Bacharelado em Sistemas de Informação
Disciplina: Linguagens Formais e Autômatos
2019.2 — Lista de exercícios 2 — GABARITO

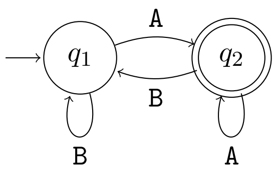
Linguagens Regulares

Questão 1 ....

Para cada um dos autômatos finitos determinísticos abaixo, apresente a sua descrição formal, determine se ele irá aceitar a palavra w apresentada, e determine qual é a linguagem reconhecida pelo mesmo.

Todos os AFDs são 5-uplas: $M = (\Sigma, Q, \delta, q_0, F)$. As componentes seguem abaixo.

(a)

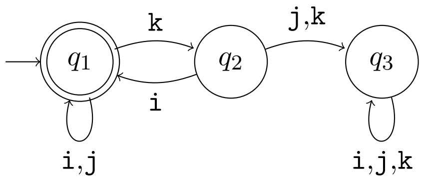


$\Sigma = \{\mathtt{A}, \mathtt{B}\}$, $Q = \{q_1, q_2\}$, $q_0 = q_1$, $F = \{q_2\}$ **e:**

δ	A	B
q_1	q_2	q_1
q_2	q_2	q_1

$w = \mathtt{AABBA}$ **é aceita;**
$L(M) = \{w\mathtt{A} \ : \ w \in \Sigma^*\}$.

(b)

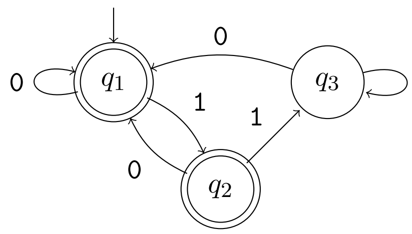


$\Sigma = \{\mathtt{i}, \mathtt{j}, \mathtt{k}\}$, $Q = \{q_1, q_2, q_3\}$, $q_0 = q_1$, $F = \{q_1\}$ **e:**

δ	i	j	k
q_1	q_1	q_1	q_2
q_2	q_1	q_3	q_3
q_3	q_3	q_3	q_3

$w = \mathtt{ikijkki}$ **é rejeitada;**
$L(M) = \{w \in \Sigma^* \ : \$ **todo** k **em** w **é seguido de um** i$\}$.

(c)

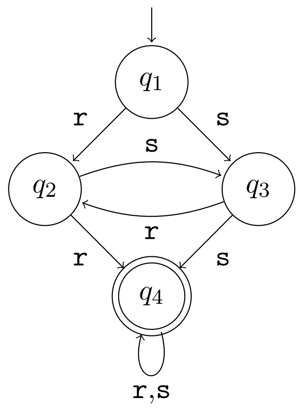


$\Sigma = \{0, 1\}$, $Q = \{q_1, q_2, q_3\}$, $q_0 = q_1$, $F = \{q_1, q_2\}$ **e:**

δ	0	1
q_1	q_1	q_2
q_2	q_1	q_3
q_3	q_1	q_3

$w = \mathtt{0101110}$ **é aceita;**
$L(M) = \{w \in \Sigma^* \ : \ w$ **não tem** 11 **como sufixo**$\}$.

(d)

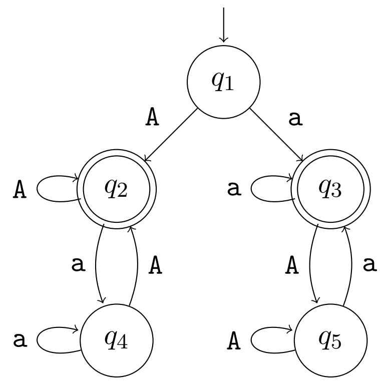


$\Sigma = \{\mathtt{r}, \mathtt{s}\}$, $Q = \{q_1, q_2, q_3, q_4\}$, $q_0 = q_1$, $F = \{q_4\}$ **e:**

δ	r	s
q_1	q_2	q_3
q_2	q_4	q_3
q_3	q_2	q_4
q_4	q_4	q_4

$w =$ **rsrssrr é aceita;**
$L(M) = \{w \in \Sigma^* \ : \ w$ **possui rr ou ss como subpalavra**$\}$.

(e)



$\Sigma = \{\mathtt{A}, \mathtt{a}\}$, $Q = \{q_1, q_2, q_3, q_4, q_5\}$, $q_0 = q_1$, $F = \{q_2, q_3\}$ **e:**

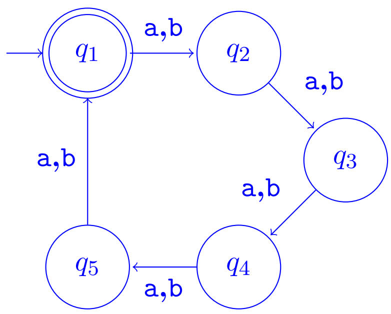
δ	A	a
q_1	q_2	q_3
q_2	q_2	q_4
q_3	q_5	q_3
q_4	q_2	q_4
q_5	q_5	q_3

$w =$ **AaaAaA é aceita;**
$L(M) = \{w \in \Sigma^* \ : \ w$ **começa e termina com o mesmo símbolo**$\}$.

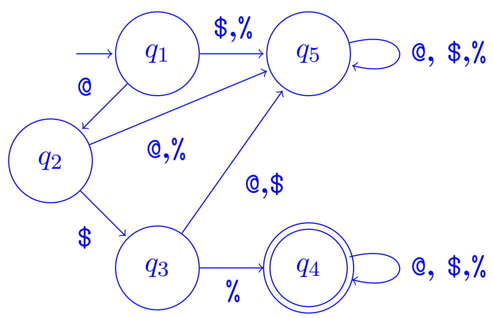
Questão 2 . . . .

Considere as linguagens a seguir. Mostre que cada uma é regular apresentando um autômato finito *determinístico* que reconheça essa linguagem. Quando o alfabeto não for especificado, considere sempre o menor alfabeto para o qual a linguagem está bem-definida.

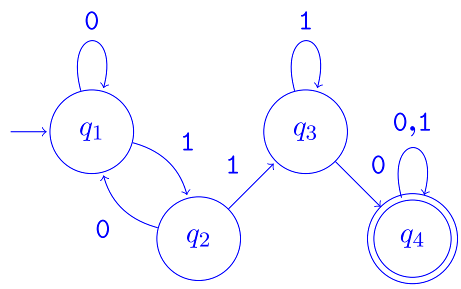
(a) $\{w \in \{\mathtt{a}, \mathtt{b}\}^* \ : \ |w|$ é divisível por 5$\}$



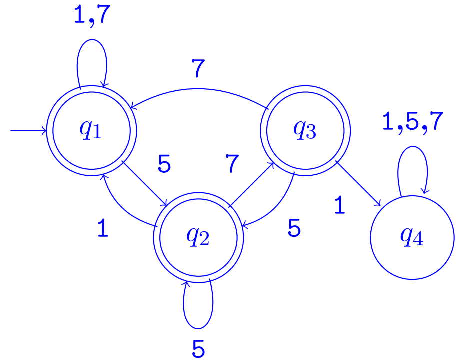
(b) $\{w \ : \ \mathtt{@\$\%}$ é prefixo de $w\}$



