

UNIVERSITÀ DEGLI STUDI "ROMA TRE"
 CORSO DI STUDI IN MATEMATICA
 IN2 - MODELLI DI CALCOLO – A.A. 2004-2005
 M. PEDICINI

FOGLIO N. 1 – CONSEGNA DELL'ELABORATO 08/11/2004

Esercizio 1. Fornire il programma per macchina RAM per la moltiplicazione di due interi non negativi $x, y \in \mathbb{Z}$, rappresentati in base 2 una cifra per ogni registro di input e separati dall'intero -1 , utilizzando l'algoritmo standard per la moltiplicazione nei sistemi posizionali.

Esempio 1.

$$\begin{array}{r}
 1\ 0\ 0\ 0\ 1\ 1\ 0\ 1\ . \\
 1\ 0\ 1 \\
 \hline
 1\ 0\ 0\ 0\ 1\ 1\ 0\ 1\ + \\
 0\ + \\
 1\ 0\ 0\ 0\ 1\ 1\ 0\ 1\ = \\
 \hline
 1\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 1
 \end{array}$$

Esercizio 2. Valutare la complessità dell'algoritmo di moltiplicazione al punto 1.

Esercizio 3. 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 4. Fornire la macchina di Turing che computa la funzione di shift di n posti a sinistra di una sequenza binaria e valutarne la complessità.

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

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

Esercizio 5. Siano u, v due rappresentazioni a $2n$ 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 $v = 2^n V_1 + V_0$.

Dimostrare che

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

Esercizio 6. 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:

$$(2) \quad T(2n) \leq 3T(n) + cn$$

(utilizzare i punti precedenti).

Esercizio 7. Dimostrare per induzione che

$$T(2^k) \leq c(3^k - 2^k) \quad k \geq 1.$$

Esercizio 8. Dimostrare che la moltiplicazione ha complessità

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