Departamento de Ciência de Computadores FCUP

Modelos de Computação CC1004 2020/2021

Folha Prática 6

Por definição, uma linguagem de alfabeto Σ é uma **linguagem regular** se existir uma expressão regular sobre Σ que a descreva. Provou-se que L é regular se e só se L pode ser reconhecida por um autómato finito, não sendo relevante indicar de que tipo é o autómato, uma vez que o poder computacional dos AFDs, AFNDs e AFNDs-ε, como reconhecedores de linguagens, é igual.

Teorema de Myhill-Nerode

Qualquer que seja a linguagem L de alfabeto Σ, as três condições seguintes são equivalentes:

- L é regular.
- L é união de classes de equivalência de alguma relação de equivalência definida em $\Sigma^\star$, invariante à direita para a concatenação e de índice finito.
- A relação de equivalência R_L é de índice finito em $\Sigma^\star$, sendo R_L definida por xR_Ly sse $\forall\, z \in \Sigma^\star\ (xz \in L \Leftrightarrow yz \in L)$.

Corolário do Teorema de Myhill-Nerode

Qualquer que seja a linguagem **regular** L de alfabeto Σ, o **AFD mínimo que reconhece** L é dado por $\mathcal{A} = (\Sigma^\star/R_L, \Sigma, \delta, [\varepsilon], F)$, com $\delta([x], a) = [xa]$, para todo $x \in \Sigma^\star$ e $a \in \Sigma$, e $F = \{[x] \mid x \in L\}$.

Exercícios

1. Em cada alínea, prove que o conjunto das classes de equivalência da relação R_L é infinito, sendo $\Sigma = \{\texttt{0}, \texttt{1}\}$, e conclua que L não é regular.

Sugestão: Em cada alínea, tente identificar uma família de palavras que têm de estar em classes distintas duas a duas.

a) $L = \{\texttt{0}^n\texttt{1}\texttt{0}^n \mid n \in \mathbb{N}\}$.

b) $L = \{ww \mid w \in \Sigma^\star\}$.

c) $L = \{wyw^R \mid w \in \{\texttt{0},\texttt{1}\}^\star, y \in \{\texttt{0},\texttt{1},\varepsilon\}\}$.

NB: w^R denota a **palavra reversa** de w, isto é, w escrita da direita para a esquerda (por exemplo, $(\texttt{1101})^R = \texttt{1011}$). Formalmente, tal operação é definida recursivamente por: $\varepsilon^R = \varepsilon$ e $(ax)^R = x^Ra$, para todo $x \in \Sigma^\star$ e $a \in \Sigma$.

2. Por aplicação do Teorema de Myhill-Nerode, averigue se cada uma das linguagens seguintes é ou não é regular, para $\Sigma = \{\texttt{0}, \texttt{1}\}$.

a) $\{x \mid$ o número de $\texttt{0}$'s em x não é múltiplo de cinco$\}$.

b) $\{x \mid x$ tem igual número de $\texttt{0}$'s e de $\texttt{1}$'s$\}$.

c) $\{x \mid$ a diferença entre o número de $\texttt{0}$'s e de $\texttt{1}$'s em qualquer prefixo de x é inferior a três$\}$.

d) $\{\texttt{1}y\texttt{1}x \mid x \in \{\texttt{0}\}^\star$ e $|y| \leq 2|x|\}$.

Sugestão: Em cada alínea, assuma que a linguagem L indicada é regular e tente construir o AFD mínimo que a aceitaria. Parta da classe de $[\varepsilon]$ e use $\delta([x], a) \stackrel{\text{def}}{=} [xa]$, para ver que estados (classes) surgiriam. Se concluir que o conjunto de estados (ou seja, o conjunto de classes de equivalência de R_L) é finito, então a máquina que obteve é o AFD mínimo que reconhece L. Se concluir que não pode ser finito então a máquina não seria um AFD (L não é regular).