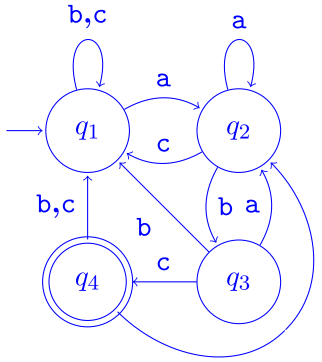
(c) $\{a\mathtt{110}b \ : \ a, b \in \{\mathtt{0}, \mathtt{1}\}^*\}$



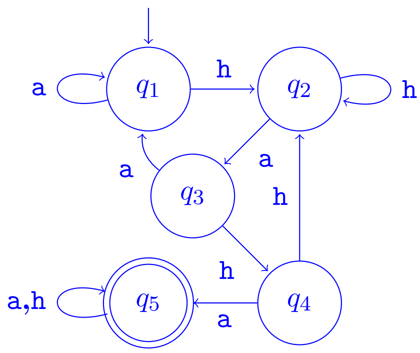
(d) $\{w \ : \ \mathtt{571}$ não é subpalavra de $w\}$



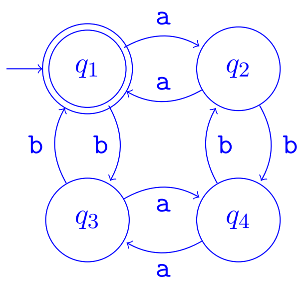
(e) $\{w \ : \ \texttt{abc} \text{ é sufixo de } w\}$



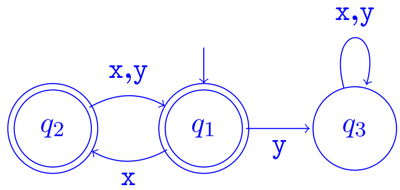
(f) $\{w \ : \ \texttt{haha} \text{ é subpalavra de } w\}$



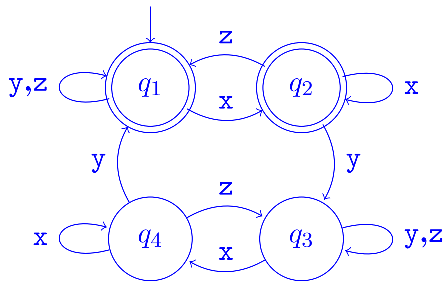
(g) $\{w \ : \ w \text{ possui número par de } \texttt{a}\text{'s e } \texttt{b}\text{'s}\}$



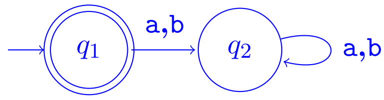
(h) $\{w \in \{\texttt{x}, \texttt{y}\}^* \ : \ w \text{ tem } \texttt{x} \text{ em todas as posições ímpares}\}$



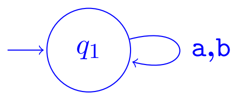
(i) $\{w \in \{\texttt{x}, \texttt{y}, \texttt{z}\}^* \ : \ w \text{ tem } \texttt{xy} \text{ como subpalavra uma quantidade par de vezes}\}$



(j) $\{\varepsilon\}$ $(\Sigma = \{\texttt{a}, \texttt{b}\})$



(k) $\emptyset$ $(\Sigma = \{\texttt{a}, \texttt{b}\})$



Questão 3 . . . .

Considere uma linguagem regular L. Explique como utilizar um AFD que reconheça L para construir um AFD que reconheça $\overline{L}$.

Resposta:

Lembre-se que $\overline{L}$ possui exatamente as palavras (em um dado alfabeto Σ) que L não possui. Se $M = (\Sigma, Q, \delta, q_0, F)$ é um AFD que reconhece L, basta inverter os estados finais e não-finais, construindo $M' = (\Sigma, Q, \delta, q_0, Q \setminus F)$. Para qualquer palavra w, a computação de w por M e por M' irá terminar no mesmo estado: logo se M aceitar uma palavra, M' irá rejeitá-la, e vice-versa.

Questão 4 ....

Considere um AFD qualquer. Como podemos determinar se ε é aceita por este AFD?

Resposta:

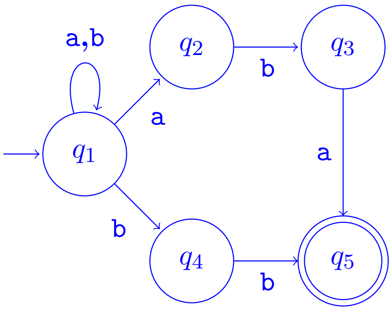
Após computar ε, qualquer autômato termina no estado inicial q_0, logo ε será aceita se e somente se q_0 for um estado final.

Questão 5 ....

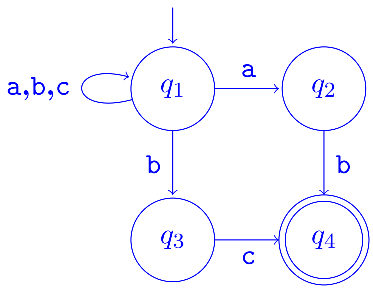
Considere as linguagens a seguir. Apresente um autômato finito não-determinístico que reconheça essa linguagem, e utilize a construção apresentada em sala para apresentar um autômato finito determinístico equivalente. Quando o alfabeto não for especificado, considere sempre o menor alfabeto para o qual a linguagem está bem-definida.

Abaixo de cada item são apresentados apenas os AFNs, exceto pelo item (e), que inclui também o AFD correspondente. Ao final, serão apresentados os AFDs correspondentes dos demais itens.

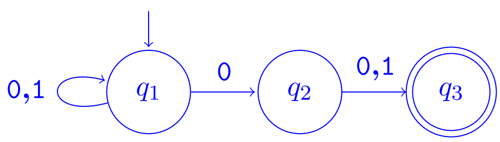
(a) $\{w$: `aba` ou `bb` é sufixo de $w\}$



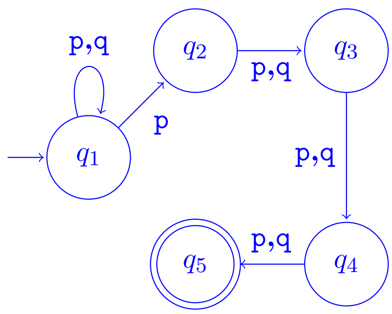
(b) $\{w$: `ab` ou `bc` é sufixo de $w\}$



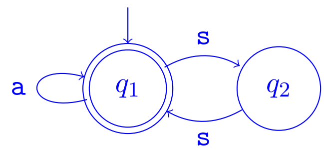
(c) $\{w \in \{0,1\}^*$: w tem `0` como penúltimo símbolo $\}$



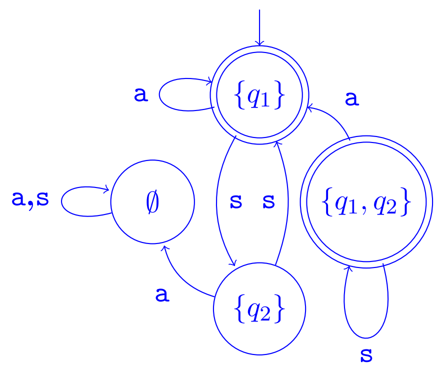
(d) $\{w \in \{\mathtt{p},\mathtt{q}\}^*$: w tem `p` como quarto último símbolo$\}$



