EQUIVALENZA TRA MACCHINE DI TURING CON NASTRO INFINITO SOLO IN UNA DIREZIONE O IN ENTRAMBE LE DIREZIONI.

Partiamo da una macchina di Turing $\mathcal{M}$ con alfabeto $A = \{s_1, \dots, s_n\}$ e stati $q_1, \dots, q_k$.

$\mathcal{M}$ calcoli una funzione parziale unaria g.

La configurazione di ingresso quando $\mathcal{M}$ inizia a calcolare $g(x)$ sarà:

$$\begin{array}{l} B\,x \\ \uparrow \\ q_1 \end{array}$$

Costruiamo adesso una macchina di Turing $\bar{\mathcal{M}}$ che calcola g su un nastro infinito in una sola direzione.

La configurazione iniziale di $\bar{\mathcal{M}}$ sarà

$$\begin{array}{l} \#\,B\,x \\ \uparrow \\ q_1 \end{array}$$

essendo $\#$ un simbolo speciale che occupa il ~~simbolo~~ quadrato più a sinistra del nastro.

L'alfabeto di $\bar{\mathcal{M}}$ sarà:

$$A \cup \{\#\} \cup \{b_j^i \mid 0 \le i, j \le n\}$$

$$b_j^i = \binom{s_i}{s_j}$$

b_j^i sta ad indicare che s_i è il simbolo che si trova sulla traccia superiore ed s_j quello sulla traccia inferiore.

Gli stati di $\overline{M}$ sono

$$\{q_1, q_2, q_3, q_4, q_5\} \cup \{\bar{q}_i , \tilde{q}_i \mid i = 1, 2, \dots, k\}$$

più alcuni stati addizionali.

Suddividiamo, come siamo ormai abituati a fare, le quadruple in tre gruppi.

La parte centrale, il CORPO della nuova macchina consisterà in una opportuna simulazione delle quadruple della macchina originaria M.

La parte iniziale servirà a preparare i dati iniziali di ingresso in una forma opportuna per la loro elaborazione da parte della macchina $\overline{M}$ a due tracce.

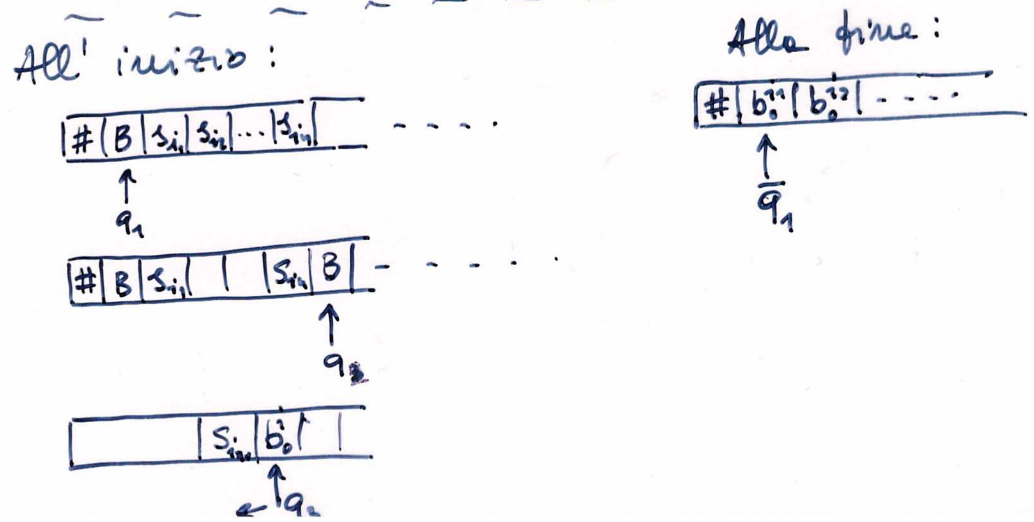
La parte finale di quadruple, infine, ha la funzione di ripresentare i valori di uscita divisi tra la traccia inferiore e quella superiore in una forma "ragionevole", cioè in una stringa dell'alfabeto di partenza.

La preparazione dei dati in modo da renderli leggibili da $\overline{\mathcal{M}}$ consiste essenzialmente nel copiare i dati di ingresso nella traccia superiore mettendo dei Blank sui quadrati corrispondenti nella traccia inferiore.

