Automi e Linguaggi Formali – A.A. 2016/17
Appello 5.9.17 Parte II

Esercizio 1. Descrivete in italiano il funzionamento della TM definita dalla seguente tabella di transizione:

	0	1	B
q_0	(q_1, B, R)	(q_5, B, R)	
q_1	$(q_1, 0, \mathrm{R})$	$(q_2, 1, \mathrm{R})$	
q_2	$(q_3, 1, \mathrm{L})$	$(q_2, 1, \mathrm{R})$	$(q_4, \mathrm{B}, \mathrm{L})$
q_3	$(q_3, 0, \mathrm{L})$	$(q_3, 1, \mathrm{L})$	$(q_0, \mathrm{B}, \mathrm{R})$
q_4	$(q_4, 0, \mathrm{L})$	$(q_4, \mathrm{B}, \mathrm{L})$	$(q_6, 0, \mathrm{R})$
q_5	(q_5, B, R)	$(q_5, \mathrm{B}, \mathrm{R})$	$(q_6, \mathrm{B}, \mathrm{R})$
$*q_6$			

Soluzione: La TM calcola l'operazione di sottrazione propria tra i due interi m e n, cioè $m \dot{-} n$=max(m-n, 0). La macchina parte con un nastro in cui è presente la stringa di input 0^m10^n, dove m e n sono rappresentati dalla loro codifica unaria (usando il simbolo zero) e divisi dal simbolo 1. La macchina si arresta con $0^{m \dot{-} n}$.

Esercizio 2. (a) Definite una macchina di Turing *M* che accetta il linguaggio costituito dalle stringhe binarie palindrome, riportando δ sia come tabella che come grafo di transizione. (b) Scrivete tre esempi di stringhe accettate dalla TM *M*, e tre esempi di stringhe non accettate da *M*.

(a) Soluzione (una tra le possibili):

	0	1	B
q_0	$(q_1, \mathrm{B}, \mathrm{R})$	$(q_2, \mathrm{B}, \mathrm{R})$	$(q_6, \mathrm{B}, \mathrm{R})$
q_1	$(q_1, 0, \mathrm{R})$	$(q_1, 1, \mathrm{R})$	$(q_3, \mathrm{B}, \mathrm{L})$
q_2	$(q_2, 0, \mathrm{R})$	$(q_2, 1, \mathrm{R})$	$(q_4, \mathrm{B}, \mathrm{L})$
q_3	$(q_5, \mathrm{B}, \mathrm{L})$		$(q_6, \mathrm{B}, \mathrm{R})$
q_4		$(q_5, \mathrm{B}, \mathrm{L})$	$(q_6, \mathrm{B}, \mathrm{R})$
q_5	$(q_5, 0, \mathrm{L})$	$(q_5, 1, \mathrm{L})$	$(q_0, \mathrm{B}, \mathrm{R})$
$*q_6$			

(b) Esempi di stringhe accettate sono 00, 101, 01010; esempi di stringhe non accettate sono 10, 001, 10011.

Esercizio 3. Indicate quali fra le seguenti istanze di PCP hanno soluzione. Ognuna è presentata sotto forma delle due liste *A* e *B*; le *i*-esime stringhe delle due liste sono corrispondenti per *i*=1, 2, etc.

(a) A = (1, 10111, 10); B = (111, 10, 0)
(b) A = (ab, aab, ba); B = (abb, ba, aa)
(c) A = (11, 1010, 01); B = (101, 10, 10)

Soluzione: (a) Ha soluzione: 2,1,1,3. (b) Non ha soluzione. La prima scelta obbligata è i=1; dopo possiamo scegliere solo 3; successivamente, ognuno dei tre indici non rende possibile proseguire; (c) Non ha soluzione. Si crea un loop: 2,1,3,3,…

Esercizio 4. (a) Date la definizione delle classi di problemi P, NP e NP-completi. (b) Quando invece possiamo definire un problema come NP-arduo? (c) Date la definizione del problema CSAT ed indicate a quale classe appartiene.

Soluzione: (a) Un problema è nella classe P se è risolvibile da una TM deterministica in tempo polinomiale. Un problema è nella classe NP se è risolvibile da una TM non-deterministica in tempo polinomiale. Un linguaggio L si dice NP-completo se (1) è in NP, e (2) per ogni altro L' in NP esiste una riduzione polinomiale di L' a L.
(b) Un problema si dice NP-arduo se è possibile dimostrare la sola condizione (2) della definizione di NP-completezza, mentre non si può dimostrare la (1). Di solito ci riferiamo a questa classe di problemi come "intrattabili".
(c) CSAT è il problema di soddisfacibilità così definito: data una espressione booleana in forma CNF, dire se è soddisfacibile. CSAT (come SAT) è NP-completo.

Esercizio 5. Dite quali tra le seguenti affermazioni è corretta:
(a) Ogni linguaggio accettato da una TM multinastro è ricorsivamente numerabile.
(b) Il linguaggio di diagonalizzazione L_d è definito come l'insieme delle stringhe w_i tali che w_i non è in $L(M_i)$ (seguendo la codifica definita a lezione).
(c) La trattazione dell'intrattabilità si basa sull'ipotesi (non dimostrata) che P=NP.
(d) L'espressione $(x \wedge \neg y \wedge \neg z) \vee (\neg x \wedge \neg y \wedge z)$ è in 3-CNF.
(e) Il linguaggio L_{ne} è ricorsivo ma non RE.

Soluzione: (a) corretta; (b) corretta; (c) non corretta (si basa sull'ipotesi, non dimostrata, che P≠ NP); (d) non corretta (AND e OR sono invertiti); (e) non corretta (è RE ma non ricorsivo).