Sommario

- Pumping lemma per i regolari
- esempi di uso del teorema per dimostrare che un linguaggio non è regolare

PUMPING LEMMA: l'idea

Dato un DFA con n stati ci poniamo la seguente domanda:

ci sono condizioni sotto le quali si ha la certezza di avere un ciclo nel cammino determinato da una parola accettata?

Basta prendere parole di lunghezza maggiore o uguale a n:

$x_1 \; x_2 \; x_3 \; x_4 \; \ldots \; x_n \; \ldots \; x_{n+m}$

qui $m \geq 0$

$p_1 \; p_2 \; p_3 \; p_4 \; p_5 \; p_n \; p_{n+1} \; \ldots \; p_{n+m+1}$

Tra gli stati $p_1, \ldots, p_{n+1}$ ce n'è uno che si ripete.

Il principio della piccionaia!

PUMPING LEMMA

Bar-Hillel, Perles, Shamir (1961)

Primo enunciato: Se L è un linguaggio accettato da un DFA con n stati, allora tutte le parole w in L di lunghezza maggiore o uguale a n si possono ottenere come concatenazione di tre sottoparole, x,y e z tali che

1. **$w = xyz$ e $|y|>0$**
3. **le parole $w_i = xy^iz$, per ogni $i \geq 0$ sono in L**

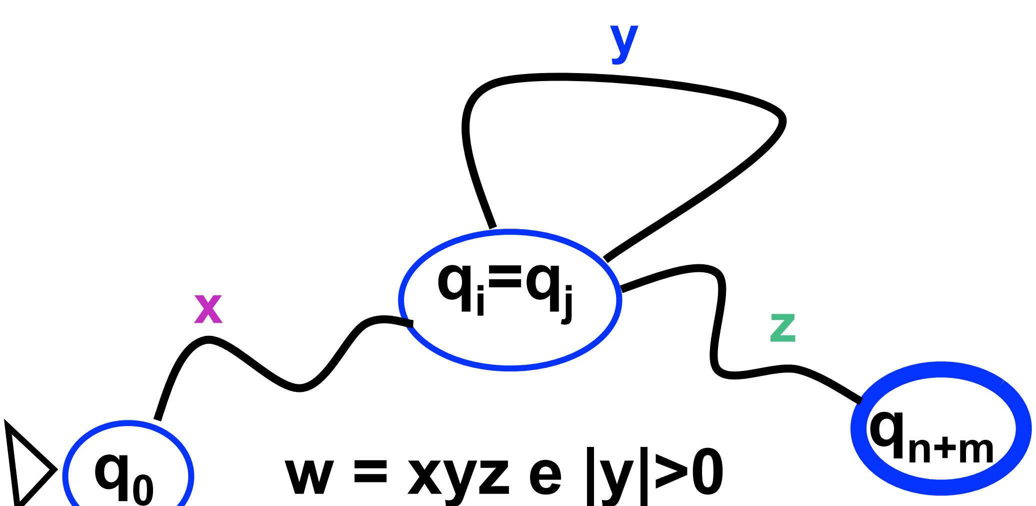
PUMPING LEMMA

Se w è di lunghezza maggiore o uguale a n:

$w = w_1 w_2 \ \dots \ w_n \ \dots \ x_{n+m}$ allora il cammino sul grafo di transizione è del tipo $q_0\ q_1\ \dots\ q_n\ q_{n+1} \dots q_{n+m}$, per un certo m≥0 e quindi c'è uno stato che si ripete.

