

## Ordenamiento

Aizhan tiene una secuencia de  $N$  enteros  $S[0], S[1], \dots, S[N - 1]$ . La secuencia está formada por números distintos entre  $0$  y  $N - 1$ . Ella desea ordenar esta secuencia de manera ascendente, para lo cual intercambiará algunos pares de elementos. Su amigo Ermek también intercambiará algunos pares de elementos - aunque esto no necesariamente ayude al objetivo de Aizhan.

Ermek y Aizhan llevarán a cabo las modificaciones a la secuencia mediante una serie de rondas. En cada ronda, Ermek comienza realizando un intercambio, y luego Aizhan realiza un intercambio. Más precisamente, la persona que realiza un intercambio elige dos índices válidos e intercambia los elementos ubicados en esos índices. Notar que los dos índices no tienen por qué ser distintos entre sí. Si son iguales, la persona en cuestión intercambia un elemento con sí mismo, lo cual no provoca ningún cambio en la secuencia.

Aizhan sabe que a Ermek no le importa ordenar la secuencia  $S$ . Ella también sabe exactamente cuáles son los índices que Ermek va a elegir. Ermek tiene planeado participar en  $M$  rondas de intercambios. Las rondas se numeran desde  $0$  hasta  $M - 1$ . Para cada  $i$  entre  $0$  y  $M - 1$  inclusive, Ermek elegirá los índices  $X[i]$  e  $Y[i]$  en la ronda  $i$ .

Aizhan desea ordenar la secuencia  $S$ . Antes de comenzar cada ronda, si Aizhan ve que la secuencia ya está ordenada ascendentemente, dará por terminado todo el proceso. Dada la secuencia original  $S$  y los índices que elegirá Ermek, se debe encontrar una secuencia de intercambios que Aizhan pueda utilizar para ordenar la secuencia  $S$ . Además, en algunas subtarefas se debe encontrar una secuencia de intercambios lo más corta posible. Se puede asumir que es posible ordenar la secuencia  $S$  en a lo sumo  $M$  rondas.

Notar que si Aizhan ve que la secuencia  $S$  se encuentra ordenada luego de un intercambio realizado por Ermek, puede elegir intercambiar dos índices iguales (por ejemplo,  $0$  y  $0$ ). El resultado será que  $S$  seguirá ordenada al finalizar la ronda, por lo que Aizhan logrará su objetivo. Notar también que si la secuencia inicial  $S$  ya está ordenada, la mínima cantidad de rondas que se necesitan para ordenarla es  $0$ .

### Ejemplo 1

Supongamos que:

- La secuencia inicial es  $S = 4, 3, 2, 1, 0$ .
- Ermek desea realizar  $M = 6$  intercambios.
- Las secuencias  $X$  e  $Y$  que describen los índices que Ermek va a elegir son  $X = 0, 1, 2, 3, 0, 1$  e  $Y = 1, 2, 3, 4, 1, 2$ . En otras palabras, los pares de índices que Ermek planea elegir son  $(0, 1)$ ,  $(1, 2)$ ,  $(2, 3)$ ,  $(3, 4)$ ,  $(0, 1)$ , y  $(1, 2)$ .

En este caso Aizhan puede ordenar la secuencia  $S$  transformándola en  $0, 1, 2, 3, 4$  en tres rondas. Puede hacerlo eligiendo intercambiar los pares de índices  $(0, 4)$ ,  $(1, 3)$ , y finalmente  $(3, 4)$ .

La siguiente tabla muestra cómo Ermek y Aizhan modifican la secuencia.

Ronda	Jugador	Par de índices intercambiados	Secuencia
inicio			4, 3, 2, 1, 0
0	Ermek	(0, 1)	3, 4, 2, 1, 0
0	Aizhan	(0, 4)	0, 4, 2, 1, 3
1	Ermek	(1, 2)	0, 2, 4, 1, 3
1	Aizhan	(1, 3)	0, 1, 4, 2, 3
2	Ermek	(2, 3)	0, 1, 2, 4, 3
2	Aizhan	(3, 4)	0, 1, 2, 3, 4

## Ejemplo 2

Supongamos que:

- La secuencia inicial es  $S = 3, 0, 4, 2, 1$ .
- Ermek desea realizar  $M = 5$  intercambios.
- Los pares de índices que Ermek elegirá son  $(1, 1)$ ,  $(4, 0)$ ,  $(2, 3)$ ,  $(1, 4)$ , y  $(0, 4)$ .

