



Buku-Buku Kuno

Kota Teheran adalah tempat di mana Perpustakaan Nasional Iran berada. Kekayaan utama perpustakaan ini terletak pada sebuah aula panjang yang memiliki n meja, dinomori dari 0 hingga $n - 1$ dari kiri ke kanan. Pada setiap meja, terdapat sebuah buku kuno tertulis tangan yang dipamerkan. Buku-buku ini diurutkan berdasarkan usia, yang membuat para pengunjung kesulitan untuk mencari buku-buku berdasarkan judul. Oleh karena itu, manajer perpustakaan memutuskan untuk mengurutkan buku-buku tersebut terurut abjad berdasarkan judul.

Aryan, seorang pustakawan, akan melakukan pengurutan tersebut. Ia telah membuat sebuah daftar p sepanjang n , yang berisi bilangan-bilangan bulat berbeda-beda dari 0 hingga $n - 1$. Daftar ini mendeskripsikan perubahan-perubahan yang diperlukan untuk mengurutkan buku-buku tersebut terurut abjad: untuk setiap $0 \leq i < n$, buku yang saat ini terletak pada meja i harus dipindahkan ke meja $p[i]$.

Aryan memulai pengurutan buku-buku tersebut pada meja s . Ia ingin kembali ke meja yang sama setelah menyelesaikan pengurutan tersebut. Karena buku-buku tersebut sangat berharga, ia tidak dapat membawa lebih dari satu buku pada suatu saat. Selama pengurutan, Aryan akan melakukan serangkaian langkah. Setiap langkah harus berupa salah satu dari berikut ini:

- Jika ia tidak sedang membawa buku dan terdapat buku pada meja di mana ia berada, ia dapat mengambil buku ini.
- Jika ia sedang membawa buku dan terdapat buku lain pada meja di mana ia berada, ia dapat menukar buku yang sedang ia bawa dengan buku yang terdapat pada meja tersebut.
- Jika ia sedang membawa buku dan ia sedang berada pada meja kosong, ia dapat meletakkan buku yang sedang ia bawa pada meja tersebut.
- Ia dapat berjalan ke meja yang mana saja. Ia dapat membawa sebuah buku saat ia melakukannya.

Untuk setiap $0 \leq i, j \leq n - 1$, jarak di antara meja-meja i dan j adalah tepat $|j - i|$ meter. Tugas Anda adalah menghitung total jarak minimum yang Aryan perlukan untuk mengurutkan seluruh buku tersebut.

Detil implementasi

Anda harus mengimplementasikan prosedur berikut ini:

```
int64 minimum_walk(int[] p, int s)
```

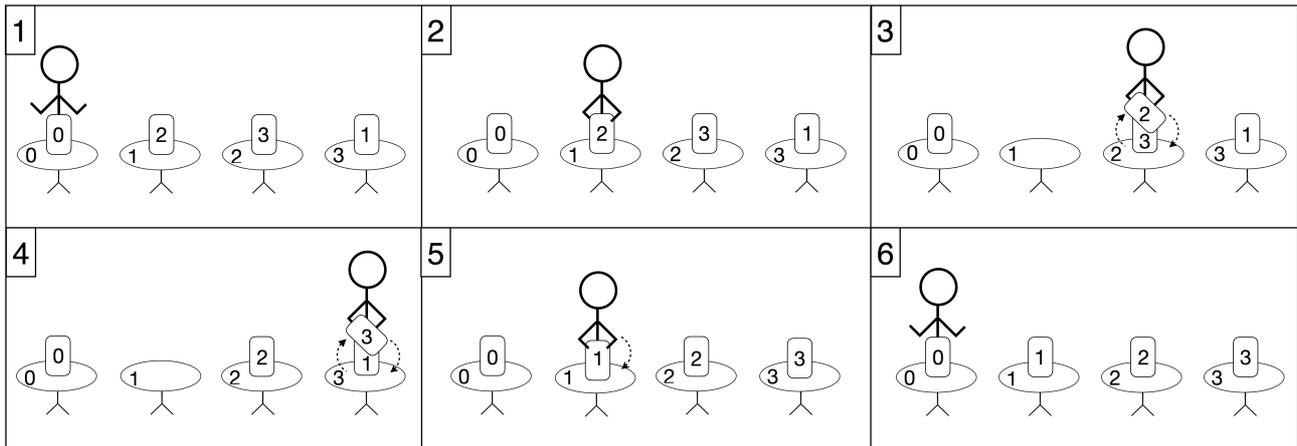
- p adalah sebuah *array* sepanjang n . Buku yang pada mulanya terletak pada meja i harus

dipindahkan oleh Aryan ke meja $p[i]$ (untuk setiap $0 \leq i < n$).

- s adalah nomor meja tempat Aryan berada pada mulanya, dan tempat di mana ia harus berada setelah pengurutan.
- Prosedur ini harus mengembalikan total jarak minimum (dalam meter) Aryan harus berjalan untuk mengurutkan buku-buku tersebut.

Contoh

```
minimum_walk([0, 2, 3, 1], 0)
```



Pada contoh ini, $n = 4$ dan Aryan pada mulanya berada pada meja 0. Ia mengurutkan buku-buku tersebut dengan cara berikut ini:

- Ia berjalan ke meja 1 dan mengambil buku yang terletak pada meja. Buku ini harus diletakkan pada meja 2.
- Kemudian, ia berjalan ke meja 2 dan menukar buku yang sedang ia bawa dengan buku yang terletak pada meja. Buku baru yang sedang ia bawa harus diletakkan pada meja 3.
- Kemudian, ia berjalan ke meja 3 dan menukar buku yang sedang ia bawa dengan buku yang terletak pada meja. Buku baru yang sedang ia bawa harus diletakkan pada meja 1.
- Kemudian, ia berjalan ke meja 1 dan meletakkan buku yang sedang ia bawa pada meja.
- Pada akhirnya, ia berjalan kembali ke meja 0.

Perhatikan bahwa buku pada meja 0 sudah berada pada tempat yang benar, yakni meja 0, sehingga Aryan tidak harus mengambilnya. Total jarak perjalanannya pada solusi ini adalah 6 meter. Ini adalah solusi optimal; maka, prosedur tersebut harus mengembalikan 6.

Batasan

- $1 \leq n \leq 1\,000\,000$
- $0 \leq s \leq n - 1$
- Array p berisi n bilangan bulat berbeda-beda di antara 0 dan $n - 1$, inklusif.

Subsoal

1. (12 poin) $n \leq 4$ dan $s = 0$
2. (10 poin) $n \leq 1000$ dan $s = 0$
3. (28 poin) $s = 0$
4. (20 poin) $n \leq 1000$
5. (30 poin) tidak ada batasan tambahan

Grader contoh

Grader contoh membaca masukan dengan format berikut ini:

- baris 1: n s
- baris 2: $p[0]$ $p[1]$ \dots $p[n - 1]$

Grader contoh mencetak sebuah baris berisi nilai kembalian dari `minimum_walk`.