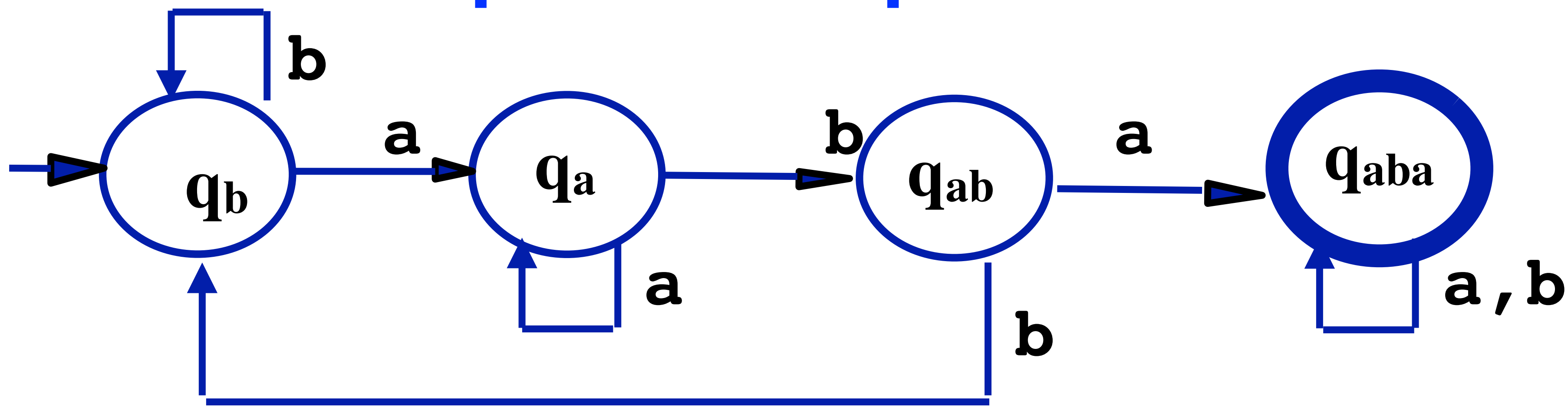
Sia $q_i = q_j$ allora si presenta una situazione come quella raffigurata, dove

$x=w_1 w_2 \ \dots \ w_i$,$y=w_{i+1} w_{i+2} \ \dots \ w_i$ e $z=w_{i+1} w_{i+2} \ \dots \ w_{n+m}$



Poiché il ciclo si può percorrere un numero qualsiasi di volte si ottiene un'infinità di parole accettate $w_i = xy^iz$, per ogni i≥0.

