Automi e Linguaggi Formali

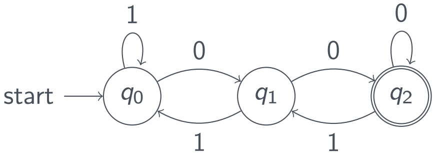
a.a. 2016/2017

LT in Informatica
20 Marzo 2017

Università degli Studi di Padova

Torniamo al secondo esercizio di ieri

- Costruiamo l'espressione regolare equivalente al seguente DFA:



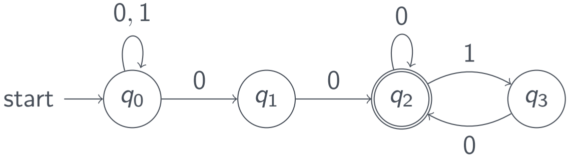
- Il risultato che abbiamo ottenuto con la procedura di eliminazione degli stati è:

$$((\mathbf{1}+\mathbf{01})+\mathbf{00}(\mathbf{0}+\mathbf{10})^*\mathbf{11})^*\,\mathbf{00}(\mathbf{0}+\mathbf{10})^*$$

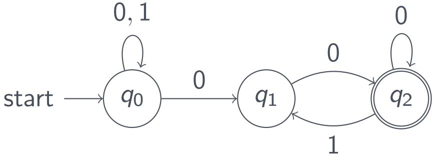
- **Domanda:** l'espressione sopra è equivalente a $(\mathbf{0}+\mathbf{1})^*\,\mathbf{00}(\mathbf{0}+\mathbf{10})^*$?

Risposta: SI

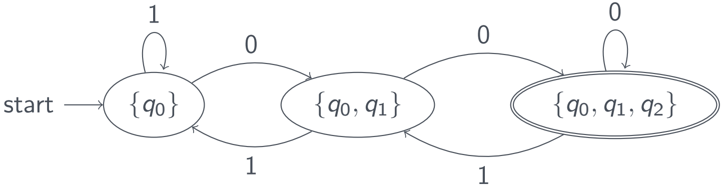
- Costruiamo un automa per $(\mathbf{0}+\mathbf{1})^* \, \mathbf{00}(\mathbf{0}+\mathbf{10})^*$:



- Lo semplifichiamo eliminando lo stato q_3:



- Alla fine, lo convertiamo in un DFA:



- E abbiamo ottenuto lo stesso automa da cui siamo partiti!

Esercizio

- Costruiamo un automa che riconosce il linguaggio:

$$L_{01} = \{a^n b^n : n \geq 0\}$$

http://etc.ch/nPjo

Dimostriamo che L_{01} non è regolare

- Supponiamo che $L_{01} = \{0^n 1^n : n \geq 0\}$ sia regolare
- Allora deve essere accettato da un DFA A con un certo numero k di stati
- Cosa succede quando A legge 0^k?
- Seguirà una qualche sequenza di transizioni:

$$\text{start} \longrightarrow \underbrace{p_0 \xrightarrow{0} p_1 \xrightarrow{0} p_2 \xrightarrow{0} \cdots \xrightarrow{0} p_k}_{k+1 \text{ stati}}$$

- Siccome ci sono $k+1$ stati nella sequenza, esiste uno stato che si ripete: esistono $i < j$ tali che $p_i = p_j$
- Chiamiamo q questo stato

Dimostriamo che L_{01} non è regolare

- Cosa succede quando l'automa A legge 1^i partendo da q?
- Se l'automa finisce la lettura in uno stato finale:
  - allora accetta, sbagliando, la parola 0^j1^i
- Se l'automa finisce la lettura in uno stato non finale:
  - allora rifiuta, sbagliando, la parola 0^i1^i
- In entrambi i casi abbiamo ingannato l'automa, quindi L_{01} non può essere regolare

Il Pumping Lemma per Linguaggi Regolari

Università degli Studi di Padova

Theorem (Pumping Lemma per Linguaggi Regolari)

Sia L un linguaggio regolare. Allora

- *esiste una lunghezza $n \geq 0$ tale che*
- *ogni parola $w \in L$ di lunghezza $|w| \geq n$*
- *può essere spezzata in $w = xyz$ tale che:*
  1. $y \neq \varepsilon$ *(il secondo pezzo è non vuoto)*
  2. $|xy| \leq n$ *(i primi due pezzi sono lunghi al max n)*
  3. $\forall k \geq 0,\ xy^k z \in L$ *(possiamo "pompare" y rimanendo in L)*

