



Uspořádání bot

Adnanovi patří největší obchod s obuví v Baku. Do obchodu právě dorazila bedna obsahující n párů bot. Každý pár sestává ze dvou bot: levé a pravé.

Adnan položil do řady všech $2n$ bot na **pozice** očíslované 0 až $2n - 1$ zleva doprava. Adnan chce nyní boty přeuspořádat do **validního uspořádání**. Uspořádání nazveme validním, právě když pro každé i ($0 \leq i \leq n - 1$) platí následující podmínky:

- Boty na pozicích $2i$ a $2i + 1$ mají stejnou velikost.
- Bota na pozici $2i$ je levá.
- Bota na pozici $2i + 1$ je pravá.

Za účelem přeuspořádání smí Adnan provést posloupnost vzájemných prohození dvou bot. Při každém prohození bot vybere dvojici **sousedních** bot a vymění je mezi sebou, tj. obě zvedne a každou položí na předchozí pozici té druhé. Sousedními botami rozumíme dvě boty, jejichž pozice se liší o 1.

Určete nejmenší počet prohození bot, které Adnan potřebuje k získání validního uspořádání všech bot.

Pokyny k implementaci

Máte za úkol implementovat následující funkci:

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

- S : pole $2n$ celých čísel. Pro každé i ($0 \leq i \leq 2n - 1$), je $S[i]$ nenulová hodnota označující botu, která je na začátku na pozici i . Absolutní hodnota $S[i]$ udává velikost boty na pozici i . Velikost boty nepřesahuje n . Jestliže $S[i] < 0$ pak je bota na pozici i levá, jinak je pravá.
- Funkce musí vrátit minimální počet prohození sousedních bot potřebný pro dosažení validního uspořádání.

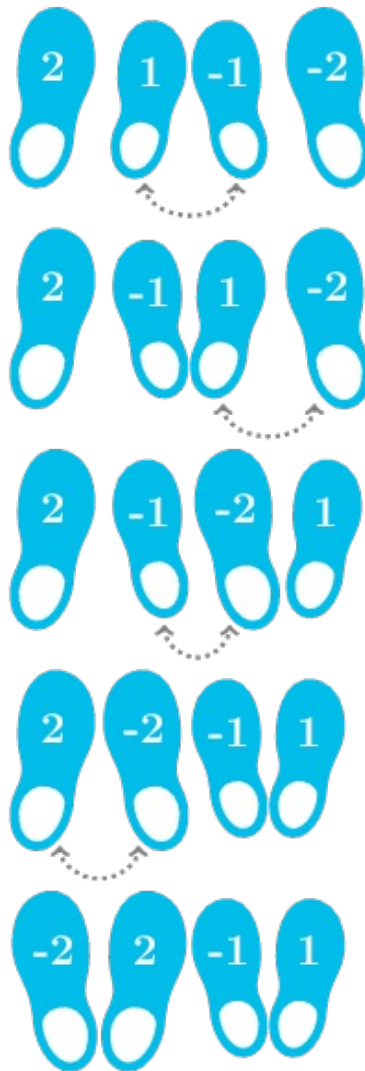
Příklady

Příklad 1

Uvažujme následující volání:

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

Adnan může dosáhnout validního uspořádání pomocí 4 prohození. Například může nejprve prohodit boty 1 a -1 , pak 1 a -2 , pak -1 a -2 a konečně 2 a -2 . Tímto postupem dostane následující validní uspořádání bot: $[-2, 2, -1, 1]$. Validního uspořádání nelze dosáhnout pomocí méně než 4 prohození. Funkce musí tudíž vrátit 4.



Příklad 2

V následujícím příkladu mají všechny boty stejnou velikost:

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

Adnan může prohodit boty na pozicích 2 a 3 a dostane validní uspořádání $[-2, 2, -2, 2, -2, 2]$, tudíž funkce musí vrátit 1.

Omezení

- $1 \leq n \leq 100\,000$
- Pro každé i ($0 \leq i \leq 2n - 1$), $1 \leq |S[i]| \leq n$. $|x|$ označuje absolutní hodnotu x .
- Validního uspořádání bot lze pomocí nějaké posloupnosti prohození dosáhnout.

Podúlohy

1. (10 bodů) $n = 1$
2. (20 bodů) $n \leq 8$
3. (20 bodů) Všechny boty mají stejnou velikost.
4. (15 bodů) Všechny boty na pozicích $0, \dots, n - 1$ jsou levé a všechny boty na pozicích $n, \dots, 2n - 1$ jsou pravé. Současně pro každé i ($0 \leq i \leq n - 1$) mají boty na pozici i a $i + n$ stejnou velikost.
5. (20 bodů) $n \leq 1000$
6. (15 bodů) Žádná další omezení.

Ukázkový vyhodnocovač

Ukázkový vyhodnocovač čte vstup v následujícím formátu:

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

Ukázkový vyhodnocovač vypíše jeden řádek obsahující výstupní hodnotu `count_swaps`.