Departamento de Ciência da Computação — ICEx/UFMG
Teoria de Linguagens — $2^{\underline{o}}$ semestre de 2017
Professor: Newton José Vieira
Primeira Lista de Exercícios/Solução do professor — Data de entrega: 19/9/2017 — Valor: 10 pontos

1. Uma versão do *problema da parada*, problema este a ser abordado no Capítulo 5, é: dado um programa (sem entrada), determinar se ele para. Lá no Capítulo 5 será visto que não existe algoritmo que, recebendo como entrada um (texto de) programa qualquer em Java (ou em outra linguagem de programação), determine se ele para ou não. Em outras palavras, o problema da parada não é computável. Sabendo disto, argumente que o problema "dado um programa qualquer em Java, determinar se ele emitirá algum sinal sonoro", também não é computável.

   Solução: Considere que a emissão de um sinal sonoro se dê via um certo comando S em Java (que comando é esse é irrelevante), de tal forma que um programa só possa emitir um sinal sonoro se ele contém uma ou mais ocorrências de S. Ora, antes de uma ocorrência de S, pode vir qualquer sequência de comandos. Como determinar se a execução dessa sequência de comandos para, de forma que S seja atingida em seguida? Impossível, pois o problema da parada não é computável!

2. Seja $\Sigma = \{\mathtt{0}, \mathtt{1}, \mathtt{\#}\}$. Sejam k e n números naturais tais que $k \leq n$.
   a) Quantas palavras há na linguagem Σ^k?
   b) Quantas palavras há na linguagem $(\Sigma \cup \{\lambda\})^k$?
   c) Quantas palavras de n símbolos contêm exatamente k ocorrências de #?
   d) Quantas palavras de n símbolos contêm no máximo k ocorrências de #?

   Solução:

   (a) Cada símbolo da palavra pode ser 0, 1 ou # (3 opções). Logo, existem 3^k palavras de k símbolos.
   (b) Número de palavras de zero a k símbolos: $\sum_{i=0}^{k} 3^i = (3^{i+1} - 1)/2$.
   (c) Número de posicionamentos para os k #s: $C(n, k)$; número de sequências dos símbolos restantes (0 e 1) nas posições restantes: 2^{n-k}. A resposta é, então, $C(n, k).2^{n-k}$.
   (d) Número de palavras com zero a k ocorrências de #: $\sum_{i=0}^{k} C(n, i).2^{n-i}$.

3. Seja Σ um alfabeto. Prove que se $x, y, w \in \Sigma^*$ e $xw = yw$, então $x = y$.

   Solução: Por indução sobre $|w|$. Para $w = \lambda$, se $x\lambda = y\lambda$, segue-se trivialmente que $x = y$. Seja $n \geq 0$, e suponha, como hipótese de indução, que para todo w tal que $|w| = n$, se $xw = yw$, então $x = y$. Seja w tal que $|w| = n + 1$. Segue-se que $w = za$, em que $z \in \Sigma^*$ e $a \in \Sigma$. Suponha que $x(za) = y(za)$. Pela associatividade da concatenação, $(xz)a = (yz)a$. Para que isso seja verdade, os prefixos xz e yz devem ser idênticos. Neste caso, pela hipótese de indução, $x = y$. Portanto, se $xw = yw$ para $|w| = n + 1$, então $x = y$.

4. Sejam $\Sigma = \{\mathtt{0}, \mathtt{1}\}$, $A = \Sigma^*\{\mathtt{0}\}$ e $B = \{\mathtt{1}\}\Sigma^*$. Descreva, em português, as linguagens a seguir. Não é necessário dizer que as palavras são de 0s e 1s. Por exemplo, A é o "conjunto das palavras que terminam com 0".

a) $A \cup B$;
b) $A \cap B$;
c) $A - B$;
d) $B - A$;
e) $\overline{A}$;
f) AB;
g) BA.

Solução:

a) $A \cup B$: conjunto das palavras que começam com `1` ou terminam com `0`.

b) $A \cap B$: conjunto das palavras que começam com `1` e terminam com `0`.

c) $A - B$: conjunto das palavras que começam e terminam com `0`.

d) $B - A$: conjunto das palavras que começam e terminam com `1`.

e) $\overline{A}$: conjunto das palavras que não terminam com `0`.

f) AB: conjunto das palavras que contêm `01`.

g) BA: conjunto das palavras que começam com `1` e terminam com `0`.

5. Sejam $A = \{\lambda, \mathtt{a}\}$ e $B = \{\mathtt{a}, \mathtt{b}\}$. Determine A^nB, AB^n e A^nB^n. Quantos elementos tem cada um desses conjuntos?

   Solução: $A^nB = \{\mathtt{a}^k s \,|\, 0 \leq k \leq n \text{ e } s \in \{\mathtt{a}, \mathtt{b}\}\}$ e $|A^nB| = 2(n+1)$; $AB^n = \{sy \,|\, s \in \{\lambda, \mathtt{a}\}, y \in \{\mathtt{a}, \mathtt{b}\}^* \text{ e } |y| = n\}$ e $|AB^n| = 2^{n+1}$; $A^nB^n = \{\mathtt{a}^k y \,|\, 0 \leq k \leq n, y \in \{\mathtt{a}, \mathtt{b}\}^* \text{ e } |y| = n\}$ e $|A^nB^n| = (n+1)2^n$.

6. Descreva, em português, as seguintes linguagens sobre o alfabeto $\{\mathtt{0}, \mathtt{1}\}$:

   a) $\{\mathtt{0}, \mathtt{1}\}^*\{\mathtt{1}\}\{\mathtt{0}, \mathtt{1}\}$;
   b) $\{\mathtt{0}\}\{\mathtt{0}, \mathtt{1}\}^* \cup \{\mathtt{0}, \mathtt{1}\}^*\{\mathtt{1}\}$;
   c) $\{\mathtt{0}, \mathtt{1}\}^*\{\mathtt{01}\}\{\mathtt{11}\}$;
   d) $\{\mathtt{01}, \mathtt{1}\}^*$.
   e) $\{\mathtt{1}, \lambda\}(\{\mathtt{0}\}\{\mathtt{0}\}^*\{\mathtt{1}\})^*\{\mathtt{0}\}^*$;
   f) $\{\mathtt{0}\}^*\{\mathtt{1}\}(\{\mathtt{0}\} \cup \{\mathtt{1}\}\{\mathtt{0}\}^*\{\mathtt{1}\})^*$.

   Solução:

   a) Conjunto das palavras cujo penúltimo símbolo é `1`.

   b) Conjunto das palavras que começam com `0` ou terminam com `1`.

   c) Conjunto das palavras que terminam com `0` imediatamente seguido de um número ímpar de `1`s.

   d) Conjunto das palavras em que todo `0` (caso haja algum) é seguido de `1`.

   e) Conjunto das palavras sem `1`s consecutivos.

   f) Conjunto das palavras com um número ímpar de `1`s.

7. Seja Σ o conjunto de todas as 26 letras do alfabeto e C o conjunto das 21 consoantes. Seja D o conjunto de todas as palavras de um dicionário da lingua portuguesa, e, para cada letra $l \in \Sigma$, seja $P_l = \{w \in D \,|\, w \text{ contém a letra } l\}$. A partir de tais conjuntos, usando operações sobre conjuntos, concatenações e fechos de Kleene, descreva o conjunto das palavras:

   a) que contêm pelo menos uma das letras A, B ou C;

b) que contêm as letras A e B, mas não C;
c) que não contêm A nem B;
d) que começam com a letra A;
e) que começam com a letra A e contêm pelo menos uma consoante;
f) que contêm a subpalavra RR.