Pumping Lemma: dimostrazione

Dimostrazione:

- Supponiamo che L sia un linguaggio regolare
- Allora è riconosciuto da un DFA con, supponiamo, n stati
- Consideriamo una parola $w = a_1 a_2 \ldots a_m \in L$ di lunghezza $m \geq n$
- Consideriamo gli stati percorsi da A mentre legge w:

$$p_i = \hat{\delta}(p_0, a_1 a_2 \ldots a_i)$$

- Siccome in $p_0, p_1, \ldots, p_n$ ci sono $n + 1$ stati, ne esiste uno che si ripete:

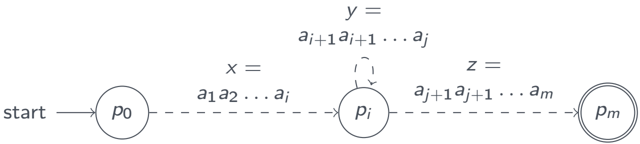
$$\text{esistono } i < j \text{ tali che } p_i = p_j \text{ e } j \leq n$$

Pumping Lemma: dimostrazione

- Possiamo spezzare w in tre parti $w = xyz$:
  1. $x = a_1 a_2 \dots a_i$
  2. $y = a_{i+1} a_{i+1} \dots a_j$
  3. $z = a_{j+1} a_{j+1} \dots a_m$
- che rispettano le condizioni del Lemma:
  - $y \neq \varepsilon$ perché $i < j$
  - $|xy| \leq n$ perché $j \leq n$

Pumping Lemma: dimostrazione

- Quindi, nel grafo delle transizioni di A:



- E di conseguenza anche $xy^k z$ viene riconosciuta dall'automa per ogni $k \geq 0$ $\square$

- Ogni linguaggio regolare soddisfa il Pumping Lemma.
- Per mostrare che un linguaggio non è regolare dobbiamo mostrare che falsifica il Pumping Lemma:
  - per ogni lunghezza $n \geq 0$
  - esiste una parola $w \in L$ di lunghezza $|w| \geq n$ tale che
  - per ogni split $w = xyz$ tale che:
    1. $y \neq \varepsilon$ (il secondo pezzo è non vuoto)
    2. $|xy| \leq n$ (i primi due pezzi sono lunghi al max n)
  - esiste un $k \geq 0$ tale che $xy^k z \in L$ (possiamo "pompare" y ed uscire da L)

- L'avversario sceglie la lunghezza k
- Noi scegliamo una parola w
- L'avversario spezza w in xyz
- Noi scegliamo i tale che $xy^iz \notin L$
- allora abbiamo vinto

1. Sia L_{ab} il linguaggio delle stringhe sull'alfabeto $\{a, b\}$ con un numero di b maggiore del numero di a. L_{ab} è regolare?

No, L_{ab} non è regolare:

- supponiamo per assurdo che lo sia
- sia n la lunghezza data dal Pumping Lemma
- consideriamo la parola $w = a^n b^{n+1}$
- prendiamo un qualsiasi split $w = xyz$ tale che $y \neq \varepsilon$ e $|xy| \leq n$:

$$w = \underbrace{aaa\ldots\ldots}_{x}\underbrace{\ldots a}_{y}\underbrace{abbb\ldots bb}_{z}$$

- per il Pumping lemma, anche $xy^2z \in L_{ab}$, ma contiene più a che b $\Rightarrow$ assurdo

2 Il linguaggio $L_{rev} = \{ww^R : w \in \{a, b\}^*\}$ è regolare?

No, L_{rev} non è regolare:

- supponiamo per assurdo che lo sia
- sia n la lunghezza data dal Pumping Lemma
- consideriamo la parola $w = a^n bba^n$
- prendiamo un qualsiasi split $w = xyz$ tale che $y \neq \varepsilon$ e $|xy| \leq n$:

$$w = \underbrace{aaa\ldots}_{x}\underbrace{\ldots a}_{y}\underbrace{abbaaa\ldots aaa}_{z}$$

- per il Pumping lemma, anche $xy^0z = xz \in L_{rev}$, ma non la posso spezzare in ww^R $\Rightarrow$ assurdo

3. Il linguaggio $L_{nk} = \{a^n b^k : n \text{ è dispari oppure } k \text{ è pari}\}$ è regolare?

Si, L_{nk} è regolare:

- è rappresentato dall'espressione regolare $\mathbf{a(aa)^*b^* + a^*(bb)^*}$
- e riconosciuto dall'automa