



Arranging Shoes 排列鞋子

阿登擁有巴庫最大的鞋店。現在剛好有一箱含有 n 對鞋子的貨品運到他的鞋店。每對鞋子包含兩隻相同大小的鞋子：一隻左腳，一隻右腳。阿登把這 $2n$ 隻鞋排成一橫行，該橫行上包含有 $2n$ 個位置，以 0 至 $2n - 1$ 從左至右加以編號之。

阿登希望把這些鞋排成一個有效的排列。一個排列是有效的若及只若對於每個 i ($0 \leq i \leq n - 1$)，以下的條件均成立：

- 在位置 $2i$ 及 $2i + 1$ 上的鞋子是同一個大小的
- 在位置 $2i$ 上的鞋是一隻左腳鞋
- 在位置 $2i + 1$ 上的鞋是一隻右腳鞋

為了達成這個目的，阿登將作出一系列的對調操作。在每一個對調操作中，阿登將選擇在當時是相鄰的兩隻鞋子並將之對調（即是把兩隻鞋子拿起然後放回在對方之前所在的位置上）。兩隻鞋子是相鄰的若他們的位置編號相差為 1。

請找出阿登最少需要進行對調操作多少次才可使鞋子的排列成一個有效排列。

實現細則

你需要實現以下的子程序：

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

- S ：為一含有 $2n$ 整數的數組。對於每一個 i ($0 \leq i \leq 2n - 1$)， $|S[i]|$ 為非 0 的數值，它表示著一隻最初在位置 i 上的鞋子的大小。在這裡， $|x|$ 表示 x 的絕對值，即若 $x > 0$ 時，其值為 x ，而若 $x < 0$ 時，其值則為 $-x$ 。若 $S[i] < 0$ ，則在位置 i 上的是一隻左腳鞋，否則它是一隻右腳鞋。
- 該子程序的返回值是要把鞋子排成有效排列的最少次對調（相鄰鞋子）操作的次數。

樣例

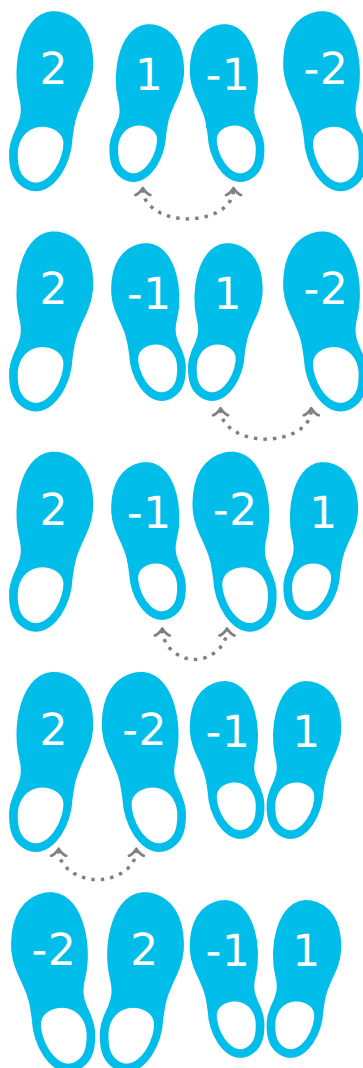
樣例 1

考慮以下的子程序呼叫：

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

阿登可以用最少 4 次對調操作就可以得到一個有效的排列。

例如他可以先對調鞋子 1 及 -1，再對調 1 及 -2，再對調 -1 及 -2，最後對調 2 及 -2。之後他就可以得到 $[-2, 2, -1, 1]$ 這麼一個有效的排列。沒有任何比 4 對調操作更少的操作方案可以得出有效的排列。因此該子程序的返回值為 4。



樣例 2

在以下的樣例中，所有的鞋子均有相同的大小：

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

阿登可以對調在位置 2 及 3 上的鞋子而獲得有效排列 $[-2, 2, -2, 2, -2, 2]$ ，因此該子程序的返回值應為 1。

限制條件

- $1 \leq n \leq 100\,000$
- 對於每個 i ($0 \leq i \leq 2n - 1$)， $1 \leq |S[i]| \leq n$ 。其中 $|x|$ 表示 x 的絕對值

- 我們一定可以通過某些對調的次序而得到有效的排列

子任務

1. (10 分) $n = 1$
2. (20 分) $n \leq 8$
3. (20 分) 所有鞋子大小都是相同的
4. (15 分) 所有在位置 $0, \dots, n - 1$ 上的鞋子都是左腳鞋，而所有在位置 $n, \dots, 2n - 1$ 上的鞋都是右腳鞋。而且對於每個 i ($0 \leq i \leq n - 1$)，在位置 i 及 $i + n$ 上的鞋子大小是相同的。
5. (20 分) $n \leq 1000$
6. (15 分) 沒有任何附加的限制條件

樣例評分程序

樣例評分程序將可以讀取合符以下格式的輸入數據:

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

樣例評分程序將輸出單一行，其上是 `count_swaps` 的返回數值。