(e) $\{w \in \{\mathtt{a},\mathtt{s}\}^*$: todos os `s`'s de w vêm em pares$\}$

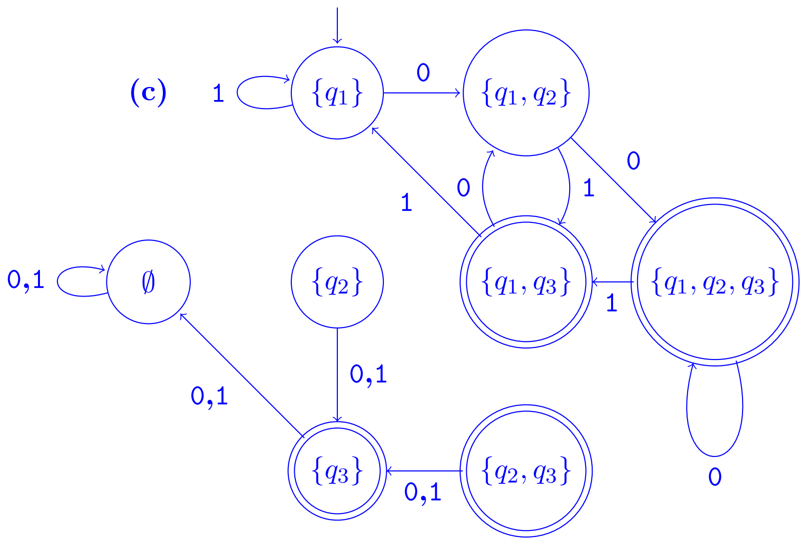


AFD:

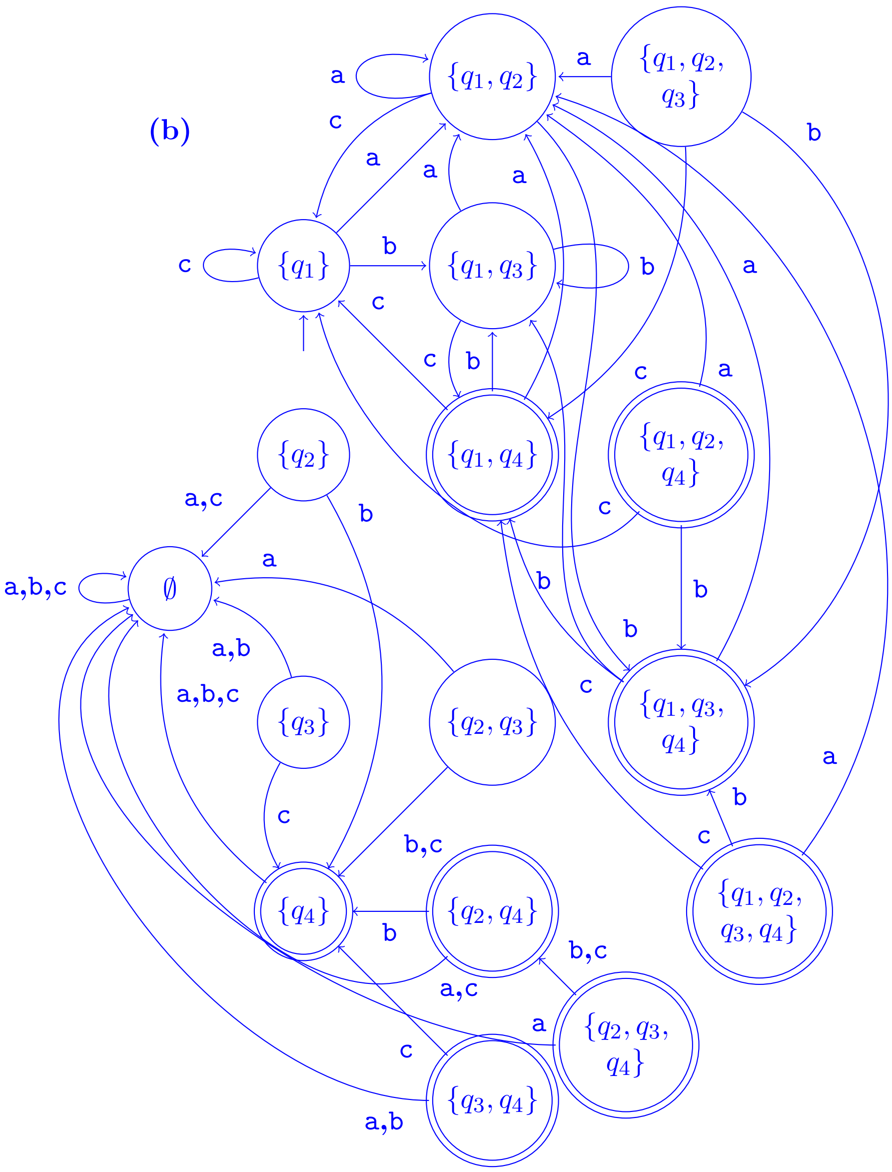


AFDs correspondentes:

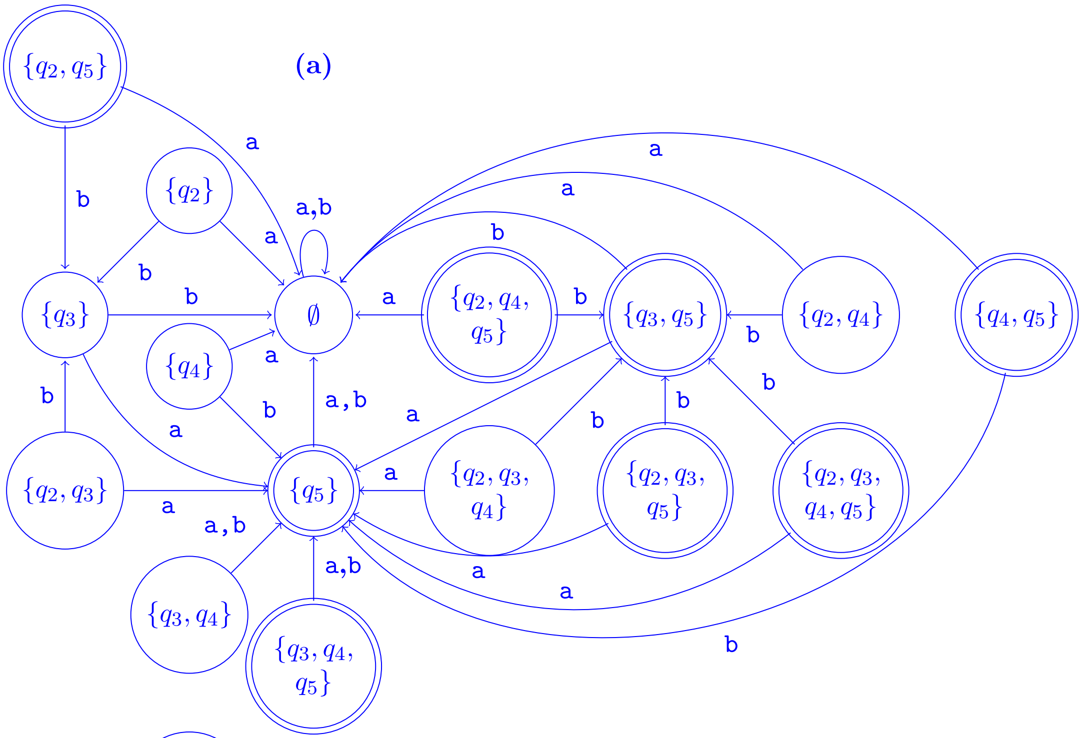
(c)