Solução:

a) $P_{\text{A}} \cup P_{\text{B}} \cup P_{\text{C}}$;
b) $(P_{\text{A}} \cap P_{\text{B}}) - P_{\text{C}}$;
c) $D - (P_{\text{A}} \cup P_{\text{B}})$;
d) $(\{\text{A}\}\Sigma^*) \cap D$;
e) $(\{\text{A}\}\Sigma^* C \Sigma^*) \cap D$;
f) $(\Sigma^*\{\text{RR}\}\Sigma^*) \cap D$.

8. Sejam A, B e C linguagens sobre um alfabeto Σ. Mostre que:
a) $A(B \cup C) = (AB) \cup (AC)$;
b) nem sempre $A(B \cap C) = (AB) \cap (AC)$.

Solução:

a) $A(B \cup C) \subseteq (AB) \cup (AC)$
Seja $w \in \Sigma^*$ tal que $w \in A(B \cup C)$. Então existem x e y tais que $w = xy$, $x \in A$ e $y \in B \cup C$. No caso em que $y \in B$, segue-se que $xy \in AB$ e no caso em que $y \in C$, segue-se que $xy \in AC$. Logo, $xy = w \in (AB) \cup (AC)$.
$(AB) \cup (AC) \subseteq A(B \cup C)$
Seja $w \in \Sigma^*$ tal que $w \in (AB) \cup (AC)$. Então $w \in AB$ ou $w \in AC$. No primeiro caso, existem x e y tais que $w = xy$, $x \in A$ e $y \in B$. Como $y \in B \cup D$ para qualquer linguagem D, segue-se que $xy = w \in A(B \cup C)$. No caso em que $w \in AC$, existem x e y tais que $w = xy$, $x \in A$ e $y \in C$. Como $y \in D \cup C$ para qualquer linguagem D, segue-se que $xy = w \in A(B \cup C)$. Conclui-se, portanto, que $w \in A(B \cup C)$.

b) Contra-exemplo: $A = \{\lambda, a\}$, $B = \{b\}$, $C = \{ab\}$.

9. Prove que não existe $x \in \{\mathtt{0}, \mathtt{1}\}^*$ tal que $\mathtt{0}x = x\mathtt{1}$.

Solução: Será mostrado que para todo $x \in \{\mathtt{0}, \mathtt{1}\}^*$ $\mathtt{0}x \neq x\mathtt{1}$ por indução sobre $|x|$. Para $x = \lambda$, tem-se: $\mathtt{0}\lambda = \mathtt{0} \neq \mathtt{1} = \lambda\mathtt{1}$. Seja $n \geq 0$ e suponha, como hipótese de indução, que $\mathtt{0}x \neq x\mathtt{1}$ para qualquer x tal que $|x| = n$. Dada uma palavra qualquer w tal que $|w| = n + 1$, tem-se que existe um símbolo $s \in \{\mathtt{0}, \mathtt{1}\}$ tal que $w = xs$ e $|x| = n$. Dois casos:
Caso 1: $s = \mathtt{0}$. Tem-se: $\mathtt{0}w = \mathtt{0}(x\mathtt{0})$. Mas $\mathtt{0}x\mathtt{0} \neq x\mathtt{0}\mathtt{1}$, pois diferem no último símbolo. Logo, $\mathtt{0}(x\mathtt{0}) \neq (x\mathtt{0})\mathtt{1}$ e, portanto, $\mathtt{0}w \neq w\mathtt{1}$.
Caso 2: $s = \mathtt{1}$. Tem-se: $\mathtt{0}w = \mathtt{0}(x\mathtt{1}) = (\mathtt{0}x)\mathtt{1}$. Pela hipótese de indução, $\mathtt{0}x \neq x\mathtt{1}$ e, portanto, $(\mathtt{0}x)\mathtt{1} \neq (x\mathtt{1})\mathtt{1}$. Como $(x\mathtt{1})\mathtt{1} = w\mathtt{1}$, segue-se que $\mathtt{0}w \neq w\mathtt{1}$.

10. Seja a gramática $G = (\{A, B\}, \{\mathtt{a}, \mathtt{b}\}, R, A)$ em que R é constituído pelas quatro regras:

$$A \to \mathtt{a}A \mid B$$
$$B \to \mathtt{b}B \mid \lambda$$

Apresente um esquema de derivação que mostre como derivar qualquer palavra de $L(G)$. Expresse o que é $L(G)$ de forma bem concisa.

Solução:

$$\begin{aligned} A &\overset{n}{\Rightarrow} \mathtt{a}^n A && \text{(regra } A \to \mathtt{a}A\ n \text{ vezes, } n \geq 0\text{)} \\ &\Rightarrow \mathtt{a}^n B && \text{(regra } A \to B\text{)} \\ &\overset{k}{\Rightarrow} \mathtt{a}^n \mathtt{b}^k B && \text{(regra } B \to \mathtt{b}B\ k \text{ vezes, } k \geq 0\text{)} \\ &\Rightarrow \mathtt{a}^n \mathtt{b}^k && \text{(regra } B \to \lambda\text{)} \end{aligned}$$

$L(G) = \{\mathtt{a}\}^*\{\mathtt{b}\}^*$.

11. Construa gramáticas para as seguintes linguagens:

a) $\{\mathtt{0}\}\{\mathtt{0},\mathtt{1}\}^*\{\mathtt{11}\}$;
b) $\{\lambda, \mathtt{0}\}\{\mathtt{11}\}^*\{\lambda, \mathtt{0}\}$;
c) $\{w \in \{\mathtt{a},\mathtt{b}\}^* \mid$ o número de $\mathtt{a}$s em w é par$\}$;
d) $\{\mathtt{a}^n\mathtt{b}^n \mid n \in \mathbf{N}\}$;
e) $\{w \in \{\mathtt{a},\mathtt{b}\}^* \mid w = w^R\}$.

Para as duas últimas, determine o número de passos para gerar uma palavra de n símbolos.

Solução:

a) $P \to \mathtt{0}X\mathtt{11}$
$X \to \mathtt{0}X \mid \mathtt{1}X \mid \lambda$

b) $P \to AXA$
$A \to \mathtt{0} \mid \lambda$
$X \to \mathtt{11}X \mid \lambda$

c) $P \to \mathtt{a}I \mid \mathtt{b}P \mid \lambda$
$I \to \mathtt{a}P \mid \mathtt{b}I$

d) $X \to \mathtt{a}X\mathtt{b} \mid \lambda$

e) $X \to \mathtt{a}X\mathtt{a} \mid \mathtt{b}X\mathtt{b} \mid \lambda \mid \mathtt{a} \mid \mathtt{b}$

12. Diga que linguagens são geradas pelas gramáticas:

a) $G_1 = (\{A\}, \{\mathtt{0},\mathtt{1}\}, R_1, A)$, sendo R_1 constituído de:
$A \to \mathtt{0}A \mid A\mathtt{0} \mid \mathtt{1}$

b) $G_2 = (\{B\}, \{\mathtt{0},\mathtt{1}\}, R_2, B)$, sendo R_2 constituído de:
$B \to \mathtt{0}B\mathtt{00} \mid \mathtt{1}$

c) $G_3 = (\{S, A, B\}, \{\mathtt{0},\mathtt{1}\}, R_3, S)$, sendo R_3 constituído de:
$S \to AA \mid B$
$A \to \mathtt{0}A \mid A\mathtt{0} \mid \mathtt{1}$
$B \to \mathtt{0}B\mathtt{00} \mid \mathtt{1}$

Solução:

(a) $\{0\}^*\{1\}\{0\}^*$
(b) $\{0^n 1 0^{2n} \mid n \geq 0\}$
(c) $\{0\}^*\{1\}\{0\}^*\{1\}\{0\}^* \cup \{0^n 1^{2n} \mid n \geq 0\}$

13. Identifique as linguagens geradas pelas gramáticas:

a) $G_1 = (\{P, X\}, \{\mathtt{a}, \mathtt{b}\}, R_1, P)$.

