

Esercizio 1 del 17/03/2020

Dato un array A di $\text{dim}A$ interi ed un intero x , vogliamo eliminare tutte le occorrenze di x in $A[0..\text{dim}A-1]$. Assumiamo che in A ci sia almeno un'occorrenza di x , allora rimarranno $\text{dim}A' < \text{dim}A$ elementi in A e dovranno essere in $A[0..\text{dim}A'-1]$. Insomma vogliamo compattare gli elementi diversi da x verso sinistra. Inoltre è richiesto che questo venga fatto in modo che i valori che restano in A mantengano l'ordine relativo che possedevano in A originale.

Esempio 1. Supponiamo che $A=[2,10,1,10,3,4,5,10,11,23,-2]$ e $x=10$, allora dopo l'eliminazione dei 3 10 presenti, rimarranno 8 elementi, quindi $\text{dim}A'=8$ e $A=[2,1,3,4,5,11,23,-2]$.

Si chiede di scrivere una funzione: `void elimina(int A[], int & dimA, int x)` che ha come effetto di trasformare A come spiegato nell'esempio e di restituire col parametro formale $\text{dim}A$, passato per riferimento, il numero di valori rimasti in A .

Si dovrebbe cercare di trovare una soluzione che garantisca di necessitare un numero di passi pari a $\text{dim}A$, anche nel caso in cui A consista di $\text{dim}A$ x , cioè, nel caso in cui tutti gli elementi di A vadano eliminati e A si vuoti completamente.

PRE e POST di `elimina` sono come segue:

PRE $\equiv (A \text{ contiene } \text{dim}A \text{ elementi}, \text{dim}A \geq 0, A = vA)$

POST $(A[0..\text{dim}A-1])$ contiene gli elementi di vA diversi da x nell'ordine relativo che essi avevano in vA

Correttezza: scrivere l'invariante del ciclo principale della funzione `elimina`, e si dimostri la correttezza del ciclo stesso. In particolare si dovrà dimostrare che all'uscita dal ciclo vale la POST data,