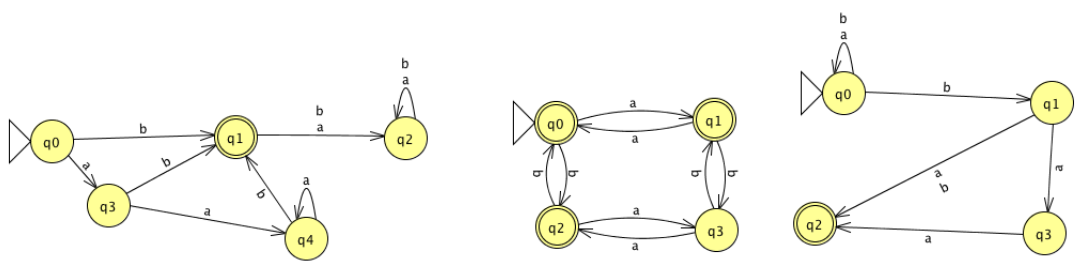
Teoria degli Automi e Calcolabilità a.a. 2019/20
Prova scritta 22 gennaio 2020

Il compito è diviso in due parti corrispondenti ai compitini: 1) Teoria degli Automi e 2) Calcolabilità. Si può consegnare una delle due parti mantenendo per l'altra il risultato del compitino. I punteggi sono indicativi.

1 Automi e Linguaggi

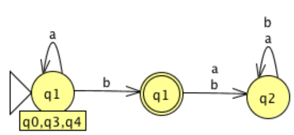
Esercizio 1 - Automi a stati finiti Si considerino i seguenti automi a stati finiti sull'alfabeto a, b.



- Per ognuno degli automi, se è non deterministico lo si renda deterministico. Se invece è deterministico si dica se è minimo o, in caso contrario, lo si minimizzi.
- Per ognuno degli automi, si descriva (in modo preciso) il linguaggio riconosciuto.

Soluzione

1. Il primo automa è deterministico. Per la minimizzazione, inizialmente abbiamo le classi $\{q_1\}$ e $\{q_0, q_2, q_3, q_4\}$. Leggendo b possiamo discriminare q_0, q_3, q_4 da q_2, abbiamo quindi $\{q_1\}$, $\{q_0, q_3, q_4\}$ e $\{q_2\}$. Non possiamo discriminare ulteriormente quindi otteniamo:



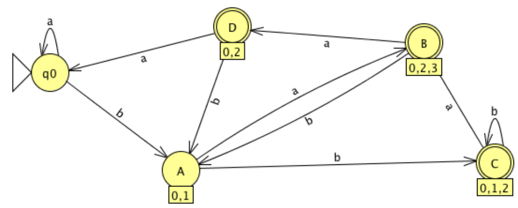
   Il linguaggio riconosciuto è $a^\star b$.

2. Il secondo automa è deterministico. Per la minimizzazione, inizialmente si hanno le classi $\{q_0, q_1, q_2\}$ e $\{q_3\}$. Leggendo a possiamo discriminare q_0, q_1 da q_2 quindi abbiamo $\{q_0, q_1\}$, $\{q_2\}$ e $\{q_3\}$. Leggendo b possiamo discriminare q_0 da q_1. L'automa è quindi già minimo. Il linguaggio riconosciuto è l'insieme delle stringhe con un numero pari di a o di b.

3. Il terzo automa è non deterministico. Scriviamolo in formato tabellare.

	a	b
$\to q_0$	q_0	q_0, q_1
q_1	q_2, q_3	q_2
$\star q_2$		
q_3	q_2	

   Applichiamo la trasformazione che elimina il non determinismo e ridenominiamo gli insiemi con più di un elemento.



	a	b
$\rightarrow q_0$	q_0	q_0, q_1
$A \equiv q_0, q_1$	$B \equiv q_0, q_2, q_3$	$C \equiv q_0, q_1, q_2$
$B \equiv \star q_0, q_2, q_3$	$D \equiv q_0, q_2$	$A \equiv q_0, q_1$
$C \equiv \star q_0, q_1, q_2$	$B \equiv q_0, q_2, q_3$	$C \equiv q_0, q_1, q_2$
$D \equiv \star q_0, q_2$	q_0	$C \equiv q_0, q_1, q_2$

Il linguaggio riconosciuto è l'insieme delle stringhe che terminano con ba, bb, oppure baa.

