Matrix Multiplication

Description

Given TWO square matrices of size ***N***×***N***, implement an **efficient algorithm** based on **Strassen's** method to multiply them?

NOTE:

- ***N*** is power of 2 (i.e. 2, 4, 8, 16, 32… 2^i)

Complexity

The complexity of your algorithm should be **less than O(N^3)**

Evaluation

Sample Cases (Correctness)	**UNSEEN Large Cases** (Efficiency)	**Total**
2 Marks	6 Marks	8 MARKS

Bonus & Competition#2

	Criteria	**BONUS**
Vs. Naïve (on Large Cases)	**Just Faster**	+1 Mark
	1x Faster	+3 Marks
	[N]x Faster	+**[N]x2** Marks
TOP5	Correct & Speed	2~4 Marks

Function: **Implement it!**

```
static public int[,] MatrixMultiply(int[,] M1, int[,] M2, int N)
```

MatrixMultiplication.cs includes this method.

Examples

<table>
<tr><th colspan="2">EX#1</th><th colspan="2">EX#2</th></tr>
<tr>
<td>M1:<table><tr><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table></td>
<td>M2:<table><tr><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table></td>
<td>M1:<table><tr><td>1</td><td>-1</td><td>1</td><td>-1</td></tr><tr><td>1</td><td>-1</td><td>1</td><td>-1</td></tr><tr><td>1</td><td>-1</td><td>1</td><td>-1</td></tr><tr><td>1</td><td>-1</td><td>1</td><td>-1</td></tr></table></td>
<td>M2:<table><tr><td>-1</td><td>1</td><td>-1</td><td>1</td></tr><tr><td>-1</td><td>1</td><td>-1</td><td>1</td></tr><tr><td>-1</td><td>1</td><td>-1</td><td>1</td></tr><tr><td>-1</td><td>1</td><td>-1</td><td>1</td></tr></table></td>
</tr>
<tr>
<td colspan="2">Res:<table><tr><td>2</td><td>1</td></tr><tr><td>1</td><td>1</td></tr></table></td>
<td colspan="2">Res:<table><tr><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td></tr></table></td>
</tr>
</table>

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);
```