SORTING COMPARISON #2

Description

Given an array A of N numbers. It's desired to sort the elements of this array using two sorting algorithms and compare them:

1. Quick sort algorithm.
2. Quick insertion sort algorithm. The quick insertion sort works as the ordinary quick sort **till the array size becomes less than or equal a specific value (threshold),** then simply stop the quicksort for such sub-arrays, and use insertion sort on the entire array.

Complexity

Complexity of your algorithm should be upper-bounded by **O(N log(N))**

Function: Implement it!

```
static float[] RequiredFunction(float[] numbers, int N, int threshold)
```

`PROBLEM_CLASS.cs` includes this method.

Where:

- `numbers`: An array of N numbers
- `N`: array size
- `threshold`: value at which quick sort will stop in case of the quick insertion sort.
- **return:** sorted array.

Example

I	Input	Output
1	N = 4, thresh = 2 Numbers: 3.1 2.2 1.6 4.4 [Quick Insertion: Quick sort starts with size 4 and when the array size ≤ 2, quick sort stopped & insertion sort is applied on entire array]	1.6 2.2 3.1 4.4
2	N = 8, thresh = 6 Numbers: 21 22 23 26 30 31 33 50 [Quick Insertion: Quick sort starts with size 8 and when the array size ≤ 6, quick sort stopped & insertion sort is applied on entire array]	21 22 23 26 30 31 33 50
3	N = 10, thresh = 1 Numbers: 100 90 80 70 60 50 40 30 20 10 [Quick sort only on an array of size 10]	10 20 30 40 50 60 70 80 90 100

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]
```