

# El mayor valor perdido

Dado un vector de números enteros ( $n > 1$ ), ordenados en orden estrictamente creciente, encuentra el valor mayor que falta en el vector.

Por ejemplo, dado el vector 5, 8, 11, 14, 17. El valor mayor que falta es el 16, porque es el mayor valor menor que 17 que no se encuentra en el vector.

Implementa una función recursiva eficiente que resuelva el problema en  $O(\log n)$ . Escribe como comentarios la ecuación de recurrencia que determina el coste del algoritmo.

## 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 ( $1 < n \leq 10.000$ ) y en la segunda sus valores. Se garantiza que en cada vector falta al menos un valor.

La entrada de datos termina con un valor cero que no debe procesarse.

## Salida

Para cada caso de prueba el programa escribirá el valor mayor que falta en el vector en una línea distinta.

## Entrada de ejemplo

```
5
5 8 11 14 17
4
2 3 5 6
8
2 3 6 8 10 14 16 17
6
1 4 5 6 7 8
0
```

## Salida de ejemplo

```
16
4
15
3
```

## Nota

Este ejercicio debe verse en el contexto de la asignatura de Fundamentos de Algoritmia (FAL), FDI-UCM. Por tanto *no* vale cualquier solución, sino sólo aquellas que utilicen los conceptos de la asignatura. Es muy posible que se den aclaraciones adicionales en clase a este respecto.

Enunciado adaptado a partir de un problema con el mismo nombre de la profesora **Isabel Pita**.