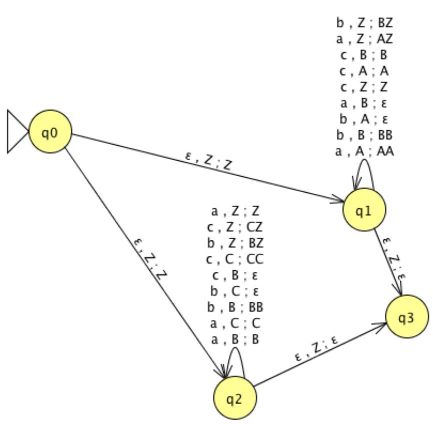
Esercizio 2 - Linguaggi context-free Per ognuno dei seguenti linguaggi sull'alfabeto $\{a, b, c\}$:

- l'insieme delle stringhe con lo stesso numero di a e b, oppure lo stesso numero di b e c (per esempio *bacaa*, *cabba*, ma non *bacaac*)
- l'insieme delle stringhe con lo stesso numero di a, b, e c (per esempio *bacaabbcc*, *cabbac*, ma non *bacaa*)

si dica se è context-free, motivando la risposta (ossia: in caso di risposta positiva si dia un PDA che riconosce il linguaggio, o una grammatica che lo generi, in caso di risposta negativa lo si provi utilizzando il pumping lemma).

Soluzione

1. Il primo linguaggio è context-free. Un PDA che lo riconosce è il seguente, che non deterministicamente riconosce le stringhe con lo stesso numero di a e di b, oppure quelle con lo stesso numero di b e di c.



2. Il secondo linguaggio non è context-free. Possiamo dimostrarlo utilizzando il pumping lemma. Infatti, preso n arbitrario, consideriamo la stringa $a^n b^n c^n$. Vi sono molti diversi casi di decomposizione di questa stringa come $uvwxy$. Tuttavia, dato che la lunghezza di vwx deve essere $\leq n$, è facile capire che tale sottostringa *non* può contenere sia a che c, perché in tal caso dovrebbe contenere tutti i b. Quindi, prendendo uv^0wx^0y, si ottiene una stringa in cui il numero di uno o due simboli è diminuito strettamente, quindi la stringa non appartiene al linguaggio.

2 Calcolabilità

Esercizio 3 - Macchine di Turing Si consideri la seguente macchina di Turing usata come riconoscitore ($q_5 \equiv$ `halt-accept` è l'unico stato finale).

	a	b	B
q_0	q_1, B, R	q_2, B, R	q_0, B, R
`check` $\equiv q_1$	q_2, a, R	q_2, b, R	q_5, B, N
`goR` $\equiv q_2$	q_2, a, R	q_2, b, R	q_3, B, L
`elim` $\equiv q_3$	q_4, B, L	q_4, B, L	
`back` $\equiv q_4$	q_4, a, L	q_4, b, L	q_0, B, R
`halt-accept` $\equiv q_5$			

Riportiamo anche la versione nel formato del simulatore:

```
0 a _ r check
0 b _ r goR
0 _ _ r 0
check a a r goR
check b b r goR
check _ _ * halt-accept
goR a a r goR
goR b b r goR
goR _ _ l elim
elim a _ l back
elim b _ l back
back a a l back
back b b l back
back _ _ r 0
```

1. Scegliendo input di lunghezza almeno due, si descrivano una computazione accettante, una computazione terminante e non accettante, e una computazione non terminante.
2. Si descriva il linguaggio accettato dalla macchina.
3. Si modifichi la macchina in modo che il linguaggio accettato resti lo stesso e tutte le computazioni siano terminanti.

Soluzione

1. Computazione accettante:

   $$\langle q_0, \epsilon, bab\rangle \to \langle q_2, \epsilon, ab\rangle \to \langle q_2, a, b\rangle \to \langle q_2, ab, \epsilon\rangle \to \langle q_3, a, b\rangle \to \langle q_4, \epsilon, a\rangle \to \langle q_4, \epsilon, Ba\rangle \to \langle q_0, \epsilon, a\rangle \to \langle q_1, \epsilon, \epsilon\rangle \to \langle q_5, \epsilon, \epsilon\rangle$$

   Computazione terminante e non accettante:

   $$\langle q_0, \epsilon, bbb\rangle \to \langle q_2, \epsilon, bb\rangle \to \langle q_2, b, b\rangle \to \langle q_2, bb, \epsilon\rangle \to \langle q_3, b, b\rangle \to \langle q_4, \epsilon, b\rangle \to \langle q_4, \epsilon, Bb\rangle \to \langle q_0, \epsilon, b\rangle \to \langle q_2, \epsilon, \epsilon\rangle \to \langle q_3, \epsilon, \epsilon\rangle$$

   Computazione non terminante:

   $$\langle q_0, \epsilon, bb\rangle \to \langle q_2, \epsilon, b\rangle \to \langle q_2, b, \epsilon\rangle \to \langle q_3, b, \epsilon\rangle \to \langle q_4, \epsilon, \epsilon\rangle \to \langle q_0, \epsilon, \epsilon\rangle \to \langle q_0, \epsilon, \epsilon\rangle \to \ldots$$

2. Il linguaggio accettato è l'insieme delle stringhe con simbolo centrale a (quindi di lunghezza dispari).
3. Basta eliminare la mossa per q_0 e B. In questo modo sulle stringhe di lunghezza pari anziché non terminare la macchina si arresta.

