Informatica Teorica

Prima prova in itinere - 11 Maggio 2007

Tempo a disposizione: 1h 45'

Esercizio 1 (punti 5/13)

Scrivere una grammatica che genera il linguaggio costruito sull'alfabeto {a,b,c} fatto di tutte e sole le stringhe x tali che esistano 2 sottostringhe disgiunte w e w', di lunghezza multiplo di 3, e $w' = w^R$ (es. w = ccacbb, w' = bbcacc).

NB: Il punteggio massimo verrà dato se la grammatica scritta sarà a potenza minima tra quelle che generano il linguaggio desiderato.

Esercizio 2 (punti 5/13)

Si abbia un impianto manifatturiero che produce 2 tipi di prodotti, p_1 e p_2, tali che ogni pezzo di tipo p_1 vale 20000 euro, ed ogni pezzo di tipo p_2 vale 30000 euro. L'impianto può produrre pezzi p_1 e p_2 in qualunque sequenza, fatto salvo che, ad ogni punto della sequenza di produzione, il ***valore totale*** dei pezzi p_2 prodotti deve essere maggiore o uguale del ***valore totale*** dei pezzi p_1 prodotti, e comunque la differenza tra i 2 valori totali non deve essere mai superiore ai 90000 euro.

Si scriva un automa che accetta tutte e sole le sequenze di produzione ammissibili dell'impianto sopracitato.

NB: Il punteggio massimo verrà dato se l'automa scritto sarà a potenza minima tra quelli che riconoscono le sequenze desiderate.

Esercizio 3 (punti 6/13)

E' noto che ogni automa trasduttore definisce una traduzione $\tau: I^* \longrightarrow O^*$. Per ogni famiglia di automi trasduttori (a stati finiti, a pila –deterministici e non–, di Turing) si dica, giustificando brevemente la risposta, se essa è chiusa rispetto alla composizione di traduzioni, ossia se, dati due automi A_1 e A_2 che definiscano rispettivamente le traduzioni τ_1 e τ_2, esista nella stessa famiglia un automa A che definisca la traduzione $\tau(x) = \tau_2(\tau_1(x))$.

Soluzioni

Esercizio1

Una grammatica (non-contestuale) che genera il linguaggio desiderato è la seguente.

$S \rightarrow QR_1Q$

$Q \rightarrow aQ \mid bQ \mid cQ \mid \varepsilon$

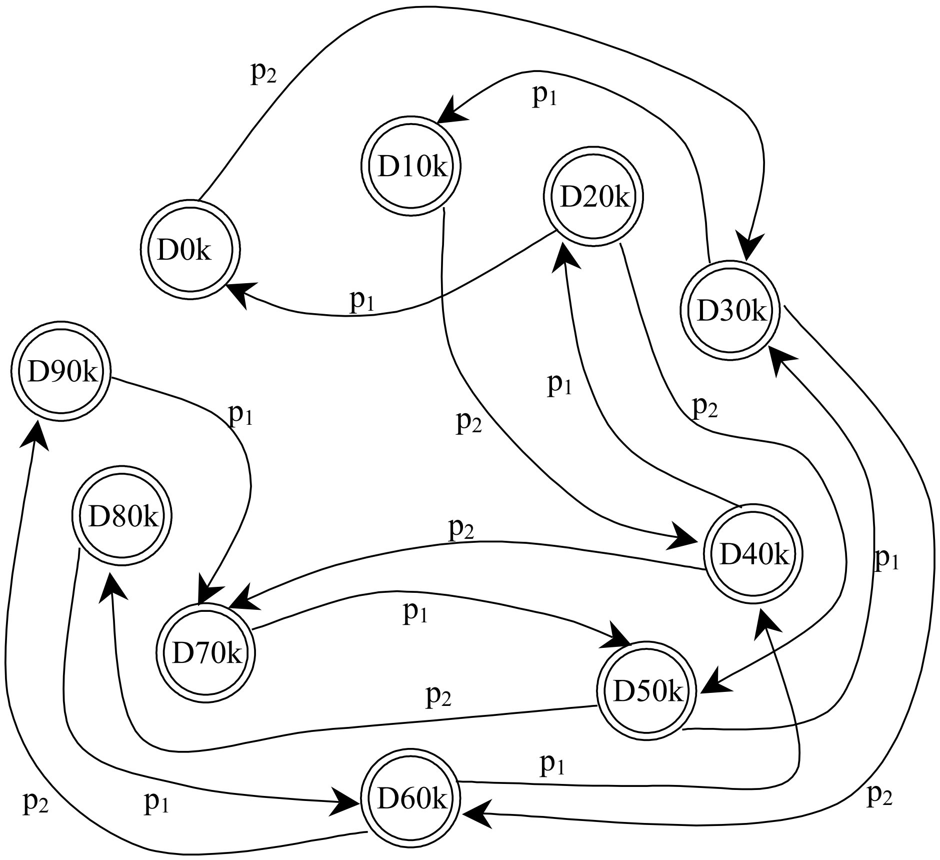
$R_1 \rightarrow aR_2a \mid bR_2b \mid cR_2c$