R_1: $P \to \mathtt{a}X \mid \mathtt{b}P \mid \lambda$
$X \to \mathtt{a}P$

b) $G_2 = (\{P, A\}, \{\mathtt{a}, \mathtt{b}\}, R_2, P)$.

R_2: $P \to \mathtt{a}P\mathtt{a} \mid A$
$A \to \mathtt{b}A \mid \mathtt{b}$

c) $G_3 = (\{A, B\}, \{\mathtt{0}, \mathtt{1}\}, R_3, A)$.

R_3: $A \to \mathtt{0}A \mid B$
$B \to B\mathtt{1} \mid \mathtt{01}$

d) $G_4 = (\{P, X\}, \{\mathtt{a}, \mathtt{b}\}, R_4, P)$.

R_4: $P \to \mathtt{a}P \mid X\mathtt{b} \mid \lambda$
$X \to \mathtt{a}P$

e) $G_5 = (\{P, X\}, \{\mathtt{a}, \mathtt{b}\}, R_5, P)$.

R_5: $P \to \mathtt{aa}P \mid X\mathtt{b} \mid \lambda$
$X \to \mathtt{a}P$

Solução:

(a) $L(G_1) = \{\mathtt{aa}, \mathtt{b}\}^*$.
(b) $L(G_2) = \{\mathtt{a}^n\mathtt{b}^k\mathtt{a}^n \mid n \geq 0 \text{ e } k \geq 1\}$.
(c) $L(G_3) = \{\mathtt{0}\}^*\{\mathtt{01}\}\{\mathtt{1}\}^*$.
(d) $L(G_4) = \{\mathtt{a}^n\mathtt{b}^k \mid n \geq k\}$.
(e) $L(G_5) = \{\mathtt{a}^{k+2n}\mathtt{b}^k \mid n, k \geq 0\}$.

14. Sejam os PDs:

a) determinar se um número natural n, para $1 \leq n \leq 200$, é um número primo;
b) dada uma equação do segundo grau de coeficientes a, b e c, determinar se suas raízes são ambas reais, se cada um dos coeficientes é um número natural entre 10 e 20 (exclusive);
c) dada uma equação do segundo grau de coeficientes a, b e c, determinar se suas raízes são ambas reais, se seus coeficientes podem ser números reais quaisquer;
d) dada uma fórmula da lógica de predicados, determinar se ela é válida;
e) dados dois conjuntos finitos, determinar se eles são disjuntos;
f) determinar se uma árvore A tem altura menor ou igual a n;
g) determinar se uma palavra w é palíndromo, se $w \in \{\mathtt{0}, \mathtt{1}\}^*$.

Dizer quantos parâmetros e quantas instâncias tem cada um.

Solução:

a) 1 parâmetro, 200 instâncias;
b) 3 parâmetros, $9^3 = 729$ instâncias.
c) 3 parãmetros, conjunto infinito de instâncias;
d) 1 parâmetro, conjunto infinito de instâncias.
e) 2 parâmetros, conjunto infinito de instâncias.
f) 2 parâmetros, conjunto infinito de instâncias;

g) 1 parâmetro, conjunto infinito de instâncias.

15. Diz-se que um PD P é *redutível* a um PD Q, se existe um algoritmo $\mathcal{R}$ que, recebendo x como entrada, produz um resultado y tal que a resposta a P para a entrada x é idêntica ou complementar[1] à resposta a Q para a entrada y, qualquer que seja a entrada x. Diz-se, com isso, que o algoritmo $\mathcal{R}$ pode ser usado para *reduzir* o problema P ao problema Q.

    Seja D um PD decidível e I um PD indecidível. O que se pode dizer de um PD X, com relação à sua decidibilidade, se:

    a) D é redutível a X?
    b) X é redutível a D?
    c) I é redutível a X?
    d) X é redutível a I?

    Solução:

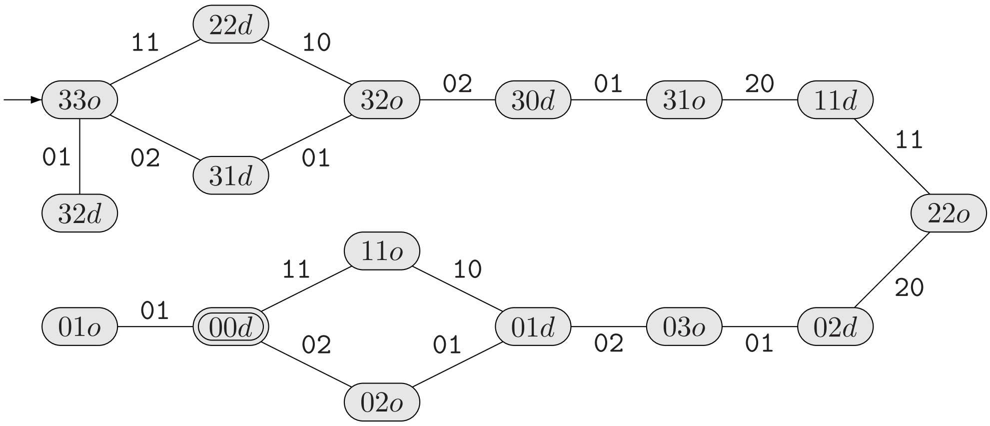
    a) Nada se pode dizer. X pode ser decidível ou não.
    b) X é decidível: para qualquer entrada x (para X), o algoritmo $\mathcal{R}$ produz uma saída y, a partir da qual um algoritmo para D obtém a resposta para x.
    c) X é indecidível: pelo item anteriror, se X fosse decidível, I seria decidível.
    d) Nada se pode dizer. X pode ser decidível ou não.

16. Faça um diagrama de estados, similar àquele do quebra-cabeça LCR, para o problema dos missionários e canibais:

    > Três missionários e três canibais devem atravessar um rio. Para isso, dispõem de uma canoa que pode transportar no máximo duas pessoas de cada vez. Durante a travessia, se o número de canibais for maior que o de missionários em qualquer uma das margens, os canibais comem os missionários. Determinar um plano para travessia em que nenhum missionário seja devorado.

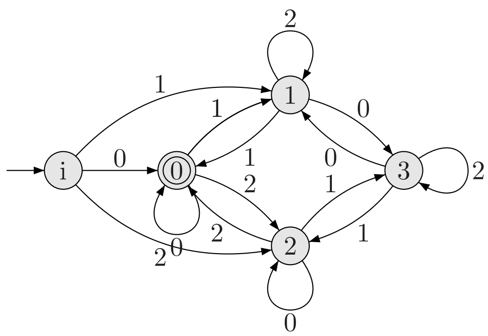
    Solução: Cada estado será uma palavra mcl, onde $m \in \{0, 1, 2, 3\}$ é o número de missionários do lado esquerdo, $c \in \{0, 1, 2, 3\}$ é o número de canibais do lado esquerdo, e $l \in \{o, d\}$ é o lado em que está a canoa (o: origem; d: destino). Cada transição terá um um rótulo da forma ij, onde $1 \leq i + j \leq 2$, sendo que i é o número de missionários e j é o número de canibais viajando na canoa. O diagrama de estados está mostrado na figura a seguir. Nele, cada aresta (v_1, r, v_2) representa duas transições: de v_1 para v_2 e de v_2 para v_1, ambas com rótulo r.