En este caso Aizhan puede ordenar la secuencia  $S$  en tres rondas, por ejemplo eligiendo los pares de índices  $(1, 4)$ ,  $(4, 2)$ , y finalmente  $(2, 2)$ . La siguiente tabla muestra cómo Ermek y Aizhan modifican la secuencia.

Ronda	Jugador	Par de índices intercambiados	Secuencia
inicio			3, 0, 4, 2, 1
0	Ermek	(1, 1)	3, 0, 4, 2, 1
0	Aizhan	(1, 4)	3, 1, 4, 2, 0
1	Ermek	(4, 0)	0, 1, 4, 2, 3
1	Aizhan	(4, 2)	0, 1, 3, 2, 4
2	Ermek	(2, 3)	0, 1, 2, 3, 4
2	Aizhan	(2, 2)	0, 1, 2, 3, 4

## Tarea

Se recibe la secuencia  $S$ , el número  $M$ , y las secuencias de índices  $X$  e  $Y$ . Se debe computar una secuencia de intercambios, que Aizhan pueda utilizar para ordenar la secuencia  $S$ . En las subtareas 5 y 6 la secuencia de intercambios hallada debe tener la mínima longitud posible.

Se debe implementar la función `findSwapPairs`:

- `findSwapPairs(N, S, M, X, Y, P, Q)` — Esta función será llamada por el evaluador exactamente una vez.
  - $N$ : la longitud de la secuencia  $S$ .

- $S$ : un arreglo de enteros con la secuencia  $S$  inicial.
- $M$ : la cantidad de intercambios que Ermek planea realizar.
- $X, Y$ : arreglos de enteros de longitud  $M$ . Dado  $0 \leq i \leq M - 1$ , en la ronda  $i$  Ermek planea intercambiar los números ubicados en los índices  $X[i]$  e  $Y[i]$ .
- $P, Q$ : arreglos de enteros. Se deben utilizar estos arreglos para reportar una posible secuencia de intercambios que Aizhan pueda utilizar para ordenar la secuencia  $S$ . Denotamos con  $R$  la longitud de la secuencia de intercambios encontrada por el programa. Para cada  $i$  entre  $0$  y  $R - 1$  inclusive, los índices que Aizhan elegirá en la ronda  $i$  se deben almacenar en  $P[i]$  y  $Q[i]$ . Se puede asumir que los arreglos  $P$  y  $Q$  tienen reservada suficiente memoria como para almacenar  $M$  elementos cada uno.
- Esta función debe retornar el valor  $R$  (definido en el párrafo anterior).

## Subtareas

subtarea	puntaje	$N$	$M$	restricciones extra para X e Y	requerimientos sobre R
1	8	$1 \leq N \leq 5$	$M = N^2$	$X[i] = Y[i] = 0$ para todo $i$	$R \leq M$
2	12	$1 \leq N \leq 100$	$M = 30N$	$X[i] = Y[i] = 0$ para todo $i$	$R \leq M$
3	16	$1 \leq N \leq 100$	$M = 30N$	$X[i] = 0, Y[i] = 1$ para todo $i$	$R \leq M$
4	18	$1 \leq N \leq 500$	$M = 30N$	ninguna	$R \leq M$
5	20	$6 \leq N \leq 2,000$	$M = 3N$	ninguna	mínimo posible
6	26	$6 \leq N \leq 200,000$	$M = 3N$	ninguna	mínimo posible

Se puede asumir que existe una solución que requiere  $M$  or menos rondas.

### Evaluador de ejemplo

El evaluador de ejemplo lee la entrada del archivo `sorting.in` de acuerdo al siguiente formato:

- línea 1:  $N$
- línea 2:  $S[0] \dots S[N - 1]$
- línea 3:  $M$
- líneas 4, ...,  $M + 3$ :  $X[i] Y[i]$

El evaluador de ejemplo imprime lo siguiente:

- línea 1: El valor  $R$  retornado por `findSwapPairs`
- línea  $2+i$ , para  $0 \leq i < R$ :  $P[i] Q[i]$