia, a palavra é rejeitada porque não foi toda consumida.

**b)** Existe uma linguagem regular $L$ tal que $(\Sigma^\star \setminus L)\{\mathtt{a}^n\mathtt{b}^n \mid n \geq 1\}$ não é independente de contexto.

***Resposta:*** Como $L$ é regular, também $(\Sigma^\star \setminus L)$ é regular e, consequentemente, $(\Sigma^\star \setminus L)$ é independente de contexto. Por outro lado, $\{\mathtt{a}^n\mathtt{b}^n \mid n \geq 1\}$ é independente de contexto, pois é gerada pela GIC $G = (\{S\}, \Sigma, \{S \to \mathtt{ab},\ S \to \mathtt{a}S\mathtt{b}\}, S)$. Portanto, a afirmação é falsa, já que a concatenação de linguagens independentes de contexto é independente de contexto.

**4.** Seguindo a caraterização do AFD mínimo dada pelo teorema de Myhill-Nerode, construa o AFD mínimo que aceita a linguagem das palavras de $\{0,1\}^\star$ que têm número par de 0's ou têm 11 como subpalavra.

***Resposta:***

**NB:** Zero é um número par. Qualquer palavra é subpalavra de si mesma.

O estado inicial do AFD mínimo é $[\varepsilon]$ e, como $\varepsilon \in L$, também é um estado final.

$\delta([\varepsilon], 0) \stackrel{\text{def}}{=} [0] \neq [\varepsilon]$, porque $0 \notin L$ e $\varepsilon \in L$ (portanto, basta tomar $z = \varepsilon$). Logo, $[0]$ é um novo estado não final.

$\delta([\varepsilon], 1) \stackrel{\text{def}}{=} [1] \neq [\varepsilon]$, porque para $z = 10$ tem-se $\varepsilon z = 10 \notin L$ e $1z = 110 \in L$; também $[1] \neq [0]$, pois $1 \in L$ e $0 \notin L$. Logo, $[1]$ é um novo estado final.

$\delta([0], 0) \stackrel{\text{def}}{=} [00] = [\varepsilon]$, porque $00z \in L$ se e só se $z$ tem número par de 0's ou 11 como subpalavra. Portanto, 00 e $\varepsilon$ são equivalentes segundo $R_L$, já que $\varepsilon z \in L$ se e só se $z \in L$.

$\delta([0], 1) \stackrel{\text{def}}{=} [01] \neq [0]$, porque para $z = 1$ tem-se $01z = 011 \in L$ e $0z = 01 \notin L$. A classe $[01]$ é um estado não final.

$\delta([1], 1) \stackrel{\text{def}}{=} [11] \neq [\varepsilon]$, pois $11z \in L$, para todo $z \in \{0,1\}^\star$ (enquanto $\varepsilon z \in L$ sse $z \in L$).
Também $[11] \neq [1]$, pois para $z = 0$ tem $1z = 10 \notin L$ e $11z = 110 \in L$. A classe $[11]$ é um novo estado final.

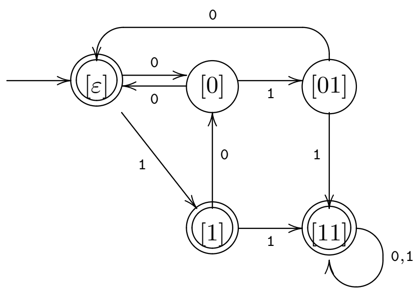
$\delta([1], 0) \stackrel{\text{def}}{=} [10] = [0]$, porque tem-se $0z \in L \Leftrightarrow 10z \in L \Leftrightarrow$ "$z$ tem número ímpar de 0's ou tem 11 como subpalavra".

$\delta([01], 0) \stackrel{\text{def}}{=} [010] = [\varepsilon]$, porque $010z \in L$ se e só se $z$ tem número par de 0's ou tem 11 como subpalavra.

$\delta([01], 1) \stackrel{\text{def}}{=} [011] = [11]$, porque $011z \in L$ se e só se $z \in \{0,1\}^\star$.

