Esempio

Dato un alfabeto $\Sigma = \{a, b\}$, l'insieme $L_3 = \{a^n b^n \mid n \geq 1\}$ è un linguaggio è composto da tutte le stringhe costruite dalla concatenazione di un certo numero di a, seguito dalla concatenazione dello stesso numero di b

$a^0 b^0 = \varepsilon \notin L_3$ $\quad a^2 b^2 = abab$ $\quad \rightarrow a^2 b^2 = aabb$

$a^3 b^3 = aaabbb$

Intersezione: $L_1 \cap L_2 = \{x \in \Sigma^* \mid x \in L_1 \wedge x \in L_2\}$

Unione: $L_1 \cup L_2 = \{x \in \Sigma^* \mid x \in L_1 \vee x \in L_2\}$

Complemento: $\overline{L_1} = \{x \in \Sigma^* \mid x \notin L_1\}$

Definizione

La potenza L^h di un linguaggi è definita come $L^h = L^1 \circ L^{h-1}$ $h \geq 1$ con la convenzione che $L^0 = \{\varepsilon\}$

$L \subseteq \Sigma$ con $h \geq 0$

Osservazione

L'insieme delle stringhe di lunghezza h sull'alfabeto Σ e lo indichiamo con Σ^h

Definizione $\rightarrow$ stella di Kleene

il linguaggio L^* viene definito come $L^* = \bigcup_{h=0}^{\infty} L^h$

$\rightarrow L^* = L^1 \cup L^2 \ldots \cup L^\infty$

L^* prende il nome di chiusura riflessiva del linguaggio L rispetto all'operazione di concatenazione

notiamo che:

$L, \varepsilon \in L^*$

$\downarrow$

perché $L^0 = \{\varepsilon\}$

$\Lambda^1 \neq \{\varepsilon\}$

$\Lambda^0 = \{\varepsilon\}$

Esempio

$L = \{ aa \} \Rightarrow L^* \{ a^{2n} \mid n \geq 0 \}$

Perché

$$L = \bigcup_{h=0}^{\infty} L^h = \bigcup_{h=0}^{\infty} \{aa\}^h = \bigcup_{h=0}^{\infty} \{a^2\}^h = \bigcup_{h=0}^{\infty} \{a^{2h}\}$$

Esempio

$L_1 = \{ BIS \} \quad L_2 = \{ NONNO \}$

ottenere $\underbrace{L_1^* L_2}$ $= \bigcup_{h=0}^{\infty} L_1^h \cdot L_2 = \{ \underset{h=0}{NONNO}, \underset{h=1}{BISNONNO}, \underset{h=2}{BISBISNONNO} \dots \}$

il simbolo di concatenazione può essere non scritto

Definizione

Si indica con L^+ la chiusura (non riflessiva) definita da

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

Data questa definizione possiamo dire che: $L^* : L^+ \cup \{ \varepsilon \}$

Osservazioni

- $\varepsilon \in L^*$ per definizione
- $\varepsilon \in L^+$ solo se $\varepsilon \in L$
- $\Delta^+ = \Delta$
- $L^* = L^+ \cup \{\varepsilon\}$ ma $L^+ \neq L^* \setminus \{\varepsilon\}$ perché se $\varepsilon \in L^+$ se si pone l'uguaglianza allora anche $L^* \setminus \{\varepsilon\}$ dovrebbe contenere ε ma per costruzione $\varepsilon \notin L^* \setminus \{\varepsilon\}$

- $\Sigma^* = \Sigma \cup \{\varepsilon\}$

Esempio:

Se $L = \{ab, bb\}$

$L^2 = L^1 \circ L^{2-1} = L^1 \circ L^1 = \{abab, bbbb, bbab, abbb\}$

$L^1 = L^{1-1} \circ L = \{\varepsilon\} L^1 = L^1$

FORMULA $L^k = L^{k-1} \circ L^1$

Provare con L^3

$L^3 = L^{3-1} \cdot L^1 =$

$L^* = \{\varepsilon\}^* = \{\varepsilon\}^0 \cup \{\varepsilon\}^1 \cup \{\varepsilon\}^2 \dots = \{\varepsilon\}$

caso finito di L^*

Linguaggi regolari

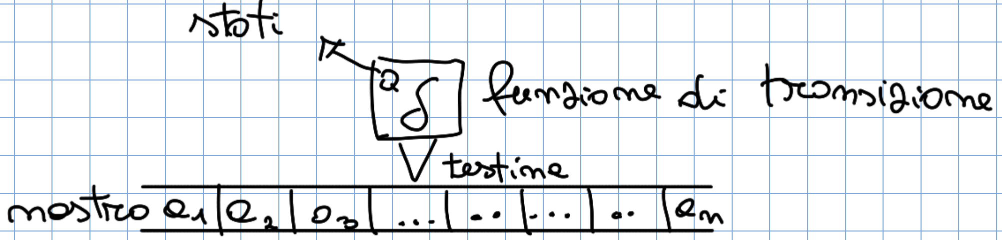
I metodi per rappresentare in modo finito i linguaggi:

- **Riconoscitivi**: metodi che prendono in input una stringa e ci dicono se essa appartiene al linguaggio oppure no. La stringa viene definita con un numero finito di elementi detti riconoscitori
- **Generativi**: metodi che partono da un simbolo iniziale e un insieme finito di regole e da lì si generano tutte le stringhe del linguaggio dette GRAMMATICHE

Riconoscitore: Strumento che riconosce insiemi di stringhe. Deve avere un nastro su cui prende in input una stringa e ci può scrivere sopra. Esso è diviso in celle con un simbolo dell'alfabeto Σ di partenza. Il simbolo di BLANK indica la mancanza di un elemento nella cella ϕ. il nastro è infinito in entrambe le direzioni, c'è la testina di lettura e/o scrittura che lavora su una cella del nastro alla volta e si muove avanti e indietro una cella alla volta. Il riconoscitore può essere definito tramite gli stati che cambiano nel tempo. Un riconoscitore ha una serie di regole che lo guidano nei movimenti e negli stati. tramite la funzione di transizione del riconoscitore. Ad ogni momento della computazione soltanto un numero finito di celle contiene simboli diversi dal simbolo BLANK. In questo caso parleremo di nastro infinito perché le celle vuote ci potrebbe servire durante la computazione e non sono occupate le celle del nastro perché esso deve

potere sostenere stringhe di qualsiasi lunghezza
Un passo è uno spostamento della testina, un eventuale cambio di stato, una lettura o un'eventuale scrittura
Una configurazione è una "fotografia" del riconoscitore e determina un'istante della computazione. Questa configurazione dipende da una successione di configurazioni che parte da una configurazione "particolare" ed eventualmente termina in un'altra

se la configurazione termina
- di accettazione se la stringa $\in$ al linguaggio
- di rifiuto se la stringa $\notin$ al linguaggio



Tipologie di dispositivo per il riconoscimento di linguaggi

Automi a stati finiti (ASF)

Definizione: un Automa a stati finiti deterministico (ASFD) è una quintupla $A = < \Sigma, Q, \delta, q_0, F >$ dove
Σ è l'alfabeto di input
$Q = \{ q_0, \dots q_m \}$ è un insieme finito di stati
$F \subseteq Q$ è un insieme di stati finali;
$q_0 \in Q$ è lo stato iniziale
$\delta : Q \cdot \Sigma \to Q$ è una funzione di transizione che ad ogni coppia di (stato, carattere in input) associa lo stato successivo.