[1] A resposta complementar a *sim* é *não*, e a *não* é *sim*.



17. Faça um diagrama de estados, usando a ideia apresentada na Seção 2.1.2, para uma máquina que determine se uma sequência ternária (com dígitos 0, 1 e 2) é divisível por 4.

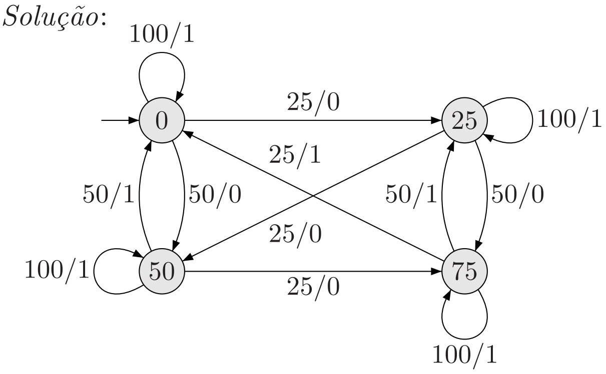
Solução:



18. Construa um diagrama de estados para uma máquina de Mealy que, dada uma sequência de moedas de 25 e 50 centavos e de 1 real, forneça uma lata de refrigerante quando a sequência totalizar 1 real ou mais. A cada moeda inserida deverá corresponder uma de duas saídas: `0`, se uma lata não pode ser (ainda) liberada, ou `1`, se uma lata deve ser liberada. Um exemplo de entrada e saída correspondente:

entrada:	50	25	50	100	25	50	25	⋯
saída:	0	0	1	1	0	1	0	⋯

Solução:



19. Construa AFDs que reconheçam:

a) $\{\lambda, \mathtt{0}\}^2$;

b) $\{\lambda, \mathtt{0}, \mathtt{1}\}^2\{\mathtt{1}\}^*$;

c) $\{011, 1\}^*$.

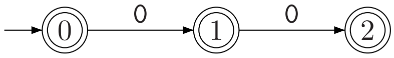
d) $\{\mathtt{0}, \mathtt{1}\}^* - \{\mathtt{0}, \mathtt{1}, \mathtt{00}, \mathtt{11}\}$;

e) $\{\mathtt{01}\}^*\{\mathtt{10}\}^*$;

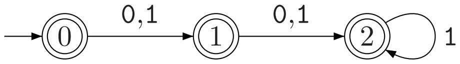
f) $\{00, 1\}^*\{0, 11\}^*$.

Solução:

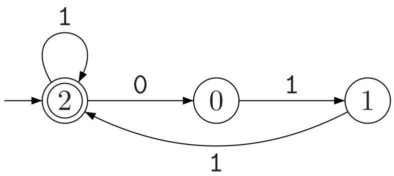
a)



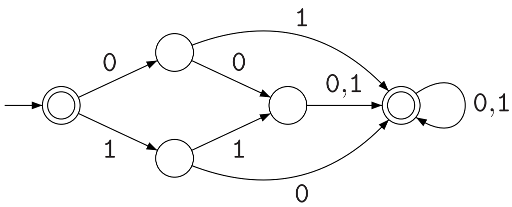
b)



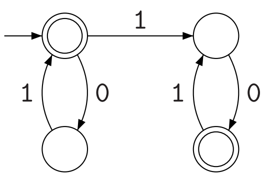
c)



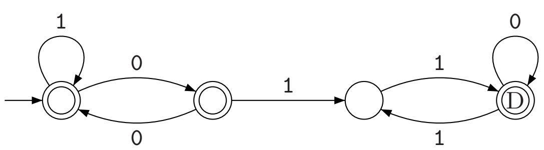
d)



e)



f)



20. Faça AFDs que reconheçam as linguagens:

a) $\{\mathtt{a}^m\mathtt{b}^n \mid m + n \geq 2\}$;

b) $\{\mathtt{a}^m\mathtt{b}^n \mid m + n \text{ é par}\}$;

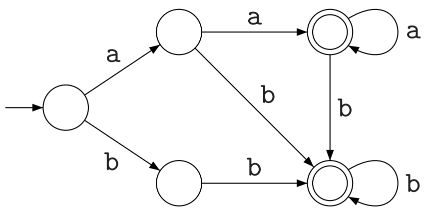
c) $\{\mathtt{a}^m\mathtt{b}^n \mid m + n \bmod 3 = 0\}$;

d) $\{\mathtt{a}^n\mathtt{b}^n \mid n \leq 3\}$.

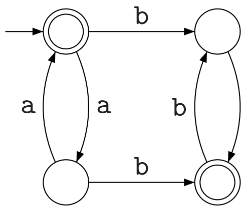
e) $\{\mathtt{a}^m\mathtt{b}^n \mid m \leq 2 \text{ e } n \geq m + 2\}$.

Solução:

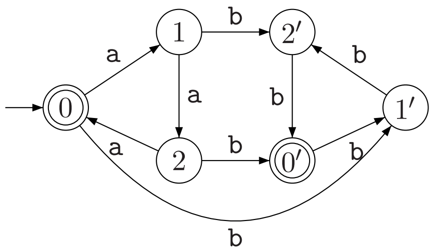
a)



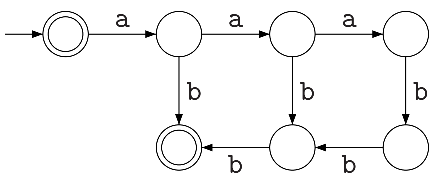
b)



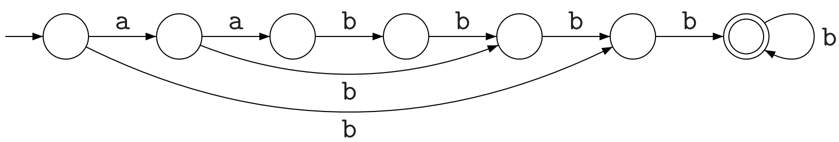
c)



d)



e)



21. Prove que $\hat{\delta}(e, xy) = \hat{\delta}(\hat{\delta}(e, x), y)$, em que δ é a função de transição de um AFD, e é um estado e x e y são palavras.

Solução: Por indução sobre $|x|$. Para $x = \lambda$, $\hat{\delta}(e, \lambda y) = \hat{\delta}(\hat{\delta}(e, \lambda), y)$, pela definição de $\hat{\delta}$. Suponha, como hipótese de indução, que $\hat{\delta}(e, xy) = \hat{\delta}(\hat{\delta}(e, x), y)$ para palavras x de tamanho n, $n \geq 0$. Para palavras de tamanho $n + 1$, az, onde a é um símbolo do alfabeto e z uma palavra de tamanho n, tem-se:

$\hat{\delta}(e, azy) = \hat{\delta}(\delta(e, a), zy)$ pela definição de $\hat{\delta}$

$$
\begin{aligned}
&= \hat{\delta}(\hat{\delta}(\delta(e,a),z),y) && \text{pela hipótese de indução} \\
&= \hat{\delta}(\hat{\delta}(e,az),y) && \text{pela definição de } \hat{\delta}.
\end{aligned}
$$

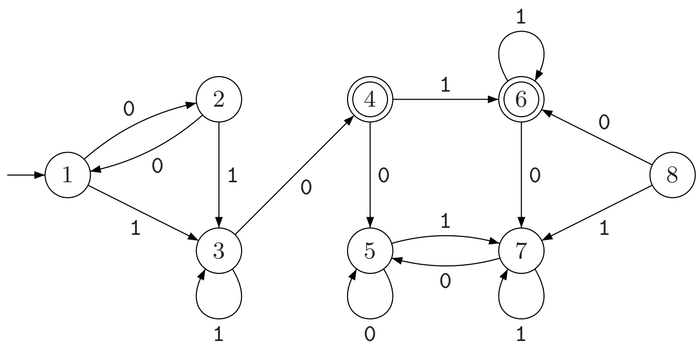
22. Dado um AFD M e um conjunto X de estados de erro em M, apresente em detalhes como seria um AFD M', equivalente a M, obtido pela substituição de todos os estados de X em M por um único estado de erro.