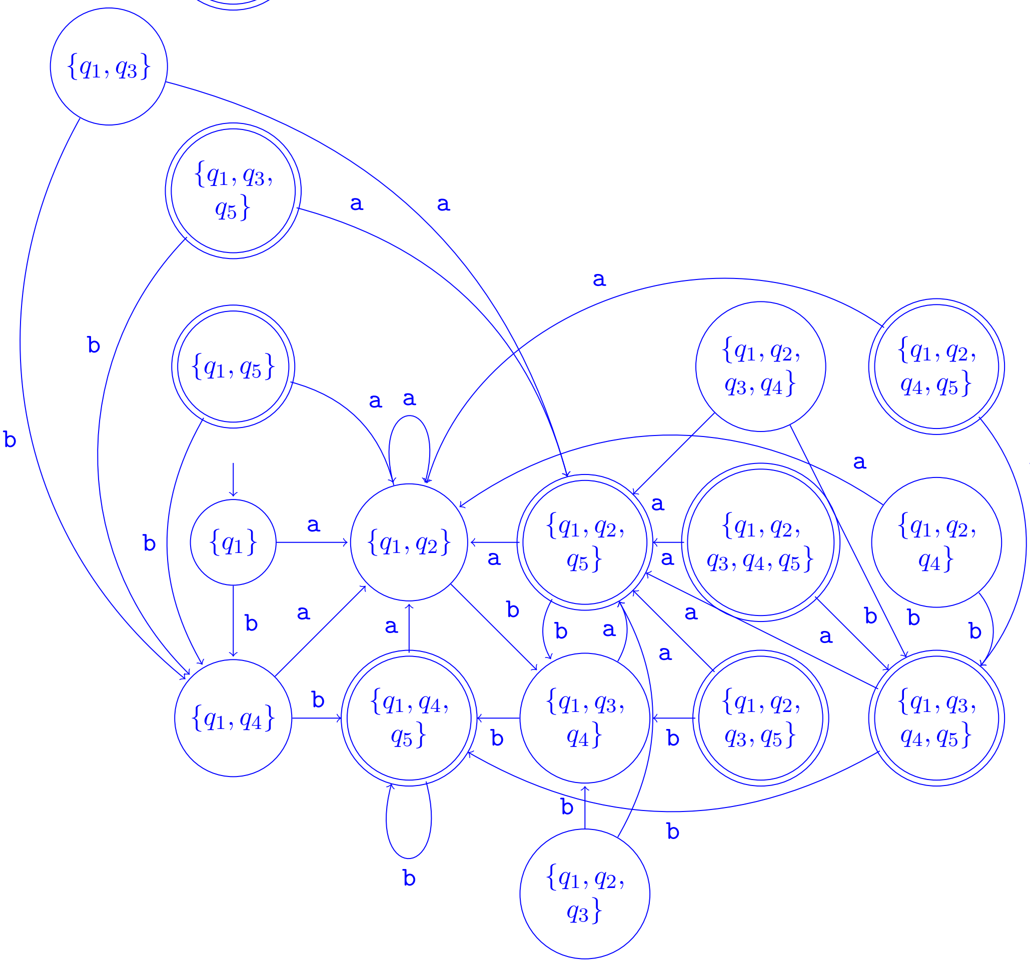


(b)

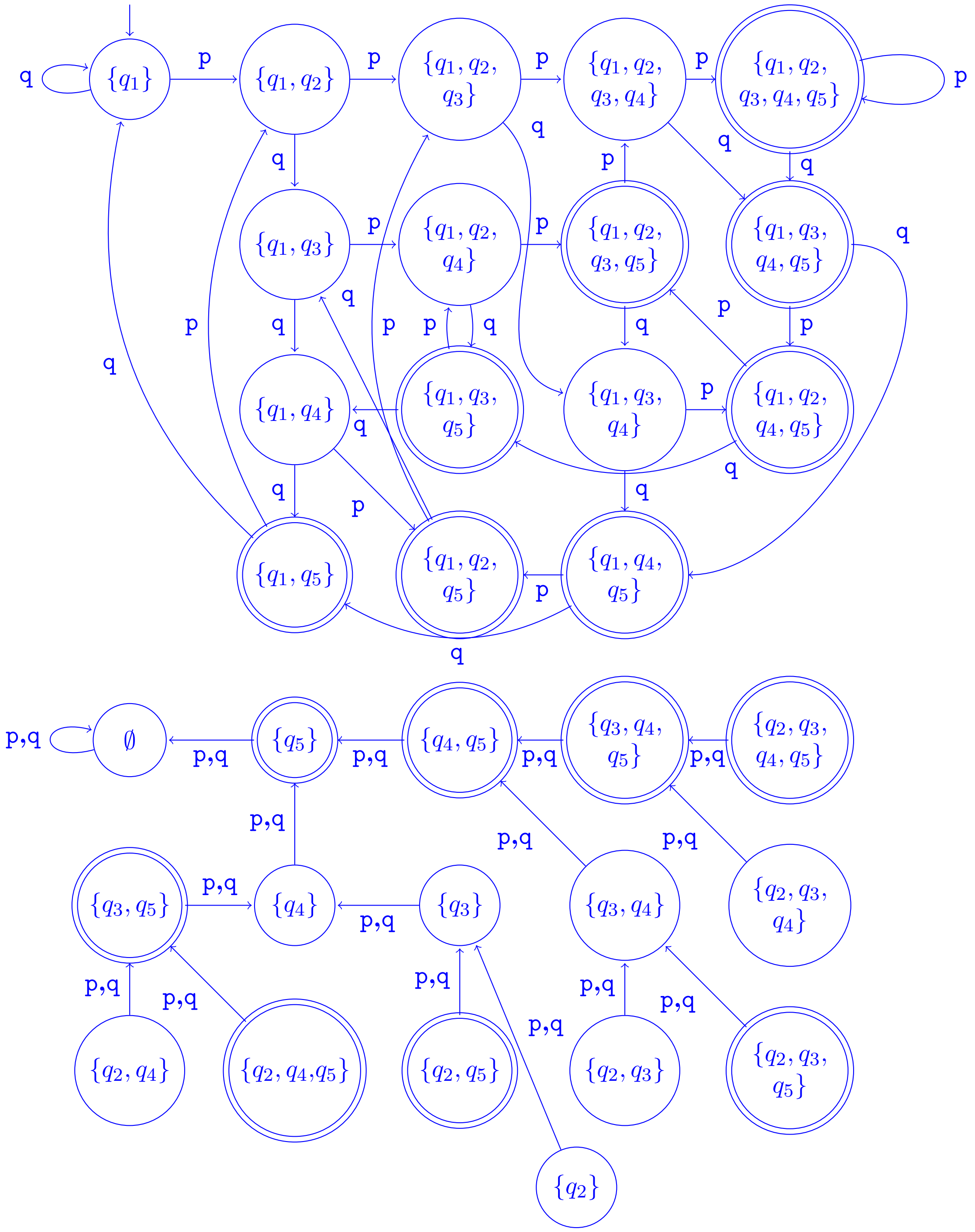


(a)


{q2, q5}
{q2}
{q3}
∅
{q4}
{q2, q4, q5}
{q3, q5}
{q2, q4}
{q4, q5}
{q2, q3}
{q5}
{q2, q3, q4}
{q2, q3, q5}
{q2, q3, q4, q5}
{q3, q4}
{q3, q4, q5}
a
b
a,b


{q1, q3}
{q1, q3, q5}
{q1, q5}
{q1}
{q1, q2}
{q1, q2, q5}
{q1, q2, q3, q4}
{q1, q2, q3, q4, q5}
{q1, q2, q4, q5}
{q1, q2, q4}
{q1, q4}
{q1, q4, q5}
{q1, q3, q4}
{q1, q2, q3, q5}
{q1, q3, q4, q5}
{q1, q2, q3}
a
b

(d)