$\delta([11], 0) \stackrel{\text{def}}{=} [110] = [11]$, porque $110z \in L$ se e só se $z \in \{0,1\}^\star$.

$\delta([11], 1) \stackrel{\text{def}}{=} [111] = [11]$, porque $111z \in L$ se e só se $z \in \{0,1\}^\star$.



**Resolva apenas uma das alíneas do problema 5. Se resolver ambas, será classificada a alínea 5a).**

**5.** Seja $T = \{0^n \mid n \geq 0\} \cup \{y2^k y^R \mid k \geq 0, y \in \{0,1\}^\star \text{ e } y \text{ tem número ímpar de 0's}\}$ uma linguagem de alfabeto $\Sigma = \{0, 1, 2\}$, onde $y^R$ designa o reverso de $y$.

**a)** Determine um autómato de pilha que reconheça a linguagem $T$ por pilha vazia. Descreva a interpretação de cada estado de forma a permitir aferir a correção do autómato (e compreender o algoritmo subjacente).

***Resposta:***

$$
\begin{array}{lcl}
\delta(s_0, \varepsilon, \mathtt{Z}) & = & \{(s_1, \varepsilon), (s_1, \mathtt{Z}), (s_2, \mathtt{Z})\} \\
\delta(s_1, \mathtt{0}, \mathtt{Z}) & = & \{(s_1, \varepsilon), (s_1, \mathtt{Z})\}
\end{array}
$$

$$
\begin{array}{lcllcllcl}
\delta(s_2, \mathtt{1}, \mathtt{Z}) & = & \{(s_2, \mathtt{B})\} & \delta(s_2, \mathtt{1}, \mathtt{B}) & = & \{(s_2, \mathtt{BB})\} & \delta(s_2, \mathtt{1}, \mathtt{A}) & = & \{(s_2, \mathtt{BA})\} \\
\delta(s_2, \mathtt{0}, \mathtt{Z}) & = & \{(s_3, \mathtt{A})\} & \delta(s_2, \mathtt{0}, \mathtt{A}) & = & \{(s_3, \mathtt{AA})\} & \delta(s_2, \mathtt{0}, \mathtt{B}) & = & \{(s_3, \mathtt{AB})\}
\end{array}
$$

$$
\begin{array}{lcllcl}
\delta(s_3, \mathtt{0}, \mathtt{B}) & = & \{(s_2, \mathtt{AB})\} & \delta(s_4, \mathtt{2}, \mathtt{A}) & = & \{(s_4, \mathtt{A})\} \\
\delta(s_3, \mathtt{0}, \mathtt{A}) & = & \{(s_2, \mathtt{AA}), (s_5, \varepsilon)\} & \delta(s_4, \mathtt{0}, \mathtt{A}) & = & \{(s_5, \varepsilon)\} \\
\delta(s_3, \mathtt{1}, \mathtt{B}) & = & \{(s_3, \mathtt{BB}), (s_5, \varepsilon)\} & \delta(s_4, \mathtt{1}, \mathtt{B}) & = & \{(s_5, \varepsilon)\} \\
\delta(s_3, \mathtt{1}, \mathtt{A}) & = & \{(s_3, \mathtt{BA})\} & \delta(s_5, \mathtt{0}, \mathtt{A}) & = & \{(s_5, \varepsilon)\} \\
\delta(s_3, \mathtt{2}, \mathtt{A}) & = & \{(s_4, \mathtt{A})\} & \delta(s_5, \mathtt{1}, \mathtt{B}) & = & \{(s_5, \varepsilon)\}
\end{array}
$$

O símbolo inicial na pilha é Z e o estado inicial é $s_0$. O conjunto de estados finais é vazio.