Exercícios sobre o algoritmo de Moore e o teorema de Myhill-Nerode
(3.–8. baseados em exames de 2014/2015 e de 2015/2016)

3. Sejam $r = (((\mathtt{aa})^\star) + ((\mathtt{bb})^\star))$ e $s = (((\mathtt{aa}) + (\mathtt{bb}))^\star)$ expressões regulares sobre $\Sigma = \{\mathtt{a}, \mathtt{b}\}$.

a) Desenhe o AFD mínimo que aceita $\mathcal{L}(r)$ e o AFD mínimo que aceita $\Sigma^\star \setminus \mathcal{L}(s)$.

b) Desenhe os diagrama de transição dos autómatos finitos que resultam da aplicação do método de Thompson às expressões regulares r e s, segundo a construção dada nas aulas.

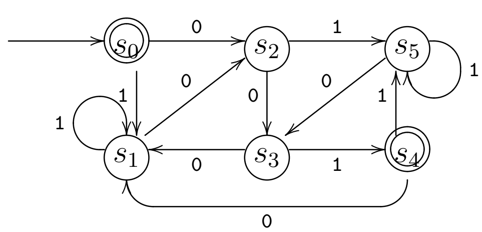
4. Seja L a linguagem de alfabeto $\Sigma = \{\mathtt{a}, \mathtt{b}\}$ constituída pelas palavras que têm $\mathtt{ab}$ como subpalavra e número ímpar de $\mathtt{a}$'s antes do $\mathtt{b}$ mais à esquerda na palavra.

a) Desenhe o diagrama do AFD mínimo que reconhece L.

b) Descreva $\mathcal{L}_s = \{x \mid x \in \Sigma^\star \text{ e } \hat{\delta}(s_0, x) = s\}$ por uma expressão regular (abreviada), para cada estado s do AFD que indicou em **4a)**, sendo s_0 o estado inicial.

c) Usando o corolário do teorema de Myhill-Nerode, prove a correção do AFD que indicou em **4a)**.

5. Aplicando o algoritmo de Moore, determine o AFD mínimo equivalente ao AFD seguinte.



Não desenhe duas tabelas. Use $\boxed{\equiv}$ e $\boxed{\mathtt{x}}$ para assinalar as entradas na *segunda* fase. Inclua as anotações intermédias.

NB: Na primeira fase (fase inicial), descartamos os estados não acessíveis do estado inicial, construimos a tabela com os restantes e assinalámos os pares (s_i, s_i), para todo i, e os pares (s_i, s_j), com $j \neq i$, em que $(s_i \in F \wedge s_j \notin F) \vee (s_i \notin F \wedge s_j \in F)$.

6. A tabela indicada abaixo, à direita, foi construída na fase inicial da aplicação do algoritmo de Moore a um dado AFD $A = (S, \{\mathtt{0}, \mathtt{1}\}, \delta, s_0, F)$, com δ função (total) de $S \times \{\mathtt{0}, \mathtt{1}\}$ em S. Justificando a resposta, analise as possibilidades de o AFD A coincidir com o AFD mínimo equivalente a A, sabendo que:

- $\delta(s_0, \mathtt{0}) = s_3 = \delta(s_1, \mathtt{0})$ e $\delta(s_2, \mathtt{0}) = s_1 = \delta(s_3, \mathtt{0})$,
- se $x \in \mathcal{L}((\mathtt{0} + \mathtt{1})^\star\mathtt{1})$ então $x \in \mathcal{L}(A)$,
- se $x \notin \mathcal{L}((\mathtt{0} + \mathtt{1})^\star\mathtt{1})$, nada nos foi dito sobre se $x \in \mathcal{L}(A)$ ou não,
- todos os estados em $S = \{s_0, s_1, s_2, s_3\}$ são acessíveis de s_0.

s_0	$\equiv$			
s_1	X	$\equiv$		
s_2		X	$\equiv$	
s_3	X		X	$\equiv$
	s_0	s_1	s_2	s_3

