

Teoria degli Automi e Calcolabilità a.a. 2024/25

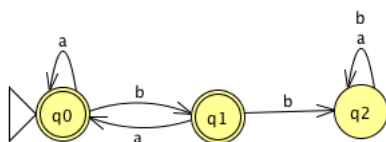
Prova scritta 3 luglio 2025

Esercizio 1 Per ognuno dei seguenti linguaggi sull'alfabeto $\{a, b\}$, si dia un automa a stati finiti *deterministico* che lo riconosca.

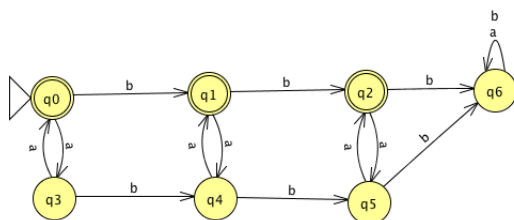
1. L'insieme delle stringhe che **non** contengono due b consecutive.
2. L'insieme delle stringhe che contengono al massimo due b e un numero pari di a .

Soluzione

1.



2.



Esercizio 2 Si consideri l'insieme delle stringhe su $\{a, b, c\}$ con un simbolo centrale (quindi di lunghezza dispari) che non compare in altre posizioni, per esempio, $abbcaab$ e cba , ma non $abbcaac$ e caa .

1. Si provi che il linguaggio non è regolare.
2. Si dia una grammatica context-free che genera il linguaggio.
3. Si dica se il linguaggio può essere riconosciuto (per pila vuota) da un PDA deterministico.

Soluzione

1. Proviamo che il linguaggio non è regolare con il pumping lemma. Fissato $n \geq 0$, consideriamo per esempio la stringa $a^n c a^n$ che appartiene al linguaggio. Decomponendo tale stringa in tre parti u, v, w tali che la lunghezza di uv sia $\leq n$ e v sia non vuota, si ha che v contiene solo a (almeno una). Allora, avendo eliminato una o più a , la stringa $uv^0 w$ non appartiene al linguaggio in quanto è di lunghezza pari oppure il simbolo centrale è a che compare anche in altre posizioni¹.

2.

S	$::=$	$A \mid B \mid C$
A	$::=$	$a \mid bAb \mid bAc \mid cAb \mid cAc$
B	$::=$	$b \mid aAa \mid aAc \mid cAa \mid cAc$
C	$::=$	$c \mid aAa \mid aAb \mid bAa \mid bAb$

3. No, in quanto esistono due stringhe appartenenti al linguaggio di cui una è prefisso dell'altra, per esempio c e cac .

¹Per $n \geq 2$. Nel caso $n = 0$ non vi sono decomposizioni che soddisfano i requisiti, e nel caso $n = 1$ l'unica decomposizione possibile di aca è con $v = a$, quindi $uv^0 w = ca$ che è di lunghezza pari.

Esercizio 3 Si consideri la seguente macchina di Turing usata come riconoscitore (q_3 è l'unico stato finale).

	a	b	B
q_0	q_1, B, R	q_2, B, R	q_3, B, N
q_1	q_1, B, R		q_3, B, N
q_2		q_2, B, R	q_3, B, N

1. Si dia un esempio di computazione non accettante di tre passi.
2. Si descriva il linguaggio accettato dalla macchina.

Soluzione

1. Per esempio:

$$q_0aaab \rightarrow q_1aab \rightarrow q_1ab \rightarrow q_1b$$

2. Il linguaggio accettato è $\{a^n \mid n \geq 0\} \cup \{b^n \mid n \geq 0\}$.

Esercizio 4 Si consideri un linguaggio A sull'alfabeto $\{a, b\}$ ricorsivamente enumerabile. Per ognuna delle seguenti affermazioni si dica se è vera o falsa, giustificando la risposta.

1. L'insieme delle stringhe palindrome che appartengono ad A è ricorsivamente enumerabile.
2. Il problema di determinare se esiste una stringa palindroma in A è semidecidibile.
3. Il problema di determinare se tutte le stringhe in A sono palindrome è semidecidibile.

Soluzione

1. Vera. Basta eseguire (in qualunque ordine) l'algoritmo che controlla se la stringa è palindroma e l'algoritmo che semidecide se la stringa appartiene ad A .
2. Vera. Basta eseguire con la tecnica a zig-zag l'algoritmo che semidecide se una stringa appartiene ad A e nel caso controllare che sia palindroma.
3. Falsa. Infatti, decidere il problema equivale a determinare se l'algoritmo che semidecide A restituisce risposta positiva solo su stringhe palindrome; questa proprietà è chiaramente estensionale e non banale, quindi non ricorsiva. Inoltre, la proprietà complementare, ossia determinare se l'algoritmo che semidecide A restituisce risposta positiva su qualche stringa non palindroma, è ricorsivamente enumerabile analogamente al caso precedente, quindi il problema di partenza non è ricorsivamente enumerabile per il teorema di Post.

Esercizio 5 Si dia una riduzione dall'insieme dei programmi che terminano sempre, ossia $\mathcal{T} = \{x \mid \phi_x \text{ totale}\}$, nell'insieme dei programmi che calcolano la funzione successore, ossia $\mathcal{S} = \{x \mid \phi_x(y) = y + 1 \text{ per ogni } y\}$. Cosa possiamo concludere su $\mathcal{S}$ come conseguenza?

Guida alla correzione

- 1.1 0 se sbagliato; -2 se non è un DFA (manca stato morto)
- 1.1 ≤ 5 se sbagliato; -2 se non è un DFA (mancano transizioni)
- 2.2 7 per grammatica che genera $(a|b)^*c(a|b)^*$ (e analoghi)
- 3.1 -1 a chi fa due passi invece di tre; -2 a chi non cancella i simboli; 5 a chi mostra computazione accettante
- 4.1,4.2,4.3 ≤ 3 per solo risposta giusta