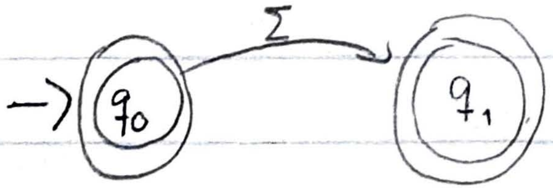
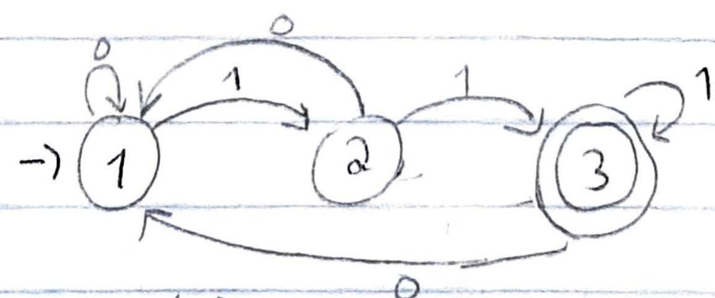
Escame 2019-01-18

Grupo I:

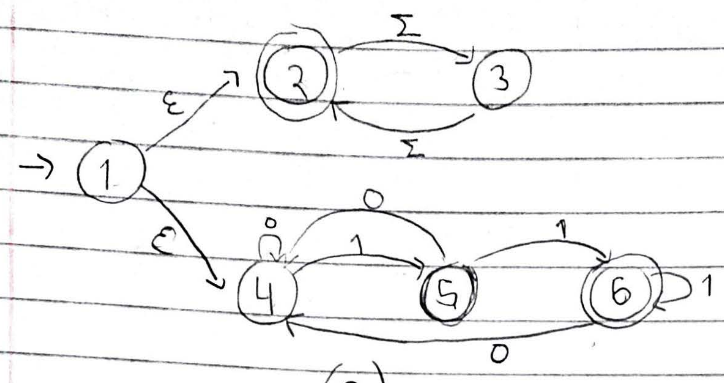
a) Um método simples de encontrar um FA para L é construir um automato que reconhece cadeias com comprimento par (1), e outro que reconhece cadeias que acabem em 11 (2), e finalmente juntar os dois adicionado um estado que passa a ser o inicial e adicionar transições - ε para os antigos estados iniciais (3).



(1)



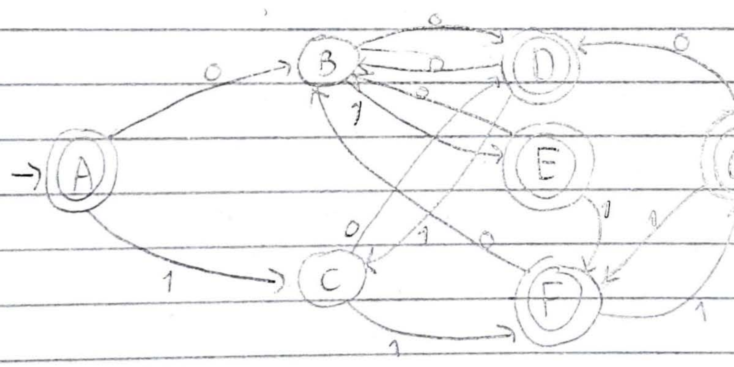
(2)



(3)

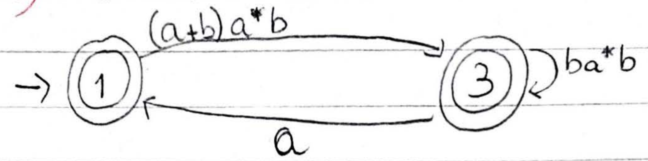
b) ε-close(1) = {1, 2, 3} ε-close(2) = {2} ε-close(3) = {3}
ε-clos(4) = {4} ε-close(5) = {5} ε-close(6) = {6}

		0	1
A	→* {1, 2, 4}	{3, 4}	{3, 5}
B	{3, 4}	{2, 4}	{2, 5}
C	{3, 5}	{2, 4}	{2, 6}
D	* {2, 4}	{3, 4}	{3, 5}
E	* {2, 5}	{3, 4}	{3, 6}
F	* {3, 6}	{3, 4}	{3, 6}
G	* {3, 6}	{2, 4}	{2, 6}



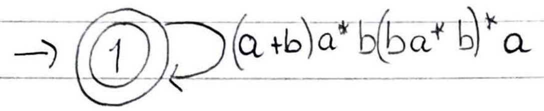
c) L_2 pode ser considerado o "reverso" de L_1. Logo, para obter um DFA de L_2, basta aplicar o método para obter o reverso de um DFA ao DFA de L1. Este método consiste em trocar o sentido de todas as transições e trocar o estado final pelo inicial. Se existir mais do que um estado final no DFA original, adicionar um novo estado, passando esse a ser inicial, com transições-ε para esses estados.

d) Eliminando o estado 2:



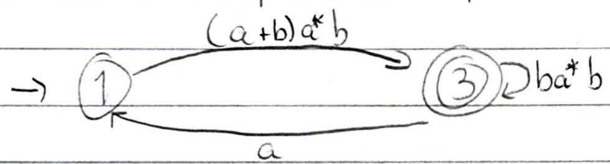
Como existem 2 estados finais (1 e 3), é necessário obter a RE admitindo que só 1 é final e, de seguida, admitindo que só 3 é final. A RE final será a junção das duas com +.