7. A tabela indicada abaixo, à direita, foi construída na fase inicial da aplicação do algoritmo de Moore a um dado AFD $A = (S, \{\mathtt{0}, \mathtt{1}\}, \delta, s_0, F)$, com δ função (total) de $S \times \{\mathtt{0}, \mathtt{1}\}$ em S. Justificando a resposta, analise as possibilidades de o AFD A coincidir com o AFD mínimo equivalente a A, sabendo que:

- $\delta(s_0, \mathtt{1}) = s_2 = \delta(s_3, \mathtt{1})$ e $\delta(s_1, \mathtt{1}) = s_3 = \delta(s_2, \mathtt{1})$,
- se $x \notin \mathcal{L}((\mathtt{0} + \mathtt{1})^\star\mathtt{0})$, nada nos foi dito sobre se $x \in \mathcal{L}(A)$ ou não,
- se $x \in \mathcal{L}((\mathtt{0} + \mathtt{1})^\star\mathtt{0})$ então $x \notin \mathcal{L}(A)$,
- todos os estados em $S = \{s_0, s_1, s_2, s_3\}$ são acessíveis de s_0.

s_0	$\equiv$			
s_1		$\equiv$		
s_2	X	X	$\equiv$	
s_3	X	X		$\equiv$
	s_0	s_1	s_2	s_3

8. Recordando a demonstração do corolário do teorema de Myhill-Nerode, que define o AFD mínimo para uma dada linguagem regular L de alfabeto Σ, explique o que garante que $|\Sigma^\star/R_L|$ é menor ou igual que o número de estados de qualquer AFD A tal que $\mathcal{L}(A) = L$.

9. Considere o AFD $A = (Q, \Sigma, \delta, q_0, \{q_0, q_5, q_6, q_8\})$ com $\Sigma = \{\mathtt{0}, \mathtt{1}, \mathtt{2}\}$, $Q = \{q_i \mid 0 \leq i \leq 9\}$ e função de transição δ dada por:

$\delta(q_0, \mathtt{0}) = q_1$	$\delta(q_0, \mathtt{1}) = q_1$	$\delta(q_0, \mathtt{2}) = q_2$	$\delta(q_1, \mathtt{0}) = q_4$	$\delta(q_1, \mathtt{1}) = q_4$
$\delta(q_1, \mathtt{2}) = q_3$	$\delta(q_2, \mathtt{0}) = q_4$	$\delta(q_2, \mathtt{1}) = q_3$	$\delta(q_2, \mathtt{2}) = q_4$	$\delta(q_3, \mathtt{0}) = q_4$
$\delta(q_3, \mathtt{1}) = q_5$	$\delta(q_3, \mathtt{2}) = q_6$	$\delta(q_4, \mathtt{0}) = q_3$	$\delta(q_4, \mathtt{1}) = q_6$	$\delta(q_4, \mathtt{2}) = q_5$
$\delta(q_5, \mathtt{0}) = q_7$	$\delta(q_5, \mathtt{1}) = q_6$	$\delta(q_5, \mathtt{2}) = q_6$	$\delta(q_6, \mathtt{0}) = q_7$	$\delta(q_6, \mathtt{1}) = q_5$
$\delta(q_6, \mathtt{2}) = q_5$	$\delta(q_7, \mathtt{0}) = q_7$	$\delta(q_7, \mathtt{1}) = q_6$	$\delta(q_7, \mathtt{2}) = q_5$	$\delta(q_8, \mathtt{0}) = q_7$
$\delta(q_8, \mathtt{1}) = q_6$	$\delta(q_8, \mathtt{2}) = q_9$	$\delta(q_9, \mathtt{0}) = q_0$	$\delta(q_9, \mathtt{1}) = q_8$	$\delta(q_9, \mathtt{2}) = q_5$

a) Desenhe o diagrama de transição de A e descreva informalmente $\mathcal{L}(A)$.

b) Aplique o algoritmo de Moore para minimizar o AFD A.

c) Assumindo que a descrição que indicou em **9a)** está correta, aplique o corolário do teorema de Myhill-Nerode para obter o AFD mínimo que reconhece $\mathcal{L}(A)$.

d) Verifique que os AFDs que obteve em **9b)** e **9c)** são iguais, a menos das designações dos estados.

