~~Grupo II~~ · Exame 2018-01-30 Recurso

Grupo I:

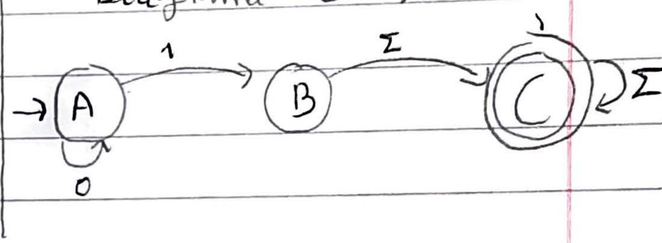
a) $\varepsilon\text{-close}(q_0) = \{q_0\}$ $\quad \varepsilon\text{-close}(q_1) = \{q_1, q_2\}$ $\quad \varepsilon\text{-close}(q_2) = \{q_1, q_2\}$

$\varepsilon\text{-close}(q_3) = \{q_3\}$

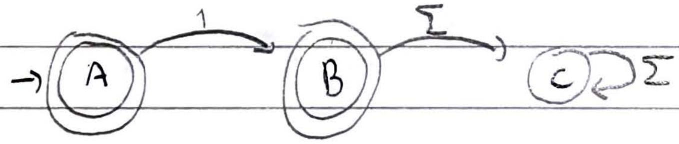
b)

		0	1
A:	$\rightarrow \{q_0\}$	$\{q_0\}$	$\{q_0, q_1, q_2\}$
B:	$\{q_0, q_1, q_2\}$	$\{q_0, q_1, q_2, q_3\}$	$\{q_0, q_1, q_2, q_3\}$
C:	$* \{q_0, q_1, q_2, q_3\}$	$\{q_0, q_1, q_2, q_3\}$	$\{q_0, q_1, q_2, q_3\}$

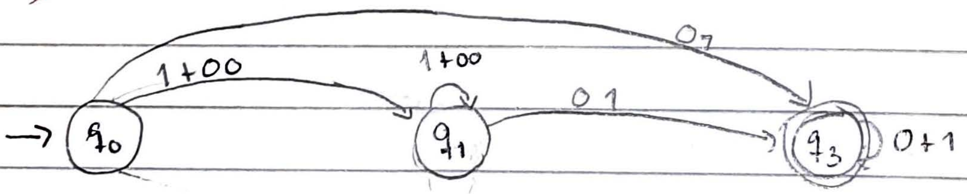
Diagrama de estados:



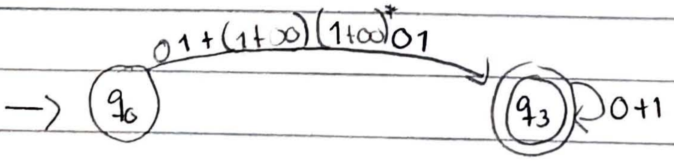
c) Basta trocar os estados finais pelos não-finais



d) Eliminando o estado q_2:



Eliminando o estado q_1:



RE: $(01 + (1+00)(1+00)^* 01)(0+1)^*$

e) Substituir todas as transições (q, x), em que q é o estado e x o caracter de entrada, por: $(q,x) = \varepsilon\text{-closure}(\delta(\varepsilon\text{-closure}(q), x))$.
Os estados finais passam a ser todos os estados em que o ε-closure contem os estados finais do ε-NFA

Grupo II:

a) A linguagem não é regular, uma vez que seria necessário guardar a informação de quantos R e A existem.
Uma possível CFG:

$$S \to RSA \mid SRA \mid \varepsilon$$

Pumping Lemma:

a) De acordo com o lema da Bombagem, existe uma cadeia $u = 0^m 1^m 0^m 1^m$, $|u| \geq m$, em que $u \in L$.
Esta cadeia pode ser divida em 3 partes: $u = xyz$, $|xy| \leq m$, $y \neq \varepsilon$

