Find Element that Appears Once

Given a sorted array in which all elements appear twice (one after one) and one element appears only once. It’s required to find this element (i.e. element that appears once)

Design an **efficient algorithm** that you can use for solving this problem.

Input: **Already Implemented**

The first line of input is an integer T (T < 30), that indicates the number of test cases. Each case consists of array items

Output: **Already Implemented**

A single integer that represents the element that appears once in the given array.

Function: **Implement it!**

public static int FindUniqueElement(int[] arr, int N)

It takes one sorted array (***which all elements appear twice except one***), it shall return the unique element in arr.

ElementOnceAppearence.cs includes this method.

# Examples

|  |  |  |
| --- | --- | --- |
| **#** | **Input Array** | **Output** |
| **1** | 1, 1, 3, 3, 4, 5, 5, 7, 7, 8, 8 | 4 |
| **2** | 1, 1, 3, 3, 4, 4, 5, 5, 7, 7, 8 | 8 |
| **3** | 1, 1, 2, 4, 4, 5, 5, 6, 6 | 2 |
| **4** | 4, 10, 10, 11, 11 | 4 |

# C# Help

## Creating 1D array

int [] array1D = new int [size]

## Creating 2D array

int [,] array2D = new int [size1, size2]

## Getting the size of 1D array

int size = array1D.GetLength(0);

## Getting the size of 2D array

int size1 = array2D.GetLength(0);

int size2 = array2D.GetLength(1);

## Sorting single array

Sort the given array "items" in ascending order

Array.Sort(items);

## Sorting parallel arrays

Sort the first array "master" and re-order the 2nd array "slave" according to this sorting

Array.Sort(master, slave);