Propriedades das linguagens regulares

- Linguagens regulares sobre Σ são, por definição, linguagens que podem ser descritas por expressões regulares sobre Σ. Vimos que as linguagens regulares podem ser reconhecidas por autómatos finitos e qualquer linguagem que pode ser reconhecida por uma autómato finito é regular.

- A **classe das linguagens regulares é fechada** para as operações de união, intersecção, concatenação, diferença, fecho de Kleene, complementação, e reverso. Ou seja, quaisquer que sejam as linguagens regulares L e M sobre Σ, as linguagens $L \cup M$, $L \cap M$, LM, $L \setminus M$, $L^\star$, $\overline{L}$ e L^R são linguagens regulares sobre Σ.

- **Autómato produto de dois AFDs**

  Um autómato produto de dois AFDs $A_1 = (S, \Sigma, \delta_1, s_0, F_1)$ e $A_2 = (Q, \Sigma, \delta_2, q_0, F_2)$ é um AFD

  $$A_1 \times A_2 = (S \times Q, \Sigma, \delta, (s_0, q_0), F),$$

  com $\delta((s, q), a) = (\delta_1(s, a), \delta_2(q, a))$, para $(s, q) \in S \times Q$, e $a \in \Sigma$. Tal AFD simula a execução de A_1 e A_2 paralelamente e, dependendo do modo como definimos F, poderá reconhecer $\mathcal{L}(A_1) \cap \mathcal{L}(A_2)$, $\mathcal{L}(A_1) \cup \mathcal{L}(A_2)$, $\mathcal{L}(A_1) \setminus \mathcal{L}(A_2)$, etc. Por exemplo:

  - se $F = F_1 \times F_2 = \{(s, q) \mid s \in F_1 \text{ e } q \in F_2\}$ então $\mathcal{L}(A_1 \times A_2) = \mathcal{L}(A_1) \cap \mathcal{L}(A_2)$;
  - se $F = (F_1 \times Q) \cup (S \times F_2) = \{(s, q) \mid s \in F_1 \text{ ou } q \in F_2\}$ então $\mathcal{L}(A_1 \times A_2) = \mathcal{L}(A_1) \cup \mathcal{L}(A_2)$.

  Se existirem estados em $S \times Q$ que não são acessíveis do estado inicial (s_0, q_0), podemos descartá-los e ficar com um AFD equivalente mas com menos estados.

- **Autómato para a linguagem reversa $\mathcal{L}(A)^R$, sendo A um AFD**

  Dado um AFD $A = (S, \Sigma, \delta, s_0, F)$, podemos definir um AFND-ε A' que reconhece $\mathcal{L}(A)^R$ assim:

  $$A' = (S \cup \{i\}, \Sigma, \delta', i, \{s_0\})$$

  sendo i é um estado novo e a função de transição δ' de $(S \cup \{i\}) \times (\Sigma \cup \{\varepsilon\})$ é dada por

  $$\begin{array}{rcl} \delta'(i, \varepsilon) & = & F \\ \delta'(s, a) & = & \{s' \mid \delta(s', a) = s\}, \text{para todo } s \in S \text{ e } a \in \Sigma \\ \delta'(s, \alpha) & = & \{\,\}, \text{para os restantes } (s, \alpha) \in (S \cup \{i\}) \times (\Sigma \cup \{\varepsilon\}) \end{array}$$

- **Existência de algoritmos de decisão sobre linguagens regulares**

  Nem todos os problemas podem ser resolvidos computacionalmente. Mas, alguns *problemas de decisão* sobre linguagens regulares e autómatos finitos podem ser resolvidos computacionalmente (ou seja, existe um *algoritmo* que produz uma resposta "sim/não" para cada instância do problema). **[NB: Mais à frente, aprofundaremos um pouco mais este tópico]**

  Alguns exemplos:

  – Existe um algoritmo para o problema de decisão "Dado um autómato finito A e dado $x \in \Sigma^\star$, determinar se $x \in \mathcal{L}(A)$".

