Risultati della prova scritta dell'esame di

`Compilatori / Linguaggi & Compilatori`

Appello del 22 giugno 2011

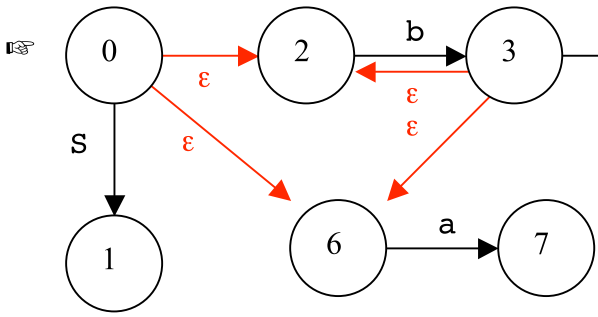
CHIRICO	28
HAIBA	21
SIRNA	30
SABELLICO	21
ZACCARDI	25

Prova scritta dell'esame di

`Compilatori / Linguaggi & Compilatori`

Appello del 22 giugno 2011

Si consideri l'automa finito **A** sull'alfabeto $\Sigma = \{$`a`, `b`, `S`$\}$ mostrato in Figura, in cui tutti gli stati sono di accettazione.



Esercizio 1. Con il metodo dei sottoinsiemi, costruire un automa finito deterministico **D** equivalente ad **A**. Quindi trovare un automa finito deterministico $\mathbf{D}_{min}$ equivalente a **D** di dimensione minima. Infine, facendo uso del lemma di Arden applicato a $\mathbf{D}_{min}$, fornire un'espressione regolare del linguaggio accettato da $\mathbf{D}_{min}$ e, quindi, da **D**.

Esercizio 2. Sia $T = \{\texttt{a}, \texttt{b}\}$ ed $N = \{\texttt{S}\}$. Determinare una grammatica acontestuale $\mathbf{G} = (T, N, \texttt{S}, P)$ tale che l'automa finito deterministico **D** dell'Esercizio 1 sia l'automa delle *preformule* (o "prefissi validi") di **G**. A tale scopo, gli stati dell'automa **A** vanno interpretati come tracce delle produzioni di una grammatica $\mathbf{G}^\# = (T, N^\#, \texttt{S}^\#, P^\#)$ con

$$N^\# = \{\texttt{S}^\#, \texttt{S}\} \quad \text{e} \qquad P^\# = \{\texttt{S}^\# \rightarrow \texttt{S}\} \cup P,$$

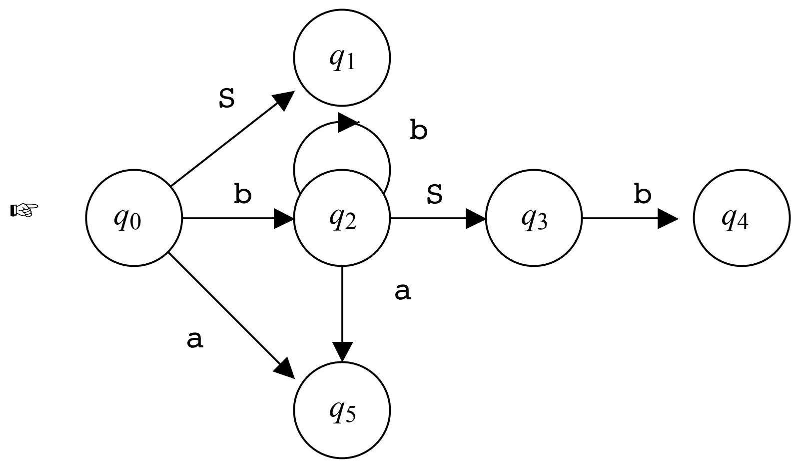
che risulterà essere la grammatica estesa di **G**.

2.1) Trovare una grammatica acontestuale **C** equivalente a **G** in forma normale di Chomsky. Verificare con l'algoritmo CYK che la stringa `bbabb` è una frase di **C** e, quindi, di **G**.

2.2) Decidere se la grammatica **G** appartiene alla classe *LR*(1). (A tale scopo, a partire da $\mathbf{G}^\#$ costruire la *tabella delle azioni*). Applicare poi il metodo induttivo (tecnica di riduzione destrorsa) per verificare che la stringa `bbabb` è una frase di **G**. Verificare infine che, nel corso del processo di riduzione della stringa `bbabb`, le voci della pila sono tutte stringhe accettate da **D**.

