

Explicación del Problema: Diverse Team

El problema "Diverse Team" plantea la tarea de formar un equipo de K estudiantes a partir de un grupo de N estudiantes, cumpliendo una condición clave:

todos los integrantes del equipo deben tener ratings distintos en la plataforma Codeforces.

Si esto es posible, debemos imprimir "YES" y los índices de los estudiantes seleccionados. En caso contrario, la salida debe ser "NO".

La entrada está compuesta por:

- Dos enteros N y K .
- Una segunda línea con N ratings, uno por cada estudiante.

Para fines de análisis, se considera una versión modificada donde:

- N puede ser tan grande como 110^8 .
- Cada rating tiene un valor máximo de 10^4 .

Ejemplo

Entrada:

5 3

15 13 15 15 12

Salida esperada:

YES

1 2 5

En este caso, necesitamos seleccionar 3 estudiantes con ratings diferentes.

Los ratings distintos presentes son: 15, 13 y 12, cuyos índices (1-based) son 1, 2 y 5.

Método 1: Enfoque ingenuo (búsqueda manual de repetidos)

Una primera forma de resolver el problema consiste en:

1. Recorrer el arreglo original de ratings.
2. Mantener un arreglo auxiliar donde guardamos los índices de los ratings distintos que encontramos.
3. Para cada nuevo rating, verificar manualmente si ya aparece en el arreglo auxiliar.

Ejemplo sobre el arreglo [15, 13, 15, 15, 12]:

- El 15 inicial se agrega.
- El 13 se agrega porque no coincide con el 15 guardado.
- El siguiente 15 se descarta, ya que ya existe.
- El 15 siguiente también se ignora.
- El 12 es nuevo, así que se agrega.

El resultado final sería: índices 1, 2 y 5.

Problema de esta solución

Requiere comparar elementos repetidamente, lo que genera una complejidad aproximada de: $O(N \times K)$

Esta solución se vuelve impráctica cuando N es muy grande.

Método 2: Solución óptima usando una tabla hash

Para mejorar eficiencia, se usa una tabla hash (o conjunto) donde cada rating queda registrado tan pronto aparece. El procedimiento es:

1. Recorrer el arreglo una sola vez.
2. Cada vez que encontremos un rating que no esté en la tabla hash:
 - Lo agregamos al conjunto.
 - Guardamos su índice.
3. Si alcanzamos K ratings distintos, detenemos el proceso.

Aplicación al ejemplo:

- 15 → no está en la tabla → agregar.
- 13 → no está → agregar.
- 15 → ya está → ignorar.
- 15 → ignorar.
- 12 → no está → agregar.

Al final tenemos los índices 1, 2 y 5, cumpliendo el requisito.

Ventajas

- Solo se recorre el arreglo una vez.
- La verificación de repetidos es instantánea gracias al hash.

Complejidad

$O(N)$

Muchísimo más eficiente para valores grandes de N y K .

Conclusión

El problema "Diverse Team" consiste en seleccionar K estudiantes con ratings diferentes, y la estrategia óptima es utilizar una tabla hash para filtrar repetidos de forma eficiente.

El método ingenuo funciona, pero su complejidad lo hace inapropiado para entradas grandes. En cambio, la versión óptima garantiza un rendimiento lineal, adecuado para los límites del problema.