  – Existe um algoritmo que resolve o problema de decisão "Dada uma expressão regular r e uma palavra $x \in \Sigma^\star$, determinar se $x \in \mathcal{L}(r)$".

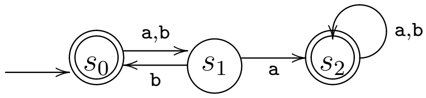
  – Existe um algoritmo que resolve o problema de decisão "Dados dois AFDs A_1 e A_2, determinar se $\mathcal{L}(A_1) \cup \mathcal{L}(A_2) = \Sigma^\star$". Também as respostas "sim/não" às questões "$\mathcal{L}(A_1) = \mathcal{L}(A_2)$?" (isto é, "os AFDs A_1 e A_2 são equivalentes?"), "$\mathcal{L}(A_1) \cap \mathcal{L}(A_2) = \emptyset$?", "$\mathcal{L}(A_1) \subseteq \mathcal{L}(A_2)$?", $\ldots$, podem ser obtidas por algoritmos.

  – Existe um algoritmo para o problema de decisão "Dadas duas expressões regulares r_1 e r_2, decidir se r_1 e r_2 são equivalentes" (i.e, decidir se "$\mathcal{L}(r_1) = \mathcal{L}(r_2)$?").

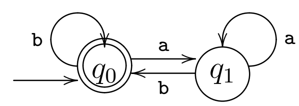
Exercícios

10. Sejam $A_1 = (\{s_0, s_1, s_2\}, \Sigma, \delta_1, s_0, \{s_0, s_2\})$ e $A_2 = (\{q_0, q_1\}, \Sigma, \delta_2, q_0, \{q_0\})$, com $\Sigma = \{\mathtt{a}, \mathtt{b}\}$ os AFDs representados pelos diagramas de transição seguintes:

AFD A_1:



AFD A_2:



Usando a construção do AFD produto, determine AFDs que reconheçam $\mathcal{L}(A_1) \cap \mathcal{L}(A_2)$, $\mathcal{L}(A_1) \cup \mathcal{L}(A_2)$, e $\mathcal{L}(A_1) \setminus \mathcal{L}(A_2)$. No autómato produto, crie apenas estados acessíveis do estado inicial (s_0, q_0).

11. Para os autómatos A_1 e A_2 indicados no exercício **10.**, determine AFDs que reconheçam $L(A_1)^R$ e $L(A_2)^R$, usando a construção para o reverso e a sua conversão para um AFD.

12. Sejam $L_1 = \mathcal{L}((\mathtt{0}+\mathtt{1})^\star\mathtt{01})$ e $L_2 = \mathcal{L}((\mathtt{0}+\mathtt{1})^\star\mathtt{10})$ linguagens de alfabeto $\Sigma = \{\mathtt{0}, \mathtt{1}\}$ e sejam A_1 e A_2 os AFDs mínimos que as reconhecem.

a) Por aplicação do corolário do teorema de Myhill-Nerode, determine A_1 e A_2.

b) Usando a construção do AFD produto de A_1 e A_2, determine um AFD que reconheça $L_1 \cup L_2$.

c) Aplique o algoritmo de Moore para minimizar o AFD que obteve em **12.b)**.

d) Descreva informalmente $L_1 \cup L_2 = \mathcal{L}((\mathtt{0}+\mathtt{1})^\star(\mathtt{01}+\mathtt{10}))$ e determine o AFD mínimo que reconhece $L_1 \cup L_2$, por aplicação do corolário do teorema de Myhill-Nerode. Verifique que corresponde ao AFD que obteve na alínea **12.c)**.