SOLUZIONI

Esercizio 1. Con il metodo dei sottoinsiemi, troviamo il seguente automa finito deterministico **D** equivalente ad **A**:



dove $q_0 = \{0, 2, 6\}$, $q_1 = \{1\}$, $q_2 = \{2, 3, 6\}$, $q_3 = \{4\}$, $q_4 = \{5\}$ e $q_5 = \{7\}$ sono tutti stati di accettazione. Con l'aggiunta di uno stato non di accettazione q_6 che sia stato finale per tutte le transizioni assenti in **D**, otteniamo un automa finito deterministico completo $\mathbf{D}'$ equivalente ad **A**. A questo punto possiamo ottenere un automa finito deterministico completo equivalente a **D** e di dimensione minima per raffinamento della partizione iniziale $\{R, S\}$ degli stati di $\mathbf{D}'$ con $R = \{q_0, q_1, q_2, q_3, q_4, q_5\}$ ed $S = \{q_6\}$. Consideriamo la sottotabella della tabella delle transizioni di $\mathbf{D}'$ indotta da R.

stato	s	a	b
q_0	R	R	R
q_1	S	S	S
q_2	R	R	R
q_3	S	S	R
q_4	S	S	S
q_5	S	S	S

Così la classe R viene scomposta nelle tre sottoclassi $X = \{q_0, q_2\}$, $Y = \{q_1, q_4, q_5\}$ e $Z = \{q_3\}$. Consideriamo prima X e poi Y. Consideriamo la sottotabella della tabella delle transizioni di $\mathbf{D}'$ indotta da X.

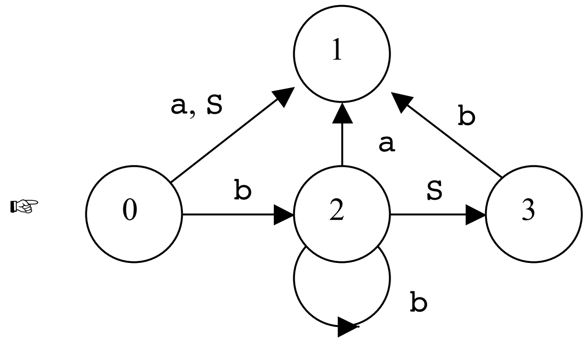
stato	s	a	b
q_0	Y	Y	X
q_2	Z	Y	X

Così la classe X viene scomposta nelle due sottoclassi $\{q_0\}$ e $\{q_2\}$. Consideriamo ora la sottotabella della tabella delle transizioni di $\mathbf{D}'$ indotta da Y.

stato	s	a	b
q_1	S	S	S
q_4	S	S	S
q_5	S	S	S

Così la classe Y non viene scomposta. In definitiva, un automa deterministico completo equivalente a **D** ha cinque stati: 0, 1, 2, 3, 4 e 5 che corrispondono rispettivamente a $\{q_0\}$, $\{q_1,$

q_4, q_5}, {q_2}, {q_3} e {q_6}, dove 5 è uno stato morto. Pertanto, $\mathbf{D}_{min}$ ha il seguente diagramma delle transizioni e tutti i suoi quattro stati sono di accettazione.



Finalmente, possiamo applicare il lemma di Arden a $\mathbf{D}_{min}$ e si ottiene il sistema di equazioni

$$\lambda_0 = \{a, S\} \cdot \lambda_1 \cup \{b\} \cdot \lambda_2 \cup \{\varepsilon\}$$
$$\lambda_1 = \{\varepsilon\}$$
$$\lambda_2 = \{b\} \cdot \lambda_2 \cup \{a\} \cdot \lambda_1 \cup \{S\} \cdot \lambda_3 \cup \{\varepsilon\}$$
$$\lambda_3 = \{b\} \cdot \lambda_1 \cup \{\varepsilon\}$$

che ha per soluzione

$$\lambda_1 = \{\varepsilon\} \qquad \lambda_3 = \{b, \varepsilon\} \qquad \lambda_2 = \{b\}^* \{a, Sb, S, \varepsilon\}$$

$$\begin{aligned}
\lambda_0 &= \{a, S\} \cup \{b\}\ \{b\}^* \{a, Sb, S, \varepsilon\} \cup \{\varepsilon\} = \\
&= \{b\}^+ \{a, Sb, S, \varepsilon\} \cup \{a, S, \varepsilon\} = \\
&= (\{b\}^+ \cup \{\varepsilon\})\ \{a, S, \varepsilon\} \cup \{b\}^+ \{Sb\} = \\
&= \{b\}^* \{a, S, \varepsilon\} \cup \{b\}^+ \{Sb\} = \\
&= \{b\}^* \{a, S, \varepsilon\} \cup \{b\}^* \{bSb\} = \\
&= \{b\}^* \{a, bSb, S, \varepsilon\}.
\end{aligned}$$

Pertanto, un'espressione regolare per il linguaggio accettato da $\mathbf{D}_{min}$ e, quindi, da **D** è

$(b)^*(a|bSb|S|\varepsilon)$.