esempi di scomposizioni



Se prendiamo

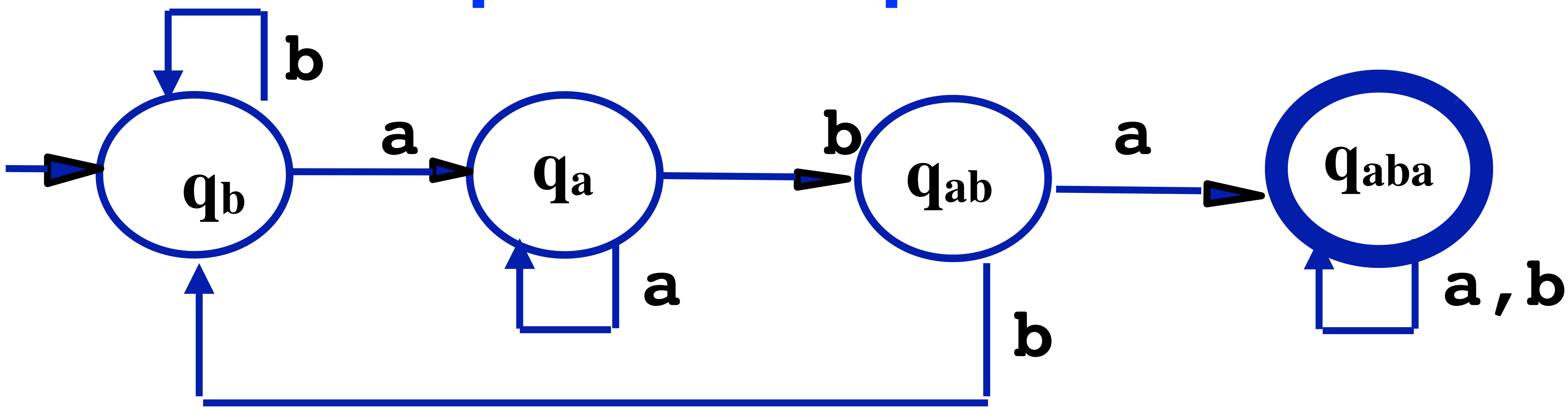
w = a b b a b a

$p_1 p_2 p_3 p_4 p_5 p_6 p_7 = q_b q_a q_{ab} q_b q_a q_{ab} q_{aba}$

$p_1 = p_4$ allora si ha la scomposizione

x = ε, y =abb e z = aba, e le parole $(abb)^i aba$ sono accettate, per ogni i≥0

esempi di scomposizioni



Se prendiamo

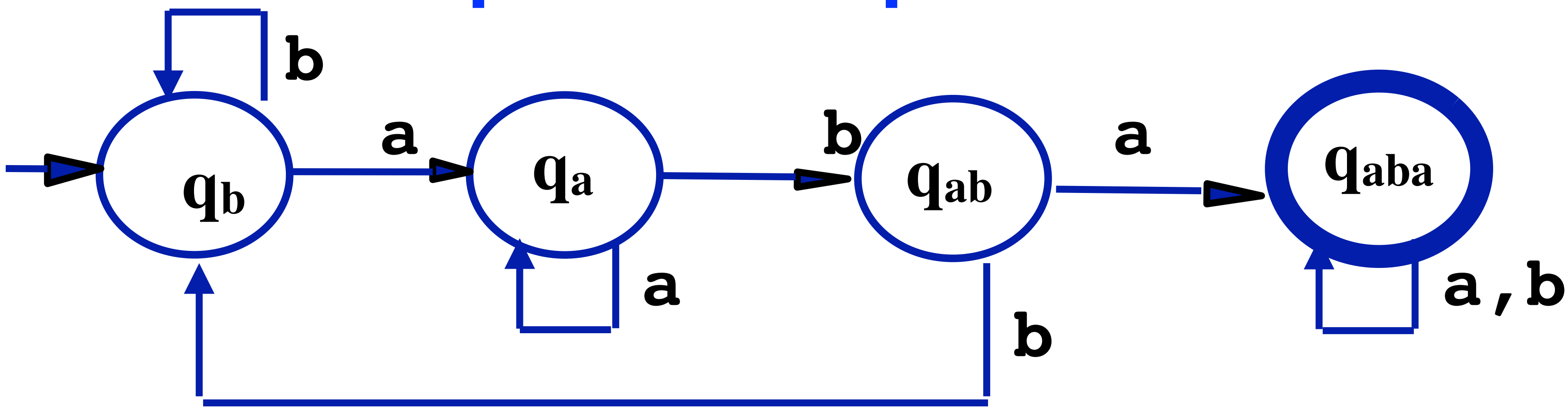
w = a b b a b a

$p_1p_2p_3p_4p_5\,p_6p_7 = q_bq_aq_{ab}q_bq_aq_{ab}q_{aba}$

$p_2 = p_5$ allora si ha la scomposizione

x = a, y = bba e z = ba e le parole $a(bba)^iba$ sono accettate, per ogni i≥0

esempi di scomposizioni



Se prendiamo

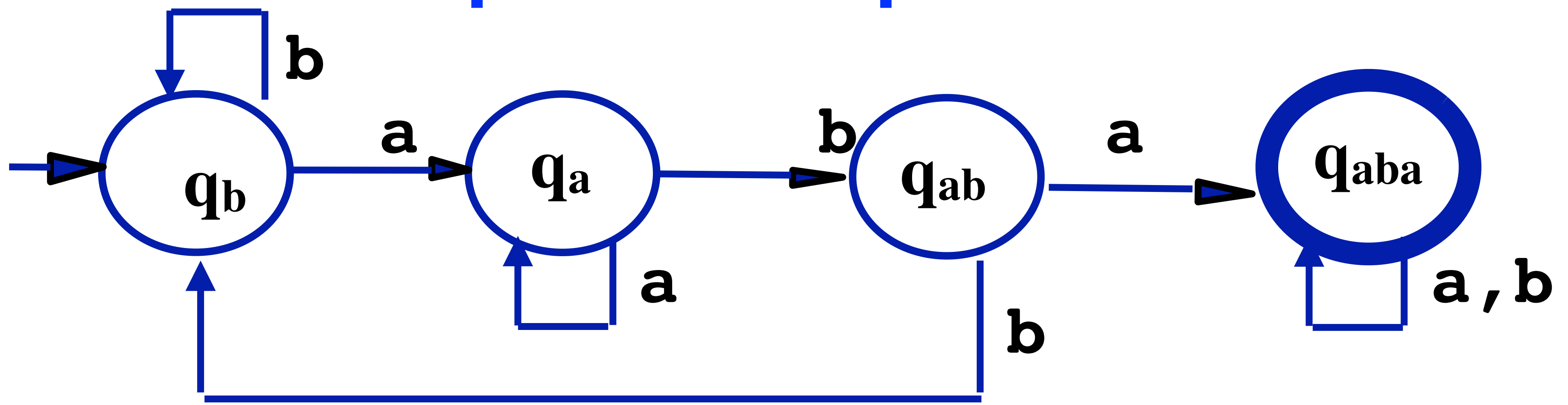
w = a b b a b   a

$p_1p_2p_3p_4p_5\,p_6p_7 = q_b q_a q_{ab} q_b q_a q_{ab} q_{aba}$

$p_2 = p_5$ allora si ha la scomposizione

x = a, y = bba e z = ba e le parole a(bba)^{i}ba sono accettate, per ogni i≥0

esempi di scomposizioni



Se prendiamo

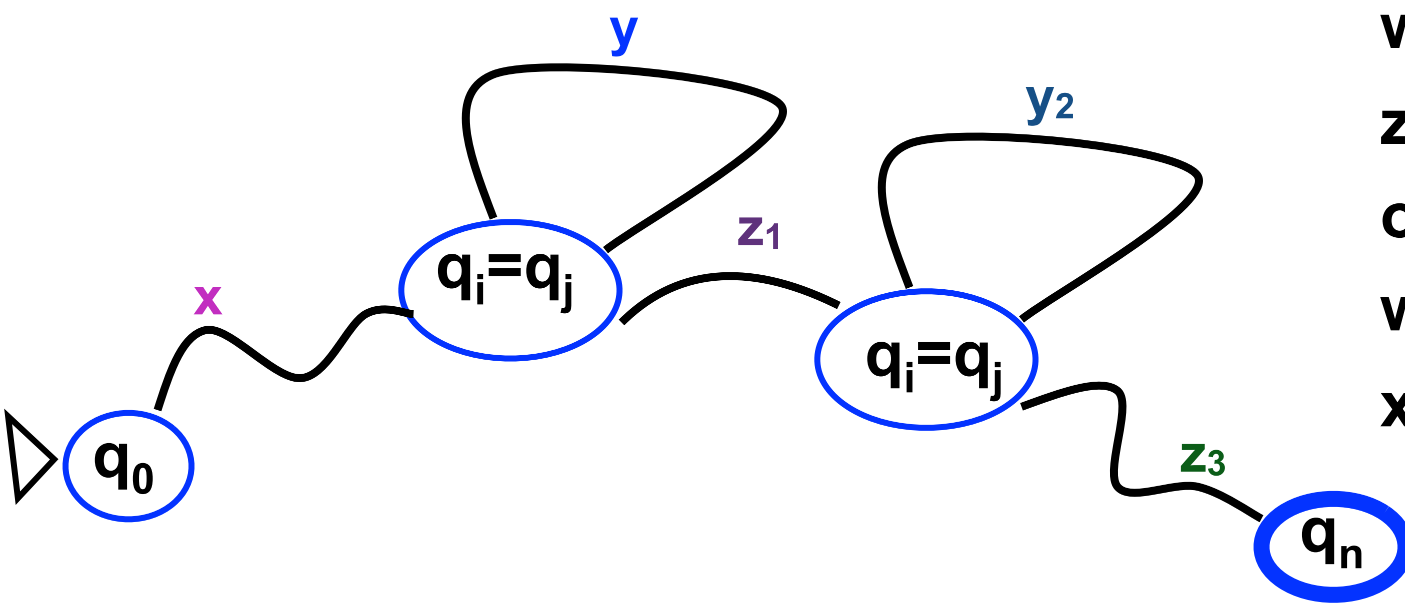
w = a b b a b   a

$p_1p_2p_3p_4p_5\,p_6p_7 = q_bq_aq_{ab}q_bq_aq_{ab}q_{aba}$

$p_3 = p_6$ **allora si ha la scomposizione**

x = ab, y = bab e z = a e le parole ab(bab)^{i}a sono accettate, per ogni i≥0

Enunciato più stringente



