

# Implementación misteriosa

Implementa la función que cumpla la postcondición siguiente. En el envío que hagas indica, además, la precondition, cota y complejidad del algoritmo. No hace falta que indices el invariante.

```
// { Pre: [completar] }  
int misterio(int v[], int n) /* return ret */  
// { Post: ret = max a, b : 0 <= a < b < n : v[a] + v[b] }
```

## Entrada

La entrada comienza con un número que indica el número de casos de prueba que vendrá a continuación.

Por cada caso de prueba hay dos líneas. La primera indica el número de elementos (hasta 200.000) y el segundo contiene cada elemento del vector separado por un espacio. Cada valor nunca superará los 100.000 en valor absoluto.

## Salida

Por cada caso de prueba se escribirá el valor devuelto por la función.

## Entrada de ejemplo

```
2  
2  
4 5  
4  
1 2 3 0
```

## Salida de ejemplo

```
9  
5
```

## Nota

Este ejercicio debe verse en el contexto de la asignatura de Fundamentos de Algoritmia (FAL), FDI-UCM 2023/2024 (prof. Marco Antonio Gómez Martín). Por tanto *no* vale cualquier solución, sino sólo aquellas que utilicen los conceptos de FAL. Es muy posible que se den aclaraciones adicionales en clase a este respecto.