Esercizio 4 Si considerino macchine di Turing che calcolano funzioni da $\mathbb{N}$ in $\mathbb{N}$, rappresentate dai loro indici in una numerazione effettiva. Per ognuna delle seguenti affermazioni si dica se è vera o falsa, giustificando la risposta.

1. L'insieme delle macchine che su qualche input eseguono più di 10 passi è ricorsivamente enumerabile.
2. L'insieme delle macchine che su qualche input tra 0 e 10 effettuano una mossa a destra è ricorsivamente enumerabile.
3. L'insieme delle macchine che sull'input 10 non producono 10 è ricorsivamente enumerabile.
4. L'insieme delle macchine che su qualche input non producono 10 è ricorsivamente enumerabile (suggerimento: utilizzare una riduzione)

Soluzione

1. Vera. Basta eseguire successivamente la macchina per 10 passi su tutti gli input, e se esiste un input sul quale la computazione non è ancora terminata questo sarà trovato.
2. Vera. Basta eseguire la macchina in interleaving per gli input da 0 a 10 e se per uno di questi input viene effettuata una mossa a destra questa sarà trovata.
3. Falsa. L'insieme e il suo complementare ($\{x \mid \phi_x(10) = 10\}$) sono estensionali e non banali quindi per il teorema di Rice non ricorsivi. L'insieme complementare è ricorsivamente enumerabile (basta eseguire la macchina sull'input 10) quindi l'insieme delle macchine che sull'input 10 non producono 10 non può esserlo per il teorema di Post.
4. Falsa. Per mostrare che l'insieme, sia $\mathcal{B}$, non è ricorsivamente enumerabile, utilizziamo una riduzione da $\overline{\mathcal{K}}$ (non possiamo utilizzare il teorema di Post perché anche il complementare non è ricorsivamente enumerabile). Dato un programma x, il programma modificato, per qualunque input, restituisce 10 se x termina sull'input x (altrimenti ovviamente non termina). Allora se $x \in \overline{\mathcal{K}}$ il programma modificato non termina, quindi non produce 10, per qualunque input, quindi appartiene a $\mathcal{B}$; se $x \notin \overline{\mathcal{K}}$, il programma modificato produce sempre 10, quindi non appartiene a $\mathcal{B}$. Alternativamente si può considerare una riduzione dall'insieme, sia $\mathcal{A}$, delle macchine che sull'input 10 non producono 10 del punto precedente. Dato un programma x, il programma modificato, per qualunque input, esegue x sull'input 10. Allora se $x \in \mathcal{A}$ il programma modificato non produce 10 per qualunque input, quindi appartiene a $\mathcal{B}$; se $x \notin \mathcal{A}$, il programma modificato produce sempre 10, quindi non appartiene a $\mathcal{B}$.

Esercizio 5 - Riduzioni Si provi che $\mathcal{K} = \{x \mid \phi_x(x) \text{ termina}\}$ è riducibile a $\mathcal{A} = \{x \mid \phi_x(y) = \phi_x(y') = 10 \text{ per qualche } y \neq y'\}$, ossia che il problema di determinare se un algoritmo termina sulla propria rappresentazione è riducibile al problema di determinare se un algoritmo produce output 10 su due input diversi.

Soluzione Dobbiamo trasformare un input x per il problema $\mathcal{K}$ (un algoritmo) in un input $x' = g(x)$ per il problema $\mathcal{A}$ in modo tale che $\phi_x(x)$ termini se e solo se ϕ'_x produce output 10 su due input diversi.
Questo si può ottenere costruendo l'algoritmo $x' = g(x)$ nel modo seguente:

input $y \to$ se $\phi_x(x)$ termina restituisco 10, altrimenti ho non terminazione

Allora: se $\phi_x(x)$ termina, $\phi_{x'}$ restituisce 10 per ogni y, quindi in particolare restituisce 10 su due inout diversi, altrimenti $\phi_{x'}$ non termina per ogni y, quindi non restituisce 10 su nessun input.