

## Esercizio 4 del 6/2

### Consegnare corretto entro il 10/2 compreso

Questo esercizio riguarda ancora un array a 1 dimensione `int A[200]`, i cui i primi  $n$  elementi sono definiti e gli altri no (con  $n \leq 200$ ), e che va visto come un array a 3 dimensioni `int X[lim1][lim2][lim3]`, dove `lim1`, `lim2` e `lim3` sono tutti maggiori di 0 e  $\text{lim1} * \text{lim2} * \text{lim3} \leq 200$ .

Anziché considerare le fette di `X` come negli esercizi 2 e 3, considereremo i tasselli di `X`. Un tassello di `X` è costituito dagli elementi `X[0][i][j]...X[m][i][j]`, che sono tutti elementi definiti di `X` e tale che o  $m = \text{lim1} - 1$  oppure `X[m+1][i][j]` è indefinito.

**Esempio.** Supponiamo che `lim1=4`, `lim2=3` e `lim3=3` e che  $n=31$ . Supponiamo anche che questi siano i 31 valori distribuiti sugli strati di `X`:

0 1 0    2 1 0    0 0 1    2 0 1

1 1 1    0 1 1    1 1 1    2

1 0 0    1 0 1    0 2 2

Il tassello di coordinate  $i=0$  e  $j=0$  è costituito dai valori 0 2 0 2 che corrispondono a `X[0][0][0]`, `X[1][0][0]`, ..., `X[3][0][0]`. Il tassello di coordinate  $i=2$  e  $j=1$  è costituito dai valori 0 0 2 che corrispondono agli elementi `X[0][2][1]`, `X[1][2][1]`, `X[2][2][1]`. Esiste un ordine totale tra tasselli: il tassello di coordinate  $[i][j]$  precede quello di coordinate  $[k][m]$  se  $i < k$  o, altrimenti, se  $i = k$ , allora  $j < m$ . Ovviamente  $i$  e  $k$  sono in  $[0..\text{lim2}-1]$  e  $j$  e  $m$  in  $[0..\text{lim3}-1]$ .

**Esercizio:** viene dato un main che apre i file "input" e "output" ed esegue le seguenti letture da "input":

- i) legge  $n > 0$  e poi  $n$  interi che inserisce nei primi  $n$  posti di `A`,
- ii) poi legge `lim1`, `lim2` `lim3`, come in esercizio 3, e anche di un ulteriore intero `z`.

Successivamente il main dichiara un array `int Tass[200][2]` ed invoca la funzione `calcola` col seguente prototipo:

```
void calcola(int*A, int Tass[][2], int & etass, int n, int lim1, int lim2, int lim3, int z)
```

che deve soddisfare le seguenti pre- e post-condizioni:

PRE=(A ha i primi n elementi definiti, Tass ha 400 elementi,  $n > 0$ ,  $\text{lim1} > 0$ ,  $\text{lim2} > 0$ ,  $\text{lim3} > 0$ ,  $\text{lim1} * \text{lim2} * \text{lim3} \leq 200$ )

POST=( le coppie di elementi di Tass,  $(\text{Tass}[0][0], \text{Tass}[0][1]), (\text{Tass}[1][0], \text{Tass}[1][1]), \dots, (\text{Tass}[\text{etass}-1][0], \text{Tass}[\text{etass}-1][1])$  ) contengono le coordinate di tutti i tasselli di  $X[\text{lim1}][\text{lim2}][\text{lim2}]$  tali che le loro parti definite contengano un uguale numero di valori minori di z e di valori maggiori di z (i valori uguali a z non contano). Inoltre questi tasselli sono in Tass in ordine crescente rispetto all'ordine definito nell'Esempio).

**Esempio:** si consideri il tassello di coordinate (0,0) del precedente esempio. I suoi valori sono 0 2 0 2. Se z fosse 1 allora questo tassello avrebbe 2 valori minori di 1 e 2 valori maggiori di 1 e quindi dovrebbe venire inserito in Tass e poiché è il tassello minimo, sarebbe certamente nella prima posizione di Tass. Quindi avremmo,  $\text{Tass}[0][0]=0$  e  $\text{Tass}[0][1]=0$

Si ricordi che vanno considerati solo i valori definiti dei tasselli.

**Correttezza:** accompagnare un invariante ed una post a ciascun ciclo del vostro programma. Ogni funzione introdotta oltre a calcola, va accompagnata da una pre- ed una post-condizione. Va dimostrata la correttezza di calcola (assumendo la correttezza delle eventuali altre funzioni rispetto alle loro pre- e post-condizioni)