



Libros Antiguos

La ciudad de Tehran es hogar de la Librería Nacional de Irán. El principal tesoro de esta librería se encuentra en un largo pasillo con n mesas, numeradas de 0 a $n - 1$ de izquierda a derecha. Cada mesa exhibe un libro antiguo escrito a mano. Estos libros están ordenados según su antigüedad, lo que hace difícil para los visitantes buscar los libros por su título. Por esto, el administrador ha decidido ordenar los libros en orden alfabético según los títulos de los libros.

Aryan, un bibliotecario, hará el trabajo. Él ha creado una lista p de tamaño n , que contiene números enteros diferentes de 0 a $n - 1$. Esta lista describe los cambios necesarios para reordenar los libros a orden alfabético: para todo $0 \leq i < n$, el libro que está en la mesa i debe ser movido a la mesa $p[i]$.

Aryan se encuentra en la mesa s y empieza a ordenar los libros. Él quiere retornar a la misma mesa luego de finalizar su trabajo. Debido a que los libros son muy valiosos, él no puede cargar más de un libro al mismo tiempo. Mientras ordena los libros, Aryan realizará una secuencia de acciones. Cada una de ellas tiene que ser alguna de las siguientes

- Si no está cargando un libro y hay un libro en la mesa en la que él se encuentra, puede levantar ese libro.
- Si está cargando un libro y hay otro libro en la mesa en la que él se encuentra, puede intercambiar el libro que está cargando con el de la mesa.
- Si está cargando un libro y él se encuentra en una mesa vacía, puede poner en la mesa el libro que está cargando.
- Él puede caminar a cualquier mesa. Puede llevar un solo libro mientras lo hace.

Para todo $0 \leq i, j \leq n - 1$, la distancia entre las mesas i y j es precisamente $|j - i|$ metros. Tu tarea es calcular la mínima distancia total que Aryan debe caminar para ordenar todos los libros.

Detalles de implementación

Se debe implementar la siguiente función:

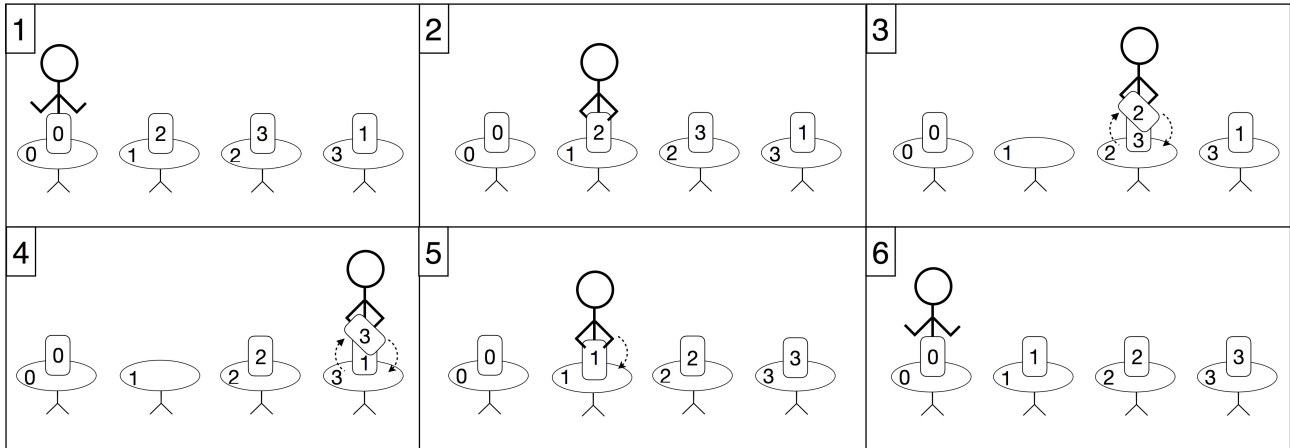
```
int64 minimum_walk(int[] p, int s)
```

- p es un arreglo de tamaño n . El libro que inicialmente se encuentra en la mesa i debe ser llevado por Aryan a la mesa $p[i]$ (para todo $0 \leq i < n$).
- s es el número de la mesa en la que Aryan empieza, y donde debe estar luego de ordenar los libros.
- La función debe retornar la mínima distancia total (en metros) que Aryan debe caminar para

ordenar los libros.

Example

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



En este ejemplo, $n = 4$ y Aryan empieza en la mesa 0. Él ordena los libros de la siguiente manera:

- Él camina a la mesa 1 y recoge el libro que está en ella, Este libro debe ser puesto en la mesa 2.
- Luego, camina a la mesa 2 e intercambia el libro que está cargando con el de la mesa. El nuevo libro debe ser puesto en la mesa 3.
- Luego, camina a la mesa 3 e intercambia el libro que está cargando con el de la mesa. El nuevo libro debe ser puesto en la mesa 1.
- Luego, camina a la mesa 1 y pone el libro que está cargando en la mesa.
- Finalmente, camina de regreso a la mesa 0.

Note que el libro en la mesa 0 ya se encuentra en su lugar correcto, así que Aryan no tiene que levantarlo. La distancia total que debe caminar en esta solución es 6 metros. Esta es la solución óptima; por ende, la función debe retornar 6.

Restricciones

- $1 \leq n \leq 1\,000\,000$
- $0 \leq s \leq n - 1$
- Arreglo p contiene n enteros diferentes entre 0 y $n - 1$, inclusive.

Subtareas

1. (12 puntos) $n \leq 4$ y $s = 0$
2. (10 puntos) $n \leq 1000$ y $s = 0$
3. (28 puntos) $s = 0$

4. (20 puntos) $n \leq 1000$
5. (30 puntos) no hay restricciones adicionales

Evaluador de ejemplo

El evaluador de ejemplo lee la entrada en el siguiente formato:

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

El evaluador de ejemplo imprime una sola línea que contiene el valor que retorna `minimum_walk`.