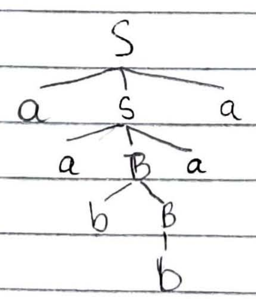
$$\begin{cases} x = 0^t, & t \geq 0 \\ y = 0^s, & s > 0 \\ z = 0^{m-s-t}\, 1^m\, 0^m\, 1^m \end{cases}$$

Então, pelo lema, todas as cadeias xy^kz, $k \geq 0$, $\in L$

Para $k = 0 \Rightarrow xz = 0^t\, 0^{m-s-t}\, 1^m\, 0^m\, 1^m = 0^{m-s}\, 1^m\, 0^m\, 1^m$.
Como $s > 0$, o numero de 0's no inicio será diferente do nº de 0's no meio, não pertencendo assim esta cadeia à linguagem. Logo L não é regular

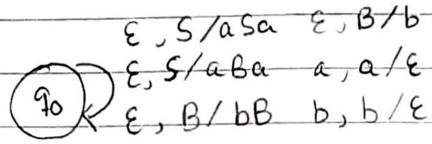
Grupo III:

a) $S \underset{lm}{\Rightarrow} aSa \Rightarrow aaBaa \Rightarrow aabBaa \Rightarrow aabbaa$



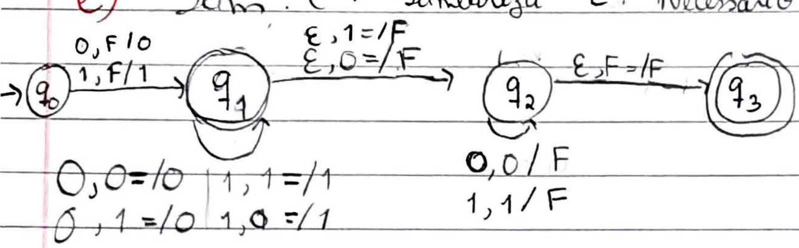
b) Não é ambígua, uma vez que não existe mais do que uma árvore de análise para cada cadeia

c)



d) S → aaAc | aCcc |
B → bB | b
A → aAc | aA | B
C → aCc | Cc | B

e) Sim. (F simboliza ε. Necessário pois é sempre preciso por algo na queue)



