Expresiones regulares

Clase 04

IIC 2223

Prof. Cristian Riveros

Recordatorio: Autómata finito no-determinista

Definición

Un autómata finito **no-determinista** (NFA) es una estructura:

$$\mathcal{A} \quad = \quad (Q, \Sigma, \Delta, I, F)$$

- Q es un conjunto finito de estados.
- Σ es el alfabeto de input.
- $F \subseteq Q$ es el conjunto de estados finales (o aceptación).

$$+$$

- $\Delta \subseteq Q \times \Sigma \times Q$ es la **relación de transición**.
- $I \subseteq Q$ es un **conjunto de estados iniciales**.

Rec: ¿cómo ejecuto un autómata no-determinista?

Sea:

- Un autómata finito no-determinista $\mathcal{A} = (Q, \Sigma, \Delta, I, F)$.
- El input $w = a_1 a_2 \dots a_n \in \Sigma^*$.

Una **ejecución** (o run) ρ de $\mathcal{A}$ sobre w es una secuencia:

$$\rho : p_0 \xrightarrow{a_1} p_1 \xrightarrow{a_2} \dots \xrightarrow{a_n} p_n$$

- $p_0 \in I$
- para todo $i \in \{0, \dots, n-1\}$, $(p_i, a_{i+1}, p_{i+1}) \in \Delta$.

Una ejecución ρ de $\mathcal{A}$ sobre w es de **aceptación** si:

$$p_n \in F.$$

Desde ahora hablaremos de **las ejecuciones** de $\mathcal{A}$ sobre w

Rec: Lenguaje aceptado por un autómata no-determinista

Sea un autómata $\mathcal{A} = (Q, \Sigma, \Delta, I, F)$ y $w \in \Sigma^*$.

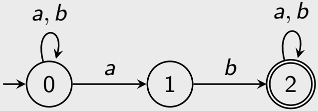
Definiciones

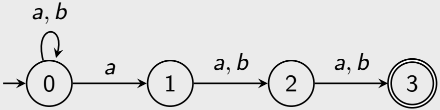
- $\mathcal{A}$ **acepta** w si **existe** una ejecución de $\mathcal{A}$ sobre w que es de aceptación.
- $\mathcal{A}$ **rechaza** w si **todas** las ejec. de $\mathcal{A}$ sobre w **NO** son de aceptación.
- El **lenguaje aceptado** por $\mathcal{A}$ se define como:

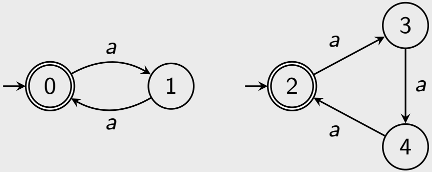
$$\mathcal{L}(\mathcal{A}) = \{w \in \Sigma^* \mid \mathcal{A} \text{ acepta } w\}$$

Rec: Lenguaje aceptado por un autómata no-determinista

¿qué lenguaje acepta cada autómata no-determinista?

1. 

2. 

3. 

Rec: ¿qué tan poderoso es el no-determinismo?

Teorema

Para todo autómata finito no-determinista $\mathcal{A}$, existe un autómata determinista $\mathcal{A}'$ tal que:

$$\mathcal{L}(\mathcal{A}) \;=\; \mathcal{L}(\mathcal{A}')$$

En otras palabras, DFA $\equiv$ NFA.

Ambos modelos computan lo mismo

Demostración

Para demostrar este resultado,
construiremos la "determinación" del autómata no-determinista $\mathcal{A}$.

Recordatorio: Determinización (subset construction)

Formalización

Para un autómata no-determinista $\mathcal{A} = (Q, \Sigma, \Delta, I, F)$,
definimos el autómata determinista (**determinización** de $\mathcal{A}$):

$$\mathcal{A}^{\text{det}} \quad = \quad (2^Q, \Sigma, \delta^{\text{det}}, q_0^{\text{det}}, F^{\text{det}})$$

