

# Viaggio in treno (finestrini)

Per aiutarti con questo task, abbiamo preparato delle **tracce di soluzione**, che includono solo le parti di lettura dell'input e scrittura dell'output (da tastiera e su schermo). Puoi decidere se leggere/scrivere su file decommentando le opportune righe di codice.

- Scarica la traccia in C: `finestrini.c`
- Scarica la traccia in C++: `finestrini.cpp`
- Scarica la traccia in Python: `finestrini.py`
- Scarica la traccia in Java: `finestrini.java`
- Scarica la traccia in C#: `finestrini.cs`
- Scarica la traccia in JavaScript: `finestrini.html`
- Scarica la traccia in JavaScript (Node.js): `finestrini.js`

## Descrizione del problema

Carmen sta andando a Biella per preparare le OII! Purtroppo, nella carrozza del treno su cui viaggia, non funziona l'aria condizionata. Aiutala a decidere quali finestrini aprire.



Figure 1: “Il treno su cui viaggia Carmen”

La carrozza ha  $N$  file di posti, ciascuna con un finestrino sul lato destro e uno sul lato sinistro. Inizialmente i finestrini sono tutti chiusi. Ogni finestrino richiede una certa energia per essere aperto:  $L_i$  per il finestrino sinistro e  $R_i$  per il finestrino destro della fila  $i$ . Spendendo meno energia possibile, Carmen vuole aprire alcuni finestrini in modo che:

- in ciascuna fila, ci sia *esattamente* un finestrino aperto, a sinistra o a destra (non entrambi, per evitare correnti d'aria),
- non ci siano tre file consecutive che, su uno stesso lato, abbiano i finestrini tutti chiusi.

Stabilisci il minimo valore di energia necessaria, in totale, per aprire i finestrini in modo da soddisfare queste condizioni.

## Dati di input

La prima riga del file di input contiene un intero  $T$ , il numero di casi di test. Seguono  $T$  casi di test, numerati da 1 a  $T$ . Ogni caso di test è preceduto da una riga vuota.

Ogni caso di test è composto da  $N + 1$  righe:

- la prima riga contiene l'intero  $N$ ;
- le successive  $N$  righe contengono ciascuna i due interi  $L_i$  e  $R_i$ .

## Dati di output

Il file di output deve contenere la risposta ai casi di test che sei riuscito a risolvere. Per ogni caso di test che hai risolto, il file di output deve contenere una riga con la dicitura:

Case #t: e

dove  $t$  è il numero del caso di test (a partire da 1) e il valore  $e$  il minimo valore di energia necessario in totale.

## Assunzioni

- $T = 24$ , nei file di input che scaricherai saranno presenti esattamente 24 casi di test.
- $1 \leq N \leq 1000$ .
- $1 \leq L_i, R_i \leq 1\,000\,000$  per ogni  $0 \leq i \leq N - 1$ .

## Esempi di input/output

---

Input:

```
2
5
1 4
1 4
4 5
1 4
3 2
10
3 10
2 5
2 8
8 8
7 4
```

2 8  
1 7  
7 10  
1 8  
5 4

---

**Output:**

Case #1: 10  
Case #2: 40

---

## Spiegazione

Nel **primo caso d'esempio** la carrozza ha  $N = 5$  file. Spendendo energia 10, possiamo aprire i seguenti finestrini:

- nella fila 0, a sinistra (spendendo energia 1)
- nella fila 1, a sinistra (spendendo energia 1)
- nella fila 2, a destra (spendendo energia 5)
- nella fila 3, a sinistra (spendendo energia 1)
- nella fila 4, a destra (spendendo energia 2)

Non è possibile soddisfare le condizioni spendendo meno di 10. Pertanto, il minimo valore di energia necessaria per soddisfare le condizioni è 10

Nel **secondo caso d'esempio** la carrozza ha  $N = 10$  file. Il minimo valore di energia necessaria per soddisfare le condizioni è 40.