A School's Quiz VI

Description

The math teacher in a primary school wrote on board a list of 'K' positive integers. The teacher gave the students another number ‘N’ and asked the students to collect from board the **TOTAL NUMBER of combinations IF EXIST** such that the sum of each combination equals to ‘N’. If no elements sum-up to N, then the output shall be ZERO.

Note:

- Array of K positive integers is **1 BASED**
- **They can use the same integer ONLY ONE TIME**.

REQUIRED: Determine the efficient algorithm that the winner used to find the **TOTAL NUMBER of different ways that numbers can be combined** from **K** to be equal to ‘N’.

Complexity

Your algorithm should take **polynomial time.**

Function: Implement it!

```
int RequiredFunction(int N, int[] numbers)
```

PROBLEM_CLASS.cs includes this method.

Example

1. Numbers = [3, 7], N = 5 -> 0 combinations.
2. Numbers = [1, 2, 3], N = 6 -> 1 combination (**1+2+3**).
3. Numbers = [2, 13, 6, 8, 4], N = 10 -> 2 combinations (**2+8**, **6+4**).

C# Help

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);
```

Creating 1D array

```
int [] array1D = new int [size]
```

Creating 2D array

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

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);
```