Solução: Seja um AFD $M = (E, \Sigma, \delta, i, F)$ e um conjunto de estados de erro $X \subseteq E$. Seja $\mathsf{x} \notin E$ e $M' = ((E - X) \cup \{\mathsf{x}\}, \Sigma, \delta', i', F)$ em que $i' = i$, se $i \notin X$, e $i' = \mathsf{x}$ se $i \in X$, e δ' é tal que

- $\delta'(e,a) = \delta(e,a)$ se $\delta(e,a) \notin X$,
- $\delta'(e,a) = \mathsf{x}$ se $\delta(e,a) \in X$, e
- $\delta'(\mathsf{x},a) = \mathsf{x}$ para todo $a \in \Sigma$.

Segue-se que $L(M') = L(M)$. Assim, em particular, o conjunto de todos os estados de erro pode ser reduzido a um só.

23. Seja o AFD com o seguinte diagrama de estados:



a) Obtenha o diagrama de estados de um AFD mínimo equivalente.

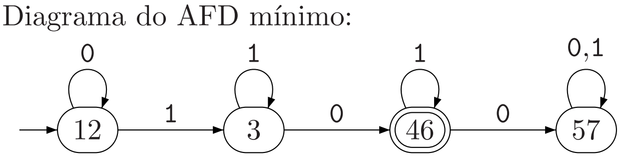
b) Que linguagem é reconhecida pelo AFD em questão?

Solução:

a) Evolução das partições do conjunto de estados após eliminação do estado 8:

S_0: $\{1,2,3,5,7\}$ $\{4,6\}$;
S_1: $\{1,2,5,7\}$ $\{3\}$ $\{4,6\}$;
S_2: $\{1,2\}$ $\{5,7\}$ $\{3\}$ $\{4,6\}$;
S_3: $\{1,2\}$ $\{5,7\}$ $\{3\}$ $\{4,6\}$.

Diagrama do AFD mínimo:



b) Linguagem: $\{\mathtt{0}\}^*\{\mathtt{1}\}\{\mathtt{1}\}^*\{\mathtt{0}\}\{\mathtt{1}\}^*$.

24. Utilizando produto de autômatos, determine AFDs que reconheçam

a) a união e

b) a interseção

das linguagens $\{w \in \{\mathtt{0},\mathtt{1}\}^* \mid n_1(w) \bmod 3 \neq 0\}$ e $\{w \in \{\mathtt{0},\mathtt{1}\}^* \mid |w| \bmod 3 = 0\}$.

Solução: O produto tem estados $E = \{0,1,2\} \times \{0,1,2\}$, em que cada estado (i,j), é atingido quando a palavra de entrada w for tal que $n_1(w) \bmod 3 = i$ e $|w| \bmod 3 = j$. O estado inicial é $(0,0)$ e a função de transição é dada por:

- $\delta((i,j),\mathtt{0}) = (i,(j+1) \bmod 3)$; e
- $\delta((i,j),\mathtt{1}) = ((i+1) \bmod 3,(j+1) \bmod 3)$.

a) Estados finais para união: $(0,0),(1,0),(1,1),(1,2),(2,0),(2,1)(2,2)$.

b) Estados finais para interseção: $(1,0),(2,0)$.

25. Sejam

- $L_1 = \{\mathtt{0},\mathtt{1}\}^*\{\mathtt{1}\}\{\mathtt{0},\mathtt{1}\}^*$;
- $L_2 = \{w \in \{\mathtt{0},\mathtt{1}\}^* \mid w$ não contém $\mathtt{00}\}$.

Construa autômatos finitos determinísticos que reconheçam:

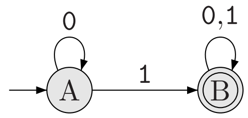
a) L_1.

b) L_2.

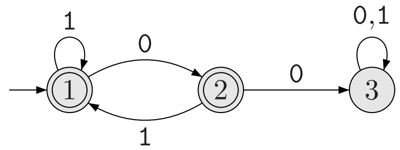
c) $L_1 - L_2$. Para isto, faça o produto dos autômatos que reconhecem L_1 e $\overline{L_2}$.

Solução:

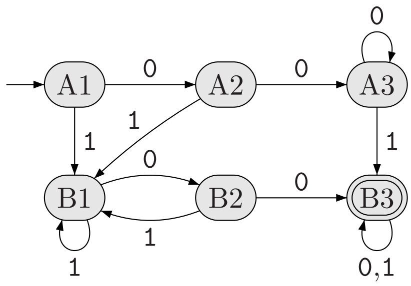
26. (a)



(b)



(c)



27. Sim ou não? Por quê?

a) Existe linguagem infinita que pode ser reconhecida por um AFD de apenas um estado.

b) Existe linguagem finita que pode ser reconhecida por um AFD de, no mínimo, um trilhão de estados.

c) Se uma linguagem pode ser reconhecida por um AFD, qualquer subconjunto dela também pode.

d) Se uma linguagem não pode ser reconhecida por um AFD e ela é subconjunto de L, então L também não pode ser reconhecida por um AFD.

e) Se um AFD M reconhece uma palavra cujo número símbolos é igual ao número de estados de M, então $L(M)$ é infinita.

f) Supondo que existam AFDs que reconheçam as linguagens A, B e C, então existirá AFD que reconhece $A - (B - C)$.

Solução:

a) Sim. O AFD $(\{i\}, \{\mathtt{0}\}, \delta, i, \{i\})$ tal que $\delta(i, \mathtt{0}) = i$ reconhece $\{\mathtt{0}\}^*$.

b) Sim. O número de estados de um AFD que reconhece uma linguagem finita deve ter no mínimo o tamanho da maior palavra reconhecida mais 1. Por exemplo, se uma palavra tiver 999.999.999.999 símbolos, o AFD terá no mínimo 1 trilhão de estados.

c) Não. $\{\mathtt{0}\}^*\{\mathtt{1}\}^*$ pode ser reconhecida, e $\{\mathtt{0}^n\mathtt{1}^n \mid n \geq 0\}$ não pode.

d) Não. Valem os exemplos do item anterior. Observe que as afirmativas são logicamente equivalentes.

e) Sim. Seja k o número de estados de M. Se M reconhece uma palavra w de k símbolos, o caminho percorrido no diagrama de estados de M tem $k + 1$ estados. Assim, algum estado é repetido no caminho, o que implica que o caminho tem um ciclo. Isto, por sua vez, implica que w é da forma xyz, em que y é a subpalavra consumida no ciclo. Como o ciclo pode ser percorrido quantas vezes se queira, tem-se que $xy^iz \in L(M)$ para todo $i \in \mathbf{N}$. Logo, $L(M)$ é infinita.

f) Sim. Observe que $A - (B - C) = A \cap \overline{B - C} = A \cap \overline{B \cap \overline{C}} = A \cap (\overline{B} \cup C)$. Como existe AFD para B, existe AFD para $\overline{B}$. Como existem AFDs para $\overline{B}$ e para C, existe AFD para $\overline{B} \cup C$. Finalmente, como existem AFDs para A e para $\overline{B} \cup C$, existe AFD para $A \cap (\overline{B} \cup C)$. Conclusão: existe AFD para $A - (B - C)$.

