



## Historische Bücher

Teheran beheimatet Irans Nationalbibliothek. Der Stolz der Bibliothek befindet sich in einem langen Saal mit  $n$  Tischen, nummeriert von 0 bis  $n - 1$  von links nach rechts. Auf jedem Tisch wird ein handgeschriebenes historisches Buch präsentiert. Geordnet sind die Bücher nach Alter, was es für die Besucher erschwert, nach einem bestimmten Buchtitel zu suchen. Die Bibliotheksleitung hat sich deshalb entschieden, die Bücher alphabetisch nach Buchtitel zu sortieren.

Diese Aufgabe wurde dem Bibliothekar Aryan anvertraut. Dazu erstellte er eine Liste  $p$  der Länge  $n$ , bestehend aus unterschiedlichen ganzen Zahlen von 0 bis  $n - 1$ . Sie beschreiben die nötigen Änderungen, um die Bücher in alphabetische Reihenfolge zu bringen. Das Buch auf Tisch  $i$  soll zu Tisch  $p[i]$  gebracht werden, für alle  $0 \leq i < n$ .

Aryan beginnt seinen Auftrag an Tisch  $s$ . Nachdem er seinen Auftrag beendet hat, möchte er an Tisch  $s$  zurückkehren. Um die wertvollen Bücher nicht zu beschädigen, darf er nicht mehr als ein Buch auf einmal tragen. Die Bücher sortiert Aryan mit einer Abfolge von Aktionen der folgenden Art:

- Falls er gerade kein Buch trägt und vor einem Tisch steht, auf dem sich ein Buch befindet, kann er dieses Buch mitnehmen.
- Falls er gerade ein Buch trägt und vor einem Tisch steht, auf dem sich ein anderes Buch befindet, kann er die beiden Bücher vertauschen.
- Falls er gerade ein Buch trägt und vor einem leeren Tisch steht, kann er das Buch auf dem Tisch ablegen.
- Er kann zu einem beliebigen Tisch gehen und dabei entweder ein oder kein Buch tragen.

Die Distanz zwischen den Tischen  $i$  und  $j$  ist genau  $|j - i|$  Meter, für alle  $0 \leq i, j \leq n - 1$ . Deine Aufgabe ist es, die minimale Gesamtdistanz zu berechnen, welche Aryan zurücklegen muss, um alle Bücher zu sortieren.

## Implementationsdetails

Du sollst folgende Funktion implementieren:

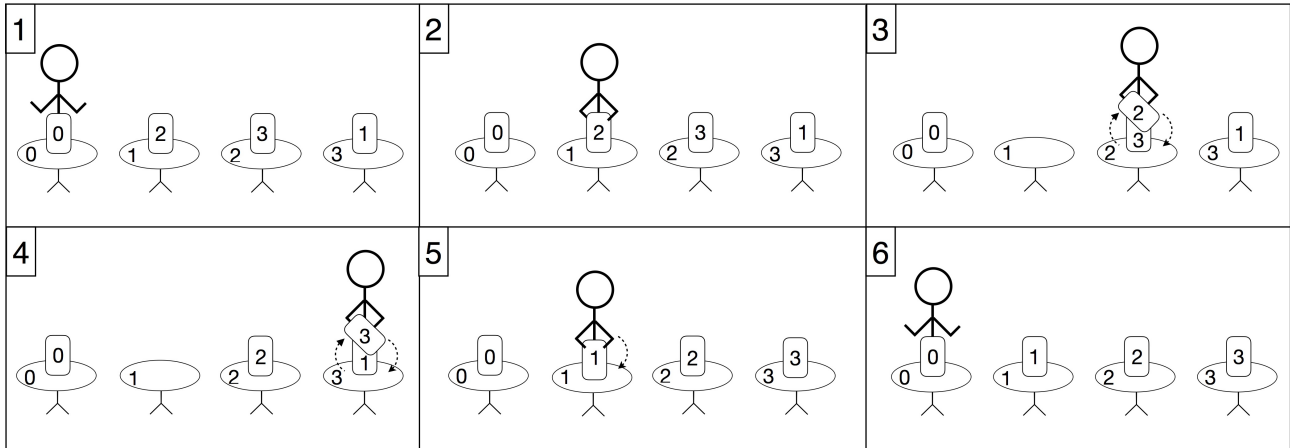
```
int64 minimum_walk(int[] p, int s)
```

- $p$  ist ein Array der Länge  $n$ . Das Buch, welches sich anfangs auf Tisch  $i$  befindet, soll von Aryan zu Tisch  $p[i]$  getragen werden (für alle  $0 \leq i < n$ ).
- $s$  ist die Nummer des Tisches, an welchem sich Aryan zu Beginn befindet und wohin er wieder zurückkehren soll, nachdem er die Bücher sortiert hat.
- Der Rückgabewert der Funktion soll die minimale Gesamtdistanz (in Metern) sein, welche

Aryan zurücklegen muss, um die Bücher zu sortieren.

## Beispiel

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



In diesem Beispiel ist  $n = 4$  und Aryan startet an Tisch 0. Er sortiert die Bücher wie folgt:

- Zuerst geht er zu Tisch 1 und nimmt das Buch an sich. Dieses Buch gehört auf Tisch 2.
- Dann begibt er sich zu Tisch 2 und tauscht das von ihm getragene Buch mit dem auf dem Tisch liegenden aus. Das neue Buch, das er trägt, gehört auf Tisch 3.
- Dann geht er zu Tisch 3 und tauscht das von ihm getragene Buch mit dem auf dem Tisch liegenden aus. Das neue Buch, das er trägt, gehört auf Tisch 1.
- Er begibt sich zu Tisch 1 und legt das Buch, das er trägt, auf dem Tisch ab.
- Zuletzt kehrt er an Tisch 0 zurück.

Bemerke, dass sich das Buch auf Tisch 0 bereits an der erwünschten Position befindet und er es deshalb nicht vom Tisch nehmen muss. Aryan legt in dieser Lösung insgesamt 6 Meter zurück. Diese Lösung ist optimal, deine Funktion soll hier also 6 zurückgeben.

## Beschränkungen

- $1 \leq n \leq 1\,000\,000$
- $0 \leq s \leq n - 1$
- Der Array  $p$  besteht aus  $n$  verschiedenen ganzen Zahlen von 0 bis  $n - 1$  inklusive.

## Subtasks

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

5. (30 Punkte) Keine weiteren Beschränkungen

## Beispiel-Grader

Der Beispiel-Grader liest die Eingabe in folgendem Format:

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

Der Beispiel-Grader gibt in einer einzelnen Zeile den Rückgabewert von `minimum_walk` aus.