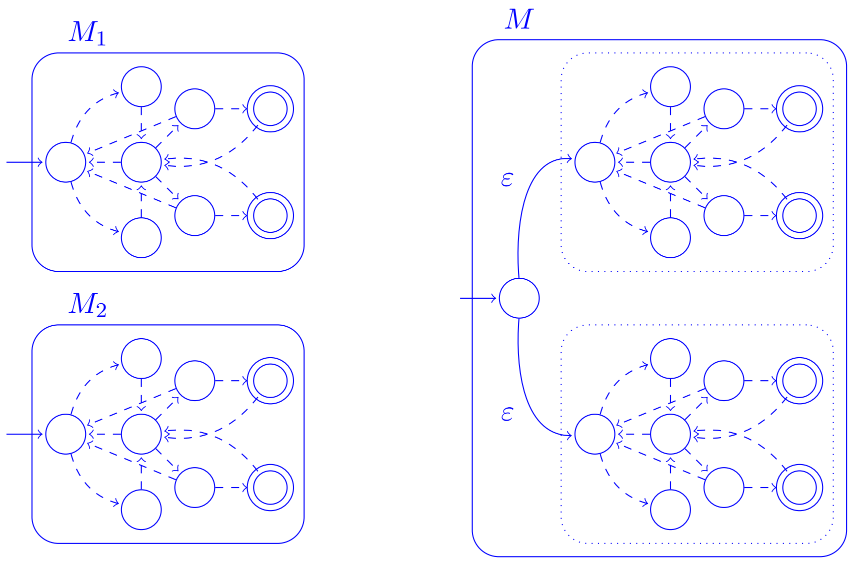


Questão 6 . . . .

Considere duas linguagens regulares L_1 e L_2. Explique graficamente como utilizar AFNs M_1 e M_2 que reconheçam L_1 e L_2, respectivamente, para construir AFNs que reconheçam as seguintes linguagens:

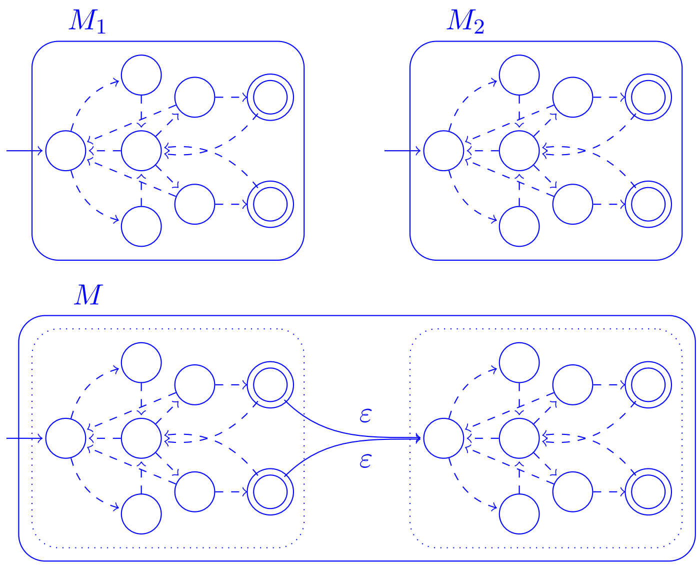
(a) $L_1 \cup L_2$

Uma palavra, para ser aceita por um autômato que reconhece $L_1 \cup L_2$, só precisa pertencer a uma das linguagens L_1 e L_2. Logo, podemos construir um autômato M que usa M_1 e M_2 como pedaços, "adivinha" se esta palavra pertence a L_1 ou a L_2 e computa a palavra utilizando o autômato correspondente:



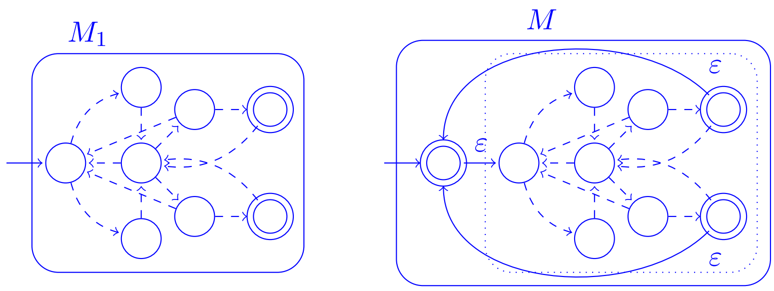
(b) $L_1 \cdot L_2$

As palavras de $L_1 \cdot L_2$ são aqueles formadas concatenando uma palavra de L_1 com outra de L_2. Assim, podemos construir M fazendo com que as palavras de entrada tenham que passar, em sequência, por M_1 e M_2: a parte inicial da palavra será aceita por M_1, que deixará o restante para M_2 aceitar.



(c) $L_1{}^*$

As palavras de $L_1{}^*$ são formadas pegando palavras de L_1 e colocando em sequência, logo, podemos construir M que faça a computação "atravessar" M_1 diversas vezes: cada vez, M_1 irá "consumir" um pedaço da palavra de entrada, até a palavra terminar.



Questão 7 . . . .

Mostre que se as linguagens L_1 e L_2 são regulares, $L_1 \cap L_2$ e $L_1 \setminus L_2$ também são regulares.

Resposta:

Pelas Leis de DeMorgan, $L_1 \cap L_2 = \overline{\overline{L_1} \cup \overline{L_2}}$. Utilizando os resultados das questões anteriores, temos que:

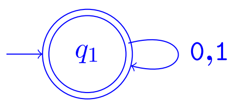
$$\begin{aligned} L_1 \text{ e } L_2 \text{ são regulares} &\implies \overline{L_1} \text{ e } \overline{L_2} \text{ são regulares} && \text{(questão 3)} \\ &\implies \overline{L_1} \cup \overline{L_2} \text{ é regular} && \text{(questão 6a)} \\ &\implies \overline{\overline{L_1} \cup \overline{L_2}} \text{ é regular} && \text{(questão 3)} \end{aligned}$$

Isto mostra que $L_1 \cap L_2$ é regular. Continuando, sabemos que $L_1 \setminus L_2 = L_1 \cap \overline{L_2}$. Como L_1 é regular e $\overline{L_2}$ é regular, pelo resultado anterior, a interseção das duas também é regular.

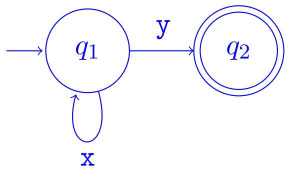
Questão 8 . . . .

Para cada expressão regular abaixo, apresente um AFN que reconheça a linguagem descrita pela mesma. Considere sempre o menor alfabeto para o qual a linguagem está bem-definida.

