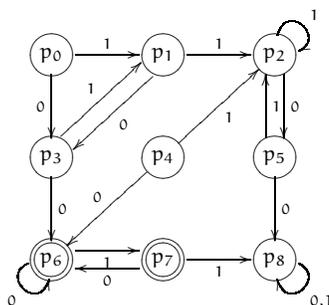


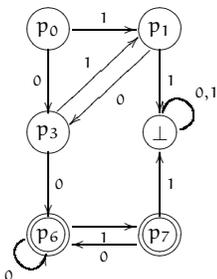
Teoria degli Automi e Calcolabilità a.a. 2021/22

Prova scritta 12 settembre 2022

Esercizio 1 Minimizzare il seguente DFA, dove p_0 è lo stato iniziale, descrivendo in modo preciso i passaggi effettuati:



Soluzione Anzitutto notiamo che p_4 non è raggiungibile a partire da p_0 quindi può essere eliminato. Inizialmente abbiamo quindi le due classi $\{p_6, p_7\}$ dei finali e $\{p_0, p_1, p_2, p_3, p_5, p_8\}$ dei non finali. Leggendo 0 possiamo discriminare $\{p_3\}$ (che va in uno stato finale) da $\{p_0, p_1, p_2, p_5, p_8\}$ (che vanno in stati non finali). Inoltre, leggendo 1 possiamo discriminare p_6 (che va in uno stato finale) da p_7 (che va in uno stato non finale). Otteniamo quindi $\{p_6\}, \{p_7\}, \{p_3\}, \{p_0, p_1, p_2, p_5, p_8\}$. A questo punto, leggendo 0 possiamo discriminare p_0 e p_1 (che vanno in $\{p_3\}$) da p_2, p_5, p_8 (che vanno in $\{p_2, p_5, p_8\}$). Otteniamo quindi $\{p_6\}, \{p_7\}, \{p_3\}, \{p_0, p_1\}, \{p_2, p_5, p_8\}$. Infine, leggendo 1 possiamo discriminare p_0 (che va in $\{p_0, p_1\}$) da p_1 (che va in $\{p_2, p_5, p_8\}$). Otteniamo quindi $\{p_6\}, \{p_7\}, \{p_3\}, \{p_0\}, \{p_1\}, \{p_2, p_5, p_8\}$. Non si può discriminare ulteriormente, quindi si ottiene il seguente DFA minimo:

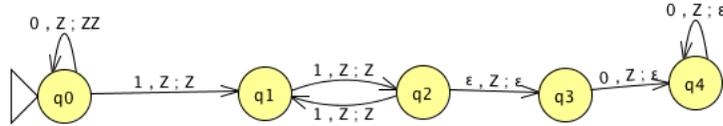


Esercizio 2 Provare che il linguaggio $\{0^m 1^n 1^n 0^m \mid n > 0\}$ non è regolare.

Soluzione Possiamo dimostrarlo utilizzando il pumping lemma. Infatti, preso $k \geq 0$ arbitrario, consideriamo la stringa $0^k 1^{2(k+1)} 0^k$ che appartiene al linguaggio ed è di lunghezza $\geq k$. Decomponendo questa stringa come uvw con $|uv| \leq k$ e $v \neq \epsilon$, si ha che sicuramente le stringhe u e v contengono solo 0. Allora la stringa uv^0w contiene un numero di 0 strettamente minore di quello in uvw , quindi non appartiene al linguaggio.

Esercizio 3 Dare un automa a pila che riconosca (per pila vuota) il linguaggio dell'esercizio precedente. È possibile dare un automa deterministico?

Soluzione Una soluzione è la seguente:



Non è possibile dare un automa deterministico in quanto il linguaggio contiene due stringhe di cui una è prefisso dell'altra: per esempio, 11 e 1111.

Esercizio 4 Dire se le seguenti affermazioni relative a proprietà dei programmi sono vere o false motivando la risposta. Consideriamo programmi C che prendono in input una stringa e restituiscono in output un numero naturale.

1. La proprietà “su nessuna stringa il programma restituisce 21” è estensionale.
2. La proprietà “su nessuna stringa di lunghezza ≤ 21 il programma restituisce 21” è ricorsivamente enumerabile.
3. La proprietà “il programma ha meno di 21 righe e restituisce 21 su almeno una stringa di lunghezza ≤ 21 ” è estensionale.
4. La proprietà “il programma ha meno di 21 righe e restituisce 21 su almeno 21 stringhe” è ricorsivamente enumerabile.

Soluzione

- Vero. Infatti si tratta di una proprietà che dipende solo dal comportamento input/output del programma.
- Falso. Infatti questa proprietà è non ricorsiva per il teorema di Rice in quanto estensionale e non banale; la proprietà complementare, ossia “esiste una stringa di lunghezza ≤ 21 sulla quale il programma restituisce 21” è ricorsivamente enumerabile in quanto basta eseguire *in interleaving* la macchina su tutte le stringhe di lunghezza ≤ 21 che sono in numero finito; quindi questa proprietà non può essere ricorsivamente enumerabile per il teorema di Post.
- Falso. Infatti dato un programma che ha meno di 21 righe (e restituisce 21 su almeno una stringa di lunghezza ≤ 21) è sempre possibile costruirne uno con lo stesso comportamento input/output che abbia più di 21 righe.
- Vero. Infatti, se il programma ha più di 21 righe si può restituire falso (o dare non terminazione). Altrimenti, si può utilizzare la tecnica a zig-zag per eseguire la macchina su tutte le stringhe, e se ci sono 21 stringhe sulle quali il programma restituisce 21 queste saranno trovate.

Esercizio 5 Si consideri l'insieme $\mathcal{EQ} = \{\langle x, y \rangle \mid \phi_x(z) = \phi_y(z)\}$, ossia le coppie di (indici di) algoritmi che danno risultati uguali su almeno un input. L'insieme è ricorsivamente enumerabile?

Soluzione Sì, infatti è sufficiente eseguire con la tecnica a zig-zag $\phi_x(z)$ e $\phi_y(z)$ su ogni z , e se per z entrambe le esecuzioni terminano possiamo controllare se i risultati sono uguali.