

# Reparto lineal

Límite de tiempo: 1s    Límite de memoria: 1 GB

## Enunciado

En una calle muy larga hay  $n$  robots repartidores. El robot  $i$  empieza en la posición  $a_i$  de la calle. También hay  $n$  puntos de entrega, en posiciones  $b_1, \dots, b_n$ .

Cada robot debe ir a exactamente un punto de entrega, y cada punto de entrega debe recibir exactamente un robot. No importa que robot va a que punto: solo queremos minimizar la distancia total recorrida.

Si un robot empieza en  $x$  y termina en  $y$ , recorre una distancia  $|x - y|$ .

Calcula la mínima suma total de distancias.

## Entrada

La primera línea contiene un entero  $n$ .

La segunda línea contiene  $n$  enteros: las posiciones iniciales de los robots.

La tercera línea contiene  $n$  enteros: las posiciones de los puntos de entrega.

## Salida

Imprime un único entero: la mínima distancia total.

## Restricciones

$1 \leq n \leq 10^5$ .

Todas las posiciones son enteros entre  $-10^9$  y  $10^9$ .

## Ejemplo

### Entrada de ejemplo

```
3
-2 10 3
4 -1 8
```

### Salida de ejemplo

```
4
```

## Nota

En el ejemplo, después de ordenar las posiciones iniciales y finales, los emparejamientos óptimos son:

$$-2 \rightarrow -1, \quad 3 \rightarrow 4, \quad 10 \rightarrow 8.$$

La distancia total es  $1 + 1 + 2 = 4$ .

Una forma de justificar el algoritmo es observar que no conviene cruzar dos asignaciones. Si  $a_i \leq a_j$  y  $b_p \leq b_q$ , emparejar en el mismo orden nunca es peor que hacer  $a_i \rightarrow b_q$  y  $a_j \rightarrow b_p$ .