



## Ancient Books

Biblioteca Națională a Iranului se află în Teheran. Atracția principală a bibliotecii este localizată de-a lungul unui coridor în care se află  $n$  mese, numerotate de la 0 la  $n - 1$  de la stânga la dreapta. Pe fiecare masă se află câte o carte antică scrisă de mână. Cărțile sunt ordonate după vârsta lor, acest lucru îngreunându-i pe vizitatori să caute cărțile după titlu. Așadar, administratorul bibliotecii a decis să sorteze cărțile în ordine alfabetică după titlu.

Aryan, un bibliotecar, este responsabil pentru această sarcină. El a creat o listă  $p$  de lungime  $n$ , care conține numere întregi de la 0 la  $n - 1$ , distincte două câte două. Această listă reprezintă modificările necesare pentru rearanjarea cărților în ordine alfabetică: pentru fiecare  $0 \leq i < n$ , cartea care se află în momentul curent pe masa  $i$  trebuie mutată pe masa  $p[i]$ .

Aryan începe sortarea cărților pornind de la masa  $s$ . După terminarea sortării cărților acesta dorește să se întoarcă la această masă. Având în vedere valoarea cărților, el nu poate avea la el mai mult de o carte în orice moment de timp. Cât timp Aryan sortează cărțile, el va efectua o secvență de acțiuni. Fiecare acțiune este una din următoarele:

- Dacă nu are la el o carte și dacă este o carte pe masa la care se află, el poate lua această carte.
- Dacă are la el o carte și o altă carte este pe masa la care se află, el poate interschimba cele două cărți.
- Dacă are la el o carte și masa la care se află este goală, el poate pune pe masă cartea pe care o are la el.
- El poate merge la orice masă. El poate căra o singură carte.

Pentru toate  $0 \leq i, j \leq n - 1$ , distanța dintre mesele  $i$  și  $j$  este de  $|j - i|$  metri. Sarcina voastră este de a-l ajuta pe Aryan să sorteze cărțile într-un mod în care distanța totală parcursă de el să fie minimă.

## Detalii de implementare

Voi trebuie să implementați următoarea procedură:

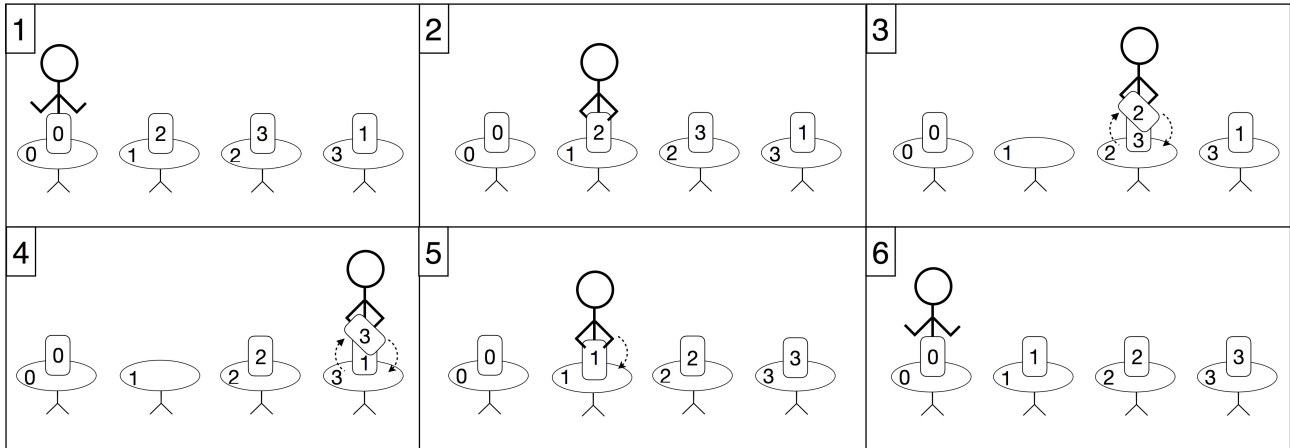
```
int64 minimum_walk(int[] p, int s)
```

- $p$  este un șir de lungime  $n$ . Cartea care se află la început pe masa  $i$  trebuie dusă de Aryan pe masa  $p[i]$  (pentru toate  $0 \leq i < n$ ).
- $s$  reprezintă poziția mesei de unde începe Aryan, și unde trebuie să se afle după sortarea cărților.

- Această procedură trebuie să returneze distanța totală minimă (în metri) parcursă de Aryan.

## Exemplu

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



În acest exemplu,  $n = 4$  și Aryan se află inițial la masa 0. El sortează cărțile după cum urmează:

- Merge la masa 1 și ia cartea de pe masă. Această carte trebuie să ajungă pe masa 2.
- Apoi, merge la masa 2 și interschimbă cartea din mână cu cea de pe masă. Cartea nouă trebuie să ajungă pe masa 3.
- Apoi, merge la masa 3 și interschimbă cartea din mână cu cea de pe masă. Cartea nouă trebuie să ajungă pe masa 1.
- Apoi, merge la masa 1 și pune cartea ce o are la el pe această masă.
- În final, el se întoarce la masa 0.

Se observă că pe masa 0 cartea se află pe poziția corectă, deci Aryan nu trebuie să o mute. Distanța totală parcursă în această soluție este de 6 metri. Aceasta este soluția optimă; așadar procedura trebuie să returneze valoarea 6.

## Restricții și precizări

- $1 \leq n \leq 1\,000\,000$
- $0 \leq s \leq n - 1$
- Șirul  $p$  conține  $n$  numere întregi distincte între 0 și  $n - 1$ , inclusiv.

## Subtask-uri

1. (12 puncte)  $n \leq 4$  și  $s = 0$
2. (10 puncte)  $n \leq 1000$  și  $s = 0$
3. (28 puncte)  $s = 0$
4. (20 puncte)  $n \leq 1000$

5. (30 puncte) fără restricții adiționale

## Evaluator local

Evaluatorul local citește datele de intrare în următorul format:

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

Evaluatorul local afișează o singură linie reprezentând valoarea returnată de `minimum_walk`.