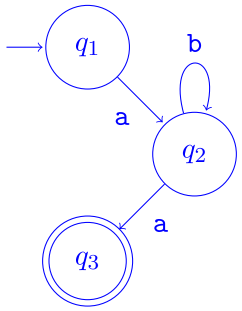
(a) `(0|1)*`



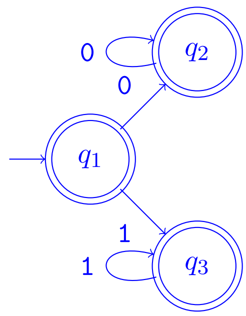
(b) `x*y`



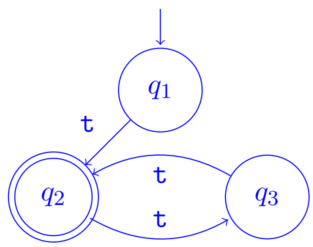
(c) `ab*a`



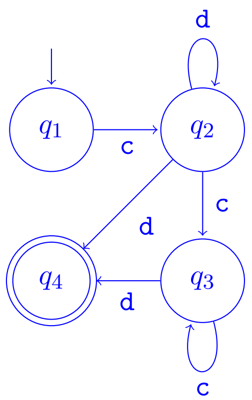
(d) `0*|1*`



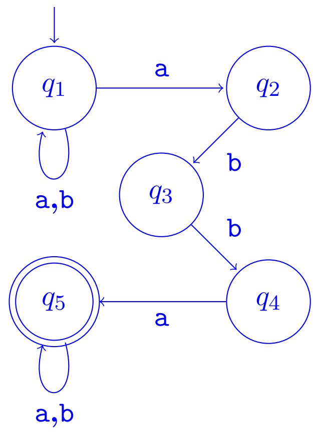
(e) `t(tt)*`



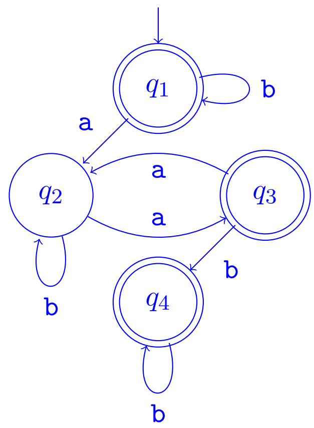
(f) `cd*c*d`



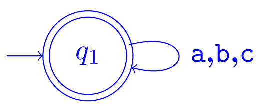
(g) `(a|b)*abba(a|b)*`



(h) `b*(ab*a)*b*`



(i) `(b|c)*(aa)*(a|b|c)*`



Questão 9 . . . .

Para cada linguagem das questões 2 e 5, apresente uma expressão regular que a descreva.

Resposta:

Questão 2:

(a) $(\Sigma\Sigma\Sigma\Sigma\Sigma)^*$
(b) Σ^*@\$%
(c) $\Sigma^*\mathtt{110}\Sigma^*$
(d) $(\mathtt{1}|\mathtt{7}|\mathtt{5}^*\mathtt{1}|\mathtt{5}^*(\mathtt{75})^*\mathtt{77})^*(\mathtt{55}^*((\mathtt{75})^*|\varepsilon)|\varepsilon)$
(e) $\Sigma^*\mathtt{abc}$
(f) $\Sigma^*\mathtt{haha}\Sigma^*$
(g) $(\mathtt{aa}|\mathtt{bb}|((\mathtt{ab}|\mathtt{ba})(\mathtt{aa}|\mathtt{bb})^*(\mathtt{ab}|\mathtt{ba}))^*$
(h) $(\mathtt{xx}\,|\,\mathtt{xy})^*(\mathtt{x}|\varepsilon)$
(i) $(\mathtt{y}|\mathtt{z}|\mathtt{x}^*\mathtt{z}^*\mathtt{xyy}|\mathtt{z}|\mathtt{x}^*\mathtt{z}^*\mathtt{xy})^*\mathtt{y}|\mathtt{z}|\mathtt{x}^*\mathtt{z}^*$
(j) ε
(k) $\emptyset$

Questão 5:

(a) $\Sigma^*(\mathtt{aba}|\mathtt{bb})$
(b) $\Sigma^*(\mathtt{ab}|\mathtt{bc})$
(c) $\Sigma^*\mathtt{0}\Sigma$
(d) $\Sigma^*\mathtt{p}\Sigma\Sigma\Sigma$
(e) $(\mathtt{a}|\mathtt{ss})^*$

Questão 10 . . . .

Para cada uma das gramáticas regulares abaixo, classifique-a como gramática regular à esquerda ou gramática regular à direita, e apresente uma expressão regular que a descreva.

(a) $G = (\{I\}, \{\mathtt{1}\}, P, I)$, com regras de produção:

$$I \to \mathtt{11}I$$
$$I \to \varepsilon$$

Gramática linear à direita; $(\mathtt{11})^*$

(b) $G = (\{X, Y\}, \{\mathtt{x}, \mathtt{y}\}, P, X)$, com regras de produção:

$$X \to \mathtt{x}Y$$
$$Y \to \mathtt{yx}Y \mid \varepsilon$$

Gramática linear à direita; $\mathtt{x}(\mathtt{yx})^*$

(c) $G = (\{R, S\}, \{\mathtt{r}, \mathtt{s}\}, P, R)$, com regras de produção:

$$R \to S\mathtt{rsrs}$$
$$S \to S\mathtt{r} \mid S\mathtt{s} \mid \varepsilon$$

Gramática linear à esquerda; $\Sigma^*\mathtt{rsrs}$

(d) $G = (\{A, B, C\}, \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}, P, A)$, com regras de produção:

$$A \to \mathtt{a}B \mid \mathtt{a}C \mid \mathtt{a}$$
$$B \to \mathtt{b}B \mid \mathtt{b}A$$
$$C \to \mathtt{c}C \mid \mathtt{c}A$$

Gramática linear à direita; $(\mathtt{a}(\mathtt{bb}^*|\mathtt{cc}^*))^*\mathtt{a}$

(e) $G = (\{A, B, C\}, \{\mathtt{0}, \mathtt{1}, \mathtt{2}\}, P, A)$, com regras de produção:

$$A \to \mathtt{0}A \mid \mathtt{1}B \mid \mathtt{2}C \mid \varepsilon$$
$$B \to \mathtt{0}B \mid \mathtt{1}C \mid \mathtt{2}A$$
$$C \to \mathtt{0}C \mid \mathtt{1}A \mid \mathtt{2}B$$

Gramática linear à direita; $(\mathtt{0}^*|\mathtt{1}(\mathtt{0}^*\mathtt{10}^*\mathtt{20}^*)^*\mathtt{2}|\mathtt{10}^*\mathtt{1}(\mathtt{0}^*\mathtt{20}^*\mathtt{10}^*)^*\mathtt{1}|\mathtt{20}^*(\mathtt{20}^*\mathtt{10}^*)^*\mathtt{1}|\mathtt{20}^*\mathtt{2}(\mathtt{0}^*\mathtt{10}^*\mathtt{20}^*)^*\mathtt{2})^*$

Questão 11 . . . .

Para cada linguagem das questões 2 e 5, apresente uma gramática regular que a gera.

Resposta:

Questão 2:

(a) $G = (\{A, B, C, D, E\}, \{\mathtt{a}, \mathtt{b}\}, P, A)$, com regras de produção:

$$A \to \mathtt{a}B \mid \mathtt{b}B \mid \varepsilon$$
$$B \to \mathtt{a}C \mid \mathtt{b}C$$
$$C \to \mathtt{a}D \mid \mathtt{b}D$$
$$D \to \mathtt{a}E \mid \mathtt{b}E$$
$$E \to \mathtt{a}A \mid \mathtt{b}A$$

(b) $G = (\{A, B\}, \{\mathtt{@}, \mathtt{\$}, \mathtt{\%}\}, P, A)$, com regras de produção:

$$A \to \mathtt{@\$\%}B$$
$$B \to \mathtt{@}B \mid \mathtt{\$}B \mid \mathtt{\%}B \mid \varepsilon$$

(c) $G = (\{A, B\}, \{\mathtt{0}, \mathtt{1}\}, P, A)$, com regras de produção:

$$A \to \mathtt{a}A \mid \mathtt{b}A \mid \mathtt{110}B$$
$$B \to \mathtt{a}B \mid \mathtt{b}B \mid \varepsilon$$

