Ejercicio 1. Sea $\Sigma_1 = \{0,1\}$ y $\Sigma_2 = \{a,b,c\}$ alfabetos. Listar 5 cadenas para cada uno de los alfabetos.

Cadenas para Σ_1

"011"
"100"
"101010"
"010"
"000001"

Cadenas para Σ_2

"abc"
"cba"
"bac"
"cab"
"baca"

Ejercicio 2. Sea $\Sigma = \{a,b\}$ un alfabeto y $\alpha = aa, \beta = bb$, obtener: $|\alpha\beta|$, $\alpha\epsilon$, $\alpha\alpha$, $\alpha\beta$, β^0, β^1, β^2, β^3, $\alpha^2\beta^2$, $(\alpha\beta)^2$, $(\alpha\beta)^R$ y el conjunto Σ^*.

$|\alpha\beta| = |\alpha| + |\beta| = 2+2 = 4$

$\alpha\alpha = aaaa$

$\alpha\beta = aabb$

¿$\beta^0 = \varepsilon$?

$\beta^1 = bb$

$\beta^2 = bbbb$

$\beta^3 = bbbbbb$

$\alpha\varepsilon = \alpha = aa$

$\alpha^2\beta^2 = aaaabbbb$

$(\alpha\beta)^2 = aabbaabb$

$(\alpha\beta)^R = bbaa$

$\Sigma^* = \{ \{\varepsilon, aa, bb\}^* \}$

Ejercicio 3. Dar una definición recursiva del operador de concatenación de cadenas y probar que es asociativa.

Veamos la definición del teórico primero.

$$\alpha.\beta = \begin{cases} \alpha & \text{si } \beta = \epsilon \\ \beta & \text{si } \alpha = \epsilon \\ a_1 \ldots a_n b_1 \ldots b_m & \text{si } \alpha = a_1 \ldots a_n \text{ y } \beta = b_1 \ldots b_m \end{cases}$$

Casos base

$\alpha = \varepsilon \Rightarrow \alpha . \beta = \beta$

$\beta = \varepsilon \Rightarrow \alpha . \beta = \alpha$

Caso recursivo Sea $\alpha = a\gamma$ donde a es el primer elemento de α e γ es el resto de la cadena.

$\alpha\beta = a(\gamma\beta)$

Demostración de asociatividad:

Queremos ver que dados $\alpha, \beta, \gamma \in \Sigma^*$ se cumple que $\alpha(\beta\gamma) = (\alpha\beta)\gamma$

Caso base $\alpha = \varepsilon$

- $\alpha(\beta\gamma) = \varepsilon(\beta\gamma) = \beta\gamma$
- $(\alpha\beta)\gamma = (\varepsilon\beta)\gamma = \beta\gamma$ ✓ se cumple el caso base.

Caso Inductivo: supongamos una cadena de longitud n en la que se cumple para $\alpha = a\lambda$

$(\alpha\beta)\gamma = (a\lambda\beta)\gamma = a(\lambda\beta\gamma) = a(\lambda(\beta\gamma)) = \alpha(\beta\gamma)$ ✓

se cumple.

Ejercicio 4. Dar una definición recursiva del operador de longitud de una cadena y probar que $|\alpha\beta| = |\alpha| + |\beta|$.

$$|\alpha| = \begin{cases} 0 & \text{si } \alpha = \epsilon \\ n & \text{si } \alpha = a_1 \ldots a_n \end{cases}$$

Veamos caso base $\alpha = \varepsilon$

$|\alpha| = 0$

Veamos caso recursivo $\alpha = a\lambda$ donde a es el primer elemento y λ el resto

$|\alpha| = |a\lambda| = 1 + |\lambda|$

Probemos $|\alpha\beta| = |\alpha| + |\beta|$

Veamos $\alpha = \varepsilon$

$|\alpha\beta| = |\beta|$

$|\alpha\beta| = |\alpha| + |\beta| = 0 + |\beta| = |\beta|$ ✓

Veamos para un $\alpha = a\lambda$ con HI $|\lambda\beta| = |\lambda| + |\beta|$

$|\alpha\beta| = |a\lambda.\beta| = 1 + |\lambda\beta| \overset{HI}{=} 1 + |\lambda| + |\beta| = (1 + |\lambda|) + |\beta| = |\alpha| + |\beta|$

Ejercicio 5. Dar una definición recursiva del operador de potencia de una cadena y probar que $\alpha^n\alpha^m = \alpha^{n+m}$.