Ideia: Separou-se a análise de $\mathcal{L}(0^\star)$ da análise das palavras da forma $y2^k y^R$ (deste modo, as palavras da forma $0^{2p}$, com $p \geq 0$, serão aceites por duas vias, mas isso não constitui um problema). Para a separação, criaram-se os estados $s_1$ e $s_2$, respetivamente. Em $s_1$, o AP pode retirar Z da pilha ou continuar a aceitar 0's. Em $s_2$, segue uma ideia semelhante à do AP dado nas aulas para reconhecimento das capícuas (de comprimento par). As diferenças decorrem da necessidade de garantir que $y$ tem número ímpar de 0's. Para isso, é criado o estado $s_3$. Se estiver em $s_3$, o prefixo consumido de $y$ tem número ímpar de 0's. Quando o autómato está (ou passa) a $s_2$, o prefixo de $y$ consumido tem número par de 0's. Em $s_3$ e $s_2$ carrega a pilha (como nas capícuas). Em $s_5$, descarrega-a de modo também análogo. O autómato é não determinístico. Como $k$ pode ser zero (i.e., a palavra pode não ter 2's), sempre que está em $s_3$ pode iniciar o descarregamento da pilha, desde que o topo e o símbolo que vai consumir emparelhem. Um A na pilha representa um 0. Um B na pilha representa um 1. O autómato pode consumir 2 em $s_3$ (mas não pode se estiver em $s_2$). Quando consome 2, passa a $s_4$, onde continuará a poder consumir 2. Em $s_4$, se for lido 0 com A no topo da pilha ou se for lido 1 com B no topo da pilha, inicia o descarregamento da pilha, passando a $s_5$.

**b)** Justifique que $T$ é uma linguagem independente de contexto não ambígua. Não é necessário escrever uma prova formal detalhada, mas a justificação não pode deixar dúvidas de que os argumentos estão corretos.

***Resposta:***

É importante começar por representar $T$ como uma união de conjuntos disjuntos.

$L_X = \{0^n \mid n \geq 0\}$

$L_Y = \{y2^k y^R \mid k \geq 0, y \in \{0,1\}^\star \text{ e } y \text{ tem número ímpar de 0's e } y \textbf{ tem algum } 1\} \cup \{0^{2n+1}22^k0^{2n+1} \mid k \geq 0, n \geq 0\}$

Esta decomposição pode ser usada para caraterizar $T$ pela seguinte gramática não ambígua, com símbolo inicial $S$:

$$
\begin{array}{lcl}
S & \to & X \mid Y \\
X & \to & 0X \mid \varepsilon \\
Y & \to & 1W1 \mid 0I0 \\
W & \to & 1W1 \mid 0K0 \\
I & \to & 0Y0 \mid 1K1 \mid D \\
K & \to & 0W0 \mid 1K1 \mid \varepsilon \mid D \\
D & \to & 2 \mid 2D
\end{array}
$$

Tem-se $Y \Rightarrow^\star x_1 I x_1^R$ se e só se $x_1$ tem número ímpar de 0's mas não tem 1's.

Tem-se $Y \Rightarrow^\star x_1 Y x_1^R$ se e só se $x_1$ tem número par de 0's mas não tem 1's.

Tem-se $Y \Rightarrow^\star x_1 W x_1^R$ se e só se $x_1$ tem número par de 0's e algum 1.

Tem-se $Y \Rightarrow^\star x_1 K x_1^R$ se e só se $x_1$ tem número ímpar de 0's e algum 1.

Em todos os casos, $x_1 \in \{0,1\}^\star$. Se se aplicar a regra $I \to D$ numa derivação, a sequência de terminais que se poderá obter é da forma $0^{2n+1}22^k0^{2n+1}$. Se se aplicar $K \to D$ ou $K \to \varepsilon$, a sequência de terminais que se pode obter é da forma $y2^k y^R$, com $k \geq 0$ e $y \in \{0,1\}^\star$ tem número ímpar de 0's e algum 1. Qualquer palavra de $T$ tem uma só derivação possível por esta GIC (e, portanto, uma só derivação pela esquerda). Logo, a gramática não é ambígua, o que implica que $T$ também não é ambígua. **(Fim)**