

Buscar un elemento en un vector rotado

Consideramos un vector $V[N]$ de números enteros, cuyo valores se han obtenido aplicando una rotación sobre un vector ordenado en orden estrictamente creciente. Implementa un algoritmo que decida si un valor pertenece al vector con una complejidad $O(\log(n))$. El número de elementos sobre los que se aplica la rotación para obtener el vector de entrada es un valor entre 0 y N y no se conoce.

Por ejemplo, un posible vector de entrada sería el vector 8 12 16 20 22 2 4 obtenido desplazando los dos primeros elementos del vector 2 4 8 12 16 20 22 al final del mismo.

Requisitos de implementación.

Se debe implementar una función recursiva (resolver) que dado el vector, con los datos de entrada ya leídos, y un valor, decida si el valor pertenece al vector en tiempo logarítmico respecto al número de elementos del vector. Se pueden utilizar más parámetros o devolver más valores como retorno de la función si se considera necesario.

Debe justificarse el coste de la función escribiendo la recurrencia empleada y realizando el desplegado de la misma.

Entrada

La entrada consta de una serie de casos de prueba. Cada caso de prueba consta de dos líneas. En la primera se indica el número de elementos del vector y el valor a buscar y en la segunda los valores del vector. La entrada finaliza con el valor -1.

El número de elementos del vector es mayor o igual que cero.

Salida

Para cada caso de prueba se escribe en una línea diferente *SI* si el elemento pertenece al vector y *NO* si el elemento no pertenece al vector.

Entrada de ejemplo

```
7 12
8 12 16 20 22 2 4
7 21
8 12 16 20 22 2 4
7 4
8 12 16 20 22 2 4
8 20
20 22 2 3 4 8 12 16
8 2
20 22 2 3 4 8 12 16
8 15
20 22 2 3 4 8 12 16
-1
```

Salida de ejemplo

```
SI
NO
SI
SI
SI
NO
```

Autor: Isabel Pita