

**2. (3 puntos)** Una *intromisión* en un vector de enteros **sin elementos repetidos** consiste en insertar en este vector un nuevo valor **que no aparece en el mismo** (el *valor intruso*).

Debes desarrollar un algoritmo “divide y vencerás” eficiente que, dados dos vectores `int a[n]` e `int b[n+1]`, donde el vector **b** representa el resultado de llevar a cabo una *intromisión* en el vector **a**, devuelva el *valor intruso*. Debes, así mismo, determinar justificadamente el coste de este algoritmo, planteando y resolviendo las recurrencias apropiadas.

La implementación deberá ir acompañada de un programa de prueba, que lea desde la entrada estandar casos de prueba, los ejecute, e imprima por la salida estándar el resultado. Cada caso de prueba consiste en tres líneas:

- La primera línea indica el tamaño **n** del vector **a** ( $n \leq 10000$ ).
- La segunda línea contiene los valores del vector **a** (**n** valores).
- La tercera línea contiene los valores del vector **b** (**n** + 1 valores).

La lista de casos de prueba termina con un -1.

Para cada caso de prueba leído se escribirá el *valor intruso* devuelto por el algoritmo.

A continuación se muestra un ejemplo de entrada/salida:

Entrada	Salida
5	10
2 1 3 5 4	5
2 10 1 3 5 4	
4	
1 3 2 4	
1 3 2 4 5	
-1	