Admitindo que 1 é final e eliminando 3:



RE: $(a+b)a^*b(ba^*b)^*a$

Admitindo que 3 é final:

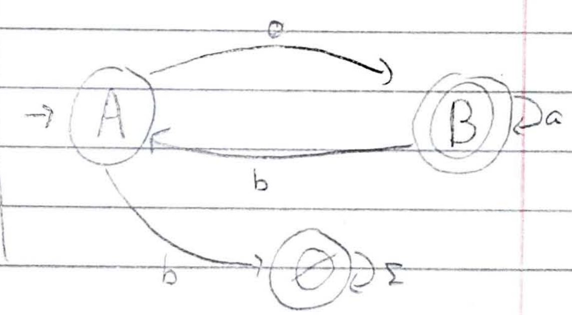


RE: $(a+b)a^*b(ba^*b)^*(a(a+b)a^*b)^*$

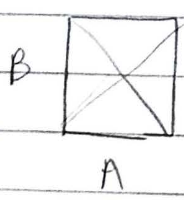
RE final: $(a+b)a^*b(ba^*b)^*a + (a+b)a^*b(ba^*b)^*(a(a+b)a^*b)^*$

e)

		a	b
A	$\{q_0\}$	$\{q_0, q_1\}$	$\emptyset$
B	$^*\{q_0, q_1\}$	$\{q_0, q_1\}$	$\{q_0\}$
	$\emptyset$	$\emptyset$	$\emptyset$



f)



A e B não podem ser equivalentes, pois B é um estado final e A não. O estado morto ($\emptyset$) não é equivalente a nenhum outro estado. Logo, o DFA já se encontra minimalizado

Grupo II:

a) $N_2 \longrightarrow$ número de 2's $\qquad N_2 = n - m \iff n = m + N_2$

Como o número de 0's é igual à soma do nº de 1 com o nº de 2, ao aplicar o seguinte homomorfismo: $h(0)=a$, $h(1)=b$, $h(2)=b$; obtém-se $L2 = \{a^n b^n \mid n \geq 0\}$ que é igual a L1. Como o homomorfismo é uma operação fechada às linguagens regulares e, ao o aplicar a L resultou numa linguagem não-regular, conclui-se que L não é regular.

b) A diferença simétrica é fechada às linguagens regulares, uma vez que realizar a diferença simétrica traduz-se em fazer a união dos dois conjuntos, seguida da diferença da interseção deles: $A \otimes B = A \cup B - A \cap B$. Como a interseção, a união, e a diferença são fechadas às linguagens regulares, a diferença simétrica também o será. Para obter uma expressão regular que representa a $\otimes$ a partir das expressões regulares de duas linguagens, pode-se converter as expressões regulares em ε-NFA's, convertê-los em DFA's e realizar o produto dos DFA's. De seguida, obter-se o DFA da união e da interseção, e realizar a diferença. Por fim, converter o DFA resultante numa expressão regular.

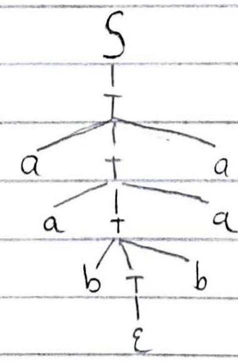
Grupo III

a) $S \rightarrow aTaV \mid T$

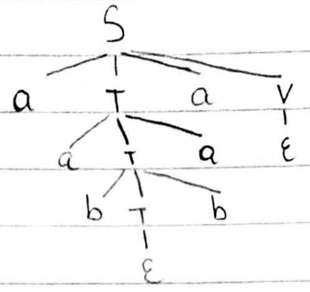
$T \rightarrow aTa \mid bTb \mid \varepsilon \mid a \mid b$

$V \rightarrow a \mid \varepsilon$

b) $S \underset{lm}{\Rightarrow} T \Rightarrow aTa \Rightarrow aaTaa \Rightarrow aabTbaa \Rightarrow aabbaa$



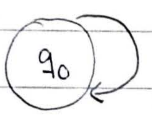
c) A CFG é ambígua, uma vez que existe mais do que uma árvore de análise para uma cadeia w. Por exemplo, a string da alínea anterior tem outra árvore binária:



Removendo a ambiguidade:

