

## Prova scritta del 12/2/2008

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

1) Si supponga di realizzare un insieme di interi tra 0 ed  $n$  come un array di  $n$  booleani in cui l' $i$ -esimo elemento dell'array ha valore `true` se e solo se il numero  $i$  appartiene all'insieme. Ad esempio, assumendo  $n=10$ , l'insieme  $\{2,3,1,5\}$  è realizzato con l'array di booleani  $\{F,T,T,T,F,T,F,F,F,F\}$  ( $F = \text{false}$ ,  $T = \text{true}$ ).

(a) Realizzare una funzione booleana di nome `intersect` che, presi tre insiemi A, B e C realizzati nel modo sopra descritto, e la loro dimensione  $n$  (uguale per tutti e tre) calcola  $C = A \cap B$ . La funzione restituisce come suo risultato il numero di elementi appartenenti a C. SUGG. Si definisca C come ulteriore parametro della funzione.

(b) Descrivere la funzione `intersect` anche tramite un diagramma di flusso.

2) Scrivere un programma che permetta di eseguire le seguenti operazioni su due insiemi S1 ed S2 realizzati come nell'esercizio (1) (dimensione max. 100):

1. `inserisci_1`: richiede all'utente un numero e lo aggiunge all'insieme S1; se già presente, stampa un opportuno messaggio;
2. `inserisci_2`: come 1, ma inserisce in S2;
3. `intersezione`: calcola e stampa l'intersezione di S1 ed S2; se è vuota, stampa un opportuno messaggio (N.B. utilizzare obbligatoriamente la funzione `intersect` dell'esercizio (1));
4. `smetti`.

Il programma presenta all'utente (su standard output) il menù delle possibili operazioni, esegue l'operazione scelta e quindi ripete dall'inizio finché non viene scelta l'operazione 4. Si richiede (obbligatoriamente) di realizzare anche una funzione di nome `inserisci` per svolgere le operazioni previste ai punti 1 e 2. Si richiede inoltre di controllare la correttezza dei numeri inseriti e di ripetere l'input nel caso risultino fuori dai limiti previsti. SUGG. Inizializzare gli array S1 ed S2 con tutti `false` ...

3) Sia S il tipo di una struttura dati costituita da due campi `c1` e `c2`, rispettivamente di tipo intero e di tipo reale. Scrivere un programma che legge da un file, il cui nome è dato in input dall'utente, una sequenza di dati di tipo S e li memorizza in un array di nome T (dim. max. 1000). Quindi, per ogni dato in T, memorizza il valore del campo `c1` ed il valore del campo `c2` in due file di output distinti, i cui nomi sono ottenuti da quello di input facendolo precedere, rispettivamente, dalle stringhe "`campi1_di_`" e "`campi2_di_`" (ad es., se il nome del file di input è "`dati.txt`", i nomi dei file di output saranno "`campi1_di_dati.txt`" e "`campi2_di_dati.txt`"). Si richiede di controllare l'esistenza del file di input ed in caso negativo di terminare il programma con un opportuno messaggio d'errore. Si supponga che il nome del file di input sia una stringa qualsiasi (contente anche spazi), di lunghezza massima 100. N.B. Utilizzare soltanto stringhe tipo C (= array di caratteri ...).