

# Copia de seguridad corrupta

UNICOS

El servidor central de la OIE ha sufrido una leve corrupción de datos en su sistema de copias de seguridad. En uno de los sectores de memoria, existe una secuencia secreta  $a$  de  $2 \cdot n + 1$  enteros con valores entre 1 y  $n$ . Por diseño, cada valor debería estar respaldado apareciendo exactamente 2 veces, pero un *glitch* ha provocado que un único valor aparezca exactamente 3 veces. Tu objetivo como auditor de sistemas es encontrar las 3 posiciones exactas de este dato corrupto.



Para no activar los protocolos de seguridad, no puedes descargar la secuencia completa. En su lugar, deberás utilizar una función de diagnóstico de la API realizando preguntas del siguiente estilo:

1. Elige un entero  $k$  y una secuencia de índices de tamaño  $k$ ,  $s_1, s_2, \dots, s_k$ , de valores **diferentes** entre 1 y  $2 \cdot n + 1$ .
2. Recibirás como respuesta el número de valores en  $\{a_{s_1}, a_{s_2}, \dots, a_{s_k}\}$  que aparecen **exactamente una vez**. A estos valores les llamamos **valores únicos**. Por ejemplo, si los valores  $a_{s_1}, \dots, a_{s_k}$  fueran  $\{2, 1, 2, 3, 2, 3, 6, 7\}$  la respuesta sería 3, ya que 1, 6 y 7 son los únicos valores que aparecen exactamente una vez. El 3 aparece dos veces y el 2 aparece tres veces, por lo que no se tienen en cuenta.

## Interacción

Cada ejecución comenzará con un entero  $T$ , el número de casos a procesar. En cada caso se establecerá una comunicación con el juez según el siguiente protocolo:

- Primero debes leer un entero  $n$ , que indica que la secuencia oculta tiene  $2 \cdot n + 1$  enteros.
- A continuación podrás realizar preguntas. Para preguntar por la secuencia de índices  $s = \{s_1, s_2, \dots, s_k\}$ , debes imprimir `? k s1 s2 ... sk`, donde **los valores de  $s$  tienen que ser diferentes dos a dos**.
- Inmediatamente después de cada pregunta, recibirás un entero: la respuesta del juez a tu pregunta.
- Si lees `-1` después de una pregunta, indica que has superado el número máximo de preguntas o que tu pregunta es inválida. **Tu programa debe terminar inmediatamente** para obtener el veredicto “Respuesta incorrecta” en CMS. De otra forma, tu programa podría obtener veredictos arbitrarios.
- Cuando hayas resuelto el caso de prueba, deberás imprimir los 3 índices especiales (las 3 posiciones del dato corrupto) en cualquier orden utilizando el formato `! i1 i2 i3`.
- Después de haber mostrado el resultado del caso, recibirás un 1 si tu respuesta es correcta y un `-1` si tu respuesta es incorrecta o inválida. Si recibes un 1, puedes pasar a resolver el siguiente caso de prueba (si existe). Pero si recibes un `-1`, **tu programa debe terminar inmediatamente** para obtener el veredicto “Respuesta incorrecta” en CMS. De otra forma, tu programa podría obtener veredictos arbitrarios.

**Recuerda que debes refrescar la salida** cada vez que imprimas datos (usando `cout << endl` o `cout << flush` en C++, `System.out.flush()` en Java, `sys.stdout.flush()` en Python).

Para todos los casos de prueba de una misma ejecución ( $T$ ), dispones de un límite global máximo de  $100 \cdot T$  preguntas. Puedes exceder las 100 preguntas en un caso particular siempre y cuando lo compenses ahorrando preguntas en otros.

La secuencia secreta  $a$  se fija antes de empezar cada caso de prueba, es decir, el juez **no es adaptativo**.

## Subtareas

- (12 puntos)  $T = 1$  y  $n = 3$ . Conseguirás los puntos si realizas no más de 100 preguntas.
- (9 puntos)  $T = 1$  y  $n = 6$ . Conseguirás los puntos si realizas no más de 100 preguntas.
- (79 puntos)  $T = 100$  y en todos los casos  $n = 100$ . Conseguirás los puntos en función de  $f$ , que depende del promedio  $q$  del número de preguntas sobre todos los casos, definida de la siguiente manera:

$$f(q) = \max(0, \min(79, 2 \cdot (63 - q)))$$

Para conseguir todos los puntos, el problema debe ser resuelto en como mucho  $23.5 \cdot T$  preguntas.

**Los casos de este problema han sido generados de manera aleatoria uniforme y para cada subtarea solo existe un único lote de casos.**

Un ejemplo (ficticio) de caso de prueba con  $T = 1, n = 2, a = [1, 2, 2, 1, 2]$  podría ser:

Juez	Concursante	Comentario
1		valor de $T$
2		valor de $n$
	? 3 1 2 3	
1		Entre los valores $a_1 = 1, a_2 = 2$ y $a_3 = 2$ , solo hay un valor único: $a_1$ .
	? 1 4	
1		
	? 3 2 3 5	
0		En $\{a_2, a_3, a_5\}$ no hay ningún valor único.
	! 2 3 5	Las posiciones de $a$ con un valor repetido 3 veces son 2, 3 y 5
1		El caso se ha resuelto correctamente.