(d) $G = (\{A, B, C\}, \{\mathtt{1}, \mathtt{5}, \mathtt{7}\}, P, A)$, com regras de produção:

$$A \to \mathtt{1}A \mid \mathtt{7}A \mid \mathtt{5}B \mid \varepsilon$$
$$B \to \mathtt{1}A \mid \mathtt{5}B \mid \mathtt{7}C \mid \varepsilon$$
$$C \to \mathtt{5}B \mid \mathtt{7}A \mid \varepsilon$$

(e) $G = (\{A\}, \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}, P, A)$, com regras de produção:

$$A \to \mathtt{a}A \mid \mathtt{b}A \mid \mathtt{c}A \mid abc$$

(f) $G = (\{A, B\}, \{\mathtt{1}, \mathtt{5}, \mathtt{7}\}, P, A)$, com regras de produção:

$$A \to \mathtt{a}A \mid \mathtt{h}A \mid \mathtt{haha}B$$
$$B \to \mathtt{a}B \mid \mathtt{h}B \mid \varepsilon$$

(g) $G = (\{A, B, C, D\}, \{\mathtt{a}, \mathtt{b}\}, P, A)$, com regras de produção:

$$A \to \mathtt{a}B \mid \mathtt{b}C \mid \varepsilon$$
$$B \to \mathtt{a}A \mid \mathtt{b}D$$
$$C \to \mathtt{a}D \mid \mathtt{b}A$$
$$D \to \mathtt{a}C \mid \mathtt{b}B$$

(h) $G = (\{A, B\}, \{\mathtt{x}, \mathtt{y}\}, P, A)$, com regras de produção:

$$A \to \mathtt{x}B \mid \varepsilon$$
$$B \to \mathtt{x}A \mid \mathtt{y}A \mid \varepsilon$$

(i) $G = (\{A, B, C, D\}, \{\mathtt{x}, \mathtt{y}, \mathtt{z}\}, P, A)$, com regras de produção:

$$A \to \mathtt{y}A \mid \mathtt{z}A \mid \mathtt{x}B \mid \mathtt{xy}C \mid \varepsilon$$
$$B \to \mathtt{x}B \mid \mathtt{z}A \mid \varepsilon$$
$$C \to \mathtt{y}C \mid \mathtt{z}C \mid \mathtt{x}D \mid \mathtt{xy}A$$
$$D \to \mathtt{x}D \mid \mathtt{z}C$$

(j) $G = (\{A\}, \{\mathtt{a}, \mathtt{b}\}, P, A)$, com regras de produção:

$$A \to \varepsilon$$

(k) $G = (\{A\}, \{\mathtt{a}, \mathtt{b}\}, P, A)$, com regras de produção:

$$A \to A$$

Questão 5:

(a) $G = (\{A\}, \{\mathtt{a}, \mathtt{b}\}, P, A)$, com regras de produção:

$$A \to \mathtt{a}A \mid \mathtt{b}A \mid \mathtt{aba} \mid \mathtt{bb}$$

(b) $G = (\{A\}, \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}, P, A)$, com regras de produção:

$$A \to \mathtt{a}A \mid \mathtt{b}A \mid \mathtt{ab} \mid \mathtt{bc}$$

(c) $G = (\{A\}, \{\mathtt{0}, \mathtt{1}\}, P, A)$, com regras de produção:

$$A \to \mathtt{0}A \mid \mathtt{1}A \mid \mathtt{01} \mid \mathtt{00}$$

(d) $G = (\{A, B, C, D\}, \{\mathtt{p}, \mathtt{q}\}, P, A)$, com regras de produção:

$$A \to \mathtt{p}A \mid \mathtt{q}A \mid \mathtt{p}B$$
$$B \to \mathtt{p}C \mid \mathtt{q}C$$
$$C \to \mathtt{p}D \mid \mathtt{q}D$$
$$D \to \mathtt{p} \mid \mathtt{q}$$

(e) $G = (\{A\}, \{\mathtt{a}, \mathtt{s}\}, P, A)$, com regras de produção:

$$A \to \mathtt{a}A \mid \mathtt{ss}A \mid \varepsilon$$

Questão 12 . . . .

Considere uma linguagem L. Explique como utilizar uma gramática regular que gera L para construir uma gramática regular que gera L^R.

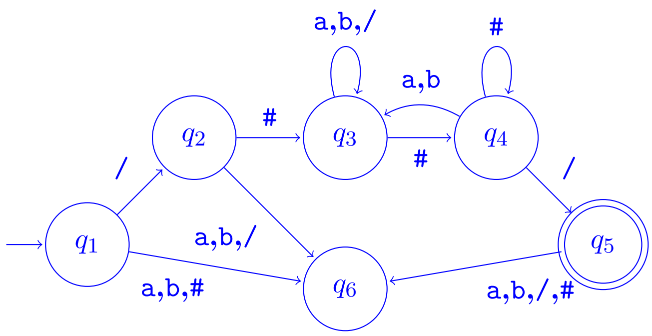
Resposta:

Se M é uma gramática regular que gera L, basta criar uma gramática M' "invertendo" as produções de todas as regras de M. Se M for uma GLD, todas as regras seguem o formato $A \to wB$, e as regras correspondentes de M' seguirão o formato $A \to Bw$, logo M' será uma GLE. Para qualquer derivação de uma palavra $w \in L$, a aplicação, na mesma ordem, das regras de produção correspondentes em M' irá derivar a palavra w^R, portanto, M' gera L^R. O mesmo vale se M for uma GLE, exceto que, neste caso, M' é uma GLD.

Questão 13 . . . .

Em certas linguagens de programação, comentários aparecem entre delimitadores como `/#` e `#/`. Seja L a linguagem de todas as strings de comentários corretamente delimitadas. Um membro de L deve iniciar com `/#` e terminar com `#/` sem possuir outros `#/` no meio. Por simplicidade, digamos que os comentários em si são escritos apenas com os símbolos `a` e `b`, logo o alfabeto de L é $\Sigma = \{\texttt{a}, \texttt{b}, \texttt{/}, \texttt{\#}\}$.

(a) Apresente um AFD que reconheça L.



(b) Apresente uma expressão regular que represente L.

$\texttt{/\#(a|b|/|\#}^*\texttt{(a|b))}^*\texttt{\#/}$

(c) Apresente uma gramática que gere L.

$G = (\{A, B, C\}, \{\texttt{a}, \texttt{b}, \texttt{/}, \texttt{\#}\}, P, A)$, **com regras de produção:**

$$A \to \texttt{/\#}B$$
$$B \to \texttt{a}B \mid \texttt{b}B \mid \texttt{/}B \mid \texttt{\#}C \mid \texttt{\#/}$$
$$C \to \texttt{\#}C \mid \texttt{a}B \mid \texttt{b}B$$

Questão 14 . . . .

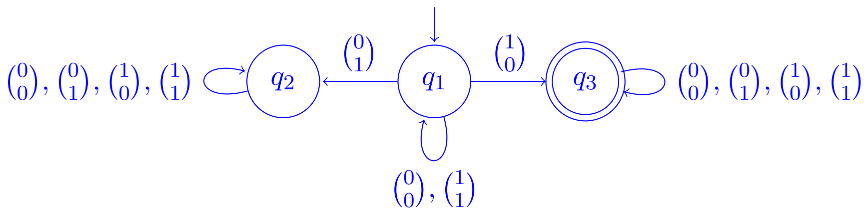
Considere $\Sigma = \left\{\binom{0}{0}, \binom{0}{1}, \binom{1}{0}, \binom{1}{1}\right\}$ contendo todas as colunas de `0`'s e `1`'s de altura 2. Uma palavra de Σ contém duas linhas de `0`'s e `1`'s. Interpretando cada linha como sendo um número binário, considere a linguagem

$$L = \{w \in \Sigma^* \ : \ \text{a linha superior de } w \text{ é um número maior do que a linha inferior de } w\}.$$

Por exemplo, $\binom{0}{0}\binom{1}{0}\binom{1}{1}\binom{0}{0} \in L$, mas $\binom{0}{0}\binom{0}{1}\binom{1}{1}\binom{0}{0} \notin L$. Mostre que L é regular.

