

# Mozzarelle di bufala (bufale)

## Descrizione del problema

Salerno è la patria delle mozzarelle di bufala. A Monica e Paola sono state regalate  $N$  mozzarelle di bufala, tutte di tipologie diverse. Monica e Paola, dopo aver trascorso vari giorni a Salerno, hanno provato tutti i tipi di mozzarelle e hanno entrambe ormai sviluppato delle preferenze. Il vostro compito è quello di aiutare Monica e Paola a dividersi le mozzarelle.

In particolare, sia Monica che Paola hanno dato un voto, espresso da un numero intero maggiore o uguale a 0, ad ogni mozzarella. I voti, ovviamente, non sono necessariamente correlati. Dovete dividere le  $N$  mozzarelle ( $N$  è un intero positivo pari) in due gruppi di  $\frac{N}{2}$  mozzarelle, uno per Monica e l'altro per Paola, in maniera che la somma complessiva dei voti (cioè, la somma dei voti di Monica relativa al gruppo di mozzarelle per Monica sommata alla somma dei voti di Paola relativa al gruppo di mozzarelle per Paola) sia massima.

Per esempio, supponiamo che ci siano le seguenti 8 mozzarelle, con i seguenti voti:

Numero mozzarella	1	2	3	4	5	6	7	8
Voto di Monica	10	2	4	6	1	7	3	4
Voto di Paola	6	6	1	0	3	8	5	7

In questo caso la soluzione migliore consiste nel dividere le mozzarelle nei due gruppi (1, 3, 4, 6) per Monica, e (2, 5, 7, 8) per Paola. In questo modo la somma totale è 48, ottenuta sommando 27 (dalle mozzarelle di Monica) con 21 (dalle mozzarelle di Paola). Non è possibile trovare una divisione delle mozzarelle che totalizzi un valore migliore.

Per risolvere il problema, dovete scrivere una funzione `solve( $N, M, P$ )`, che restituisca un singolo numero intero: il massimo valore ottenibile suddividendo tra Monica e Paola le  $N$  mozzarelle, sapendo che la  $i$ -esima mozzarella è valutata  $M_i$  da Monica e  $P_i$  da Paola.

## Subtask

- **Subtask 0 [5 punti]:** caso di esempio.
- **Subtask 1 [7 punti]:**  $2 \leq N \leq 5\,000$  e tutti i voti sono 0 oppure 1.
- **Subtask 2 [12 punti]:**  $2 \leq N \leq 1\,000\,000$  e tutti i voti sono 0 oppure 1.
- **Subtask 3 [33 punti]:**  $2 \leq N \leq 5\,000$  e tutti i voti sono compresi tra 0 e 10.
- **Subtask 4 [35 punti]:**  $2 \leq N \leq 1\,000\,000$  e tutti i voti sono compresi tra 0 e 10.
- **Subtask 5 [8 punti]:**  $2 \leq N \leq 10\,000\,000$  e tutti i voti sono compresi tra 0 e 10\,000\,000.

## Dettagli di implementazione

Dovrai sottoporre esattamente un file con estensione: `.c`, `.cpp` o `.pas`. Questo file deve implementare la funzione `solve` utilizzando uno dei seguenti prototipi.

## Programma in C o C++

```
long long solve(int N, int* M, int* P);
```

## Programma in Pascal

```
function solve(N: longint; var M: array of longint; var P: array of longint): int64;
```

La funzione implementata dovrà comportarsi come sopra descritto. Naturalmente sei libero di implementare altre routine ausiliarie per uso interno. La tua sottoposizione non deve effettuare direttamente operazioni di input o output, via console o via file di testo.

## Funzionamento del grader di esempio

Nella directory relativa a questo problema è presente una versione semplificata del grader usato durante la correzione, che potete usare per testare le vostre soluzioni in locale. Il grader di esempio legge i dati di input dal file `input.txt`, chiama la funzione che dovete implementare, e scrive il risultato restituito dalla vostra funzione sul file `output.txt`. Il file `input.txt` deve essere composto da  $N + 1$  righe, in questo formato:

- nella prima riga deve essere presente un singolo intero,  $N$ ;
- nella  $i$ -esima delle successive  $N$  righe devono essere presenti 2 interi:  $M_i$  e  $P_i$ .

## Esempio di input/output

File input.txt	File output.txt
8 10 6 2 6 4 1 6 0 1 3 7 8 3 5 4 7	48