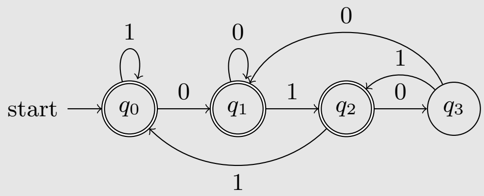
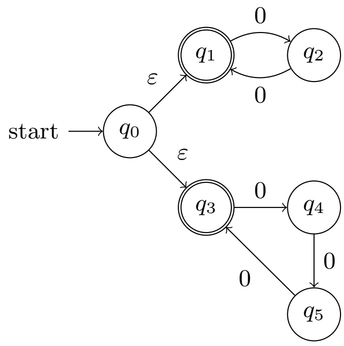
Automi e Linguaggi Formali
Esame del 5 Luglio 2018 – Parte I

Tempo a disposizione: 1 h 30 min

1. Considerate il linguaggio di tutte le parole sull'alfabeto $\{0, 1\}$ che *non* terminano con 010 (incluse ε e le parole di lunghezza < 3). Definire un'espressione regolare oppure un automa a stati finiti che riconoscano questo linguaggio.



2. Considerate il seguente ε-NFA che riconosce stringhe sull'alfabeto $\Sigma = \{0, 1\}$:

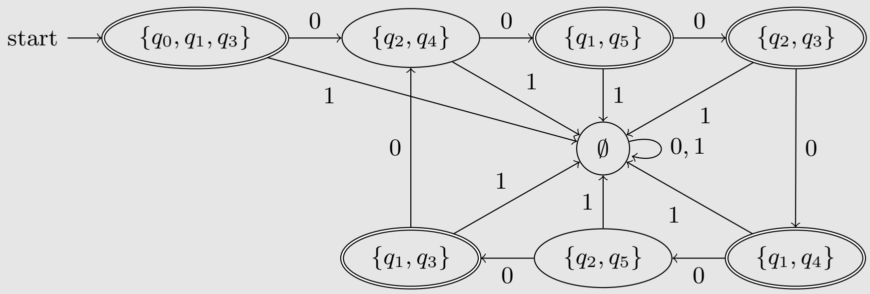


(a) Descrivete il linguaggio riconosciuto dall'automa.

L'automa riconosce il linguaggio di tutte le sequenze di zeri di lunghezza divisibile per 2 o per 3:

$$L = \{0^n \mid n \text{ è divisibile per 2 o per 3}\}$$

(b) Convertite l'automa in un DFA equivalente.



3. Considerate il linguaggio $L_1 = \{0^n 1^m 0^m : n, m > 0\}$. Questo linguaggio è regolare? Dimostrare formalmente la risposta.

Il linguaggio non è regolare. Supponiamo per assurdo che lo sia:

- sia h la lunghezza data dal Pumping Lemma;
- consideriamo la parola $w = 01^h 0^h$, che appartiene ad L_1 ed è di lunghezza maggiore di h;
- sia $w = xyz$ una suddivisione di w tale che $y \neq \varepsilon$ e $|xy| \leq h$;
- poiché $|xy| \leq h$, allora xy è completamente contenuta nel prefisso 01^h di w. Ci sono due casi possibili.

- Se $x \neq \varepsilon$ allora y è composta solo da 1. Inoltre, siccome $y \neq \varepsilon$, possiamo dire che $y = 1^p$ per qualche valore $p > 0$. Allora la parola xy^2z è nella forma $01^{h+p}0^h$, e quindi non appartiene al linguaggio perché il numero di 1 è diverso dal numero di 0 nell'ultima parte della parola.
- Se $x = \varepsilon$ allora, siccome $y \neq \varepsilon$, possiamo dire che $y = 01^p$ per qualche valore $p \geq 0$. Notate in questo caso lo zero iniziale è compreso in y (perché x è vuota). Allora la parola xy^0z è nella forma $1^{h-p}0^h$, e quindi non appartiene al linguaggio perché non inizia con 0, mentre tutte le parole di L_1 devono iniziare con 0 perché $n > 0$.

In entrambi i casi abbiamo trovato un assurdo quindi L_1 non può essere regolare.

4. Sia L un linguaggio regolare su un alfabeto Σ e dimostrate che il seguente linguaggio è regolare:

$$\mathit{suffixes}(L) = \{y \mid xy \in L \text{ per qualche stringa } x \in \Sigma^*\}$$

Intuitivamente, $\mathit{suffixes}(L)$ è il linguaggio di tutti i suffissi delle parole che stanno in L.

Per dimostrare che $\mathit{suffixes}(L)$ è regolare basta mostrare come costruire un automa a stati finiti che riconosce $\mathit{suffixes}(L)$ a partire dall'automa a stati finiti che riconosce L.

Sia quindi $A = (Q, \Sigma, q_0, \delta, F)$ un automa a stati finiti che riconosce il linguaggio L. Costruiamo un ε-NFA $B = (Q \cup \{q_0'\}, \Sigma, q_0', \delta_B, F)$ che ha uno stato in più di A. Chiamiamo q_0' questo nuovo stato e gli assegnamo il ruolo di stato iniziale di B. La funzione di transizione del nuovo automa aggiunge una ε-transizione dal nuovo stato iniziale q_0' verso ogni stato di Q che è raggiungibile dal vecchio stato iniziale. Il resto delle transizioni rimane invariato. Gli stati finali sono gli stessi di A.

5. Costruire una CFG che genera il linguaggio

$$L = \{a^n b^m c^k \mid \text{con } n = m \text{ o } m = k \text{ e } n, m, k \geq 0\}.$$

$$\begin{aligned}
S &\to S_1C \mid AS_2 \\
S_1 &\to \varepsilon \mid aS_1b \\
S_2 &\to \varepsilon \mid bS_2c \\
A &\to \varepsilon \mid aA \\
C &\to \varepsilon \mid cC
\end{aligned}$$