$R_2 \rightarrow aR_3a \mid bR_3b \mid cR_3c$

$R_3 \rightarrow aR_1a \mid bR_1b \mid cR_1c \mid aQa \mid bQb \mid cQc$

Esercizio2

Un automa a stati finiti che accetta tutte e sole le sequenze di produzione desiderate è il seguente (laddove ogni stato rappresenta la differenza fino ad ora accumulata tra il totale del valore dei pezzi p_2 ed il totale del valore dei pezzi p_1):



Esercizio 3

1.

Le traduzioni a stati finiti sono chiuse rispetto alla composizione: un procedimento che, dati due traduttori $T_1 = <Q_1, I_1, \delta_1, q_{01}, F_1, \eta_1, O_1>$ e $T_2 = <Q_2, I_2, \delta_2, q_{02}, F_2, \eta_2, O_2>$ (con $O_1 = I_2$) produca un traduttore T che definisca $\tau(x) = \tau_2(\tau_1(x))$ può essere intuitivamente descritto con la seguente costruzione, derivata dalla costruzione per l'intersezione di automi.

T è una tupla $<Q, I_1, \delta, q_0, F, \eta, O_2>$ in cui $Q = Q_1 \times Q_2$ e le funzioni di transizione e output di T, $<\delta, \eta>$, sono definite secondo la seguente regola: se in T_1 esiste una coppia

$<\delta_1, \eta_1>(q_1, a) = <q_1', w_1>$, con $a \in I_1$ e $w_1 \in O_1^*$, e per T_2 si ha

$<\delta_2^*, \eta_2^*>(q_2, w_1) = <q_2', w_2>$, con $w_2 \in O_2^*$

si definisca la coppia $<\delta, \eta>(<q_1,q_2>, a) = <<q_1' q_2'>, w_2>$.

L'insieme di stati di accettazione F di T sono tutti e soli quelli nella forma $< q_1, q_2 >$ tali che $q_1 \in F_1$ e $q_2 \in F_2$. Lo stato iniziale di T è infine lo stato $<q_{01}, q_{02}>$.

2.

Le traduzioni a pila (sia deterministiche che non) non sono chiuse rispetto alla composizione.

Infatti la traduzione $\tau(w \cdot c) = w \cdot c \cdot w$, può essere ottenuta componendo le due traduzioni $\tau_1(w \cdot c) = w \cdot c \cdot w^R \cdot d$ e $\tau_2(w_1 \cdot c \cdot w_2 \cdot d) = w_1 \cdot c \cdot w_2^R$. Essa però non può essere ottenuta da un solo automa a pila, rigorosamente legato alla disciplina LIFO nella gestione del proprio input.

Un altro controesempio possibile è il seguente: la traduzione $\tau(a^n \cdot b^n \cdot c^n) = d$ può essere ottenuta componendo le due traduzioni $\tau_1(a^n \cdot b^n \cdot c^*) = b^n \cdot c^*$ e $\tau_2(b^n \cdot c^n) = d$. Anche in questo caso la traduzione, che necessita dell'accettazione del linguaggio $\{a^n \cdot b^n \cdot c^n\}$, non può essere effettuata da un automa a pila semplice.

Più in generale, si osserva che componendo "in serie" due trasduttori a pila è possibile fare in modo che la composizione dei due accetti un linguaggio uguale all'intersezione dei due linguaggi accettati da ognuno dei due trasduttori di partenza. Poiché è noto che la classe dei linguaggi accettati dagli automi a pila ***non*** è chiusa rispetto all'intersezione, in generale non esiste quindi un automa a pila equivalente alla composizione dei due di partenza.

3.

La composizione di due traduzioni effettuate da macchine di Turing può essere banalmente ottenuta da una macchina che memorizzi in un nastro di memoria l'output della prima e poi lo usi come fosse l'input della seconda.