

Fondamenti di Informatica

Esercitazione 5

7 ottobre 2024

Costrutti iterativi (bis)

5.1 [Tema d'esame] Scrivere un programma che riceve in ingresso due valori interi senz'altro positivi; il programma calcola e restituisce un nuovo intero ottenuto alternando le cifre dei due numeri a partire dalle meno significative. Nel caso un valore in ingresso abbia più cifre dell'altro, le rimanenti cifre vengono appese a sinistra del valore risultante. Non ci si preoccupi di eventuali problemi di overflow. Si parta con la cifra del primo dei due valori ricevuti in ingresso. Esempio: dati i due valori **121** e **35365**, il programma restituisce **35316251**.

5.2 Si scriva un **programma C** per calcolare la media pesata dei voti di uno studente. Il programma chiede innanzitutto all'utente quanti esami vuole inserire (massimo 60). Quindi, per ciascun esame, **memorizza** il voto (controllando che sia compreso tra 18 e 30) e il numero di crediti. Infine, stampa la media pesata, calcolata secondo questa formula:

$$media\ pesata = \frac{\sum_{i=1}^n voto_i * crediti_i}{\sum_{i=1}^n crediti_i}.$$

Suggerimento: utilizzare un array per i voti e uno per i crediti

Bonus: l'esercizio richiede esplicitamente di memorizzare i dati inseriti. Questo è davvero necessario per il funzionamento del programma?

5.3 Si scriva un programma in linguaggio C che converta un numero binario in un numero decimale. Il numero binario è rappresentato su N bit, e il valore di N è inserito da tastiera. L'utente inserisce le cifre del numero binario un bit alla volta, partendo dal bit meno significativo (ossia dal bit di peso 2^0). Il programma visualizzerà il numero decimale corrispondente. Suggerimento. Per calcolare le potenze di 2 utilizzare la funzione pow, includendo la libreria math.h. Ad esempio per calcolare 2^5 si scriverà pow(2,5). In generale, data una base a, per calcolare $y = a^b$, si scrive $y = \text{pow}(a,b)$ includendo la libreria math.h.

5.4 Scrivere un programma in linguaggio C che riceva in ingresso un numero binario rappresentato in complemento a 2 su N bit. Inizialmente l'utente inserisce il numero N di bit. Quindi inserisce le cifre del numero binario un bit alla volta, partendo dal bit meno significativo. Il programma calcola l'opposto del numero binario ricevuto in ingresso. Tale numero sarà visualizzato partendo dalla cifra meno significativa. *Suggerimento.* Per poter effettuare il calcolo del risultato, utilizzare il metodo secondo il quale si considerano le cifre del numero binario in complemento a due a partire dalla meno significativa alla più significativa (ossia da destra verso sinistra). Si ricopiano in uscita tutti gli zeri fino al primo 1 compreso. Dopo si invertono i restanti bit.

5.5 Si considerino due numeri binari rappresentati in binario puro su N bit. Il valore di N viene inserito da tastiera. I due numeri sono inseriti da tastiera un bit alla volta a partire dal bit meno significativo (LSB). Si scriva un programma in linguaggio C per eseguire la somma dei due numeri. Il programma deve visualizzare il risultato delle somme, ed indicare se si è verificata la condizione di overflow.

5.6 Scrivere un programma in linguaggio C che converta un numero decimale D in un numero binario rappresentato su N bit. L'utente inserisce un numero decimale intero positivo D e il numero N di bit su cui il numero decimale deve essere rappresentata. Il programma visualizzerà i bit che compongono il numero binario partendo dal bit meno significativo. Il programma segnalerà un errore se il numero N di bit inserito dall'utente non è sufficiente per rappresentare il numero decimale.

Suggerimento. Per effettuare la conversione usare il metodo delle divisioni successive. Ad esempio, per convertire il numero decimale $D=19$ su $N=7$ bit, si avrà:

Numero	Resto	Cifra binaria	Peso
19	1 (19%2)	1	0
9 (19/2)	1 (9%2)	1	1
4 (9/2)	0 (4%2)	0	2
2 (4/2)	0 (2%2)	0	3
1 (2/2)	1 (1%2)	1	4
0 (1/2)	0 (0%2)	0	5
0 (0/2)	0 (0%2)	0	6

Nota: nell'applicazione del metodo delle divisioni successive, l'iterazione termina quando è stato assegnato un valore a ciascuno degli N bit.

5.7 [Tema d'esame]

Scrivere un sottoprogramma che, ricevuto in ingresso un valore intero, restituisce un valore intero di una singola cifra che corrisponde alla somma delle cifre che costituiscono il valore in ingresso, ripetuta anche su valori intermedi fino a quando la somma non restituisce un valore di una sola cifra. Si assuma che il

valore in ingresso possa essere arbitrariamente grande (senza generare problemi di overflow) e sia senz'altro positivo. Per esempio se il sottoprogramma riceve in ingresso il valore 12345, il sottoprogramma calcola inizialmente 15, somma poi le cifre di 15 e calcola e restituisce 6. Se il sottoprogramma riceve in ingresso 1011001, il valore restituito è 4.

5.8 [Tema d'esame]

Scrivere un sottoprogramma che, ricevuto in ingresso un valore intero positivo `val` (è senz'altro così) e un carattere `car`, visualizza `val` nel seguente formato: per ogni cifra, a partire dalla più significativa, visualizza su una riga un numero di caratteri `car` pari al valore della cifra. Per esempio, se il sottoprogramma riceve in ingresso 2451 e * il sottoprogramma visualizza

```

* *
* * * *
* * * * *
*
```

5.8 [Tema d'esame]

Scrivere un sottoprogramma che riceve in ingresso un valore intero `val` ed un carattere `ch` e visualizza un istogramma orizzontale che mostra per ogni cifra presente in `val` il numero di occorrenze, ordinate in senso crescente (si veda l'esempio).

ingresso	uscita
2584541 e #	1: #
	2: #
	4: ##
	5: ##
	8: #

5.9 [Tema d'esame]

Scrivere un sottoprogramma che riceve in ingresso un valore intero `val` ed un carattere `ch` e visualizza un istogramma orizzontale che mostra per ogni cifra presente in `val` il numero di occorrenze, ordinate in senso crescente (si veda l'esempio).

ingresso	uscita
2584541 e #	1: #
	2: #
	4: ##
	5: ##
	8: #