La macchina di Turing $\overline{\mathcal{M}}$ allora deve esaminare la stringa iniziale e sostituire ad ogni simbolo s_i il simbolo b_0^i.

Questo viene realizzato dalle quadruple:

$$
\begin{array}{llll l}
q_1 & B & R & q_2 & \\
q_2 & s_i & R & q_2 & i = 1, 2, \dots, n. \\
q_2 & B & L & q_3 & \\
q_3 & s_i & b_0^i & q_3 & i = 0, 1, \dots, n. \\
q_3 & b_0^i & L & q_3 & i = 0, 1, \dots, n. \\
q_3 & \# & R & \bar{q}_1 &
\end{array}
$$



Gruppo di quadruple che simulano direttamente le quadruple di $\mathcal{M}$.

		Quadruple of $\mathcal{M}$	Quadruple of $\overline{\mathcal{M}}$	
	(a)	$q_i\ s_j\ s_k\ q_l$	$\bar{q}_i\ b^j_m\ b^k_m\ \bar{q}_l$	$m = 0, 1, \ldots, n$
			$\tilde{q}_i\ b^m_j\ b^m_k\ \tilde{q}_l$	$m = 0, 1, \ldots, n$
	(b)	$q_i\ s_j\ R\ q_l$	$\bar{q}_i\ b^j_m\ R\ \bar{q}_l$	$m = 0, 1, \ldots, n$
			$\tilde{q}_i\ b^m_j\ L\ \tilde{q}_l$	$m = 0, 1, \ldots, n$
	(c)	$q_i\ s_j\ L\ q_l$	$\bar{q}_i\ b^j_m\ L\ \bar{q}_l$	$m = 0, 1, \ldots, n$
			$\tilde{q}_i\ b^m_j\ R\ \tilde{q}_l$	$m = 0, 1, \ldots, n$
Sostituiscono il B con un "doppio" blank. →	(d)	———	$\bar{q}_i\ B\ b^0_0\ \bar{q}_i$	$i = 1, 2, \ldots, K$
			$\tilde{q}_i\ B\ b^0_0\ \tilde{q}_i$	$i = 1, 2, \ldots, K$
fanno passare da una traccia all'altra. →	(e)	———	$\bar{q}_i\ \#\ R\ \tilde{q}_i$	$i = 1, 2, \ldots, K$
			$\tilde{q}_i\ \#\ R\ \bar{q}_i$	$i = 1, 2, \ldots, K$

Osservazioni

- gli stati $\bar{q}_i$ corrispondono ad azioni sulla traccia superiore, quelli $\tilde{q}_i$ ad azioni sulla traccia inferiore.

Gruppo finale

Dopo che la macchina $\mathcal{M}$ si è fermata ci troveremo in una certa situazione ma, in generale, non in quella in cui l'occhio della macchina guarda il quadrato Blank più a sinistra.

La prima cosa da fare è quindi far spostare la macchina a sinistra.
Per fare ciò è sufficiente passare ad uno stato (nuovo) che dia l'ordine di spostarsi a sinistra finché non raggiunge un blank o il termine del nastro (cioè #)
Tutto questo senza toccare niente del calcolo precedente.
Possiamo allora aggiungere delle quadruple

$$\bar{q}_i \; b^j_m \; b^j_m \; q_4$$
$$\tilde{q}_i \; b^m_j \; b^m_j \; q_4$$

in tutti i casi in cui non esistono in $\mathcal{M}$ quadruple che iniziano con $q_i S_j$.
(cosa che corrisponde ai casi che a noi interessano perché la macchina $\mathcal{M}$ si è fermata).

Le quadruple precedenti hanno il solo scopo di portare ad un nuovo stato che non è stato utilizzato prima.