$\delta: Q \cdot \Sigma \to Q$ tabella di transizione alle cui righe associo gli stati, alle colonne i caratteri in input e gli elementi rappresentano il risultato dell'applicazione della δ allo stato identificato dalla riga e dal carattere associato alla colonna della tabella

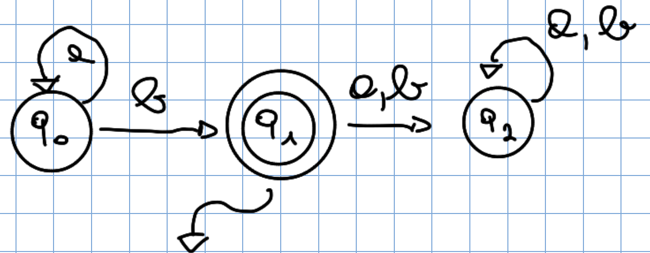
δ	a	b
q_0	q_0	q_1
q_1	q_2	q_2
q_2	q_2	q_2

$\delta: Q \cdot \Sigma \to Q$

$(q_0, a) \to q_0$

$(q_1, b) \to q_2$

Altra rappresentazione è il grafo di transizione in cui l'automa è rappresentato mediante un grafo orientato in cui i nodi rappresentano gli stati, mentre gli archi rappresentano le transizioni e sono etichettati con il carattere la cui lettura determina la transizione. Gli stati finali sono rappresentati da nodi con doppio cerchio mentre quello iniziale è individuato tramite una freccia



finito perché non va in loop

l'automa ad ogni passo legge il carattere successivo della stringa e applica la funzione di transizione terminata la lettura della stringa, essa viene accettata se lo stato attuale è uno stato finale (che appartiene all'insieme F) altrimenti viene rifiutata

N.B

Si nota che ASFD termina sempre le sue computazioni

Definizione (Pag 73)

Dato un automa a stati finiti $A = \langle \Sigma, Q, \delta, q_0, F \rangle$ una configurazione di A è una coppia $\langle q, x \rangle$ con $q \in Q$ e $x \in \Sigma^*$

Definizione: Una configurazione $\langle q, x \rangle$ con $q \in Q$ e $x \in \Sigma^*$ di A è detta:

- iniziale se $q = q_0$
- finale se $x = \varepsilon$
- accettante se $x = \varepsilon$ e $q \in F$

La funzione di transizione permette di definire la relazione di transizione che indichiamo con $\vdash$

$\Downarrow$

tra configurazioni

Questa relazione associa ad una configurazione, la configurazione successiva, nel seguente modo:

Definizione: Dato un ASFD $A = \langle \Sigma, Q, \delta, q_0, F \rangle$ e due configurazione (q, x) e (q', y) di A abbiamo che $(q, x) \vdash (q', y) \iff$ valgono le due condizioni:

1) $\exists e \in \Sigma$ tale che $x = ey$
2) $\delta(q, e) = q'$

si legge

$\overset{\vdash}{A}$ è associata tramite una funzione di transizione dell'ASFD A

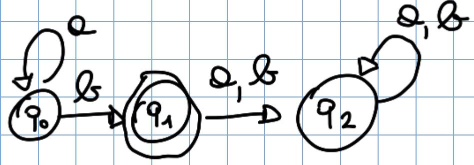
ASFD $A = < \Sigma, Q, \delta, q_0, F >$ $x \in \Sigma^*$ è accettata da $A \iff (q_0, x) \vdash (q, \varepsilon)$ con $q \in F$

e possiamo definire il linguaggio riconosciuto

$\mathcal{L}(A) = \{ x \in \Sigma^* \mid (q_0, x) \vdash^*_A (q, \varepsilon), q \in F$

Esempio

stringa aab è accettata dall'ASFD →



configurazione iniziale (q_0, aab)

l'automa raggiunge la configurazione di accettazione (q_1, ε)

$(q_0, aab) \vdash (q_0, ab) \vdash (q_0, b) \vdash (q_1, \varepsilon)$

leggiamo a si rimane in q_0

leggiamo a si rimane in q_0

leggiamo b e andiamo in q_1 → finiamo