



## LASTGAMLA BÖCKER

TEHRAN HUSERAR IRANS NATIONALBIBLIOTEK. BIBLIOTEKETETS HUVUDSKATT ÄR PLACERAD I EN LÅNG KORRIDOR MED EN RAD BESTÅENDE AV  $n$  BORD, ETIKETTERADE 0 TILL OCH MED  $n - 1$  FRÅN VÄNSTER TILL HÖGER. VARJE BORD INNEHÅLLER EN LASTGAMMAL BOK. BÖCKERNA ÄR ORDNADE EFTER ÅLDER, VILKET GÖR DET SVÅRT FÖR BESÖKARNA ATT HITTA BOKEN DE SÖKER. NATIONALBIBLIOTEKSCHEFEN HAR BESLUTAT ATT BÖCKERNA ISTÄLLET SKA SORTERAS I BOKSTAVSORDNING.

BIBLIOTIKARIEN ARYAN SKA GÖRA JOBBET. HAN HAR SKAPAT EN LISTA  $p$  AV LÄNGD  $n$  SOM INNEHÅLLER PARVIS OLIKA Heltal FRÅN 0 TILL  $n - 1$ . LISTAN BESKRIVER ÄNDRINGARNA SOM BEHÖVS FÖR ATT ARRANGERA BÖCKERNA I BOKSTAVSORDNING: FÖR VARJE  $0 \leq i < n$  SKA BOKEN SOM JUST NU FINNS PÅ PLATS  $i$  FLYTTAS TILL PLATS  $p[i]$ .

ARYAN BÖRJAR SORTERINGEN PÅ BORD  $s$ . HAN VILL TILLBAKA HIT EFTER ATT HA SLUTFÖRT JOBBET. EFTERSOM BÖCKERNA ÄR VÄLDIGT VÄRDEFULLA KAN HAN BÄRA HÖGST EN BOK ÅT GÅNGEN. NÄR HAN SORTERAR BÖCKERNA UTFÖR HAN EN SEKVENNS AV HANDLINGAR, TAGNA AV FYRA MÖJLIGA:

- OM HAN INTE BÄR EN BOK OCH DET FINNS EN BOK PÅ BORDET HAN ÄR VID KAN HAN PLOCKA UPP BOKEN.
- OM HAN BÄR EN BOK OCH DET FINNS EN ANNAN BOK PÅ BORDET HAN ÄR VID KAN HAN PLATS PÅ DE TVÅ BÖCKERNA.
- OM HAN BÄR EN BOK OCH ÄR VID ETT TOMT BORT KAN HAN LÄGGA BOKEN PÅ BORDET.
- HAN KAN GÅ TILL VILKET BORT SOM HELST, MÖJLIGTVIS BÄRANDEÅP EN BOK.

FÖR ALLA  $0 \leq i, j \leq n - 1$  ÄR AVSTÅNDET MELLAN BORD  $i$  OCH  $j$  EXAKT  $|j - i|$  METER. DIN UPPGIFT ÄR ATT HJÄLPA ARYAN ATT SORTERA BÖCKERNA PÅ ETT SÅDANT SÄTT ATT DET TOTALA AVSTÅNDET HAN GÅR BLIR SÅ LITET SOM MÖJLIGT.

## IMPLEMENTATION DETAILS

YOU SHOULD IMPLEMENT THE FOLLOWING PROCEDURE:

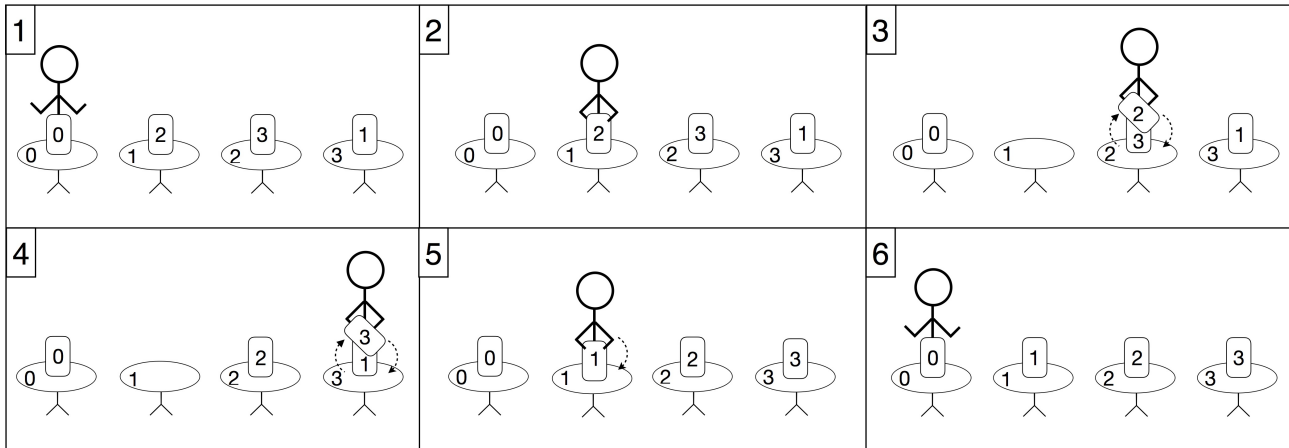
```
int64 minimum_walk(int[] p, int s)
```

- $p$  IS AN ARRAY OF LENGTH  $n$ . THE BOOK THAT IS ON TABLE  $i$  AT THE BEGINNING SHOULD BE TAKEN BY ARYAN TO TABLE  $p[i]$  (FOR ALL  $0 \leq i < n$ ).

- $s$  IS THE LABEL OF THE TABLE WHERE ARYAN IS AT THE BEGINNING, AND WHERE HE SHOULD BE AFTER SORTING THE BOOKS.
- THIS PROCEDURE SHOULD RETURN THE MINIMUM TOTAL DISTANCE (IN METERS) ARYAN HAS TO WALK IN ORDER TO SORT THE BOOKS.

## EXAMPLE

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



IN THIS EXAMPLE,  $n = 4$  AND ARYAN IS AT TABLE 0 AT THE BEGINNING. HE SORTS THE BOOKS AS FOLLOWS:

- WALKS TO TABLE 1 AND PICKS UP THE BOOK LYING ON IT. THIS BOOK SHOULD BE PUT ON TABLE 2.
- THEN, HE WALKS TO TABLE 2 AND SWITCHES THE BOOK HE IS CARRYING WITH THE BOOK ON THE TABLE. THE NEW BOOK HE IS CARRYING SHOULD BE PUT ON TABLE 3.
- THEN, HE WALKS TO TABLE 3 AND SWITCHES THE BOOK HE IS CARRYING WITH THE BOOK ON THE TABLE. THE NEW BOOK HE IS CARRYING SHOULD BE PUT ON TABLE 1.
- THEN, HE WALKS TO TABLE 1 AND PUTS THE BOOK HE IS CARRYING ON THE TABLE.
- FINALLY, HE WALKS BACK TO TABLE 0.

NOTE THAT BOOK ON TABLE 0 IS ALREADY IN THE CORRECT PLACE, TABLE 0, SO ARYAN DOES NOT HAVE TO PICK IT UP. THE TOTAL DISTANCE HE WALKS IN THIS SOLUTION IS 6 METERS. THIS IS THE OPTIMAL SOLUTION; HENCE, THE PROCEDURE SHOULD RETURN 6.

## CONSTRAINTS

- $1 \leq n \leq 1\,000\,000$
- $0 \leq s \leq n - 1$

- ARRAY  $p$  CONTAINS  $n$  DISTINCT INTEGERS BETWEEN 0 AND  $n - 1$ , INCLUSIVE.

## SUBTASKS

1. (12 POINTS)  $n \leq 4$  AND  $s = 0$
2. (10 POINTS)  $n \leq 1000$  AND  $s = 0$
3. (28 POINTS)  $s = 0$
4. (20 POINTS)  $n \leq 1000$
5. (30 POINTS) NO ADDITIONAL CONSTRAINTS

## SAMPLE GRADER

THE SAMPLE GRADER READS THE INPUT IN THE FOLLOWING FORMAT:

- LINE 1:  $n$   $s$
- LINE 2:  $p[0]$   $p[1]$  ...  $p[n - 1]$

THE SAMPLE GRADER PRINTS A SINGLE LINE CONTAINING THE RETURN VALUE OF `minimum_walk`.