$w = xyz$ con

$z = z_1 y_2 z_3$

oppure

$w = x_1 y_2 z_3$ con

$x_1 = xyz_1$

Poiché basta una sola scomposizione, posso prendere il primo ciclo che si incontra sul cammino dallo stato iniziale.

Quanto al più può essere lungo il prefisso di una parola di lunghezza maggiore o uguale a n, perché contenga un ciclo? Ancora n.

Quindi possiamo imporre che $|xy| \leq n$.

PUMPING LEMMA

Enunciato definitivo: Se L è un linguaggio regolare, allora <u>esiste</u> un n (il numero degli stati di un DFA per L) tale che <u>tutte le</u> parole w in L di lunghezza maggiore o uguale a n si possono ottenere come concatenazione di tre sottoparole, x,y e z, w = xyz, e

1. $|y|>0$

é l'etichetta di un ciclo che almeno comprende due stati

2. $|xy| \leq n$

si assicura scegliendo il primo ciclo!

3. le parole $w_i = xy^iz$, per ogni $i \geq 0$ sono in L

PUMPING LEMMA

Enunciato definitivo: Se L è un linguaggio regolare, allora <u>esiste</u> un n (il numero degli stati di un DFA per L) tale che <u>tutte le</u> parole w in L di lunghezza maggiore o uguale a n si possono ottenere come concatenazione di tre sottoparole, x,y e z, w = xyz, e

1. $|y|>0$ 2. $|xy|\leq n$

