

## Esercizio 1

Scrivere un programma che dato un vettore  $A$  di  $n$  interi stabilisca se  $A$  contiene o meno un elemento (detto maggioritario) tale che occorra almeno  $n/2 + 1$  volte. La prima fase del programma deve prevedere la lettura da tastiera della dimensione  $n$  ( $> 0$ ) e degli elementi di  $A$ . Si assuma che ciascun intero inserito da tastiera sia seguito da '\n'. Il programma dunque scrive a video **Sì**  $x$ \n dove  $x$  è l'elemento maggioritario o **No**\n nel caso in cui un tale elemento non sia presente in  $A$ .

Rispettare le specifiche per quanto riguarda input e output. Ad esempio, si eviti di stampare a video frasi del tipo "Inserire n:", "L'elemento maggioritario è:" ecc.

### Esempi

(L'input da tastiera è segnalato dal *corsivo*)

```
10  
2  
5  
5  
2  
2  
2  
2  
6  
2  
2  
8  
Sì 2
```

```
10  
1  
5  
5  
2  
2  
2  
6  
2  
2  
8  
No
```

## Esercizio 2

Scrivere tre programmi che eseguano rispettivamente la soluzione cubica, quadratica e lineare del problema della sottosequenza (segmento) di somma massima in un array  $A$  (vedere libro di testo [CGG] sezione 2.3 pagine 32–35).

Il programma deve leggere da tastiera il numero di elementi contenuti nell'array  $A$  e deve generare casualmente ciascuno di tali elementi nell'intervallo  $[-1000, 1000]$ . Il programma deve stampare in output  $l..r \setminus n$  dove  $l$  e  $r$  sono rispettivamente l'indice in  $A$  del primo e dell'ultimo elemento del segmento individuato.

Riportare in una tabella i tempi (in secondi) richiesti per l'esecuzione delle tre soluzioni su istanze di 10, 100, 1.000 e 10.000 elementi. Utilizzare il comando ‘‘time’’ per stabilire i tempi di esecuzione.

Rispettare le specifiche per quanto riguarda input e output. Ad esempio, si eviti di stampare a video frasi del tipo *“Inserire n:”*, *“La somma nel segmento è:”* ecc.