Universidade Federal Fluminense
Disciplina: Teoria da Computação
Professor: Luís Felipe

Gabarito (com algumas ideias de argumentos) da Lista de exercícios 1 – Definições iniciais e AFDs

1. Considerando o alfabeto $\Sigma = \{a, b\}$, responda as questões abaixo:

   (a) Seja $L_1 = \{b, baa, baaaa, baaaaaa, \ldots\}$. Descreva L_1 por meio da propriedade que a defina.

   Resposta: $L_1 = \{w | w$ começa por b e tem um único b e tem um número par de a's$\}$

   (b) $L_2 = \emptyset$. L_2 é linguagem em Σ? e $L_3 = \{\varepsilon\}$?

   Resposta: L_2 e L_3 são linguagens de Σ, pois ambas são subconjuntos de Σ^*.

   (c) Qual diferença entre L_2 e L_3?

   Resposta: L_2 é uma linguagem que não possui elementos, enquanto L_3 possui um único elemento, a saber, ε.

2. Dados L_1 e L_2 arbitrárias, mostre que $|L_1L_2| \leq |L_1||L_2|$. Existe caso da desigualdade ser estrita? Quando? Justifique.

   Resposta: Pela definição de concatenação de linguagens, temos que em L_1L_2 cada palavra de L_1 é concatenada com todas de L_2, sendo assim, há $|L_1||L_2|$ palavras pelo Princípio Multiplicativo. Porém, eventualmente existem palavras $x, y \in L_1$ e $z, t \in L_2$, tais que $xz = yt$, associando assim, elementos repetidos correspondendo a multiplicação $|L_1||L_2|$. Dessa forma, $|L_1L_2| \leq |L_1||L_2|$. Um exemplo onde a desigualdade é estrita aparece em $L_1 = \{a, aa\}$, $L_2 = \{a, \varepsilon\}$. Note que $L_1L_2 = \{aa, a, aaa\}$.

3. Mostre, por indução em n, que se $L_0, \cdots, L_n$ são linguagens no alfabeto Σ então $L_0(L_1 \cup \cdots \cup L_n) = (L_0L_1) \cup \ldots \cup (L_0L_n)$.

   Resposta: Vamos provar por indução em n a seguinte propriedade $P(n) : L_0(L_1 \cup \cdots \cup L_n) = (L_0L_1) \cup \ldots \cup (L_0L_n)$.

   Para a base de indução, verificamos que $P(1) : L_0(L_1) = L_0L_1$ como consequência da definição de concatenação entre linguagens.

   Assumimos, por hipótese de indução (HI), que para qualquer $k > 1$, $P(k-1)$ seja verdadeiro, ou seja: $P(k-1) : L_0(L_1 \cup \cdots \cup L_{k-1}) = (L_0L_1) \cup \ldots \cup (L_0L_{k-1})$.

Queremos provar que $P(k)$ é verdadeira, ou seja, $P(k) : L_0(L_1 \cup \cdots \cup L_k) = (L_0L_1) \cup \ldots \cup (L_0L_k)$.

Consideremos a expressão do lado direito da igualdade, queremos concluir a expressão do lado esquerdo. Sendo assim:

$$\begin{aligned}
(L_0L_1) \cup \ldots \cup (L_0L_k) &= (L_0L_1) \cup \ldots \cup (L_0L_{k-1}) \cup (L_0L_k) \\
&= \underbrace{(L_0L_1) \cup \ldots \cup (L_0L_{k-1})}_{H.I} \cup (L_0L_k) \\
&= L_0 \quad \underbrace{(L_1 \cup \cdots \cup L_{k-1})}_{\substack{\text{Como a união de linguagens} \\ \text{é também uma linguagem,} \\ \text{chamemos essa união de L'}}} \quad \cup (L_0L_k) \\
&= \underbrace{L_0L' \cup L_0L_k}_{\text{por H.I}} \\
&= L_0(L' \cup L_k) \\
&= L_0(L_1 \cup \cdots \cup L_{k-1} \cup L_k).
\end{aligned}$$

Dessa forma, temos $P(k)$ verdadeira, e portanto, $P(n)$ é verdadeira para todo $n \in \mathbb{N}$.

