

Prova d'esame di Laboratorio di Calcolo I
per il corso di laurea in Matematica
19 Settembre 2007

Tema d'esame: Calcolo di alcune operazioni aritmetiche fondamentali (quali la sottrazione e la divisione) tramite algoritmi che agiscono sulle rappresentazioni binarie dei numeri per mezzo dei soli operatori logici.

Descrizione del metodo di calcolo della sottrazione

Siano A e B due numeri interi non negativi, tali che $A \geq B$. Inoltre, siano rispettivamente $\{a_i\}_{i=0}^{30}$ e $\{b_i\}_{i=0}^{30}$ le cifre delle loro rappresentazioni binarie, in modo tale che valgano le uguaglianze

$$A = (a_{30} a_{29} \dots a_1 a_0)_2, \quad B = (b_{30} b_{29} \dots b_1 b_0)_2,$$

dove $(\dots)_2$ indica che il numero deve essere letto in base 2 e, ovviamente, $a_i = 0, 1$ e $b_i = 0, 1 \forall i = 0, \dots, 30$.

Osservazione: per semplicità, l'estensione delle rappresentazioni binarie è stata fissata a 31 cifre, così come accade per i numeri positivi che possono essere memorizzati sul computer in celle da 4 bytes.

Le cifre $\{c_i\}_{i=0}^{30}$ della rappresentazione binaria del numero $C = A - B$, possono essere determinate grazie al seguente algoritmo iterativo di calcolo (che utilizza anche le cifre $\{r_i\}_{i=0}^{30}$ che indicano i riporti, o prestiti, della sottrazione).

(I) Per quanto riguarda le cifre (binarie) del numero C , $\forall i = 0, \dots, 30$, si ponga

$$c_i = \begin{cases} 0 & \text{se } a_i = b_i \\ 1 & \text{se } a_i \neq b_i \end{cases}$$

(II) Per quanto riguarda le cifre dei riporti, si ponga $r_0 = 0$ e, $\forall i = 0, \dots, 29$,

$$r_{i+1} = \begin{cases} 0 & \text{se } a_i \geq b_i \\ 1 & \text{se } a_i < b_i \end{cases}$$

(III) $\forall i = 0, \dots, 30$, si ponga $a_i = c_i$ e $b_i = r_i$.

(IV) Se qualcuno dei riporti r_i è uguale a 1, si torni al punto (I) e si ripeta il procedimento; altrimenti, il risultato della sottrazione è proprio dato da $C = (c_{30} c_{29} \dots c_1 c_0)_2$, dove le cifre $\{c_i\}_{i=0}^{30}$ sono le ultime calcolate.

Obiettivo (intermedio) 1:

si scriva un programma in linguaggio **C** che ha il solo scopo di effettuare una sottrazione. Il programma deve contenere:

- (a) l'*input* di due numeri interi non negativi A e B tali che $A \geq B$;
- (b) una *function* che dato un numero intero restituisce un vettore contenente la sua rappresentazione binaria;
- (c) una *function* che, dati due vettori contenenti le rappresentazioni binarie di due numeri, restituisce la rappresentazione binaria del risultato della loro sottrazione (utilizzando l'algoritmo di calcolo descritto sopra);
- (d) una *function* che dato un vettore contenente una rappresentazione binaria restituisce il numero ad essa corrispondente;

- (e) la stampa sul video del risultato (in formato decimale) della sottrazione, così come calcolata dalla *function* richiesta al punto (c).

Alcuni consigli

Sia per quanto riguarda l'obiettivo 1 che i seguenti, è sicuramente comodo (e *prudente*) utilizzare delle funzioni o parti di programma, che sono incluse in altri programmi precedentemente scritti dagli studenti stessi o dal docente (e reperibili in rete).

Obiettivo (intermedio) 2:

si modifichi il programma descritto all'obiettivo 1, in modo tale da aggiungere il confronto tra due rappresentazioni binarie. Oltre a ciò che era già stato richiesto dall'obiettivo 1, il programma deve contenere:

- una *function* che, dati due vettori contenenti le rappresentazioni binarie di due numeri, restituisce 1 se il primo numero è maggiore o uguale del secondo e 0 altrimenti; si osservi che le cifre delle rappresentazioni binarie devono essere confrontate in ordine inverso, ovvero prima a_{30} con b_{30} , poi a_{29} con b_{29} , etc. (*non è ammesso il calcolo dei numeri corrispondenti alle rappresentazioni binarie prima di effettuare il confronto*);
- la chiamata della *function* richiesta al punto (a), *prima del calcolo della sottrazione*;
- la stampa su video di un messaggio che afferma quale dei due numeri immessi in *input* è maggiore dell'altro (nel caso i due numeri siano uguali, per semplicità, il messaggio può affermare che il maggiore è il primo).

