

Desaparecidos

En una secuencia en la que inicialmente teníamos varios números consecutivos, nos han desaparecido algunos. ¿Cuál es el número más grande que podemos asegurar que ha desaparecido?

Por ejemplo, en la secuencia [5 8 11 14 17], el valor más grande que echamos en falta es el 16.

Diseña un algoritmo recursivo eficiente que, dado un vector de $n \geq 2$ elementos que contiene una secuencia como la descrita, escriba el número más grande que falta en la misma. Debes, así mismo, determinar la complejidad de dicho algoritmo, planteando y resolviendo las recurrencias apropiadas.

Trabajo a realizar

Debe diseñarse el algoritmo pedido, completando los apartados indicados entre comentarios en el archivo `plantilla.cpp` que se proporciona como apoyo. Debe implementarse, además, el algoritmo. El punto de entrada al mismo será la función `num_perdido`. Si se considera necesario, deberá definirse e implementarse una generalización adecuada, y definir el algoritmo pedido como una inmersión de dicha generalización. Aparte de llevar a cabo el diseño del algoritmo, e implementar el mismo, deberás determinar justificadamente su complejidad.

Programa de prueba

Se proporcionan un programa de prueba que lee por la entrada estándar casos de prueba, los resuelve invocando a `num_perdido`, e imprime los resultados. Cada caso de prueba consta de 2 líneas:

- La primera contiene la longitud de la secuencia.
- La segunda contiene, en orden, los elementos de la secuencia.

La lista de casos de prueba termina con un -1. A continuación se muestra un ejemplo de E/S:

Entrada	Salida
5	16
5 8 11 14 17	7
5	
3 8 9 10 11	
-1	

El archivo completo debe entregarse a través del juez en línea de la asignatura.