Resposta:

Podemos mostrar que L é regular de três formas diferentes: apresentando um AFD ou AFN que a reconheça, uma expressão regular que a represente, ou uma gramática que a gere. Para tentar qualquer uma das três estratégias, precisamos notar que, nas palavras de L, um símbolo $\binom{1}{0}$ aparece antes de qualquer símbolo $\binom{0}{1}$, enquanto nas palavras de $\overline{L}$ vale o oposto: ou um símbolo $\binom{0}{1}$ aparece antes de qualquer símbolo $\binom{1}{0}$, ou nenhum dos dois símbolos aparece.
O seguinte AFN reconhece L e mostra que L é regular:



A seguinte expressão regular representa L e mostra que L é regular:

$$\left(\binom{0}{0}\middle|\binom{1}{1}\right)^* \binom{1}{0}\Sigma^*$$

A seguinte gramática regular gera L e mostra que L é regular:
$G = (\{A, B\}, \{\binom{0}{0}, \binom{0}{1}, \binom{1}{0}, \binom{1}{1}\}, P, A)$, com regras de produção:

$$\begin{aligned} A &\to \binom{0}{0}A \;\Big|\; \binom{1}{1}A \;\Big|\; \binom{1}{0}B \\ B &\to \binom{0}{0}B \;\Big|\; \binom{0}{1}B \;\Big|\; \binom{1}{0}B \;\Big|\; \binom{1}{1}B \;\Big|\; \varepsilon \end{aligned}$$

Questão 15 . . . .

Mostre que $l = 3$ é comprimento de bombeamento para a linguagem `a(aa)*|b(bb)*`.

Resposta:

A linguagem L representada por esta expressão regular é formada pelas palavras de comprimento ímpar formadas apenas por um tipo de símbolo (ou todos os símbolos são iguais a `a` ou todos são iguais a `b`). Precisamos mostrar que qualquer palavra $w \in L$ com comprimento 3 ou maior pode ser bombeada.
Pegue uma palavra w com comprimento $n \geq 3$: certamente é verdade que $w = x^n$, onde $x \in \{\mathtt{a}, \mathtt{b}\}$, e n é ímpar. Podemos escrever $w = pms$, onde: $p = \varepsilon$, $m = xx$ e $s = x^{n-2}$. Note que $|m| = 2 \geq 1$ e $pm = 2 \leq 3$, logo podemos utilizar esta decomposição para bombear w. Neste bombeamento, obtemos as palavras pm^is, com i qualquer inteiro positivo ou igual a 0. Pela nossa decomposição, $pm^is = x^{2i}x^{n-2} = x^{2i+n-2} = x^{n+2(i-1)}$. Note que $n + 2(i - 1)$ é um número ímpar (já que n é ímpar) e positivo (pois $n \geq 3$). Portanto, a palavra pm^is possui um número ímpar de símbolos iguais, logo pertence a L.
Assim, nossa tentativa de bombeamento funciona para w. Como w pode ser qualquer palavra de comprimento 3 ou maior, isso mostra que $l = 3$ é comprimento de bombeamento para L.

Questão 16 ....

Utilize o Lema do Bombeamento para demonstrar que as linguagens a seguir não são regulares. Considere sempre o menor alfabeto para o qual a linguagem está bem-definida.

Resposta:

O Lema do Bombeamento garante que toda linguagem regular possui algum inteiro n que serve de comprimento de bombeamento, o que significa que qualquer linguagem que não possua um comprimento de bombeamento deve, obrigatoriamente, não ser regular. Dessa forma, nossa estratégia para todos os casos será a mesma: precisamos mostrar que nenhum número inteiro n pode servir de comprimento de bombeamento para estas linguagens. O desafio aqui, então, é garantir que podemos descartar qualquer inteiro n como candidato a comprimento de bombeamento. Para isso, precisamos encontrar alguma palavra de comprimento n ou maior que não pode ser bombeada.

(a) $\{\texttt{a}^n\texttt{b}^n \ : \ n \geq 0\}$

Para um n possível comprimento de bombeamento, escolhemos a palavra $w = \texttt{a}^n\texttt{b}^n$, que pertence a L. Para tentar bombear w, precisamos dividir w em três partes: $w = pms$, com $|m| \geq 1$ e $|pm| \leq n$. Agora, como $|pm| \leq n$, para qualquer divisão permitida, p e m devem ser formadas apenas por a's. Escrevendo $P = |p|$ e $M = |m|$, isto significa que $p = \texttt{a}^P$, $m = \texttt{a}^M$ e $s = \texttt{a}^{n-P-M}\texttt{b}^n$.

Após bombearmos, teremos $pm^is = \texttt{a}^P\texttt{a}^{Mi}\texttt{a}^{n-P-M}\texttt{b}^n = \texttt{a}^{n+M(i-1)}\texttt{b}^n$. Esta palavra só irá pertencer a L se $n + M(i-1) = n$, ou seja, se $i = 1$. Isto significa que qualquer tentativa de bombeamento não funciona, logo w não pode ser bombeada. Portanto, L não é regular.

(b) $\{w \in \{\texttt{r}, \texttt{s}\}^* \ : \ w$ tem a mesma quantidade de r's e s's$\}$

Para um n possível comprimento de bombeamento, escolhemos a palavra $w = \texttt{r}^n\texttt{s}^n$, que pertence a L. Pelo mesmo argumento do item anterior, qualquer tentativa de dividirmos $w = pms$ e bombearmos resultará na palavra $pm^is = \texttt{r}^{n+M(i-1)}\texttt{s}^n$ (onde $M = |m|$). Esta palavra só irá pertencer a L se $i = 1$, logo w não pode ser bombeada. Portanto, L não é regular.

(c) $\{w \in \{\texttt{x}, \texttt{y}\}^* \ : \ w$ é um palíndromo$\}$

Para um n possível comprimento de bombeamento, escolhemos a palavra $w = \texttt{x}^n\texttt{yy}\texttt{x}^n$, que pertence a L. Para tentar bombear w, precisamos dividir w em três partes: $w = pms$, com $|m| \geq 1$ e $|pm| \leq n$. Agora, como $|pm| \leq n$, para qualquer divisão permitida, p e m devem ser formadas apenas por x's. Escrevendo $P = |p|$ e $M = |m|$, isto significa que $p = \texttt{x}^P$, $m = \texttt{x}^M$ e $s = \texttt{x}^{n-P-M}\texttt{yy}\texttt{x}^n$.

Após bombearmos, teremos $pm^is = \texttt{x}^P\texttt{x}^{Mi}\texttt{x}^{n-P-M}\texttt{yy}\texttt{x}^n = \texttt{x}^{n+M(i-1)}\texttt{yy}\texttt{x}^n$. Esta palavra só irá pertencer a L se $n+M(i-1) = n$, ou seja, se $i = 1$. Isto significa que qualquer tentativa de bombeamento não funciona, logo w não pode ser bombeada. Portanto, L não é regular.