

Forza mediana

Il problema

Un nuovo esperimento spaziale coinvolge N oggetti, etichettati da 1 a N . Il numero N è dispari. Ogni oggetto ha una forza sconosciuta distinta espressa da un numero naturale. Per ogni forza Y , vale $1 \leq Y \leq N$. L'oggetto con forza mediana è l'oggetto X tale che il numero di oggetti che hanno forza minore di X sia uguale al numero di oggetti che hanno forza maggiore di X . Dovete scrivere un programma che determini l'oggetto con forza mediana. Sfortunatamente, il solo modo per confrontare le forze consiste nell'usare uno strumento che, dati tre oggetti distinti, determina quale dei tre oggetti ha forza mediana.

La libreria

Avete a disposizione una libreria di nome `device` con tre operazioni:

- `GetN`, che va chiamata una volta sola all'inizio, senza argomenti; restituisce il valore N .
- `Med3`, che va chiamata con le etichette di tre oggetti distinti come argomenti; restituisce l'etichetta dell'oggetto avente forza mediana.
- `Answer`, che va chiamata una volta sola alla fine, con l'etichetta di un oggetto come argomento; l'etichetta è quella dell'oggetto X avente forza mediana; la chiamata termina correttamente l'esecuzione del vostro programma.

La libreria `device` produce due file di testo: `MEDIAN.OUT` e `MEDIAN.LOG`. La prima riga del file `MEDIAN.OUT` contiene un numero intero: l'etichetta dell'oggetto passata alla libreria nella chiamata a `Answer`. La seconda riga contiene un intero: il numero di chiamate a `Med3` effettuate dal programma. Il dialogo fra il vostro programma e la libreria è registrato nel file `MEDIAN.LOG`.

Istruzioni per i programmatori Pascal: includete lo statement di importazione `uses device;` nel codice sorgente.

Istruzioni per i programmatori C/C++: usate la direttiva

```
#include "device.h"
```

nel codice sorgente, create un progetto di nome `MEDIAN.PRJ` e aggiungete al progetto i file `MEDIAN.C` (o `MEDIAN.CPP`) e `DEVICE.OBJ`.

Sperimentazione

Potete fare esperimenti con la libreria creando un file di testo di nome `DEVICE.IN`. Il file deve contenere due righe. La prima riga deve contenere un intero: il numero di oggetti N . La seconda riga deve contenere gli interi da 1 a N in qualche ordine: l' i -esimo intero è la forza dell'oggetto con etichetta i .

Esempio

`DEVICE.IN`

5
2 5 4 3 1

Il file `DEVICE.IN` indicato sopra descrive un input con 5 oggetti a cui sono assegnate forze come segue:

Etichetta	1	2	3	4	5
Forza	2	5	4	3	1

Quella che segue è una sequenza corretta di 5 chiamate alla libreria:

1. `GetN` (in Pascal) o `GetN()` (in C/C++) restituisce 5;
2. `Med3(1, 2, 3)` restituisce 3;
3. `Med3(3, 4, 1)` restituisce 4;
4. `Med3(4, 2, 5)` restituisce 4;
5. `Answer(4)`.

Vincoli

- Per il numero di oggetti N , abbiamo $5 \leq N \leq 1499$ e N è dispari.
- Per le etichette i degli oggetti, abbiamo $1 \leq i \leq N$.
- Per le forze Y degli oggetti, abbiamo $1 \leq Y \leq N$ e tutte le forze sono distinte.
- Il nome del file di libreria Pascal è `device.tpu`.
- Le funzioni e le procedure Pascal sono dichiarate come segue:

```
function GetN: integer;  
function Med3(x,y,z:integer):integer;  
procedure Answer(m:integer);
```

- I nomi dei file di libreria C/C++ sono `device.h` e `device.obj` (usate il modello di memoria “large”).
- I prototipi delle funzioni C/C++ sono come segue:

```
int GetN(void);  
int Med3(int x, int y, int z);  
void Answer(int m);
```

- Non sono consentite più di 7777 chiamate alla funzione `Med3` durante un'esecuzione.
- Il vostro programma non deve leggere né scrivere alcun file.