3. le parole $w_i = xy^iz$, per ogni $i\geq 0$ sono in L

Chiamiamo queste tre condizioni le condizioni del pumping lemma.

Uso Pumping Lemma

Si usa per provare che un linguaggio NON è regolare.

Sia L un linguaggio che si sospetta non regolare.

Si deve far vedere che non esiste un DFA che lo accetta.

Quindi che comunque si prende un numero n (il numero degli stati di un DFA la cui esistenza vogliamo negare) si trova una parola di lunghezza maggiore di n, per la quale però non si è in grado di determinare quel ciclo su un cammino di computazione a uno stato finale a uno di accettazione che consente di fattorizzare la parola in modo da rispettare la condizione del pumping lemma.

Uso Pumping Lemma

Si tratta quindi di dimostrare che per ogni n siamo in grado di individuare una parola w in L di lunghezza almeno n tale che per ogni sua scomposizione nella concatenazione di tre sottoparole x,y,z con $|y|>0$, e $|xy|\leq n$ non siamo in grado di concludere che $w_i = xy^iz$ è in L per ogni $i\geq 0$.

Questo vuol dire che siamo in grado di individuare almeno un indice j tale che $w_j = xy^jz$ non è in L.

Esempio

L = $\{a^n b^n \mid n \geq 0\}$ non è regolare.