13. Prove a veracidade ou falsidade de cada uma das afirmações seguintes, assumindo que L e M são linguagens quaisquer de alfabeto Σ, com $|\Sigma| \geq 1$.

a) $L \setminus M$ é regular se L é regular.

b) $L \setminus M$ é regular só se L e M forem regulares.

c) $\overline{L^R}$ é regular se e só se L é regular.

d) Se $\overline{L}$ é regular então $L^\star$ e $L\overline{L}$ são regulares.

e) Se $\overline{L}$ não é regular então $L^\star$ e $L\overline{L}$ não são regulares.

f) Existe uma linguagem L tal que $LL = L^\star$.

g) Se L é regular então $L \cup M$ é regular.

h) Se L é regular e $M \subseteq L$ então M é regular.

i) Existe uma linguagem L, cujo complementar não é regular.

j) Para todo $a \in \Sigma$ e L regular, a linguagem $\{a^n x a^n \mid x \in L, n \in \mathbb{N}\}$ não é regular.

k) O subconjunto das linguagens regulares de alfabeto Σ é finito.

Lema da Repetição para Linguagens Regulares

Lema da repetição para linguagens regulares: Seja L uma linguagem regular de alfabeto Σ. Então existe $n \in \mathbb{Z}^+$ tal que, qualquer que seja $z \in L$ com $|z| \geq n$, existem $u, v, w \in \Sigma^\star$ tais que $z = uvw$, $|uv| \leq n$, $v \neq \varepsilon$ e $\forall_{i\geq 0}\ uv^iw \in L$.

$$L \text{ regular} \ \Rightarrow\ \left(\ \exists_{n\in\mathbb{Z}^+}\forall_{z\in\Sigma^\star}\ \ (z \in L \wedge |z| \geq n) \ \Rightarrow\ \exists_{u,v,w\in\Sigma^\star}\ \ (z = uvw \wedge |uv| \leq n \wedge v \neq \varepsilon \wedge \forall_{i\geq 0}\ \ uv^iw \in L)\ \right)$$

Observação:

- Na prova deste lema, partiu-se de um AFND $\mathcal{A}$ que reconhecia L (de facto, do AFND com menos estados) e tomou-se n igual o número de estados de $\mathcal{A}$. A prova seria análoga se $\mathcal{A}$ fosse um AFND ou AFD qualquer.
- A condição indicada no lema tem de ser satisfeita por **todas** as palavras "$z \in L$ com $|z| \geq n$", que existirem em L. As palavras "$z \in L$ com $|z| < n$", se existirem, são irrelevantes.
- Se L for finita, a condição verifica-se trivialmente se definirmos n como $1 + \max_{x\in L} |x|$, ou um inteiro maior do que este. Não existindo $z \in L$ com $|z| \geq n$, nada há que provar.
  Recorde que implicação $p \Rightarrow q$ é verdadeira se p é falso, pois equivale a $\neg p \vee q$.
- Se uma dada linguagem L **não satisfizer a condição** indicada no lema da repetição então L não é regular. Mas, importa salientar que existem linguagens que satisfazem a condição indicada e que não são regulares (por exemplo, a linguagem R definida no exercício **14.c)**).
- Não satisfazer a condição indicada quer dizer que: para todo o inteiro positivo n, existe $z \in L$ com $|z| \geq n$ e tal que, para **todas** as decomposições de z na forma $z = uvw$, com $|uv| \leq n$ e $v \neq \varepsilon$, se tem $uv^iw \notin L$, para algum $i \geq 0$.

  $$\forall_{n\in\mathbb{Z}^+}\exists_{z\in\Sigma^\star}\ \ (z \in L \wedge |z| \geq n)\ \ \wedge\ \ \forall_{u,v,w\in\Sigma^\star}\ ((z = uvw \wedge |uv| \leq n \wedge v \neq \varepsilon) \Rightarrow \exists_{i\geq 0}\ \ uv^iw \notin L)$$

  Assim, L **não é uma linguagem regular** se para todo o inteiro positivo n, existir $z \in L$ com $|z| \geq n$ e tal que, para todas as decomposições de z na forma $z = uvw$, com $|uv| \leq n$ e $v \neq \varepsilon$, se tem $uv^iw \notin L$, para algum $i \geq 0$. **Notar que a palavra z terá de ficar dependente de n.**

