



Ordenando

Aizhan tiene una secuencia de N enteros $S[0], S[1], \dots, S[N - 1]$. La secuencia se compone de enteros distintos entre 0 y $N - 1$. Ella está tratando de ordenar esta secuencia de forma ascendente intercambiando pares de elementos. Su amigo Ermek también hará algunos intercambios, no necesariamente para ayudarla.

Ermek y Aizhan modificarán la secuencia en una serie de rondas. En cada ronda, primero Ermek hace un intercambio y después Aizhan hace otro intercambio. Siendo precisos, la persona que hace un intercambio elige dos índices válidos e intercambia los elementos que se encuentran en dichos índices. Nota que los dos índices no tienen que ser distintos: si son iguales, la persona intercambiará un elemento con sí mismo, lo cual no altera la secuencia.

Aizhan sabe que a Ermek no le importa ordenar la secuencia S . Ella además sabe exactamente qué índices Ermek elegirá. Ermek está dispuesto a formar parte en M rondas de intercambios. Las rondas están numeradas de 0 a $M - 1$. Para cada i entre 0 y $M - 1$ inclusive, Ermek elegirá los índices $X[i]$ y $Y[i]$ para intercambiar en la ronda i .

Aizhan quiere ordenar la secuencia S . Antes de cada ronda, si Aizhan ve que la secuencia ya ha sido ordenada en orden ascendente, terminará el proceso. Dada la secuencia original S y los índices que Ermek elegirá, tu tarea consiste en encontrar una secuencia de intercambios que pueda usar Aizhan para ordenar la secuencia S . Adicionalmente, en algunas subtarefas se te pedirá que la secuencia de intercambios sea la más corta posible. Puedes asumir que es posible ordenar la secuencia S en M o menos rondas.

Nota que si Aizhan ve que la secuencia S está ordenada después del intercambio de Ermek, ella puede elegir intercambiar dos índices iguales (por ejemplo 0 y 0). Como resultado la secuencia S estará también ordenada al final de la ronda, de modo que Aizhan cumple su objetivo. Además, nota que si la secuencia inicial S ya se encuentra ordenada, el mínimo número de rondas necesarias para ordenarla es 0 .

Ejemplo 1

Suponiendo que:

- La secuencia inicial es $S = 4, 3, 2, 1, 0$.
- Ermek está dispuesto a hacer $M = 6$ intercambios.
- Las secuencias X y Y que describen los índices que Ermek elegirá son $X = 0, 1, 2, 3, 0, 1$ y $Y = 1, 2, 3, 4, 1, 2$. 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 esta configuración Aizhan puede ordenar la secuencia S en el orden $0, 1, 2, 3, 4$ en tres rondas. Ella puede hacerlo eligiendo los índices $(0, 4)$, $(1, 3)$, y $(3, 4)$.

La siguiente tabla muestra como 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

Suponiendo que:

- La secuencia inicial es $S = 3, 0, 4, 2, 1$.
- Ermek está dispuesto a hacer $M = 5$ intercambios.
- Los pares de índices que Ermek planea elegir son $(1, 1)$, $(4, 0)$, $(2, 3)$, $(1, 4)$, y $(0, 4)$.

En esta configuración Aizhan puede ordenar la secuencia S en tres rondas, por ejemplo eligiendo los pares de índices $(1, 4)$, $(4, 2)$, y $(2, 2)$. La siguiente tabla muestra como 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

Problema

Dada la secuencia S , el número M y las secuencias de índices X y Y . Calcula una secuencia de intercambios que Aizhan pueda usar para ordenar la secuencia S . En las subtareas **5** y **6** la secuencia de intercambios que encuentres deberá ser la más corta posible.

Deberás 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 que contiene la secuencia inicial S .
- M : el número de intercambios que Ermek planea hacer.
- X, Y : arreglos de enteros de longitud M . Para $0 \leq i \leq M - 1$, en la ronda i Ermek planea intercambiar los números en los índices $X[i]$ y $Y[i]$.
- P, Q : Arreglos de enteros. Usa estos arreglos para reportar una posible secuencia de intercambios que Aizhan puede hacer para ordenar la secuencia S . Considera R como la longitud de la secuencia de intercambios que tu programa ha encontrado. Para cada i entre 0 y $R - 1$ inclusive, los índices que Aizhan debe elegir en la ronda i deben ser guardados en $P[i]$ y $Q[i]$. Puedes asumir que los arreglos P y Q ya tienen memoria asignada para almacenar M elementos cada uno de ellos.
- Esta función deberá regresar el valor de R (definido arriba).

Subtareas

subtarea	puntos	N	M	restricciones extra para X, Y	Requerimientos para R
1	8	$1 \leq N \leq 5$	$M = N^2$	$X[i] = Y[i] = 0$ para toda i	$R \leq M$
2	12	$1 \leq N \leq 100$	$M = 30N$	$X[i] = Y[i] = 0$ para toda i	$R \leq M$
3	16	$1 \leq N \leq 100$	$M = 30N$	$X[i] = 0, Y[i] = 1$ para toda 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	El mínimo posible
6	26	$6 \leq N \leq 200,000$	$M = 3N$	Ninguna	El mínimo posible

Puedes asumir que existe una solución que requiere M o menos rondas.

Evaluador de Ejemplo

El evaluador de ejemplo lee la entrada del archivo `sorting.in` en el 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 entregará la siguiente salida:

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