Expresiones regulares

Clase 05

IIC2223 / IIC2224

Prof. Cristian Riveros

Outline

Problemas algorítmicos (clase anterior)

Expresiones regulares

Algunos problemas algorítmicos sobre autómatas

1. Dado un autómata $\mathcal{A}$, ¿cómo determinar si $\mathcal{A}$ es **trivial**?

$$\mathcal{L}(\mathcal{A}) = \varnothing$$

2. Dado autómatas $\mathcal{A}$ y $\mathcal{A}'$, ¿cómo saber si $\mathcal{A}$ y $\mathcal{A}'$ calculan **lo mismo**?

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

3. Dado autómatas $\mathcal{A}$ y $\mathcal{A}'$, ¿cómo saber si $\mathcal{A}$ es más **restrictivo** que $\mathcal{A}'$?

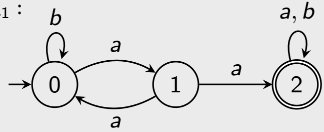
$$\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{A}')$$

$\mathcal{L}(\mathcal{A}) = \varnothing$

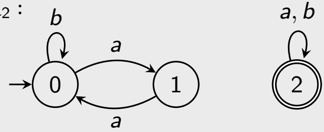
`Problema:`	EMPTYNESS
`Input:`	Un NFA $\mathcal{A} = (Q, \Sigma, \Delta, I, F)$
`Output:`	`TRUE` si, y solo si, $\mathcal{L}(\mathcal{A}) = \varnothing$.

¿cuál de los autómatas cumple que $\mathcal{L}(\mathcal{A}) = \varnothing$?

$\mathcal{A}_1$:



$\mathcal{A}_2$:



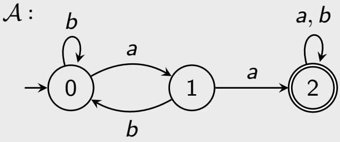
¿cómo podemos determinar si existe una palabra $w \in \Sigma^*$ tal que $w \in \mathcal{L}(\mathcal{A})$?

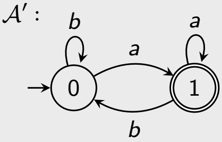
(ejercicio)

$\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{A}')$

`Problema:`	CONTAINMENT
`Input:`	Dos NFAs $\mathcal{A} = (Q, \Sigma, \Delta, I, F)$ y $\mathcal{A}' = (Q', \Sigma, \Delta', I', F')$
`Output:`	`TRUE` si, y solo si, $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{A}')$.

¿es verdad que $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{A}')$?





¿cómo determinar si $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{A}')$?

¿cómo determinar si $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{A}')$?

Dado dos autómatas $\mathcal{A}$ y $\mathcal{A}'$, tenemos que:

$$\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{A}') \quad \text{ssi} \quad \mathcal{L}(\mathcal{A}) \cap \mathcal{L}(\mathcal{A}')^C = \varnothing$$

Por lo tanto, los pasos a seguir son los siguientes:

1. Construir un autómata $\mathcal{B}$ tal que $\mathcal{L}(\mathcal{B}) = \mathcal{L}(\mathcal{A}')^C$.
2. Construir un autómata $\mathcal{C}$ tal que $\mathcal{L}(\mathcal{C}) = \mathcal{L}(\mathcal{A}) \cap \mathcal{L}(\mathcal{B})$.
3. Usar nuestro algoritmo de EMPTYNESS para verificar si $\mathcal{L}(\mathcal{C}) = \varnothing$.

¿cuál es el tiempo de este algoritmo?

Outline

Problemas algorítmicos (clase anterior)

Expresiones regulares

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 REGEX 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 $R \in \textsc{Regex}$ 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:

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

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

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

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

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

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 REGEX?

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

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**.

$$
\begin{aligned}
(R_1 + R_2) + R_3 \quad &\equiv \quad R_1 + (R_2 + R_3) \\
(R_1 \cdot R_2) \cdot R_3 \quad &\equiv \quad R_1 \cdot (R_2 \cdot R_3)
\end{aligned}
$$

Demostración: ejercicio.

Más ejemplos de expresiones regulares

¿cuál es el lenguaje definido por las siguientes REGEX?

- $\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 $R \in \textsc{Regex}$ y $\Sigma = \{a_1, \ldots, a_n\}$.

Más ejemplos de expresiones regulares

¿cuál es el lenguaje definido por las siguientes $\textsc{Regex}$?

- $\mathcal{L}(\ \Sigma^* \cdot b \cdot \Sigma^*\ ) \ = $ todas las palabras con una sola b.
- $\mathcal{L}\left(\ b^* \cdot (a \cdot b^*)^5\ \right) \ = $ todas las palabras con 5 a's.
- $\mathcal{L}\left(\ a^* \cdot (b + c)^?\ \right) \ = $ 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

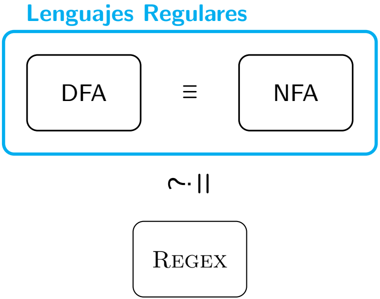
Defina una REGEX para los siguientes lenguajes

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**?

Cierre de clase

En esta clase vimos:

1. Problemas de EMPTYNESS y CONTAINMENT.
2. Sintaxis y semántica formal de expresiones regulares.
3. Especificación de lenguajes usando expresiones regulares.

Próxima clase: NFA con ϵ-transiciones