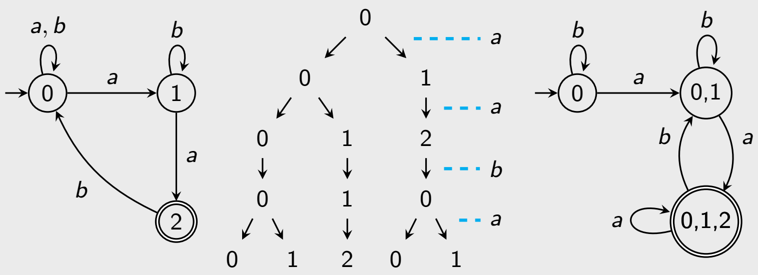
- $2^Q = \{S \mid S \subseteq Q\}$ es el conjunto potencia de Q.
- $q_0^{\text{det}} = I$.
- $\delta^{\text{det}} : 2^Q \times \Sigma \to 2^Q$ tal que:

$$\delta^{\text{det}}(S, a) = \{q \in Q \mid \exists p \in S.\ (p, a, q) \in \Delta\}$$

- $F^{\text{det}} = \{S \in 2^Q \mid S \cap F \neq \varnothing\}$.

Recordatorio: Determinización (subset construction)

Ejemplo



Recordatorio: Determinización (subset construction)

Proposición

Dado un autómata no-determinista $\mathcal{A} = (Q, \Sigma, \Delta, I, F)$ se tiene que:

$$\mathcal{L}(\mathcal{A}) \;=\; \mathcal{L}(\mathcal{A}^{\text{det}})$$

¿cómo demostramos que ambos autómatas definen el mismo lenguaje?

Recordatorio: Determinización (subset construction)

Demostración: $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{A}^{\text{det}})$

Sea $w = a_1 a_2 \ldots a_n \in \mathcal{L}(\mathcal{A})$.

Existe una ejecución ρ de $\mathcal{A}$ sobre w:

$$\rho : p_0 \xrightarrow{a_1} p_1 \xrightarrow{a_2} \ldots \xrightarrow{a_n} p_n$$

- $p_0 \in I$.
- $(p_i, a_{i+1}, p_{i+1}) \in \Delta \quad \forall\, i \in \{0, \ldots, n-1\}$.
- $p_n \in F$.

Como $\mathcal{A}^{\text{det}}$ es determinista, entonces existe una ejec. ρ' de $\mathcal{A}^{\text{det}}$ sobre w:

$$\rho' : S_0 \xrightarrow{a_1} S_1 \xrightarrow{a_2} \ldots \xrightarrow{a_n} S_n$$

- $S_0 = I$.
- $\delta^{\text{det}}(S_i, a_{i+1}) = S_{i+1} \quad \forall\, i \in \{0, 1, \ldots, n-1\}$.

¿qué debemos demostrar?

Recordatorio: Determinización (subset construction)

Demostración: $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{A}^{\text{det}})$

PD: $p_i \in S_i$ para todo $i \in \{0, 1, \ldots, n-1\}$.

Por **inducción** sobre i.

Caso base: $p_0 \in S_0$ ✓ (¿por qué?)

Inducción: Suponemos que $p_i \in S_i$ y demostramos para $i+1$.

Como sabemos que:

- $\delta^{\text{det}}(S_i, a_{i+1}) = S_{i+1} = \{q \in Q \mid \exists p \in S_i.\ (p, a, q) \in \Delta\}$ y
- $(p_i, a_{i+1}, p_{i+1}) \in \Delta$.

Entonces $p_{i+1} \in S_{i+1}$ (¿por qué?). ✓

Como $p_n \in S_n \overset{?}{\Rightarrow} S_n \cap F \neq \varnothing \overset{?}{\Rightarrow} S_n \in F^{\text{det}}$.

Por lo tanto, $w \in \mathcal{L}(\mathcal{A}^{\text{det}})$.

Determinización (subset construction)

Demostración: $\mathcal{L}(\mathcal{A}^{\text{det}}) \subseteq \mathcal{L}(\mathcal{A})$

Sea $w = a_1 a_2 \ldots a_n \in \mathcal{L}(\mathcal{A}^{\text{det}})$.

