



Schoenen ordenen

Adnan heeft de grootste schoenenwinkel in Baku. Een doos met n paar schoenen is zojuist gearriveerd in de winkel. Elk paar bestaat uit twee schoenen van dezelfde maat: een linker en een rechter. Adnan legt alle $2n$ schoenen in een rij van $2n$ **posities** die van links naar rechts genummerd zijn van 0 tot en met $2n - 1$.

Adnan wilt de schoenen op een **geldige ordening** leggen. Een ordening is geldig dan, en slechts dan als, voor iedere i ($0 \leq i \leq n - 1$) aan de volgende voorwaarden wordt voldaan:

- De schoenen op positie $2i$ en $2i + 1$ hebben dezelfde maat.
- De schoen op positie $2i$ is een linkerschoen.
- De schoen op positie $2i + 1$ is een rechterschoen.

Om dit te bereiken kan Adnan schoenen wisselen. Iedere keer dat hij wisselt kiest hij twee schoenen die op dat moment **naast** elkaar liggen, en verwisselt hij ze (hij pakt ze dus beide op, en legt ze op de oude positie van de andere schoen). Twee schoenen liggen naast elkaar als hun plek één verschilt.

Bepaal het minimaal aantal keer dat Adnan schoenen moet verwisselen om een geldige ordening te verkrijgen.

Implementatiedetails

Implementeer de volgende procedure:

```
int64 count_swaps(int[] S)
```

- S : een array van $2n$ integers. Voor elke i ($0 \leq i \leq 2n - 1$) is $|S[i]|$ een waarde ongelijk aan nul, die de maat van de schoen beschrijft die initieel ligt op positie i . Hier wordt met $|x|$ de absolute waarde van x bedoeld, welke gelijk is aan x wanneer $x > 0$ en gelijk is aan $-x$ wanneer $x < 0$. Als $S[i] < 0$ dan is de schoen op plek i een linkerschoen, anders is het een rechterschoen.
- De procedure moet teruggeven wat het minimale aantal verwisselingen (van naast elkaar liggende schoenen) is, om een geldige ordening te krijgen.

Voorbeelden

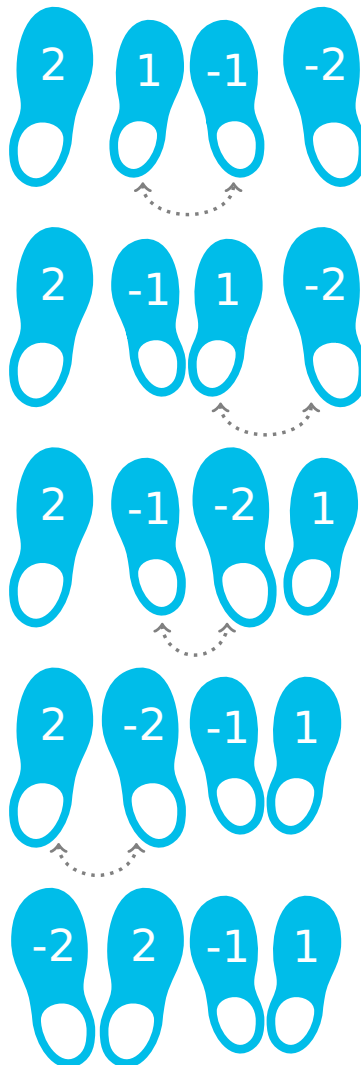
Voorbeeld 1

Kijk naar de volgende aanroep:

```
count_swaps([2, 1, -1, -2])
```

Adnan kan een geldige ordening behalen na 4 keer wisselen.

Hij kan bijvoorbeeld eerst schoenen 1 en -1 verwisselen, dan 1 en -2 , dan -1 en -2 , en tenslotte 2 en -2 . Hij bereikt dan de volgende geldige ordening: $[-2, 2, -1, 1]$. Het is niet mogelijk om een geldige ordening te behalen in minder dan 4 keer wisselen. De procedure moet dus 4 teruggeven.



Voorbeeld 2

In het volgende voorbeeld hebben alle schoenen dezelfde grootte:

```
count_swaps([-2, 2, 2, -2, -2, 2])
```

Adnan kan de schoenen op plek 2 en 3 wisselen om de geldige ordening $[-2, 2, -2, 2, -2, 2]$ te krijgen, dus de procedure moet 1 teruggeven.

Randvoorwaarden

- $1 \leq n \leq 100\,000$
- Voor elke i ($0 \leq i \leq 2n - 1$), $1 \leq |S[i]| \leq n$.
- Een geldige ordening van de schoenen kan altijd gemaakt worden door een volgorde van verwisselingen uit te voeren.

Subtaken

1. (10 punten) $n = 1$
2. (20 punten) $n \leq 8$
3. (20 punten) Alle schoenen zijn even groot.
4. (15 punten) Alle schoenen op posities $0, \dots, n - 1$ zijn linkerschoenen, en alle schoenen op posities $n, \dots, 2n - 1$ zijn rechterschoenen. Voor alle i ($0 \leq i \leq n - 1$), zijn de schoenen op i en $i + n$ even groot.
5. (20 punten) $n \leq 1000$
6. (15 punten) Geen aanvullende voorwaarden.

Voorbeeld grader

De voorbeeld grader leest de input in het volgende formaat:

- Lijn 1: n
- Lijn 2: $S[0] S[1] S[2] \dots S[2n - 1]$

De voorbeeld grader schrijft een enkele regel met de waarde verkregen van `count_swaps`.