28. Construa AFNs para as seguintes linguagens sobre $\{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$:

a) o conjunto das palavras com, no mínimo, três ocorrências de `abc`;

b) o conjunto das palavras com, no mínimo, três ocorrências de `a`s ou três ocorrências de `b`s ou três de `c`s;

c) o conjunto das palavras com sufixo `abc` ou `bca`;

d) o conjunto das palavras em que existem duas ocorrências de `abc` com um número ímpar de símbolos entre elas;

e) o conjunto das palavras em que o último símbolo seja idêntico ao primeiro;

f) o conjunto das palavras em que o último símbolo seja diferente do primeiro;

g) o conjunto das palavras em que o último símbolo tenha ocorrido antes;

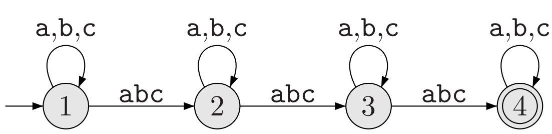
h) o conjunto das palavras em que o último símbolo tenha ocorrido antes no máximo uma vez;

i) o conjunto das palavras em que o último símbolo não tenha ocorrido antes;

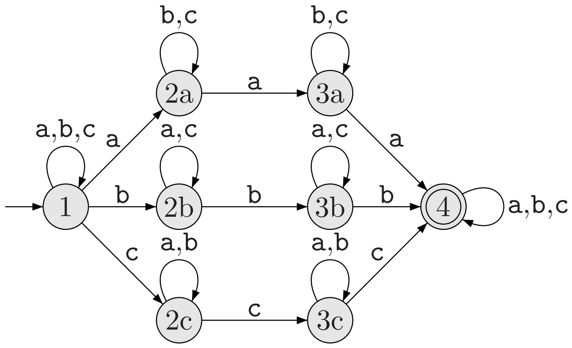
j) o conjunto das palavras com um único `a` ou um único `b` ou um único `c`.

Solução:

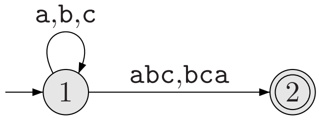
a)



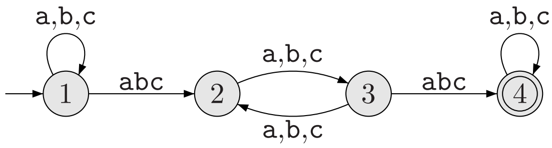
b)


b,c
b,c
2a
a
3a
a,b,c
a
a,c
a,c
a
1
b
2b
b
3b
b
4
a,b,c
c
a,b
a,b
c
2c
c
3c

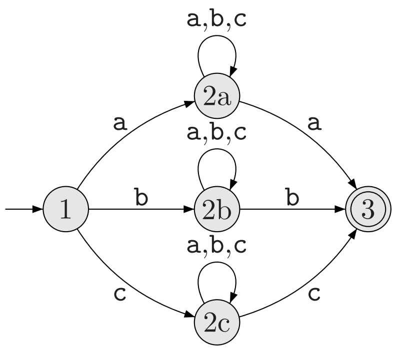
c)


a,b,c
1
abc,bca
2

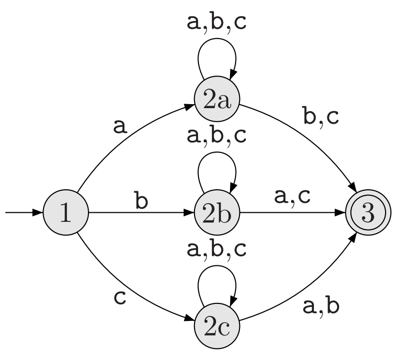
d)


a,b,c
a,b,c
a,b,c
1
abc
2
3
abc
4
a,b,c

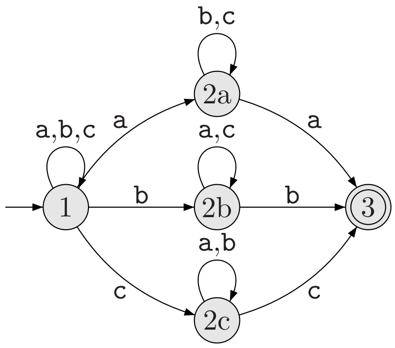
e)


a,b,c
2a
a
a
a,b,c
1
b
2b
b
3
a,b,c
c
2c
c

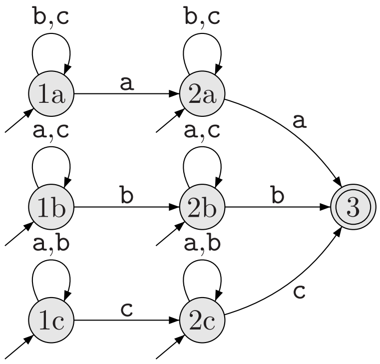
f)


a,b,c
2a
a
b,c
a,b,c
1
b
2b
a,c
3
a,b,c
c
2c
a,b

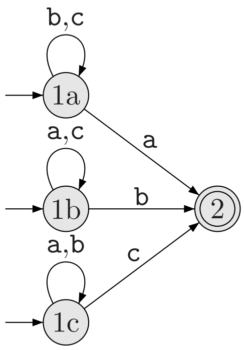
g)



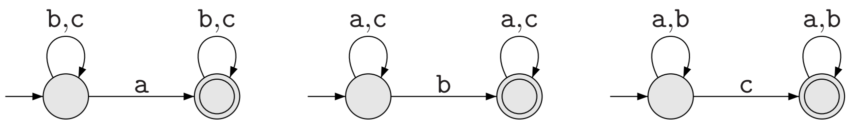
h)



i)



j)



29. Mostre que para todo AFN existe um AFN equivalente com um único estado inicial, para isto indicando um mínimo de alterações no AFN original. Exemplifique com o AFN da Figura 1.

Solução: Seja um AFN $M = (E, \Sigma, \delta, I, F)$. Um AFN com único estado inicial que reconhece $L(M)$ seria $M' = (E \cup \{i\}, \Sigma, \delta', i, F')$, onde i é um estado que não pertence a E e:

- $F' = F$ se $I \cap F = \emptyset$; caso contrário, $F' = F \cup \{i\}$;
- $\delta'(e, a) = \delta(e, a)$ para todo $(e, a) \in E \times \Sigma$;
- $\delta'(i, a) = \bigcup_{e \in I} \delta(e, a)$ para todo $a \in \Sigma$.

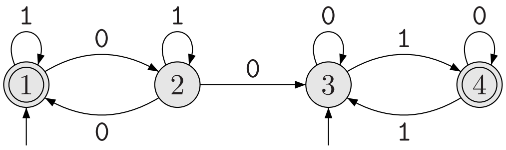
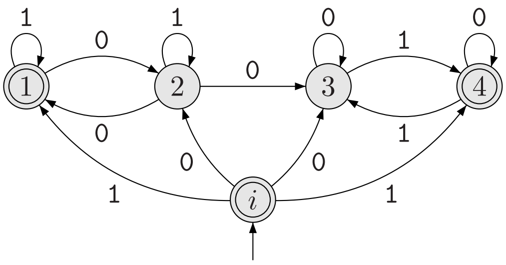


Figura 1: AFN para Exercícios 29 e 30.

