Tutoraggio di Linguaggi di Programmazione

C.d.L. in Informatica (3 anni)
A.A. 2011/2012

tutor: Annalina Caputo
{acaputo@di.uniba.it}

Alcuni esercizi svolti sul Pumping Lemma per Linguaggi Regolari

1. Dimostrare formalmente che il seguente linguaggio:

$$L = \{a^i b^j c^k : k = i + j, \quad i, j, k \geq 0\}$$

non è lineare destro.

Soluzione.
Per assurdo, supponiamo che L sia lineare destro.
Per il teorema di Kleene si ha che $\mathcal{L}_3 \equiv \mathcal{L}_{FSL} \equiv \mathcal{L}_{REG}$, quindi se L è lineare è anche regolare ed anche a stati finiti, dunque

$$\exists M = (Q, \delta, q_0, F) \text{ di alfabeto } X = \{a, b, c\} : \quad L = T(M).$$

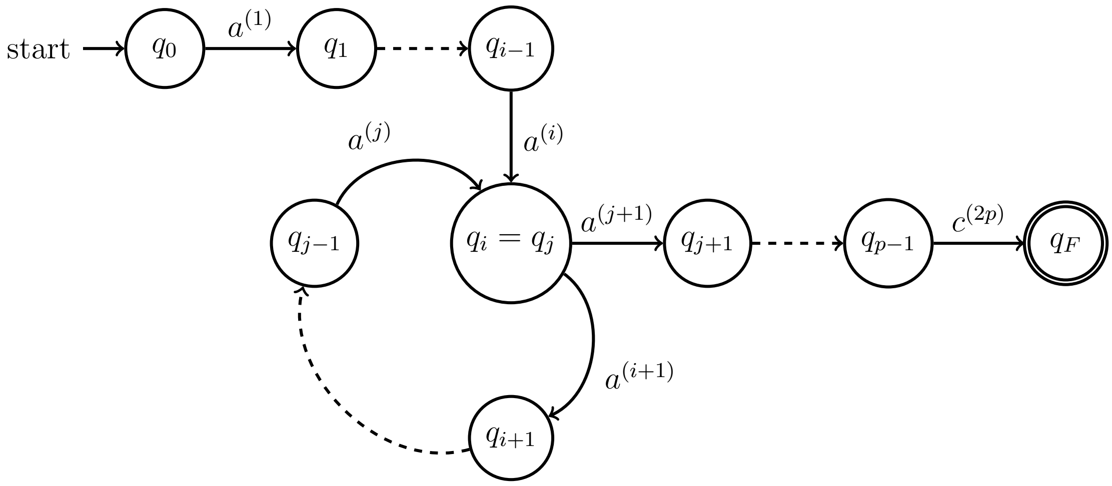
Indichiamo con p il numero di stati di M: $p = |Q|$.
Per il pumping lemma sui linguaggi regolari abbiamo che $\forall z \in L, |z| \geq p$

$$z = uvw \tag{1}$$
$$|uv| \leq p \tag{2}$$
$$v \neq \lambda \tag{3}$$
$$\forall k \in \mathbf{N} \quad uv^k w \in T(M) \tag{4}$$

Consideriamo la parola di L $z = a^p b^p c^{2p}$, si ha che $|z| = 4p > p$ ed inoltre z deve essere accettata da M. Per riconoscere/contare le prime p a, l'automa ha bisogno di transitare in $p + 1$ stati:

$$\left.\begin{array}{r} q_0 \xrightarrow{a} q_1 \\ q_1 \xrightarrow{a} q_2 \\ \dots \\ q_{p-1} \xrightarrow{a} q_p \end{array}\right\} p \text{ a}$$

Poiché M ha solo p stati, si deve verificare un ciclo:



Quindi z si può scrivere come $z = uvw = \underbrace{a^i}_{\text{u}}\underbrace{a^{j-i}}_{\text{v}}\underbrace{a^{p-j}b^p c^{2p}}_{\text{w}}$. Dalla 2) e dalla 3) abbiamo che:

$$|a^i a^{j-i}| \le p \tag{5}$$
$$a^{j-i} \neq \lambda, \quad 0 < j - i \le p \tag{6}$$

Poniamo $k = 0$, per il lemma abbiamo che la parola $uv^0w \in T(M)$

$$t = uv^0w = a^{p-(j-i)}b^p c^{2p} \in T(M)$$

ma $uv^0w \notin L$ poichè $\#_t(c) \neq \#_t(a) + \#_t(b)$. Contraddizione.
Quindi L non è regolare in quanto non esiste M tale che T(M)=L, e quindi L non lineare destro.

2. Stabilire se il linguaggio:
$$L = \{b^k a^n c^n : n \ge k > 0\}$$
è Lineare Destro giustificando la risposta.

Soluzione.
Per assurdo, supponiamo che L sia lineare destro.
Per il teorema di Kleene si ha che $\mathcal{L}_3 \equiv \mathcal{L}_{FSL} \equiv \mathcal{L}_{REG}$, quindi se L è lineare è anche regolare ed anche a stati finiti, dunque

$$\exists M = (Q, \delta, q_0, F) \text{ di alfabeto}: \quad X = \{a, b, c\} \quad L = T(M).$$

Indichiamo con p il numero di stati di M: $p = |Q|$. Per il pumping lemma sui linguaggi regolari abbiamo che, $\forall z \in L, |z| \ge p$

$$z = uvw \tag{7}$$
$$|uv| \le p \tag{8}$$
$$v \neq \lambda \tag{9}$$
$$\forall k \in \mathbf{N} \quad uv^k w \in T(M) \tag{10}$$

Consideriamo la parola di L $z = b^p a^p c^p$, si ha che $|z| = 3p > p$, ed inoltre z deve essere accettata da M.
Per il pumping lemma z si può scrivere come $z = uvw$, ma Q è composto da p stati differenti e per riconoscere le prime p b servono $p + 1$ stati, quindi due stati di Q devono coincidere.
Consideriamo gli stati $q_i = q_j$ per cui $i < j$.

Si avrà:
$u = b^i$
$v = b^{j-i}$
$w = b^{p-j}a^p c^p$
Pompando la stringa v k volte si otterrà la seguente stringa:

$$t = uv^k w = b^{p+k(j-i)}a^p c^p \in T(M)$$

Ma dalla 8) e dalla 9) si ha che la quantità $k(j-i) > 0$, quindi dovremo aggiugere *almeno* una b, risulterà quindi che $\#_t(b) > \#_t(a)$
La parola così ottenuta non appartiene al linguaggio, e quindi L non è lineare destro.