Per ogni n consideriamo le parole w = $a^n b^n$ (dato il linguaggio non c'è molta scelta!). Consideriamo le possibili scomposizioni w = xyz. Poiché $|xy| \leq n$, sia x che y sono di sole a quindi

1. y = a^m, per m>0, allora $w_i = x(a^m)^i z$ ha meno a di b se i=0.

E abbiamo considerato tutte le scomposizioni che rispettano il vincolo $|xy| \leq n$

Esempio

Se volessimo dimostrare che $L = \{a^n b^n \mid n \geq 0\}$ non è regolare applicando il pumping lemma ma ignorando il vincolo 2, dovremmo considerare per ogni n le parole $w = a^n b^n$ e tutte le possibili scomposizioni $w = xyz$:

1. $y = a^m$, per $m>0$, allora $w_i=x(a^m)^i z$ ha meno a di b se $i=0$.

2. $y = b^m$, analogo

3. $y = a^m b^p$, con $m,p \geq 1$, allora $y = x(a^m b^p)^i z$ contiene una a che segue una b per ogni $i>0$.

Molto lavoro in più anche in questo semplice caso!

STRATEGIA DI PROVA

Sia L un linguaggio che si sospetta non regolare:

- **si prende un intero positivo k,**

- **si sceglie (questa è la parte creativa) z in L tale che $|z| \geq k$**

- **si mostra che per <u>ogni</u> fattorizzazione $z = uvw$ con $|v| \geq 1$ e $|uv| \leq k$, <u>esiste</u> un $i \geq 0$ tale che $uv^iw \notin L$.**

Esempio

$L = \{a^n b^m \mid n>m \geq 0\}$ non è regolare.

Per ogni n consideriamo le parole $w = a^{n+1}b^n$. Consideriamo le possibili scomposizioni $w = xyz$, con $|xy| \leq n$. Allora sia x che y sono formate di sole a, se prendiamo i=0, consideriamo la parola xz nella quale il numero delle a è minore o uguale a quello delle b (uguale se $|y|=1$)

Formalmente $x = a^r$, per $r \geq 0$ e $y = a^s$, per $s>0$ e $z= a^t b^n$

con $r+s+t = n+1$ e $t \geq 1$

Poi basta osservare che $xy^0 = a^r$, e

allora $w_0=xz = a^r a^t b^n$,ma $r+t \leq n$

perché $s \geq 1$ e $r+s+t = n+1$, quindi w_0 non è in L.

Servono tutte le scomposizioni

L = {ww^{rev} | w in {0,1}* e w^{rev} è w letta da destra verso sinistra} non è regolare.

Per ogni n se prendessimo w = a^{2n}, che è in L, non riusciremmo a dimostrare la non regolarità infatti comunque prendo x,y e z tali che w = xyz, sono sempre di sole a, quindi non è vero che per ogni scomposizione trovo un valore di i≥0 tale che w = xy^{i}z non è in L.

Infatti basta che |y| sia pari e non ho alcun i che sia tale che xz non sia in L.

Mentre è chiaro che se si sceglie y di lunghezza dispari, si può prendere i=0

Esercizi

Si spieghi dove fallisce la prova di non regolarità, basata sul dumping lemma, per il linguaggio L = {00,11}*.

Per ogni n, per ogni parola w di lunghezza maggiore o uguale a n e per ogni scomposizione di w=xyz, con |y| di lunghezza pari non si trova un indice i per cui xy^iz non è in L.
Quindi non è vero che per ogni n esiste una parola w in L di lunghezza maggiore di n tale che per ogni scomposizione w=xyz si trova un indice i ≥0 tale che xy^iz non è in L.

Esercizi

Si dimostri, utilizzando il pumping lemma che i seguenti linguaggi non sono regolari.

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

$$L = \{a^p \mid p \text{ è un numero primo}\}$$

$$L = \{w\#x \mid w,x \text{ in } \{0,1\}^* \text{ e } x \text{ è una sottostringa di } w\}$$

$$L = \{ww^R \mid w \text{ in } \{0,1\}^* \text{ e } w^R \text{ è } w \text{ rovesciata}\}$$