$S \to aSaV \mid U$
$U \to bTb \mid \varepsilon$
$T \to aTa \mid bTb \mid a \mid b \mid \varepsilon$
$V \to a \mid \varepsilon$

d)



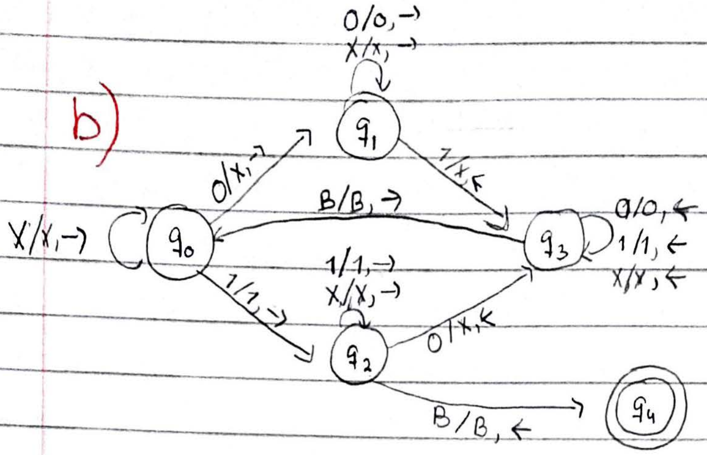
$\varepsilon, S / aSaV$
$\varepsilon, S / U$
$\varepsilon, U / bTb$
$\varepsilon, U / \varepsilon$
$\varepsilon, T / aTa$
$\varepsilon, T / bTb$
$\varepsilon, T / \varepsilon$
$\varepsilon, T / a$
$\varepsilon, T / b$
$\varepsilon, V / a$
$\varepsilon, V / \varepsilon$
$a, a / \varepsilon$
$b, b / \varepsilon$

e) $(q_0, aabbbaa, S) \vdash (q_0, aabbaa, aSaV) \vdash (q_0, abbaa, SaV)$
$\vdash (q_0, abbaa, aSaVaV) \vdash (q_0, bbaa, SaVaV)$
$\vdash (q_0, bbaa, UaVaV) \vdash (q_0, bbaa, bTbaVaV)$
$\vdash (q_0, baa, TbaVaV) \vdash (q_0, baa, baVaV) \vdash (q_0, aa, aVaV)$
$\vdash (q_0, a, VaV) \vdash (q_0, a, aV) \vdash (q_0, \varepsilon, V) \vdash (q_0, \varepsilon, \varepsilon)$

Grupo IV:

a) Primeiro, verifica-se se o primeiro caracter é um 0 ou um 1, marcando-o com um X. Se for um 0, procura-se na cadeia um 1. Se for um 1, procura-se na cadeia um 0. Ao encontrar o caracter pretendido, marca-se o com um X. Volta-se ao início da cadeia, e repete-se para os próximos caracteres. Se ao procurar um 0 (ou seja, processou-se um 1), chegamos ao fim da cadeia, vai para o estado de aceitação. Se ao procurar um 1 (ou seja, foi processado um 0), chegamos ao fim da cadeia, não a aceita. Se, ao procurar o próximo caracter a processar, chegarmos ao fim da cadeia (número de 1's igual ao número de 0's), não a aceitamos também.

b)



c) $q_0\, 101 \vdash X\, q_2\, 01 \vdash q_3\, X X 1 \vdash q_3\, B X X 1 \vdash q_0\, X X 1 \vdash X\, q_0\, X 1$
$\vdash X X\, q_0\, 1 \vdash X X X\, q_2\, B \vdash X X\, q_4\, X$

Grupo V :

a) ?

b) Verdadeiro. Consegue-se pensar numa gramática para L

$$S \rightarrow aaSbbb \mid \varepsilon$$

c) Falso. A gramática não é regular, uma vez que não é consistemente linear à esquerda ou consistemente linear à direita

? d) Verdadeiro. Todos os subconjuntos de uma linguagem devem seguir o lema da bombagem

e) Verdadeiro. Seja L uma linguagem não-regular. Se L^c fosse regular (ou seja, aplicando o complemento numa linguagem não-regular, obtem-se uma linguagem regular), significaria que $(L^c)^c = L$ é regular, o que não é o caso.

f) Falso. Seja $L1 = \{a^i b^j \mid i \geq j\}$ e $L2 = \{a^i b^j \mid i < j\}$, com $i, j \geq 0$. Ambas são não-regulares. Ao aplicar a união:
$L = L1 \cup L2 = \{a^* b^*\}$, que é regular

g) Verdadeiro. Se a pilha é finita, então o PDA só reconhece linguagens finitas. Como todas as linguagens finitas são regulares, e todas as linguagens regulares podem ser representadas por um DFA, a afirmação é verdadeira

h) ?

i) Verdadeira. Transformar uma gramática ambígua numa TM resulta numa TM não determinista.