Automi e Linguaggi Formali

a.a. 2016/2017

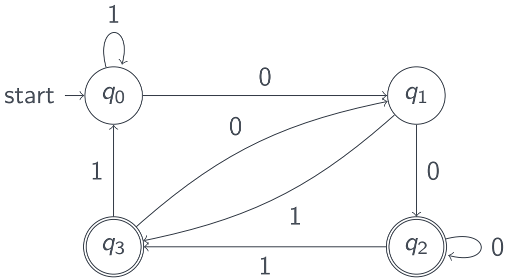
LT in Informatica
13 Marzo 2017

Università degli Studi di Padova

Compitini

- Due compitini, uno a metà del corso e uno alla fine.
- I compitini sostituiscono l'esame
  - devono essere entrambi sufficenti
- Per gli appelli di Giugno e Luglio:
  - compito diviso in due parti
  - si può recuperare un compitino insufficente
- Dagli appelli di Settembre in poi:
  - si deve fare l'esame completo

Costruire un Automa a Stati Finiti Deterministico (DFA) che riconosce il linguaggio di tutte le stringhe sull'alfabeto $\{0, 1\}$ tali che il penultimo simbolo è 0



Esercizio 2 del 3 Marzo

Descrivere il linguaggio riconosciuto dal seguente NFA

	0	1
$\rightarrow P$	$\{P, Q\}$	$\{Q\}$
Q	$\{R\}$	$\{R\}$
R	$\{S\}$	$\emptyset$
$*S$	$\{S\}$	$\{S\}$

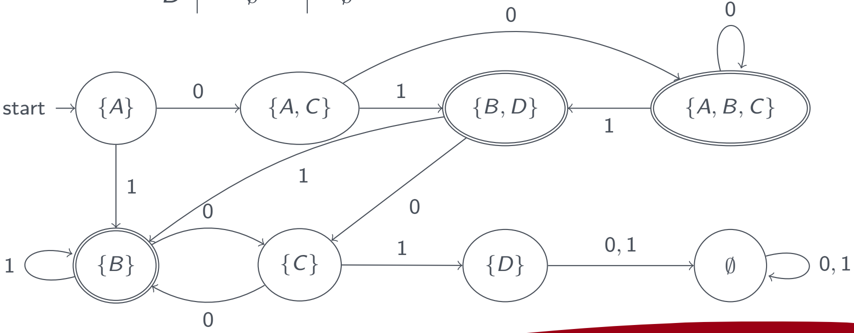
Tutte le stringhe che si possono scomporre in *uvw* dove

- u è una sequenza (anche vuota) di 0
- v è una stringa di lunghezza 3 che termina con 0
- w è una qualsiasi stringa in $\{0, 1\}^*$

Esercizio 3 del 3 Marzo

Convertire il seguente NFA in DFA:

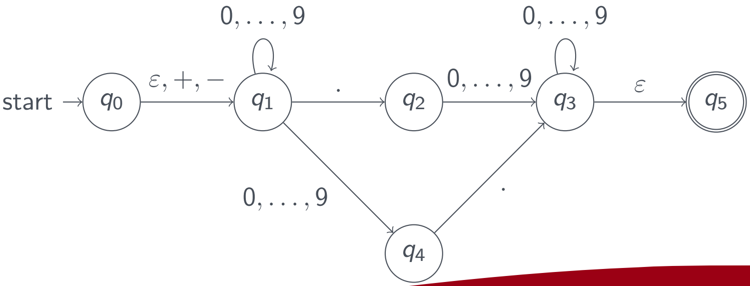
	0	1
$\rightarrow A$	$\{A, C\}$	$\{B\}$
$*B$	$\{C\}$	$\{B\}$
C	$\{B\}$	$\{D\}$
D	$\emptyset$	$\emptyset$



Esercizio: costruiamo un NFA che accetta numeri decimali:

1. Un segno $+$ o $-$, opzionale
2. Una stringa di cifre decimali $\{0, \ldots, 9\}$
3. un punto decimale .
4. un'altra stringa di cifre decimali

Una delle stringhe (2) e (4) può essere vuota, ma non entrambe



ε-NFA: definizione

Un Automa a Stati Finiti Non Deterministico con ε-transizioni (ε-NFA) è una quintupla

$$A = (Q, \Sigma, \delta, q_0, F)$$

dove:

- Q, Σ, q_0, F sono definiti come al solito
- δ è una funzione di transizione che prende in input:
  - uno stato in Q
  - un simbolo nell'alfabeto $\Sigma \cup \{\varepsilon\}$

  e restituisce un sottoinsieme di Q

Esempio di ε-NFA

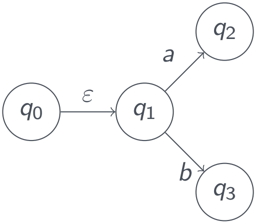
L'automa che riconosce le cifre decimali è definito come

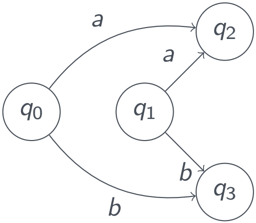
$$A = (\{q_0, q_1, \dots, q_5\}, \{+, -, ., 0, \dots, 9\}, \delta, q_0, \{q_5\})$$

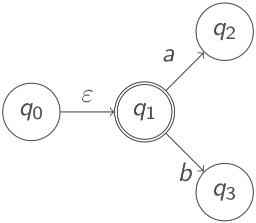
dove δ è definita dalla tabella di transizione

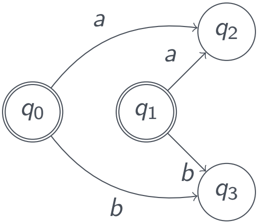
	ε	$+, -$	$.$	$0, \dots, 9$
$\to q_0$	$\{q_1\}$	$\{q_1\}$	$\emptyset$	$\emptyset$
q_1	$\emptyset$	$\emptyset$	$\{q_2\}$	$\{q_1, q_4\}$
q_2	$\emptyset$	$\emptyset$	$\emptyset$	$\{q_3\}$
q_3	$\{q_5\}$	$\emptyset$	$\emptyset$	$\{q_3\}$
q_4	$\emptyset$	$\emptyset$	$\{q_3\}$	$\emptyset$
$*q_5$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$

Eliminare le ε-transizioni









Epsilon chiusura: definizione

L'eliminazione delle ε-transizioni procede per ε-chiusura degli stati:

- tutti gli stati raggiungibili da q con una sequenza $\varepsilon\varepsilon\dots\varepsilon$

La definizione di $\mathrm{ECLOSE}(q)$ è per induzione:

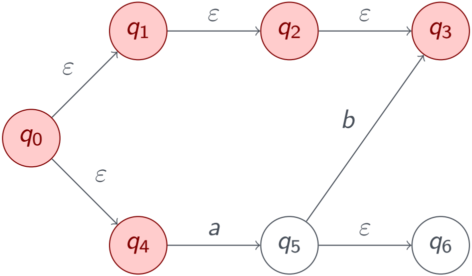
Caso base:

$$q \in \mathrm{ECLOSE}(q)$$

Caso induttivo:

$$\text{se } p \in \mathrm{ECLOSE}(q) \text{ e } r \in \delta(p, \varepsilon) \text{ allora } r \in \mathrm{ECLOSE}(q)$$

Epsilon chiusura: esempio



$$\mathrm{ECLOSE}(q_0) = \{q_0, q_1, q_4, q_2, q_3\}$$

Procedura di eliminazione delle ε-transizioni

Dato un ε-NFA

$$E = (Q, \Sigma, q_0, \delta_E, F_E)$$

costruiremo un NFA senza ε-transizioni

$$N = (Q, \Sigma, q_0, \delta_N, F_N)$$

che accetta lo stesso linguaggio

- E ed N hanno lo stesso insieme di stati e lo stesso stato iniziale
- Per ogni $q \subseteq Q$ e per ogni $a \in \Sigma$

$$\delta_N(q, a) = \bigcup_{p \in \mathrm{ECLOSE}(q)} \delta_E(p, a)$$

- $F_N = \{q \in Q : \mathrm{ECLOSE}(q) \cap F_E \neq \emptyset\}$
  Uno stato è finale se c'è almeno uno stato finale nella ε-chiusura

Esercizio

1. Costruiamo un ε-NFA che riconosce le parole costituite da
   - zero o più *a*
   - seguite da zero o più *b*
   - seguite da zero o più *c*
2. Eliminare le ε-transizioni dal risultato
3. Convertire l'NFA senza ε-transizioni in DFA