UNIVERSITÀ DEGLI STUDI "ROMA TRE"
CORSO DI STUDI IN MATEMATICA
IN2 - MODELLI DI CALCOLO – A.A. 2005-2006

M. PEDICINI

FOGLIO N. 1 – CONSEGNA DELL'ELABORATO 10/12/2005

Esercizio 1. *Fornire le macchine di Turing che computano la somma e la differenza di interi positivi rappresentati sul nastro come sequenze binarie e valutarne la complessità.*

Esercizio 2. *Dopo aver ricordato che la Turing computabilità di una funzione totale implica la Turing decidibilità del suo grafico; mostrare*

a) che in generale non vale che la Turing computabilità con tempo di arresto lineare *di una funzione totale implica la Turing decidibilità con tempo di arrestro lineare del suo grafico.*
b) che, d'altra parte, si può dimostrare che la Turing computabilità con tempo di arresto polinomiale *di una funzione totale implica la Turing decidibilità con tempo di arrestro polinomiale del suo grafico.*

Esercizio 3. *Fornire la macchina di Turing che computa la funzione di shift di n posti a sinistra di una sequenze binaria e valutarne la complessità.*

Attenzione la funzione di shift di k posti è la funzione $s^k : \mathbb{N} \to \mathbb{N}$ tale che

$$s^k(n) = n\,2^k.$$

Esercizio 4. *Siano u, v due rappresentazioni a $2\,n$ bit considerare i numeri U_1 e U_0 ed analogamente V_1 e V_0 tali che $u = 2^n U_1 + U_0$ e similmente $u = 2^n V_1 + V_0$.*

Dimostrare che

$$uv = (2^{2n} + 2^n)U_1V_1 + 2^n(U_1 - U_0)(V_0 - V_1) + (2^n + 1)U_0V_0 \tag{1}$$

Esercizio 5. *Dimostrare che se $T(n)$ è la complessità della moltiplicazione in funzione del numero di bit della rappresentazione binaria dei numeri da moltiplicare allora vale la seguente maggiorazione:*

$$T(2n) \leq 3T(n) + cn^2 \tag{2}$$

(utilizzare i punti precedenti).

Esercizio 6. *Dimostrare per induzione che*

$$T(2^k) \leq c(3^{2k} - 2^{2k}) \qquad k \geq 1.$$

Esercizio 7. *Dimostrare che la moltiplicazione ha complessità*

$$T(n) \leq 3cn^{2\log_2 3}.$$