Exemplo:

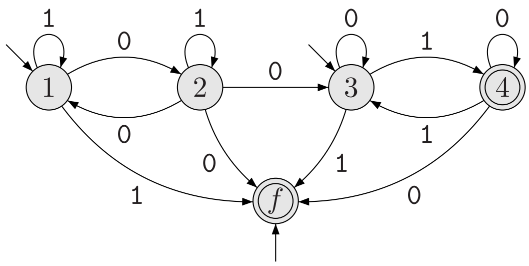


30. Mostre que para todo AFN existe um AFN equivalente com um único estado final. Exemplifique com o AFN da Figura 1.

Solução: Seja um AFN $M = (E, \Sigma, \delta, I, F)$. Um AFN com único estado final que reconhece $L(M)$ seria $M' = (E \cup \{f\}, \Sigma, \delta', I', \{f\})$, onde f é um estado que não pertence a E e:

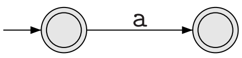
- $I' = I$ se $I \cap F = \emptyset$; caso contrário, $I' = I \cup \{f\}$;
- $\delta'(e, a) = \delta(e, a)$ se $\delta(e, a) \cap F = \emptyset$; caso contrário, $\delta'(e, a) = \delta(e, a) \cup \{f\}$;

Exemplo:

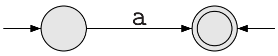


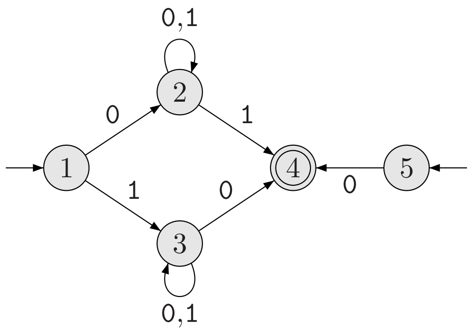
31. Mostre que existe linguagem regular que não pode ser reconhecida por AFN que tenha apenas um estado inicial **e** um estado final.

Solução: Para reconhecer $\{\lambda, \mathtt{a}\}$, um AFN de um único estado inicial tem que ter no mínimo dois estados finais. Por exemplo:

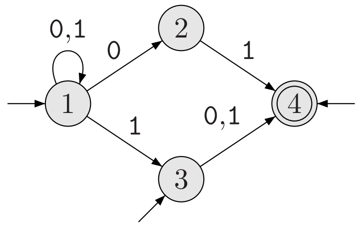


Para a mesma linguagem, um AFN de um único estado final tem que ter no mínimo dois estados iniciais. Por exemplo:

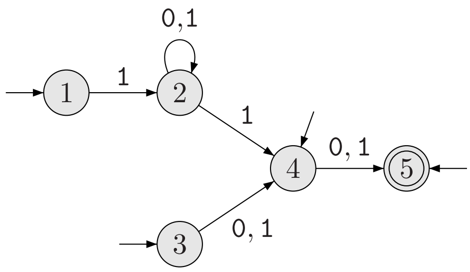




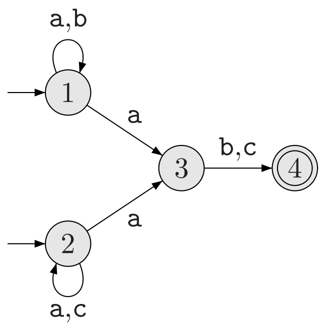
(a) AFN 1.



(b) AFN 2.



(c) AFN 3.



(d) AFN 4.

Figura 2: AFNs para Exercício 33.

32. Mostre que toda linguagem regular que não contenha a palavra λ pode ser reconhecida por AFN que tenha apenas um estado inicial **e** um estado final.

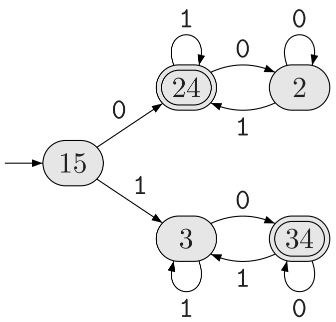
Solução: Sendo $(E, \Sigma, \delta, I, F)$ um AFN que reconhece uma linguagem sem a palavra vazia, então $I \cap F = \emptyset$. Um AFN equivalente de um único estado inicial e um único estado final seria $(E \cup \{i, f\}, \Sigma, \delta', \{i\}, \{f\})$ em que $i \notin E$, $f \notin E$ e δ' é assim definida:

- $\delta'(i, a) = \cup_{e \in I} \delta(e, a)$ para todo $a \in \Sigma$ (como $I \cap F = \emptyset$, i não é estado final);
- se $\delta(e, a) \notin F$ então $\delta'(e, a) = \delta(e, a)$, para todo $(e, a) \in E \times \Sigma$; e
- se $\delta(e, a) \in F$ então $\delta'(e, a) = \delta(e, a) \cup \{f\}$, para todo $(e, a) \in E \times \Sigma$ (como $I \cap F = \emptyset$, f não é estado inicial).

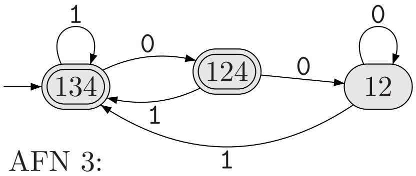
33. Sejam os AFNs com os diagramas de estados da Figura 2. Obtenha AFDs equivalentes aos AFNs utilizando o método de construção de subconjuntos.

Solução:

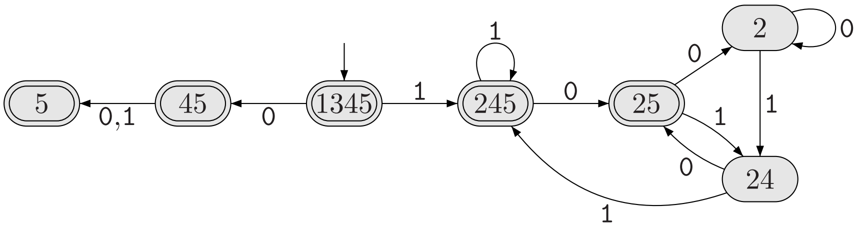
a) Para o AFN 1:



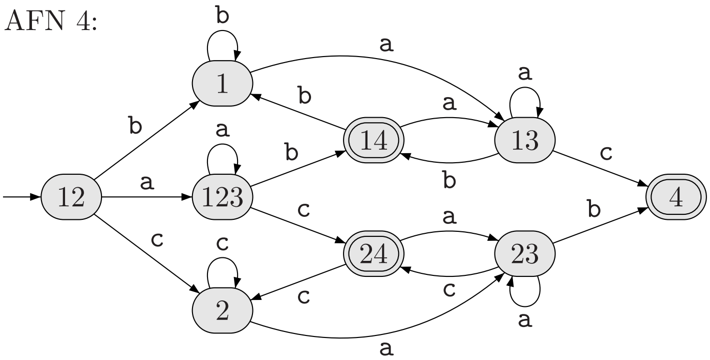
b) Para o AFN 2:



c) Para o AFN 3:



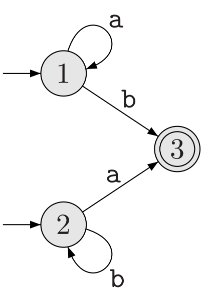
d) Para o AFN 4:



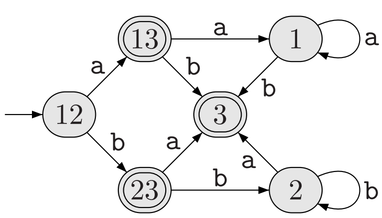
34. Construa um AFN de 3 estados que reconheça $\{\texttt{a}\}^*\{\texttt{b}\} \cup \{\texttt{b}\}^*\{\texttt{a}\}$. Em seguida, obtenha um AFD equivalente pelo método da construção de subconjuntos.