Existe una ejecución ρ de $\mathcal{A}^{\text{det}}$ sobre w:

$$\rho : S_0 \stackrel{a_1}{\rightarrow} S_1 \stackrel{a_2}{\rightarrow} \ldots \stackrel{a_n}{\rightarrow} S_n$$

- $S_0 = I$.
- $\delta^{\text{det}}(S_i, a_{i+1}) = S_{i+1} \quad \forall\, i \in \{0, 1, \ldots, n-1\}$.
- $S_n \in F^{\text{det}}$. $(S_n \cap F \neq \emptyset)$

¿cómo demostramos una **ejecución de aceptación** de $\mathcal{A}$ sobre w?

Determinización (subset construction)

Demostración: $\mathcal{L}(\mathcal{A}^{\text{det}}) \subseteq \mathcal{L}(\mathcal{A})$

PD: Para todo $i \leq n$ y para todo $p \in S_i$, existe una ejecución:

$$\rho : p_0 \xrightarrow{a_1} p_1 \xrightarrow{a_2} \dots \xrightarrow{a_i} p_i = p$$

1. $p_0 \in I$.
2. $(p_j, a_{j+1}, p_{j+1}) \in \Delta \quad \forall\, j \in \{0, \dots, i-1\}$.

Por **inducción** sobre i.

Caso base: Si $p \in S_0 = I$, entonces la ejec. $\rho : p$ cumple 1. y 2. ✓

Determinización (subset construction)

Demostración: $\mathcal{L}(\mathcal{A}^{\text{det}}) \subseteq \mathcal{L}(\mathcal{A})$

PD: Para todo $i \leq n$ y para todo $p \in S_i$, existe una ejecución:

$$\rho : p_0 \xrightarrow{a_1} p_1 \xrightarrow{a_2} \dots \xrightarrow{a_i} p_i = p$$

1. $p_0 \in I$.
2. $(p_j, a_{j+1}, p_{j+1}) \in \Delta \quad \forall\, j \in \{0, \dots, i-1\}$.

Inducción: Supongamos que se cumple para todo $p \in S_i$. Sea $q \in S_{i+1}$.

Como $\delta^{\text{det}}(S_i, a_{i+1}) = S_{i+1} = \{q \in Q \mid \exists p \in S_i.\ (p, a, q) \in \Delta\}$ y $q \in S_{i+1}$
entonces existe $p \in S_i$ tal que $(p, a_{i+1}, q) \in \Delta$.

Por **HI**, existe $\rho : p_0 \xrightarrow{a_1} p_1 \xrightarrow{a_2} \dots \xrightarrow{a_i} p_i = p$ que satisface 1. y 2.

Por lo tanto, $\rho' : p_0 \xrightarrow{a_1} p_1 \xrightarrow{a_2} \dots \xrightarrow{a_i} p_i \xrightarrow{a_{i+1}} q$ también satisface 1. y 2. ✓

Determinización (subset construction)

Demostración: $\mathcal{L}(\mathcal{A}^{\text{det}}) \subseteq \mathcal{L}(\mathcal{A})$

Por lo tanto: Para todo $i \leq n$ y para todo $p \in S_i$, existe una ejecución:

$$\rho : p_0 \overset{a_1}{\rightarrow} p_1 \overset{a_2}{\rightarrow} \ldots \overset{a_i}{\rightarrow} p_i = p$$

1. $p_0 \in I$.
2. $(p_j, a_{j+1}, p_{j+1}) \in \Delta \quad \forall\, j \in \{0, \ldots, i-1\}$.

Como $S_n \cap F \neq \varnothing$, (¿por qué?)
para $p \in S_n \cap F$ existe una ejecución de acept. de $\mathcal{A}$ sobre w.

Por lo tanto, $w \in \mathcal{L}(\mathcal{A})$. ∎

Outline

Expresiones regulares

Definición (Sintaxis)

R es una **expresión regular** sobre Σ si R es igual a:

1. a para alguna letra $a \in \Sigma$.
2. ϵ
3. $\varnothing$
4. $(R_1 + R_2)$ donde R_1 y R_2 son expresiones regulares.
5. $(R_1 \cdot R_2)$ donde R_1 y R_2 son expresiones regulares.
6. (R_1^*) donde R_1 es una expresión regular.