Caso base: $n = 0$

$\alpha^0 = \varepsilon$

Caso recursivo α^{n+1}

$\alpha^{n+1} = \alpha . \alpha^n$

Probar $\alpha^n \alpha^m = \alpha^{n+m}$

Veamos $n=0$

$\alpha^0 \alpha^m = \varepsilon \alpha^m = \alpha^m = \alpha^{0+m}$ ✓

Supongamos que para un $n=k$ vale $\alpha^k \alpha^m = \alpha^{k+m}$, veamos $n=k+1$

$\alpha^{k+1} . \alpha^m = (\alpha . \alpha^k) \alpha^m = \alpha (\alpha^k . \alpha^m) \overset{HI}{=} \alpha . \alpha^{k+m} \overset{def}{=} \alpha^{k+m+1} = \alpha^{(k+1)+m}$ ✓

Ejercicio 6. Dar una definición recursiva del operador de reversa de una cadena y probar que $(\alpha\beta)^R = \beta^R\alpha^R$.

Caso base $\alpha = \varepsilon$

$\varepsilon^R = \varepsilon \qquad \alpha^R = \alpha$

Caso Recursivo: $\alpha = a\lambda$

$\alpha^R = (\lambda^R) a$

Probar que $(\alpha\beta)^R = \beta^R \alpha^R$

Caso base $\alpha = \varepsilon$

$(\alpha \beta)^R = \beta^R = \beta^R . \varepsilon^R = \beta^R . \varepsilon$ ✓

Caso inductivo: supongamos que para una cadena de longitud n vale que $(\lambda\beta)^R = \beta^R \lambda^R$, veamos para $\alpha = a\lambda$

$(\alpha\beta)^R = (a\lambda\beta)^R \overset{R\ def}{=} (\lambda\beta)^R a \overset{HI}{=} \beta^R \lambda^R a = \beta^R \alpha^R$ ✓

Ejercicio 7. Sea $\Sigma = \{a, b\}$ un alfabeto, definir formalmente (mediante comprensión de conjuntos) los siguientes lenguajes:

a) • L_1 es el lenguaje de todas las cadenas de longitud 2.

b) • L_2 es el lenguaje de todas las cadenas que comienzan con dos a's.

c) • L_3 es el lenguaje de todas las cadenas que tienen exactamente una sola b.

d) • L_4 es el lenguaje de todas las cadenas que comienzan y terminan con a.

e) • L_5 es el lenguaje de todas las cadenas que contienen solamente b's.

f) • L_6 es el lenguaje de todas las cadenas que tienen una cantidad par de a's.

g) • L_7 es el lenguaje de todas las cadenas tal que la cantidad de a's es multiplo de la cantidad de b's.

h) • L_8 es el lenguaje de todas las cadenas capicuas de longitud par.

a) $L_1 = \{\varphi \in \Sigma^* \mid |\varphi| = 2\} = \{aa, ab, ba, bb\}$

b) $L_2 = \{\varphi \in \Sigma^* \mid \varphi = aa.\psi, \psi \in \Sigma^*\}$

c) $L_3 = \{\varphi \in \Sigma^* \mid \varphi = a^* b a^*\}$

d) $L_4 = \{\varphi \in \Sigma^* \mid a.\psi.a, \psi \in \Sigma^*\}$

e) $L_5 = \{\varphi \in \Sigma^* \mid \varphi = b^*\} = b^*$

f) $L_6 = \{ \varphi \in \Sigma^* \mid$ El número de a, en φ es par $\}$

g) $L_7 = \{ \varphi \in \Sigma^* \mid \#(a \in \varphi) = k . \#(b \in \varphi), k \in \mathbb{N} \}$

h) $L_8 = \{ \varphi \in \Sigma^* \mid \varphi = \psi . \psi^R, \psi \in \Sigma^* \wedge |\varphi| \bmod 2 = 0 \}$

Ejercicio 8. Sea $\Sigma = \{a, b, c\}$ un alfabeto, $L_1 = \{b, ab, ac\}$, $L_2 = \{b, b^2\}$, $L_3 = \{ba, bc\}$ y $L_4 = \{b^n : n \geq 0\}$ lenguajes, obtener los lenguajes $L_1 \cup L_2$, $L_1 \cap L_2$, $L_1 - L_2$, L_1L_2, L_3^2, L_3L_4, L_1^R y L_3^*.

$L_1 \cup L_2 = \{ b, ab, ac, b, b^2 \} = \{ b, ab, ac, bb \}$

$L_1 \cap L_2 = \{ b \}$

$L_1 - L_2 = \{ ab, ac \}$

$L_1 . L_2 = \{ bb, bbb, abb, abbb, acb, acbb \}$

$L_3^2 = \{ baba, babc, bcba, bcbc \}$

$L_3 L_4 = \{ ba.b^n, bc.b^n ; n \geq 0 \}$

$L_1^R = \{ b^R, ab^R, ac^R \} = \{ ac^R, ab^R, b^R \} = \{ ca, ba, b \}$

$L_3^* = \{ \varepsilon, \{ba, bc\}^* \}$ No estoy seguro.