Esercizio 2. Gli stati di **A** corrispondono alle tracce delle produzioni della grammatica $\mathbf{G}^\#$:

P0: $S^\# \rightarrow S$ P1: $S \rightarrow bSb$ P2: $S \rightarrow a$

Esplicitamente si ha:

stato di **A**	*traccia*
0	(P0, 0)
1	(P0, 1)
2	(P1, 0)
3	(P1, 1)
4	(P1, 2)
5	(P1, 3)
6	(P2, 0)
7	(P2, 1)

Pertanto, la grammatica **G** ha due sole produzioni: `S → bSb | a`

2.1) La grammatica **C** ha produzioni

`S → AB | a    A → BS    B → b`

Le sottostringhe di `bbabb` sono di seguito riportate.

11 b	12 bb	13 bba	14 bbab	15 bbabb
21 b	22 ba	23 bab	24 babb	
31 a	32 ab	33 abb	-	
41 b	42 bb	-	-	
51 b				

Tabella delle sottostringhe di `bbabb`

Ora, con l'algorimo CYK calcoliamo i generatori di ogni sottostringa di `bbabb`, cioè i simboli nonterminali da cui ogni sottostringa di `bbabb` è derivabile.

11 B	12 ∅	13 ∅	14 A	15 S
21 B	22 A	23 S	24 ∅	
31 S	32 ∅	33 ∅	-	
41 B	42 ∅	-	-	
51 B	-	-	-	

Tabella dei generatori delle sottostringhe di bbabb

Visto che il simbolo speciale S compare nella casella (1, 5), possiamo concludere che la stringa bbabb è una frase di **G**.

2.2) Per costruire la tabella delle azioni, va innanzitutto ripreso l'automa finito **D** i cui stati vengono interpretati come insiemi di tracce delle produzioni della grammatica $\mathbf{G}^{\#}$:

q_0 = {0, 2, 6} = {(P0, 0), (P1, 0), (P2, 0)}
q_1 = {1} = {(P0, 1)}
q_2 = {2, 3, 6} = {(P1, 0), (P1, 1), (P2, 0)}
q_3 = {4} = {(P1, 2)}
q_4 = {5} = {(P1, 3)}
q_5 = {7} = {(P2, 1)}

Consideriamo le tre regole che determinano le azioni di trasferimento (T), di riduzione (R) e di accettazione (A).

(1) Se q contiene una traccia ($A \rightarrow \alpha$, t) tale che $t < |\alpha|$ ed il (t+1)-esimo simbolo in α è un simbolo terminale, diciamo a, allora alla coppia (q, a) viene associata l'azione T.

Le tracce coinvolte dalla regola (1) sono (P1, 0) (in q_0 e q_2), (P1, 2) (in q_3) e (P2, 0) (in q_0 e q_2). Pertanto, abbiamo

stato	a	b	#
q_0	T	T	
q_1			
q_2	T	T	
q_3		T	

q_4			
q_5			

(2) Se q contiene una traccia $(A \to \alpha, t)$ di **A** tale che $t = |\alpha|$ ed $A \neq S^{\#}$, allora, per ogni simbolo terminale $a \in \mathcal{J}(A)$, alla coppia (q, a) viene associata l'azione R$(A \to \alpha)$.

Le tracce coinvolte dalla regola (2) sono (P1, 3) (in q_4) e (P2, 1) (in q_5). Pertanto, dopo aver calcolato $\mathcal{J}$(S) = {b, #}, abbiamo

stato	a	b	#
q_0	T	T	
q_1			
q_2	T	T	
q_3		T	
q_4		R(P1)	R(P1)
q_5		R(P2)	R(P2)

(3) Se q contiene la traccia $(S^{\#} \to S, 1)$, allora alla coppia $(q, \#)$ viene associata l'azione A.

La traccia coinvolta dalla regola (3) è (P0, 1) (in q_1). Pertanto, abbiamo

stato	a	b	#
q_0	T	T	
q_1			A
q_2	T	T	
q_3		T	
q_4		R(P1)	R(P1)
q_5		R(P2)	R(P2)

Visto che la tabella delle azioni non contiene celle multiple, possiamo concludere che la grammatica **G** appartiene alla classe $LR(1)$.

Infine, per la stringa bbabb abbiamo il processo di riduzione

pila	*buffer*	*azione*
q_0	bbabb#	T
q_0bq_2	babb#	T
q_0bq_2bq_2	abb#	T
q_0bq_2bq_2aq_5	bb#	R (S → a)
q_0bq_2bq_2Sq_3	bb#	T
q_0bq_2bq_2Sq_3bq_4	b#	R (S → bSb)
q_0bq_2Sq_3	b#	T
q_0bq_2Sq_3bq_4	#	R (S → bSb)
q_0Sq_1	#	A

e le voci della pila nel corso della riduzione sono nell'ordine: ε, b, bb, bba, bbS, bbSb, bS, bSb ed S. Queste sono tutte stringhe della forma (b)*(a|bSb|S|ε).