Denotaremos como ExpReg el conjunto de
todas las expresiones regulares sobre Σ

Expresiones regulares

Ejemplos de expresiones regulares

- $(a + b)$
- $((a \cdot b) \cdot c)$
- (a^*)
- $(b \cdot (a^*))$
- $((a + b)^*)$
- $((a \cdot ((b \cdot a)^*)) + \epsilon)$
- $((a \cdot ((b \cdot a)^*)) + \varnothing)$

Expresiones regulares

Para reducir la cantidad de paréntesis, se define el **orden de precedencia**:

1. estrella $(\cdot)^*$
2. concatenación $\cdot$
3. unión $+$

Ejemplos

Considere el alfabeto $\Sigma = \{a, b, c\}$.

- $a \cdot b + a^* \quad = \; ?$
- $(a + b) \cdot c + a \quad = \; ?$

Cada expresión regular define un lenguaje

Definición (Semántica)

Para una expresión regular R cualquiera,
se define el lenguaje $\mathcal{L}(R) \subseteq \Sigma^*$ **inductivamente** como:

1. $\mathcal{L}(a) = \{a\}$ para toda letra $a \in \Sigma$.
2. $\mathcal{L}(\epsilon) = \{\epsilon\}$.
3. $\mathcal{L}(\varnothing) = \varnothing$.
4. $\mathcal{L}(R_1 + R_2) = \mathcal{L}(R_1) \cup \mathcal{L}(R_2)$ donde R_1 y R_2 son expresiones regulares.

Cada expresión regular define un lenguaje

Definición (Semántica)

- Para dos lenguajes $L_1, L_2 \subseteq \Sigma^*$, se define el **producto** de L_1 y L_2:

$$L_1 \cdot L_2 \;=\; \left\{\, w_1 \cdot w_2 \;\middle|\; w_1 \in L_1 \;\wedge\; w_2 \in L_2 \,\right\}$$

¿cuál es el resultado del producto de estos lenguajes?

- $\{aa, bb\} \cdot \{cc, dd\}$
- $\{a, ab, \epsilon\} \cdot \{ba, a\}$
- $\{a\}^* \cdot \{b\}^*$
- $\{a\}^* \cdot \varnothing$

Cada expresión regular define un lenguaje

Definición (Semántica)

- Para dos lenguajes $L_1, L_2 \subseteq \Sigma^*$, se define el **producto** de L_1 y L_2:

$$L_1 \cdot L_2 \;=\; \left\{ w_1 \cdot w_2 \;\middle|\; w_1 \in L_1 \wedge w_2 \in L_2 \right\}$$

- Para un lenguaje $L \subseteq \Sigma^*$ se define la **potencia** a la $n \geq 0$:

$$L^n \;=\; \left\{ w_1 \cdot w_2 \cdot \ldots \cdot w_n \;\middle|\; \forall i \leq n.\; w_i \in L \right\}$$

¿cuál es el resultado de la potencia de estos lenguajes?

- $\{0, 1\}^{32}$
- $\{aa, \epsilon\}^{10}$
- $\left(\{a\}^*\right)^4$
- $\left(\{a\}^*\right)^0$

Cada expresión regular define un lenguaje

Definición (Semántica)

- Para dos lenguajes $L_1, L_2 \subseteq \Sigma^*$, se define el **producto** de L_1 y L_2:

$$L_1 \cdot L_2 \;=\; \left\{\, w_1 \cdot w_2 \;\middle|\; w_1 \in L_1 \,\land\, w_2 \in L_2 \,\right\}$$

- Para un lenguaje $L \subseteq \Sigma^*$ se define la **potencia** a la $n \geq 0$:

$$L^n \;=\; \left\{\, w_1 \cdot w_2 \cdot \ldots \cdot w_n \;\middle|\; \forall i \leq n.\; w_i \in L \,\right\}$$

- Para un lenguaje $L \subseteq \Sigma^*$ se define la **potencia** a la 0:

$$L^0 \;=\; \{\epsilon\}$$

Cada expresión regular define un lenguaje

Definición (Semántica)

Para una expresión regular R cualquiera,
se define el lenguaje $\mathcal{L}(R) \subseteq \Sigma^*$ **inductivamente** como:

1. $\mathcal{L}(a) = \{a\}$ para toda letra $a \in \Sigma$.
2. $\mathcal{L}(\epsilon) = \{\epsilon\}$.
3. $\mathcal{L}(\varnothing) = \varnothing$.
4. $\mathcal{L}(R_1 + R_2) = \mathcal{L}(R_1) \cup \mathcal{L}(R_2)$ donde R_1 y R_2 son expresiones regulares.
5. $\mathcal{L}(R_1 \cdot R_2) = \mathcal{L}(R_1) \cdot \mathcal{L}(R_2)$ donde R_1 y R_2 son expresiones regulares.
6. $\mathcal{L}(R_1^*) = \bigcup_{k=0}^{\infty} \mathcal{L}(R_1)^k$ donde R_1 es una expresión regular.