Solução:

AFN:



AFD:



35. Seja o AFNλ $M = (\{0,1,2\}, \{\mathtt{a},\mathtt{b},\mathtt{c}\}, \delta, \{0\}, \{2\})$, sendo δ dada por:

δ	a	b	c	λ
0	$\{0\}$	$\emptyset$	$\emptyset$	$\{1\}$
1	$\emptyset$	$\{1\}$	$\emptyset$	$\{2\}$
2	$\emptyset$	$\emptyset$	$\{2\}$	$\emptyset$

a) Determine $f\lambda(e)$ para $e = 0, 1, 2$.

b) Determine um AFN M' equivalente a M, usando a técnica para eliminação de transições λ.

c) Determine um AFD equivalente a M', usando o método de construção de subconjuntos.

Solução:

a) $f\lambda(0) = \{0,1,2\}$; $f\lambda(1) = \{1,2\}$; $f\lambda(2) = \{2\}$.

b) AFN $M' = (\{0,1,2\}, \{\mathtt{a},\mathtt{b},\mathtt{c}\}, \delta', \{0,1,2\}, \{2\})$, sendo δ' dada por:

δ'	a	b	c
0	$\{0,1,2\}$	$\emptyset$	$\emptyset$
1	$\emptyset$	$\{1,2\}$	$\emptyset$
2	$\emptyset$	$\emptyset$	$\{2\}$

c) AFD $M'' = (\{012, 12, 2, \emptyset\}, \{\mathtt{a},\mathtt{b},\mathtt{c}\}, \delta'', 012, \{012, 12, 2\})$, sendo δ'' dada por:

δ''	a	b	c
012	012	12	2
12	$\emptyset$	12	2
2	$\emptyset$	$\emptyset$	2
$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$

36. Sejam $L_1 = \{1, 01\}^+$ e $L_2 = \{0, 10\}^*$.

a) Construa AFNs (AFDs sem estados de erro) que reconheçam L_1 e L_2.

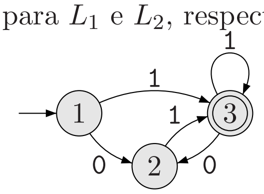
b) Obtenha um AFNλ que reconheça L_1L_2 usando transição λ, como exemplificado no Exemplo 37 da Seção 2.3.

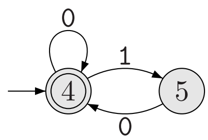
c) Obtenha um AFN equivalente usando a técnica para eliminação de transições λ.

d) Obtenha um AFD equivalente pelo método da construção de subconjuntos.

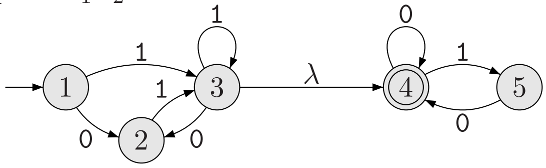
Solução:

a) AFNs para L_1 e L_2, respectivamente:

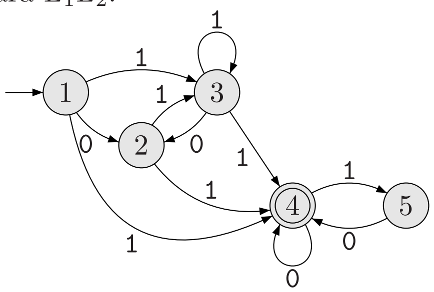




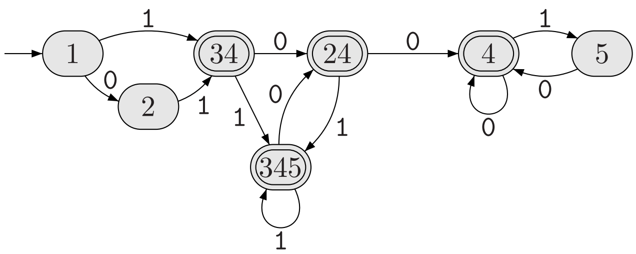
b) AFNλ para L_1L_2:



c) AFN para L_1L_2:



d) AFD para L_1L_2:



37. Espelhando-se nos Exemplos 37 a 40 da Seção 2.3 construa, a partir de AFNλs, AFDs que reconheçam:

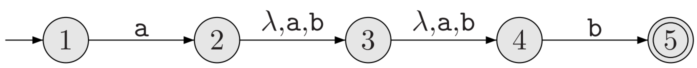
a) $\{\mathtt{a}\}\{\lambda, \mathtt{a}, \mathtt{b}\}^2\{\mathtt{b}\}$;

b) $\{\mathtt{a}\}^*(\{\mathtt{b}\}^* \cup \{\mathtt{c}\}^*)\{\mathtt{a}\}^*$;

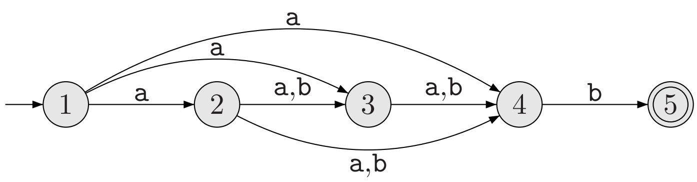
c) $\{\mathtt{aa}\}^*\{\mathtt{bb}\}^*\{\mathtt{cc}\}^*$.

Solução:

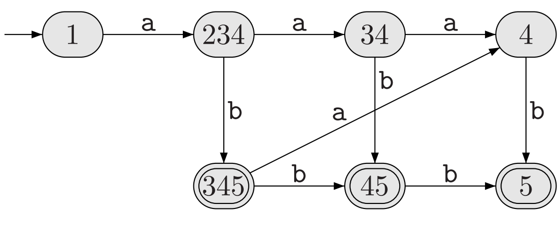
a) AFNλ:



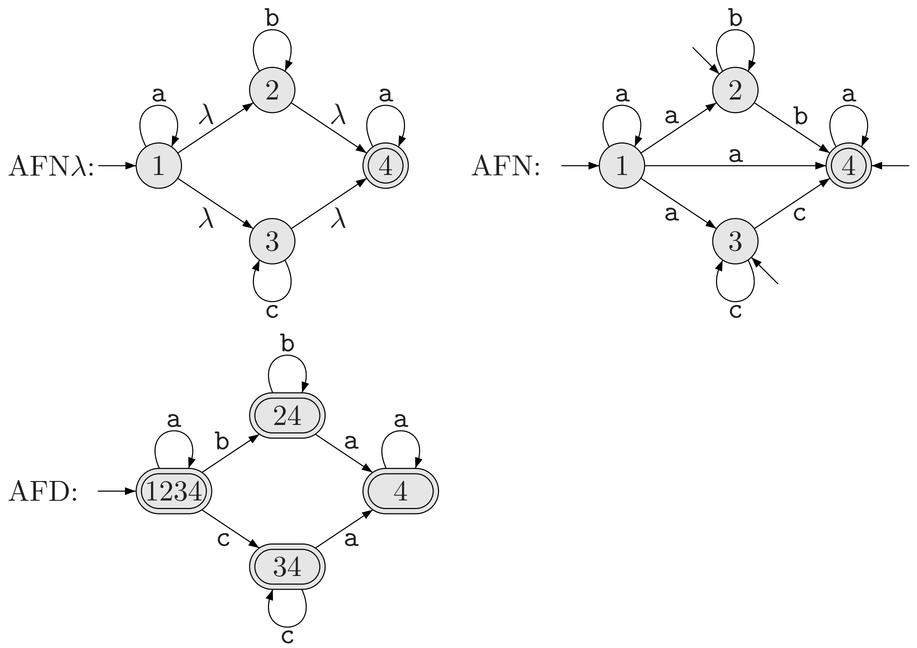
AFN:



AFD:



b)



c)