4. Mostre que $(L^*)^* = L^*$.

   Resposta (Uma ideia de argumento por inclusão, não única): Primeiramente vamos mostrar que $L^* \subseteq (L^*)^*$. Para isso, note que, pela consequência da definição da operação estrela, todo elemento de A pertence a A^*, para qualquer conjunto A arbitrário. Logo, tomando $A = L^*$, temos que todo elemento de L^* pertence a $(L^*)^*$. Vamos agora mostrar que $(L^*)^* \subseteq L^*$. Note que qualquer elemento em $(L^*)^*$ é obtido por concatenações arbitrárias de elementos em L^*. Se houvesse algum elemento de L^* que não fosse elemento de $(L^*)^*$ então existiria ao menos um elemento de L^* que não pertenceria a $(L^*)^*$, mas contrariaria o fato que qualquer elemento em $(L^*)^*$ possa ser obtido por concatenações arbitrárias de L^*.

5. Mostre que para qualquer linguagem regular L, se $\varepsilon \in L$, então todo AFD que reconhece L satisfaz $q_0 \in F$.

   Resposta: Como $\varepsilon \in L$, essa palavra deve ser consumida pelo AFD associado a L e concluir sua leitura num estado de final. Como ε não possui símbolo do alfabeto, portanto o estado inicial q_0 deve também ser um estado final.

6. Cada uma das linguagens a seguir é uma interseção de duas linguagens mais simples. Em cada caso, construa AFDs para as linguagens mais simples, e depois combine-os para obter o diagrama de estados de um AFD para a linguagem dada. Em todos os casos, $\Sigma = \{a, b\}$.

   (a) $\{w \mid w \text{ tem pelo menos três } a\text{'s e pelo menos dois } b\text{'s}\}$.

(b) $\{w \mid w$ tem um número par de a's e cada a é seguido por pelo menos um $b\}$.

Resposta (dicas): Para os dois itens, lembre dos algoritmos de fechamento em AFD de complemento e interseção.

7. Determine AFDs que reconheçam as seguintes linguagens com alfabeto $\{0, 1\}$:

   (a) $\{w \mid w$ acaba em $00\}$.

   (b) $\{w \mid$ cada 0 está entre dois $1\}$.

   (c) $\{w \mid w$ não contém subcadeia $01\}$.

   Resposta (dicas): Reveja os autômatos das aulas e atenção: todo estado num AFD deve possuir **uma única** seta saindo com cada símbolo.

8. Considere as seguintes linguagens com alfabeto $\Sigma = \{a, b\}$ e obtenha os AFDs que as reconheça.

   (a) $A = \{w \mid w$ tem exatamente dois a's ou não tem exatamente dois b's$\}$.

   (b) $A' = \{w \mid w$ não tem exatamente dois a's e tem exatamente dois b's$\}$.

   Resposta (dicas): Para os dois itens, lembre dos algoritmos de fechamento em AFD de complemento, união e interseção.

9. Considere duas linguagens regulares L_1 e L_2 com alfabetos distintos Σ_1 e Σ_2, respectivamente. Mostre que as linguagens obtidas tanto pela união quanto pela interseção de L_1 e L_2 são regulares.

   Resposta (ideia bem geral de uma das possíveis soluções): Uma das formas de obter esses AFD's é transformar L_1 e L_2 em linguagens com um mesmo alfabeto $\Sigma_1 \cup \Sigma_2$. Isso pode ser feito transformando cada AFD em um novo com mais um único estado cada. Todo estado terá agora transições de cada novo símbolo indo para o novo estado. Além disso, haverá um *loop* nesse novo estado com todos os símbolos de $\Sigma_1 \cup \Sigma_2$. Além disso, este não será um estado de aceitação.

   Após estas transformações, note que ambos os AFD's terão o mesmo alfabeto e essa transformação não altera o reconhecimento das linguagens associadas. Portanto, podemos agora aplicar os algoritmos de união e interseção vistos em aula.

10. Mostre, por indução em n, condições necessárias e suficientes para que uma palavra de tamanho n seja concluída em cada estado do autômato da Figura 1. Como consequência, determine a linguagem que seja reconhecida por esse autômato:

    Resposta: Reveja a indução trabalhada em aula.

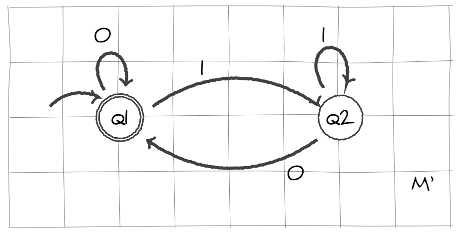

Figura 1: Autômato do Exercício 10.

11. Dê exemplo de uma linguagem que é aceita por um AFD com mais de um estado final, mas que não é aceita por nenhum AFD com apenas um estado final. Justifique.

    Resposta: Um exemplo pode ser visto no item 1 da AVON 1, a saber, $L = \{w \mid w = \varepsilon$ ou w possui um número ímpar de a's e w só possui a's$\}$.

    Argumento para responder a pergunta deste enunciado foi pedido na AVON.