

## Prova scritta del 20/9/2011

*Strutturare adeguatamente i programmi ed evidenziarne la strutturazione mediante indentazione.  
Inserire anche adeguati commenti*

1) (a) Realizzare una funzione booleana di nome `ordinato` che, presi come suoi parametri un array di interi `A` ed il numero `n` di elementi in `A`, determina se `A` è ordinato in senso crescente.

(b) Realizzare una funzione di nome `elimina_zeri` che, presi come suoi parametri un array di interi `V` ed il numero `n` di elementi in `V`, elimina da `V` tutti gli elementi il cui valore è zero. La funzione restituisce come suo risultato il numero di zeri eliminati da `V`.

(c) Realizzare un programma principale che legge da standard input una sequenza di numeri interi (lunghezza max. 1000) e determina, utilizzando opportunamente le funzioni realizzate ai punti precedenti, se la sequenza risulta ordinata in senso crescente, a meno degli eventuali zero presenti in essa. Ad esempio, la sequenza 0 3 0 0 6 0 7 9 0 è ordinata, mentre la sequenza 1 0 0 3 0 2 4 non lo è. SUGG.: Si memorizzi la sequenza letta in un array `S` che viene quindi passato come parametro alle funzioni `ordinato` e `elimina_zeri` ...

2) Scrivere un programma principale che legge da due file distinti due sequenze di numeri reali e per ogni coppia di numeri corrispondenti (= stessa posizione nei due file) scrive su un terzo file la loro somma. La lettura termina appena si raggiunge l'*end-of-file* di uno dei due file di input. I nomi dei file di input sono forniti all'inizio del programma dall'utente tramite standard input (massimo 32 caratteri); il nome del file di output è invece ottenuto concatenando i nomi dei file di input e la stringa "somma" come illustrato dal seguente esempio: se "alfa" e "beta" sono i nomi dei file di input, il nome del file di output sarà "somma\_alfa\_beta". SUGG.: non memorizzare i numeri letti dai file di input in un array, ma eseguire direttamente la somma per ogni coppia di numeri letti. N.B. Si utilizzino soltanto stringhe "tipo C".

3) Sia `S` il tipo di una struttura dati (`struct`) costituita da 3 campi di tipo intero `x1`, `x2`, `x3` e si consideri la seguente relazione d'ordine tra due elementi qualsiasi `a` e `b` di tipo `S`: `a` precede `b` se e solo se la somma dei campi di `a` è minore della somma dei campi di `b`. Modificare la funzione `ordinato` dell'esercizio (1) in modo che possa effettuare il controllo di array di elementi di tipo `S` (invece che di numeri interi). Si richiede di definire (obbligatoriamente) una funzione di nome `precede` che realizza la relazione di precedenza tra due elementi di tipo `S` sopra indicata, e di utilizzarla in modo opportuno all'interno della funzione `ordinato`.

Realizzare anche un programma principale di prova che legge i dati di tre strutture di tipo `S`, li memorizza in un array di strutture di tipo `S` di nome `dati`, e quindi verifica se `dati` è ordinato, utilizzando la (nuova) funzione `ordinato`. Ad es., l'array `dati` ottenuto leggendo la sequenza 12 3 2 8 9 7 10 10 5 risulta ordinata in senso crescente.