Espressioni regolari

- Un FA (NFA o DFA) è un metodo per costruire una macchina che riconosce linguaggi regolari.
- Le *espressioni regolari* una notazione algebrica per descrivere, in modo dichiarativo, i linguaggi regolari.
- Esempio: $\mathbf{01}^* + \mathbf{10}^*$
- Le espressioni regolari sono usate, ad esempio,
  - nella descrizione di particolari tipi di sequenze (pattern) nei testi
  - nei comandi UNIX (`grep`)
  - negli strumenti per l'analisi lessicale di UNIX (Lex (Lexical analyzer generator) e Flex (Fast Lex)).

Operazioni sui linguaggi

Le espressioni regolari usano le seguenti operazioni.

- **Unione:**
  $L \cup M = \{w : w \in L \text{ o } w \in M\}$
  Esempio: $\{0, 10\} \cup \{1, 10, 011\} = \{0, 10, 1, 10, 011\}$
- **Concatenazione:**
  $L.M = \{w : w = xy, x \in L, y \in M\}$
  Esempio:
  $\{0, 10\} \cup \{1, 10, 011\} = \{01, 010, 0011, 101, 1010, 10011\}$
- **Potenze:**
  $L^0 = \{\epsilon\},\ L^1 = L,\ L^{k+1} = L.L^k$
- **Chiusura di Kleene:**
  $$L^* = \bigcup_{i=0}^{\infty} L^i$$
  Esempio: $\{0, 10\}^* = \{\epsilon, 0, 10, 00, 010, 100, 1010, \ldots\}$

Definizione induttiva di espressioni regolari

Se E è un'espressione regolare, $L(E)$ denota il linguaggio che E definisce.

- **Base:**
  - $\boldsymbol{\epsilon}$ e $\boldsymbol{\emptyset}$ sono espressioni regolari. $L(\boldsymbol{\epsilon}) = \{\epsilon\}$ e $L(\boldsymbol{\emptyset}) = \emptyset$.
  - Se a è un simbolo in Σ, allora $\boldsymbol{a}$ è un'espressione regolare. $L(\boldsymbol{a}) = \{a\}$.
- **Induzione:**
  - Se E e F sono espressioni regolari, allora $E + F$ è un'espressione regolare. $L(E + F) = L(E) \cup L(F)$.
  - Se E e F sono espressioni regolari, allora $E.F$ è un'espressione regolare. $L(E.F) = L(E).L(F)$.
  - Se E è un'espressione regolare, allora (E) è un'espressione regolare. $L((E)) = L(E)$.
  - Se E è un'espressione regolare, allora E^* è un'espressione regolare. $L(E^*) = (L(E))^*$.

Ordine di precedenza per gli operatori

1. Chiusura (*)
2. Concatenazione (.)
3. Più $(+)$

Esempio: $\mathbf{01}^* + \mathbf{1}$ è raggruppato in $(\mathbf{0}(\mathbf{1})^*) + \mathbf{1}$

Esempi

- $L(\mathbf{ab}) = L(\mathbf{a}).L(\mathbf{b}) = \{ab\}$
- $L(\mathbf{ab} + \mathbf{b}) = L(\mathbf{ab}) \cup L(\mathbf{b}) = \{ab, b\}$
- $L(\mathbf{a}(\mathbf{ab} + \mathbf{b})) = L(\mathbf{a}).(L(\mathbf{ab}) \cup L(\mathbf{b})) = L(\mathbf{a}).L(\mathbf{ab}) \cup L(\mathbf{a}).L(\mathbf{b}) = \{aab, ab\}$
- $L(\mathbf{ab}^*) = (L(\mathbf{a})).(L(\mathbf{b}))^* = \{a, ab, abb, abbb, ...\}$
- $L(\mathbf{(ab)}^*) = (L(\mathbf{ab}))^* = \{\epsilon, ab, abab, ababab, ...\}$

Esempio

Espressione regolare per
$L = \{w \in \{0,1\}^* : \text{ 0 e 1 alternati in } w\}$

$$(\mathbf{01})^* + (\mathbf{10})^* + \mathbf{0}(\mathbf{10})^* + \mathbf{1}(\mathbf{01})^*$$

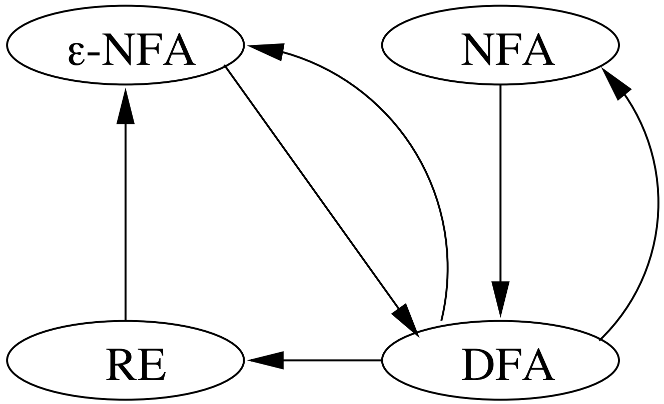
o, equivalentemente,

$$(\epsilon + \mathbf{1})(\mathbf{01})^*(\epsilon + \mathbf{0})$$

Notare che il linguaggio accetta anche le stringhe 0 e 1.

Equivalenza di FA e espressioni regolari

Abbiamo già mostrato che DFA, NFA, e ϵ-NFA sono tutti equivalenti.



Per mostrare che gli FA sono equivalenti alle espressioni regolari, mostreremo che

1. Per ogni DFA A possiamo trovare (costruire, in questo caso) un'espressione regolare R, tale che $L(R) = L(A)$.
2. Per ogni espressione regolare R esiste un ϵ-NFA A, tale che $L(A) = L(R)$.