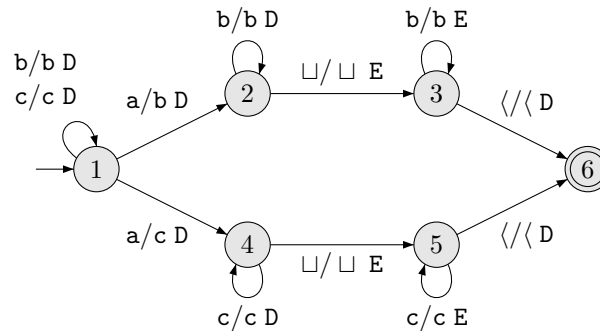


1. Seja o seguinte diagrama de estados de uma MT não determinística com alfabeto de entrada $\{a, b, c\}$:



- a) Que linguagem ela reconhece?
 b) Construa uma MT equivalente determinística.
2. Seja a linguagem de listas, LL, cuja sintaxe pode ser descrita pela gramática $(\{L, S, X\}, \{a, (,)\}, R, L)$, sendo R constituído das regras:

$$L \rightarrow (S)$$

$$S \rightarrow XS \mid \lambda$$

$$X \rightarrow a \mid L$$

Exemplos de listas: $()$, (a) , $(())$, $(a ())$, $((a) a)$, $(a ((a a) a a))$ etc.

Construa o diagrama de estados de uma MT de duas fitas que reconheça $L(LL)$.

3. Construa uma gramática irrestrita que gere $\{0^{n+1}1^n0^{n+1} \mid n \geq 0\}$. Caso a gramática seja sensível ao contexto, você ganha mais um ponto.
4. Responda sim ou não. Para cada resposta certa você ganha meio ponto e para cada resposta errada perde meio ponto:
- a) Qualquer linguagem sensível ao contexto é recursivamente enumerável.
 b) Os autômatos linearmente limitados podem ser simulados por meio de máquinas de Turing determinísticas.
 c) O complemento de uma linguagem recursiva pode não ser recursiva.
 d) As linguagens de programação (Java, C, Pascal etc.) são recursivamente enumeráveis.
 e) Se L_1 não é recursiva, L_1L_2 não é recursiva, mesmo que L_2 seja recursiva.
 f) Se L é recursivamente enumerável, mas não recursiva, não existe máquina de Turing que reconheça \bar{L} .
 g) Uma máquina de Turing com uma única fita, sendo esta ilimitada em ambas as direções, pode ser usada para simular uma não determinística de duzentas fitas ilimitadas em ambas as direções.
 h) Gramáticas irrestritas só geram linguagens recursivas.