Ejercicio 9. Sean L_1, L_2, L_3 lenguajes cualesquiera, probar:

a. Asociatividad de la concatenación: $L_1(L_2L_3) = (L_1L_2)L_3$.

b. Distributividad de la concatenación por izquierda con respecto a la unión: $L_1(L_2 \cup L_3) = L_1(L_2 \cup L_1L_3)$.

c. La concatenación no es distributiva por izquierda con respecto a la intersección.

a) Probar $L_1(L_2L_3)$

Si $w \in L_1(L_2L_3) \Rightarrow \exists\, x \in L_1, y \in L_2L_3$ tq $w = x.y$

podemos escribir $y = y_2 y_3$ tq $y_2 \in L_2$ $y_3 \in L_3$

$\therefore\ w = x\, y_2 y_3 = (x\, y_2)\, y_3$

por teorema $= (L_1 L_2) L_3$

b) Probar $L_1(L_2 \cup L_3) = L_1(L_2 \cup L_1L_3)$ → Entiendo que debería ser $(L_1L_2 \cup L_1L_3)$

Sea $w \in L_1(L_2 \cup L_3)$, sea $y \in (L_2 \cup L_3)$ sea $x \in L_1$

$\Rightarrow\ w = x.y$

y puede pertenecer a L_2 o a L_3

$\Rightarrow\ w = xy \in L_1L_2 \ \vee\ w = xy \in L_1L_3$

Pero esto es la definición de $(L_1L_2 \cup L_1L_3)$

c) Probar: La concatenación no es distributiva por izquierda con respecto a la intersección.

Basta con ver un contraejemplo.

$L_1 = \{a, b\}$
$L_2 = \{a\}$
$L_3 = \{b\}$

¡Dejar para chequear!

Ejercicio 10. Sea L un lenguaje cualquiera, probar:

a. $L^* = \bigcup_{i=0}^{\infty} L^i$ y $L^+ = \bigcup_{i=1}^{\infty} L^i$

b. $L^+ = LL^*$

c. $L^*L^* = L^*$

d. $(L^*)^* = L^*$

a)

Demostración de $L^* = \bigcup_{i=0}^{\infty} L^i$:

- L^* incluye todas las concatenaciones posibles de elementos de L, desde L^0 (la cadena vacía) hasta concatenaciones de cualquier longitud.
- La expresión $\bigcup_{i=0}^{\infty} L^i$ representa la unión de todas las potencias de L, desde $i = 0$ hasta el infinito.
- Por definición, esto es exactamente lo que contiene L^*: todas las posibles combinaciones de concatenaciones de L, incluyendo la cadena vacía.

Por lo tanto:

$$L^* = \bigcup_{i=0}^{\infty} L^i$$

Demostración de $L^+ = \bigcup_{i=1}^{\infty} L^i$:

- L^+ incluye todas las concatenaciones posibles de elementos de L, pero excluye L^0 (la cadena vacía).
- La expresión $\bigcup_{i=1}^{\infty} L^i$ representa la unión de todas las potencias de L, desde $i = 1$ en adelante, que es lo que incluye L^+ por definición.

Por lo tanto:

$$L^+ = \bigcup_{i=1}^{\infty} L^i$$

b) $L^+ = LL^*$

Demostración:

- Cualquier cadena en L^+ puede escribirse como una concatenación de al menos una cadena en L.
- Esto equivale a tomar una cadena de L y seguirla con cualquier cadena de L^* (incluyendo la cadena vacía).
- Por lo tanto, L^+ incluye exactamente las mismas cadenas que LL^*, y podemos concluir que:

$$L^+ = LL^*$$

L pisa ε

c) $L^*L^* = L^*$

- Concatenar dos elementos de L^* simplemente produce una cadena que aún pertenece a L^*.

Demostración:

- Si tomamos dos elementos arbitrarios de L^* y los concatenamos, el resultado sigue siendo una cadena que podría generarse en L^*, ya que L^* ya incluye todas las concatenaciones posibles.
- Esto implica que:

$$L^*L^* = L^*$$

d) $(L^*)^* = L^*$

- Dado que L^* ya es cerrado bajo concatenación, aplicar el operador de Kleene star nuevamente no añade nuevas cadenas.
- Por lo tanto:

$$(L^*)^* = L^*$$