

## Esercizio del 15 ottobre 2014

Si tratta di un esercizio in 2 parti che richiede l'uso degli array:

1) Il main deve dichiarare un array A di 20 interi e poi deve leggere da cin degli interi da inserire negli elementi di A (in A[0], A[1], eccetera) e la lettura deve continuare fino a quando non viene letta la sentinella 0 e fino a che A non sia pieno cioè, al massimo per 20 letture. La sentinella 0 (se viene trovata) non va inserita in A.

Per realizzare questa operazione è necessario un ciclo while con una condizione di permanenza che consiste della congiunzione di 2 condizioni. Alla fine del ciclo, il programma deve fornire una variabile, diciamo N, con il numero di valori letti in A (escludendo la eventuale sentinella 0).

La PRE e POST di questo ciclo sono:

**PRE\_1**=(cin contiene (almeno) 20 interi) || (cin contiene 0)

|| è l'OR logico. Questa PRE garantisce che il programma non debba mai trovarsi nella situazione in cui cerchi di leggere un intero che non è su cin.

**POST\_1**=((se cin contiene (almeno) 20 valori senza 0) => (A sarà riempito con i primi 20 valori di cin e N=20)) && ((se cin contiene 0 preceduto da  $0 \leq n < 20$  valori diversi da 0) => (A contiene i primi n valori di cin e N=n))

Definite un invariante per il ciclo e dimostrate che attraverso di esso, riuscite a dimostrare che la POST\_1 è sempre vera alla fine del ciclo.

### Input/Output

Di seguito consideriamo qualche input particolare e l'output corrispondente. Il vostro programma per ogni input dovrà produrre l'output dato.

input= 0   output N=0 e niente è inserito in A.

input= 2 1 0   output= 2 e A[0..1]=[2,1].

input= 1 2 3 4 5 6 7 8 9 10 1 2 3 4 5 6 7 8 9 10 0   output= N=20 e A[0..19]=[ 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10] inoltre lo 0 su cin non dovrebbe essere stato letto.

(2) Assumiamo di avere N valori distinti in A. Si deve scrivere un successivo pezzo di programma che calcoli il terzo tra questi valori. In caso N sia minore di 3, il problema non ha senso e il programma deve eseguire un opportuno output. Altrimenti deve stampare il valore del terzo valore tra gli N considerati.

**PRE\_2**=(N>=0, A[0..N-1] contiene interi)

**POST\_2**=(se N<3 => stampa "problema mal posto") &&(se n>=3 => stampa il terzo tra i valori in A[0..N-1])

Anche per questa parte, associate un invariante al ciclo e usatelo per dimostrare che all'uscita del ciclo vale la POST\_2.

Input/Output

N=0 output= "problema mal posto"

N=3 , A[0..2]=[5,2,4] output= 5

N=10 A[0..9]=[9,1,8,2,7,3,6,4,5,10] output= 3