



Drevne Knjige

Grad Teheran je mjesto gdje je dom našla Narodna biblioteka u Iranu. Glavno blago ove biblioteke nalazi se u dugoj dvorani sa nizom od n stolova, označenih brojevima od 0 do $n - 1$, slijeva udesno. Na svakom stolu se nalazi jedna drevna knjiga, vrijedan rukopis. Ove knjige su poređane na osnovu njihove starosti, što otežava posjetiocima da pretražuju knjige po naslovu. Zbog toga je menadžer biblioteke odlučio da knjige sortira po abecednom redosljedu njihovih naslova.

Aryan, bibliotekar, će raditi ovaj posao. Kreirao je listu p dužine n , koja sadrži različite cijele brojeve od 0 do $n - 1$. Ova lista opisuje promjene potrebne za preraspoređivanje knjiga po abecednom redosljedu: za sve $0 \leq i < n$, knjiga koja je trenutno na stolu i treba biti premještena na sto $p[i]$.

Aryan počinje sortiranje knjiga krenuvši od stola s . On mora da se vrati na isti, početni sto nakon završenog posla. Pošto su knjige veoma vrijedne, u svakom trenutku on može nositi najviše jednu knjigu. Dok sortira knjige Aryan će izvršiti niz radnji. Svaka od tih radnji može biti neka od sljedećih:

- Ako ne nosi knjigu a knjiga stoji na stolu do kojeg je on došao, Aryan može uzeti tu knjigu.
- Ako nosi jednu knjigu a postoji neka druga knjiga na stolu do kojeg je došao, on može da zamijeni onu knjigu koju nosi knjigom koja je na stolu.
- Ako nosi knjigu a došao je do praznog stola, on onda može staviti tu knjigu na sto.
- Aryan može doći do bilo kojeg stola. U svakom trenutku može nositi najviše jednu knjigu .

Za sve $0 \leq i, j \leq n - 1$, rastojanje između stolova i i j je tačno $|j - i|$ metara. Vaš zadatak je da pomognete Aryanu da sortira (odnosno ispremješta) knjige na takav način da ukupna udaljenost koju treba preći treba bude minimalna.

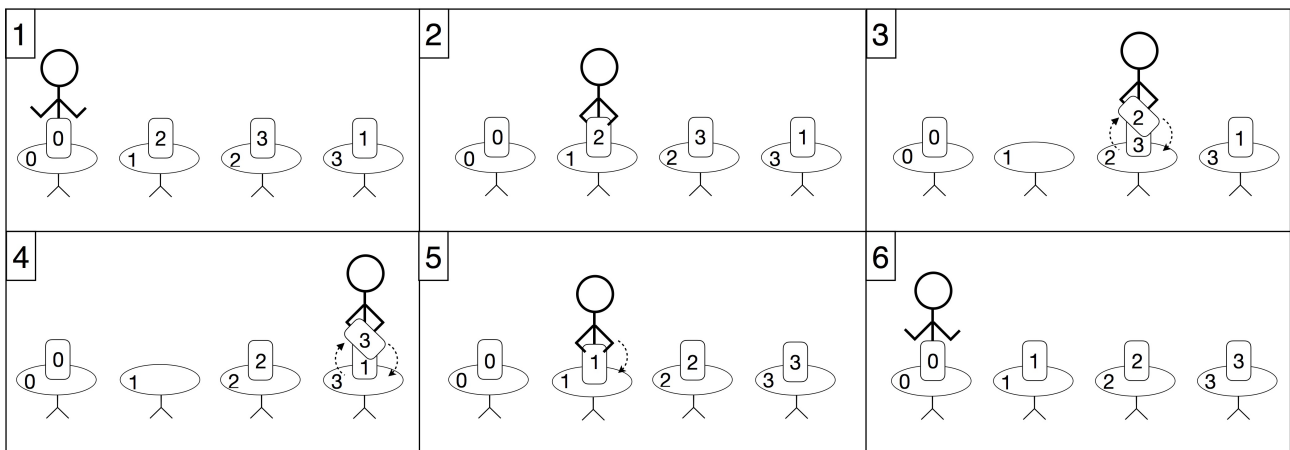
Detalji implementacije

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

- p je niz dužine n . Knjiga koja je bila na početku na stolu i Aryan treba odneti na sto $p[i]$ (za sve $0 \leq i < n$).
- s je oznaka stola sa kojeg Aryan počinje, gdje treba biti nakon što sortira sve knjige.
- Ova procedura treba vratiti najmanju moguću udaljenost (u metrima) koju Aryan mora prepješačiti kako bi sortirao sve knjige.

Primjer

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



U ovom primjeru, $n = 4$ i Aryan se na početku nalazi kod stola 0. Aryan sortira knjige na sljedeći način:

- ode do stola 1 i uzme knjigu koja je na tom stolu. Ovu knjiga treba staviti na sto 2.
- Onda, Aryan ode do stola 2 i zamijeni knjigu koju nosi knjigom koja se nalazi na tom stolu. Nova knjiga koju sada nosi treba staviti na sto 3.
- Nakon toga, Aryan ode do stola 3 i zamijeni knjigu koju nosi knjigom koja se nalazi na tom stolu. Nova knjiga koju sada nosi treba staviti na sto 1.
- Zatim, Aryan ode do stola 1 i spusti knjigu koju nosi na sto.
- Konačno, kako se mora vratiti do početnog stola on prošetava do stola 0.

Primjetimo da je knjiga na stolu 0 već na dobrom mjestu, na stolu 0, tako da Aryan ne mora uzimati knjigu sa tog stola. Ukupna udaljenost koju je prešao u ovom rješenju je 6 metara. Ovo je optimalno rješenje, pa prema tome, procedura treba vratiti 6.

Ograničenja

- $1 \leq n \leq 1\,000\,000$
- $0 \leq s \leq n - 1$
- Niz p sadrži n različitih cijelih brojeva između 0 i $n - 1$, uključivo.

Podzadaci

1. (12 bodova) $n \leq 4$ i $s = 0$
2. (10 bodova) $n \leq 1000$ i $s = 0$
3. (28 bodova) $s = 0$
4. (20 bodova) $n \leq 1000$
5. (30 bodova) nema dodatnih ograničenja

Primjer grejdera

Dati grejder učitava ulazne podatke u sljedećem formatu:

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

Dati grejder ispisuje samo vrijednost koju vraća funkcija `minimum_walk`.