Obiettivo (intermedio) 3:

si modifichi ulteriormente il programma in modo tale che sia possibile determinare

$$\bar{j} = \max \{j : A \geq 2^j B\} .$$

A questo scopo, si proceda come segue:

- si determini j^* tale che $b_{j^*} = 1$ e $b_j = 0 \forall 30 \geq j > j^*$;
- per i valori di j che cominciano da $j = 0$ fino a $j = 30 - j^*$, si effettuino le operazioni descritte ai seguenti punti (III)–(IV);
- si determini la rappresentazione binaria di $C = 2^j B$, in modo tale che $c_i = 0 \forall i = 0, \dots, j - 1$, $c_{i+j} = b_i \forall i = 0, \dots, j^*$, $c_i = 0 \forall i = j + j^* + 1, \dots, 30$;
- si confrontino le rappresentazioni binarie dei numeri A e C , utilizzando la *function* richiesta al punto (a) dell'obiettivo 2: se $A < C$ si esca immediatamente dal ciclo e si passi all'operazione (V);
- si decrementi di 1 il valore di j , così facendo, \bar{j} sarà uguale all'ultimo valore calcolato di j .

Si modifichi il programma richiesto dall'obiettivo 2, in modo tale che

- l'*input* dei due numeri interi A e B imponga ora anche la più restrittiva condizione $B > 0$;
- si includa una *function* che, date le rappresentazioni binarie di due numeri A e B , restituisce il valore di \bar{j} , calcolato seguendo l'algoritmo descritto appena sopra;
- si aggiunga la stampa sul video di \bar{j} .

Descrizione del metodo di calcolo della divisione

Come in precedenza, siano rispettivamente $\{a_i\}_{i=0}^{30}$ e $\{b_i\}_{i=0}^{30}$ le rappresentazioni binarie di due numeri $A \geq 0$ e $B > 0$.

Le cifre $\{q_i\}_{i=0}^{30}$ e $\{r_i\}_{i=0}^{30}$ della rappresentazione binaria del quoziente Q e del resto R della divisione intera $A/B = Q \text{ resto } R$, possono essere determinate grazie al seguente algoritmo iterativo di calcolo (che utilizza anche le cifre $\{c_i\}_{i=0}^{30}$ e $\{d_i\}_{i=0}^{30}$ di alcune quantità intermedie definite dall'algoritmo stesso).

- (I) $\forall i = 0, \dots, 30$, si ponga inizialmente $q_i = 0$ e $r_i = a_i$.
- (II) Si determini \bar{j} così come descritto all'interno dell'obiettivo 3.
- (III) Per i valori di j che cominciano da $j = \bar{j}$ fino a $j = 0$ (si proceda decrementando via via il valore di j), si effettuino le operazioni descritte ai seguenti punti (IV)–(VI.b).
- (IV) Si determini la rappresentazione binaria di $C = 2^j B$, così come descritto nell'operazione (III) che fa parte dell'algoritmo all'interno dell'obiettivo 3.
- (V) Si confrontino le rappresentazioni binarie corrispondenti al valore attuale del resto R e della quantità C : si ponga

$$q_j = \begin{cases} 0 & \text{se } R < C \\ 1 & \text{se } R \geq C \end{cases}$$

- (VI) Qualora la cifra r_j determinata al punto precedente sia uguale a 1, si eseguano le due seguenti operazioni:
 - (VI.a) si calcoli la rappresentazione binaria $\{d_i\}_{i=0}^{30}$ del numero $D = R - C$;
 - (VI.b) si riaggiorni il valore del resto R , ponendo $r_i = d_i \forall i = 0, \dots, 30$.

Obiettivo (finale) 4:

si modifichi il programma descritto all'obiettivo 3, in modo tale che:

- (a) nella fase di *input* si richieda ora solo che $A \geq 0$ e $B > 0$;
- (b) onde evitare confusione, si tolga dalla *main function* la chiamata alla *function* che effettua la sottrazione;
- (c) si aggiunga una *function* che, date le rappresentazioni binarie di due numeri, ne restituisce altre due contenenti il quoziente e il resto della divisione intera dei primi due numeri; tale *function* deve tradurre in pratica l'algoritmo di calcolo descritto precedentemente e, laddove necessario, deve utilizzare altre *function* richieste dagli obiettivi (1), (2) e (3);
- (d) alla fine della *main function* si inserisca la chiamata alla *function* richiesta al punto (c) e la stampa sul video del risultato (in formato decimale) della divisione.