

Truffa contabile (truffa)

La *SteamPower S.P.A.*, azienda leader mondiale nel campo delle macchine a vapore portatili, è ancora una volta in crisi nonostante le oculate manovre messe in atto nell'anno passato. Ora è il momento di stilare il bilancio di fine anno, che è di nuovo in passivo. Per non turbare gli azionisti, il *CEO* ha ricontattato il massimo esperto mondiale in campo di finanza creativa (il cui nome non possiamo rivelare).

Questa volta l'esperto ha elaborato un nuovo stratagemma: con un'audace manovra detta "*la sfangata*" una voce in uscita può diventare una voce in entrata. Questa manovra può essere ripetuta fino a che il bilancio non diventi in attivo, ma per minimizzare i rischi conviene effettuarla il *minor numero di volte*. Quante volte al minimo è necessario effettuare una sfangata affinché il bilancio diventi in attivo?

Implementazione

Dovrai sottoporre esattamente un file con estensione `.c`, `.cpp` o `.pas`.

 Tra gli allegati a questo task troverai un template (`truffa.c`, `truffa.cpp`, `truffa.pas`) con un esempio di implementazione da completare.

Se sceglierai di utilizzare il template, dovrai implementare la seguente funzione:

C/C++	<code>int sfangate(int N, int V[]);</code>
Pascal	<code>function sfangate(N: longint; var V: array of longint): longint;</code>

In cui:

- L'intero N rappresenta il numero di voci del bilancio.
- L'array V , indicizzato da 0 a $N - 1$, contiene le voci V_i del bilancio (positive le entrate e negative le uscite).
- La funzione dovrà restituire il minor numero possibile di sfangate necessarie a rendere il bilancio in attivo, che verrà stampato sul file di output.

Dati di input

Il file `input.txt` è composto da due righe. La prima riga contiene l'unico intero N . La seconda riga contiene gli N interi V_i separati da uno spazio.

Dati di output

Il file `output.txt` è composto da un'unica riga contenente un unico intero, la risposta a questo problema.

Assunzioni

- $1 \leq N \leq 100\,000$.
- $-10\,000 \leq V_i \leq 10\,000$, $V_i \neq 0$ per ogni $i = 0 \dots N - 1$.

Assegnazione del punteggio

Il tuo programma verrà testato su diversi test case raggruppati in subtask. Per ottenere il punteggio relativo ad un subtask, è necessario risolvere correttamente tutti i test relativi ad esso.

- **Subtask 1 [10 punti]:** Casi d'esempio.
- **Subtask 2 [20 punti]:** $N \leq 10$.
- **Subtask 3 [40 punti]:** $N \leq 1000$.
- **Subtask 4 [30 punti]:** Nessuna limitazione specifica.

Esempi di input/output

input.txt	output.txt
5 -5000 -10000 3000 500 -4000	1
10 100 -1000 -5000 200 -2000 400 -1800 -400 -3000 50	2
2 -700 700	1

Spiegazione

Nel **primo caso di esempio**, è sufficiente “sfangare” la voce $-10\,000$ in $10\,000$.

Nel **secondo caso di esempio**, si possono ad esempio “sfangare” le voci -5000 e -1800 .

Nel **terzo caso di esempio**, si deve “sfangare” la voce -700 .