Cada expresión regular define un lenguaje

¿cuál es el lenguaje definido por las siguientes ExpReg?

- $\mathcal{L}(\ (a+b)^*\ ) \ = \ \{a,b\}^*$
- $\mathcal{L}\big(\ (a\cdot b)\cdot(b\cdot a)\ \big) \ = \ \{abba\}$
- $\mathcal{L}(\ a\cdot(b\cdot a)+b\cdot a+(a\cdot b)\cdot a\ ) \ = \ \{aba, ba\}$

Simplificación de expresiones regulares

Definición

- R_1 es **equivalente** a R_2 si, y solo si, $\mathcal{L}(R_1) = \mathcal{L}(R_2)$.
- Si R_1 es equivalente a R_2, escribiremos $R_1 \equiv R_2$.

Lema

Los operadores de unión $+$ y producto $\cdot$ son **asociativos**.

$$(R_1 + R_2) + R_3 \equiv R_1 + (R_2 + R_3)$$

$$(R_1 \cdot R_2) \cdot R_3 \equiv R_1 \cdot (R_2 \cdot R_3)$$

Demostración: ejercicio.

Más ejemplos de expresiones regulares

¿cuál es el lenguaje definido por las siguientes ExpReg?

- $\mathcal{L}(\ a^* \cdot b \cdot a^*\ ) =$ todas las palabras con una sola b.
- $\mathcal{L}(\ (a+b)^* \cdot b \cdot (a+b)^*\ ) =$ todas las palabras con una o más b's.

Abreviaciones útiles para expresiones regulares

Definición

Usamos las siguientes **abreviaciones** de expresiones regulares:

$$
\begin{aligned}
R^+ &\equiv R \cdot R^* \\
R^k &\equiv R \cdot \overset{k}{\cdots} \cdot R \\
R^? &\equiv R + \epsilon \\
\Sigma &\equiv a_1 + \ldots + a_n
\end{aligned}
$$

para $R \in \text{ExpReg}$ y $\Sigma = \{a_1, \ldots, a_n\}$.

Más ejemplos de expresiones regulares

¿cuál es el lenguaje definido por las siguientes ExpReg?

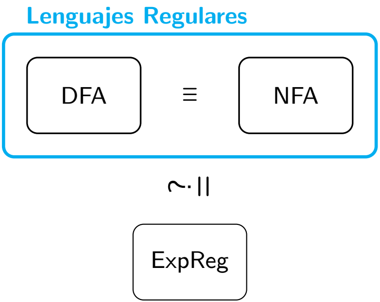
- $\mathcal{L}(\ \Sigma^* \cdot b \cdot \Sigma^*\ )$ = todas las palabras con una sola b.
- $\mathcal{L}\Big(\ b^* \cdot (a \cdot b^*)^5\ \Big)$ = todas las palabras con 5 a's.
- $\mathcal{L}\Big(\ a^* \cdot (b + c)^?\ \Big)$ = todas las palabras de a's y terminadas en b o c.
- $\mathcal{L}(\ (a \cdot b^+)^+\ )$ = todas las palabras que empiezan con a y donde cada a esta seguida de al menos una b.

Más ejemplos de expresiones regulares

Defina una ExpReg para los siguientes lenguajes

1. Todas las palabras sobre $\{a, b\}$ cuya ante-penúltima letra es una a-letra.
2. Todas las palabras sobre $\{a, b\}$ con una cantidad par de a-letras.
3. Todas las palabras sobre $\{a, b\}$ con a lo mas un par de a-letras consecutivas.

Mapa actual de nuestros modelos de computación



¿son las **ExpReg** equivalentes a los **lenguajes regulares**?