



Esercitazione di laboratorio n. 8

(Le soluzioni verranno valutate in Laboratorio. Solo se caricate nella Sezione Elaborati del Portale entro e non oltre il 02/12/2014 concorreranno all'assegnazione dei punti supplementari)

Esercizio n. 1: Gestione di tabelle di simboli - Versione "statica"

Si implementi un ADT di I categoria "tabella di simboli", basata su vettore, adatta a contenere oggetti di tipo Item.

Il tipo Item sia a sua volta un ADT di I categoria contenente i seguenti campi:

- codice alfanumerico univoco di 10 caratteri, con funzione di chiave di ricerca
- nome e cognome, due stringhe senza spazi di massimo 30 caratteri
- data di nascita, nel formato gg/mm/aaaa

Si scriva un programma in C in grado di memorizzare i contenuti di un file di testo in una apposita tabella di simboli.

Il nome del file sia passato da linea di comando, e sia esso organizzato come segue:

- sulla prima riga è presente il numero N di individui
- sulle 4N righe successive, in ragione di uno ogni 4 righe, sono riportate le informazioni dei singoli individui. I quattro campi sono contenuti in 4 righe consecutive.

La struttura dati deve poter essere istanziata della dimensione opportuna e permettere ricerche dicotomiche sull'elenco di individui, sulla base della chiave sopra indicata.

Si consideri la struttura dati immutabile nei contenuti: non sono ammessi inserimenti, cancellazioni e/o modifiche ai dati una volta letti da file.

Si preveda, inoltre, la possibilità di distruggere la tabella di simboli, contestualmente alla scelta di terminare il programma.

NOTA BENE: qualora sia necessario fare uso di un algoritmo di ordinamento, è richiesto che si usi il **QuickSort (realizzato appositamente, senza ricorrere al qsort della libreria C).**

Esercizio n. 2: Gestione di tabelle di simboli - Versione "dinamica"

Si estenda quanto richiesto per l'**Esercizio n.1** generalizzando le funzionalità della struttura dati, da non considerarsi più immutabile.

Si permetta l'inserimento di nuovi individui, le cui informazioni siano lette da tastiera, la cancellazione e la modifica di informazioni già presenti. Non vi è più l'obbligo di ricerca dicotomica. Si consiglia di realizzare la tabella mediante lista linkata di Item.

Esercizio n. 3:

Si supponga di avere 3 numeri interi di n cifre, dove n è noto, tali per cui la somma dei primi 2 dà il terzo. Nove delle cifre dei 3 numeri però sono ignote e sono state sostituite da uno carattere di spazio ' ', come nel seguente esempio per n=6 (lo spazio è rappresentato ■ da per leggibilità):

$$\begin{array}{rcccccc} 7 & \blacksquare & 3 & \blacksquare & 6 & \blacksquare & + \\ \blacksquare & 6 & \blacksquare & 8 & \blacksquare & 9 & = \\ \hline 9 & \blacksquare & 9 & \blacksquare & 8 & \blacksquare & \end{array}$$



Il gioco consiste nell'identificare per ciascuno spazio (■) la cifra decimale tra 1 e 9 tale per cui sia soddisfatta l'operazione. Le cifre da 1 a 9 devono essere utilizzate tutte una e una sola volta, quindi ad ogni spazio (■) corrisponde una cifra diversa e mai ripetuta. Nell'esempio considerato la soluzione, evidenziata in grassetto, è:

$$\begin{array}{rcccccc} 7 & \mathbf{3} & 3 & \mathbf{6} & 6 & \mathbf{8} & + \\ \mathbf{2} & 6 & \mathbf{5} & 8 & \mathbf{1} & 9 & = \\ \hline 9 & \mathbf{9} & 9 & \mathbf{4} & 8 & \mathbf{7} & \end{array}$$

Si scriva un programma C che, dopo aver letto da riga di comando il valore di n e le tre stringhe di caratteri, generi automaticamente una soluzione e la visualizzi. Nelle tre stringhe passate come parametro, i caratteri mancanti siano rappresentati mediante underscore '_'.