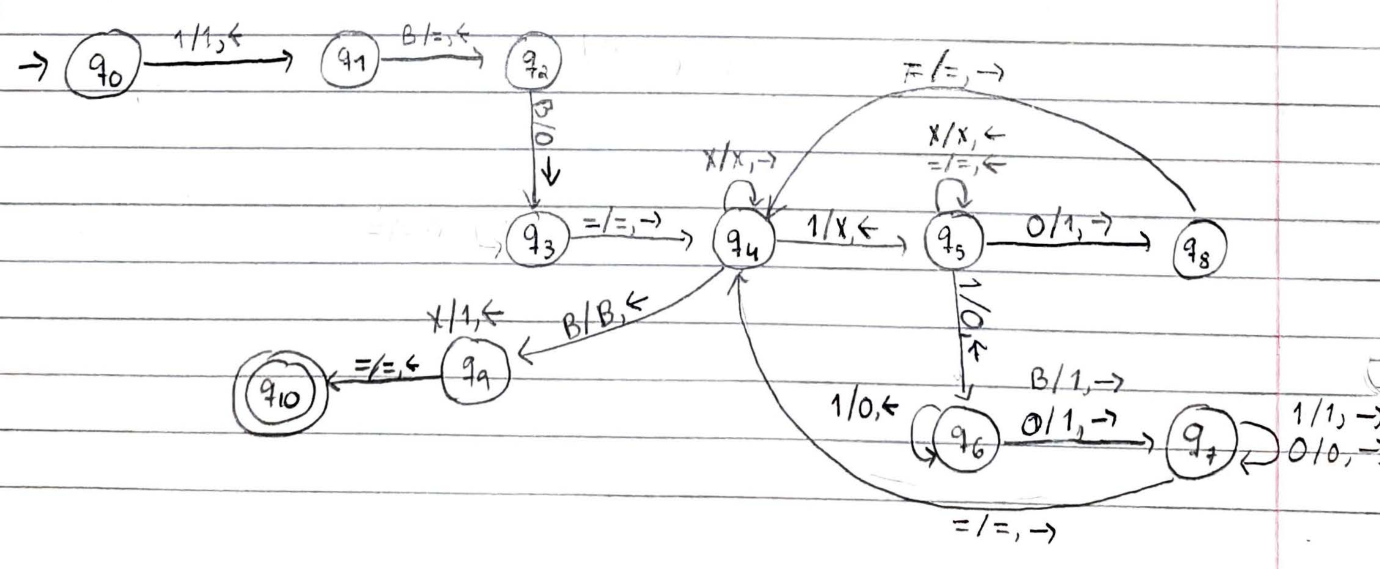
F também é o conteúdo inicial da queue

f) Não é possível.

Grupo IV:

a) Primeiro, adicionar 0= ao início da string. Depois, para cada 1 existente na parte unária, realizar a soma de 1 com o número à esquerda do igual e marcar o 1 processado com um X. Finalmente, trocar todos os X por 1's

b)



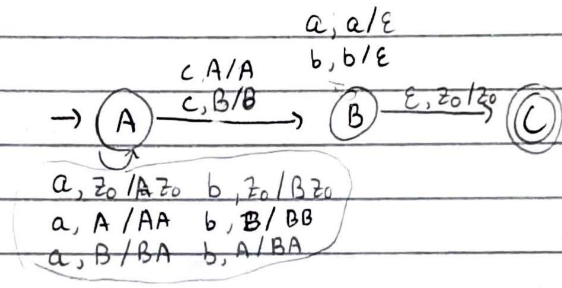
c) $q_0 11 \vdash q_1 B11 \vdash q_2 B=11 \vdash 0q_3=11 \vdash 0=q_4 11 \vdash 0q_5=X1$
$\vdash q_5 0=X1 \vdash 1q_8=X1 \vdash 1=q_4 X1 \vdash 1=Xq_4 1 \vdash 1=q_5 XX$
$\vdash 1q_5=XX \vdash q_5 1=XX \vdash q_6 B0=XX \vdash 1q_7 0=XX$ (...)

Grupo V:

a) Falso. Considerando $L1 = \{a^i b^j \mid i \geq j\}$ e $L2 = \{a^i b^j \mid i = j\}$
Ambas são CFL's. Da interseção resulta:
$L = \{a^i b^j \mid i = j\}$ e L é um CFL

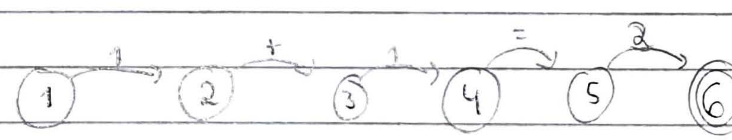
b) Verdadeiro. Por exemplo, $L = \{a^n \mid n > 1\}$, O complemento de L será $\bar{L} = \{a^n \mid n \leq 1\} = \{a, aa, \varepsilon\}$, que continua a ser uma CFL

c) Falso.



d) Falso. $L1 = \{a^n b^n \mid n \geq 0\}$ $L2 = \{a^* b^*\}$
L1 é uma CFL e L2 é uma linguagem regular. A interseção das duas: $L = L1 \cap L2 = \{a^n b^n \mid n \geq 0\}$ é uma CFL

e) Verdadeiro.



f) Falso. Os PDA representam CFL's, e as linguagens aceites pela TM são um super-set das CFL's

g) Verdadeiro. Como a parte que se repete (w) na cadeia é finita, é possível construir um DFA que aceite todas as combinações possíveis

h) Verdadeiro. Possível gramática:

$S \rightarrow NSN \mid Y$
$Y \rightarrow AYA \mid \varepsilon$
$N \rightarrow 0 \mid 1$ $\quad A \rightarrow a \mid b$