Ciò fatto possiamo fare intervenire le due quadruple successive che faranno il lavoro richiesto:

$$q_4 \, b_j^i \, L \, q_4$$

$$q_4 \, \# \, B \, q_5$$

Sotto l'effetto di queste stringhe la configurazione del nastro sarà del tipo:

$$\underset{\substack{\uparrow \\ q_5}}{B} \; b_{j_1}^{i_1} \, b_{j_2}^{i_2} \ldots b_{j_k}^{i_k}$$

Ciò che rimane adesso da fare è di trasformare la stringa precedente nella stringa

$$s_{j_k} \, s_{j_{k-1}} \ldots s_{j_1} \,|\, s_{i_1} \, s_{i_2} \ldots s_{i_k}$$

I passi elementari da compiere per ottenere questo risultato sono:

- partire dal simbolo $b_{j_1}^{i_1}$ più a sinistra, riscriverlo come $s_{j_1} s_{i_1}$ (ponendo $s_0 = \#$)
- passare al simbolo successivo $b_{j_2}^{i_2}$ e scriverlo come $s_{j_2} \, s_{j_1} \, s_{i_1} \, s_{i_2}$ attorno ai due simboli di prima.

Alla fine i # verranno sostituiti da Blank.

B	$b^{i_1}_{j_1}$	$b^{i_2}_{j_2}$	...	$b^{i_k}_{j_k}$	...

$\uparrow$
q_5

questo lo interpretiamo come:

B	s_{i_1}	s_{i_2}	...	s_{i_k}	...
	s_{j_1}	s_{j_2}	...	s_{j_k}	...

e vorremmo quindi poterlo leggere

come:

B	s_{j_k}	$s_{j_{k-1}}$	...	s_{j_1}	s_{i_1}	...	$s_{i_{k-1}}$	s_{i_k}	...

[D]	RIGHT TO NEXT BLANK	
	MOVE BLOCK RIGHT	
	RIGHT	
[C]	RIGHT	
	IF b^i_j GOTO A^i_j	$(0 \leq i, j \leq n)$
	IF B GOTO F	
	GOTO C	
[A^i_j]	PRINT s_i	$(0 < i \leq n, 0 \leq j \leq n)$
	GOTO B_j	
[A^0_j]	PRINT #	$(0 \leq j \leq n)$
	GOTO B_j	
[B_j]	LEFT TO NEXT BLANK	$(0 < j \leq n)$
	PRINT s_j	
	GOTO D	
[B_0]	LEFT TO NEXT BLANK	
	PRINT #	
	GOTO D	
[F]	LEFT	
	IF s_j GOTO F	$(0 < j \leq n)$
	IF # GOTO G	
	IF B GOTO E	
[G]	PRINT B	
	GOTO F	

$[D]$	RIGHT TO NEXT BLANK	
	MOVE BLOCK RIGHT	
	RIGHT	
$[C]$	RIGHT	
	IF b^i_j GOTO A^i_j	$(0 \leq i, j \leq n)$
	IF B GOTO F	
	GOTO C	
$[A^i_j]$	PRINT s_i	$(0 < i \leq n, 0 \leq j \leq n)$
	GOTO B_j	
$[A^0_j]$	PRINT #	$(0 \leq j \leq n)$
	GOTO B_j	
$[B_j]$	LEFT TO NEXT BLANK	$(0 < j \leq n)$
	PRINT s_j	
	GOTO D	
$[B_0]$	LEFT TO NEXT BLANK	
	PRINT #	
	GOTO D	
$[F]$	LEFT	
	IF s_j GOTO F	$(0 < j \leq n)$
	IF # GOTO G	
	IF B GOTO E	
$[G]$	PRINT B	
	GOTO F	

$B\ b^{i_1}_{j_1}\ b^{i_2}_{j_2} \cdots b^{i_k}_{j_k}$ (iniz.)

$\uparrow$

$B\ b^{i_1}_{j_1} \cdots b^{i_k}_{j_k}\ B$ (dopo la 1

$\uparrow$

$B\ B\ b^{i_1}_{j_1} \cdots b^{i_k}_{j_k}\ B$ (dopo la 2

$\uparrow$

$B\ B\ b^{i_1}_{j_1} \cdots b^{i_k}_{j_k}\ B$

$\uparrow$

$B\ B\ s_{i_1} b^{i_2}_{j_2} \cdots b^{i_k}_{j_k}\ B$

$\uparrow$

$B\ B\ s_{i_1} b^{i_2}_{j_2} \cdots b^{i_k}_{j_k}\ B$

$\uparrow$

$B\ s_{i_1} s_{i_1} b^{i_2}_{j_2} \cdots b^{i_k}_{j_k}\ B$

$\uparrow$