

Ritrovo a Brambillia (**brambillia**)

Difficoltà $D = 2$

Descrizione del problema

Nell'isola di Brambillia, vi sono N città numerate da 1 a N e collegate attraverso una ferrovia circolare, le cui tratte sono anch'esse numerate da 1 a N e possono essere percorse in entrambe le direzioni: la tratta ferroviaria j collega direttamente la città j alla città $j+1$ (e, percorsa nella direzione opposta, collega $j+1$ a j) dove $j = 1, 2, \dots, N-1$; la tratta N collega la città N alla città 1 (e, percorsa nella direzione opposta, collega 1 a N). Il biglietto ferroviario per ciascuna tratta ha un costo prestabilito.

Date due qualunque città p e q , è possibile andare da p a q attraverso due percorsi ferroviari alternativi (ipotizzando che $1 \leq p < q \leq N$, un percorso attraversa le tratte $p, p+1, \dots, q-1$ mentre l'altro attraversa, nella direzione opposta, le tratte $p-1, p-2, \dots, 1, N, N-1, \dots, q$; per andare da q a p , attraversiamo tali percorsi ma in direzione opposta). Il biglietto ferroviario per ciascuno dei percorsi ha un costo pari alla somma dei costi delle singole tratte che lo compongono.

Gli abitanti di Brambillia intendono utilizzare la ferrovia circolare per ritrovarsi in occasione della sagra annuale dell'isola e devono scegliere la città presso cui organizzare tale sagra minimizzando il costo totale dei biglietti. Per questo motivo hanno contato, per ogni città, quante persone vogliono parteciparvi, visto che è necessario acquistare un biglietto ferroviario per persona al costo descritto sopra (per gli abitanti della città che verrà scelta, il costo sarà nullo perché non dovranno prendere il treno). In base a tale conteggio, individuate la città in cui organizzare la sagra, tenendo presente che le persone possono giungervi attraverso uno dei due percorsi a loro disposizione nella ferrovia circolare.

Dati di input

Il file `input.txt` è composto da $2N+1$ righe.

La prima riga contiene un intero positivo che rappresenta il numero N delle città.

Le successive N righe contengono ciascuna un intero positivo: quello nella j -esima di tali righe rappresenta il costo del biglietto ferroviario per la tratta j , dove $1 \leq j \leq N$.

Le ulteriori N righe contengono ciascuna un intero positivo o nullo: quello nella j -esima di tali righe è il numero delle persone della città j che intendono partecipare alla sagra, per $1 \leq j \leq N$.

Dati di output

Il file `output.txt` è composto da una riga contenente un solo intero j che rappresenta la città j presso cui organizzare la sagra. Come osservato in precedenza, tale città rende minimo il costo totale, ottenuto sommando i costi dei biglietti ferroviari di tutti i partecipanti.

Assunzioni

- $1 < N < 100$.

- I dati in `input.txt` garantiscono che la soluzione è unica (esiste una sola città in cui organizzare la sagra).

Esempi di input/output

File <code>input.txt</code>	File <code>output.txt</code>
4 45 34 18 40 3 0 4 2	4

Note

- Un programma che restituisce sempre lo stesso valore, indipendentemente dai dati in `input.txt`, non totalizza alcun punteggio rilevante.
- Attenzione se usate la piattaforma di sviluppo software basata sul compilatore Turbo Pascal e sul sistema operativo Windows: visto che i vostri programmi potrebbero essere valutati in una piattaforma diversa dalla vostra, avete garanzia di uniformità di comportamenti soltanto se utilizzate *sempre* il tipo `LongInt` al posto del tipo `Integer` (quest'ultimo permette di rappresentare gli interi nell'intervallo `[-32768...32767]` mentre `LongInt` ne permette la rappresentazione in `[-2147483648...2147483647]`).