Exercícios

14. Sejam L, M e R as linguagens de alfabeto $\Sigma = \{0, 1, 2\}$ dadas por

$$L = \{x2y \mid x, y \in \{0,1\}^\star\} \qquad M = \{2^m w2w \mid w \in \{0,1\}^\star, m \geq 1\}$$
$$R = \mathcal{L}((0+1)^\star 2^\star (0+1)^\star) \cup M$$

a) Justifique que L é regular e, a seguir, mostre que satisfaz a condição do lema da repetição para linguagens regulares para $n = 2$ mas que não satisfaz a condição para $n = 1$.

b) Por aplicação do Lema de Repetição para linguagens regulares, prove que M não é regular.

c) Mostre que R satisfaz a condição do Lema da Repetição para linguagens regulares, para $n = 1$.

d) Por aplicação do teorema de Myhill-Nerode, prove que R não é regular.

15. Para cada uma das linguagens seguintes, de alfabeto $\Sigma = \{\mathtt{0}, \mathtt{1}, \mathtt{2}\}$, verifique se satisfaz ou não a condição imposta pelo *Lema da Repetição para Linguagens Regulares*.

Na justificação deve **mostrar diretamente se a condição se verifica ou não para** L. Nos casos em que é satisfeita, diga, justificando, se a linguagem é regular.

a) $L = \{y\mathtt{2}y \mid y \in \{\mathtt{0}, \mathtt{1}\}^\star$ e $|y| \geq 1\}$

b) $L = \{\mathtt{000}(\mathtt{10})^{2n} \mid n > 0\}$

c) $L = \{\mathtt{11}, \mathtt{102}, \varepsilon\}$

d) $L = \{\mathtt{22}^m w\mathtt{2}w^R \mid w \in \{\mathtt{0}, \mathtt{1}\}^\star$ e $m \geq 0\}$

e) $L = \{x\mathtt{22}y \mid x, y \in \{\mathtt{0}, \mathtt{1}\}^\star\{\mathtt{1}\}$ e $|x| > |y|\}$

f) $L = \{(\mathtt{22})^m y\mathtt{2}x \mid x, y \in \{\mathtt{0}, \mathtt{1}\}^\star, m \geq 1$ e $|x| = |y|\}$

g) $L = \{xzy \mid x, y \in \{\mathtt{0}\}^\star, z \in \{\mathtt{1}, \mathtt{2}\}^\star \setminus \{\varepsilon\}$ e $|x| - |y|$ é ímpar$\}$

h) $L = \{x \mid x \in \{\mathtt{0}, \mathtt{1}, \mathtt{2}\}^\star$ e x tem igual número de $\mathtt{0}$'s e $\mathtt{1}$'s$\}$

i) $L = \{x \mid x \in \{\mathtt{0}, \mathtt{1}, \mathtt{2}\}^\star$ e x não tem igual número de $\mathtt{0}$'s e $\mathtt{1}$'s$\}$

j) $L = \{x \mid x \in \{\mathtt{0}, \mathtt{1}, \mathtt{2}\}^\star$ e x tem igual número de $\mathtt{0}$'s, $\mathtt{1}$'s e $\mathtt{2}$'s$\}$

k) $L = \{\mathtt{0}^n \mid n$ primo$\}$

l) $L = \{a^{n^2} \mid n \geq 0, a \in \{\mathtt{0}, \mathtt{1}, \mathtt{2}\}\} \cup \{\mathtt{2}^3, \mathtt{1}^6\}$

m) $L = \{\mathtt{10}^n \mid n$ primo$\} \cup \{\mathtt{0}^n \mid n \in \mathbb{N}\}$

n) $L = \{\mathtt{0}^n \mid n$ primo$\}^\star$

o) $L = \{x \mid x$ começa por $\mathtt{1}$ e o número de $\mathtt{0}$'s em x é um quadrado perfeito$\}$