**7 kyu**

**Simple Fun #9: Array Packing**

8591% of 7621 of210[myjinxin2015](https://www.codewars.com/users/myjinxin2015)

C#

* [TRAIN AGAIN](https://www.codewars.com/kata/simple-fun-number-9-array-packing/train/csharp)
* [NEXT KATA](https://www.codewars.com/trainer/csharp)

Details

[Solutions](https://www.codewars.com/kata/simple-fun-number-9-array-packing/solutions/csharp)

[Discourse (18)](https://www.codewars.com/kata/simple-fun-number-9-array-packing/discuss/csharp)

* Add to Collection
* |
* Share this kata:

**Task**

You are given an array of up to four non-negative integers, each less than 256.

Your task is to pack these integers into one number M in the following way:

The first element of the array occupies the first 8 bits of M;

The second element occupies next 8 bits, and so on.

Return the obtained integer M as unsigned integer.

* Note:

the phrase "first bits of M" refers to the least significant bits of M - the right-most bits of an integer. For further clarification see the following example.

**Example**

For a = [24, 85, 0], the output should be 21784

An array [24, 85, 0] looks like [00011000, 01010101, 00000000] in binary.

After packing these into one number we get 00000000 01010101 00011000 (spaces are placed for convenience), which equals to 21784.

**Input/Output**

* [input] integer array a

Constraints: 1 ≤ a.length ≤ 4 and 0 ≤ a[i] < 256

* [output] an unsigned integer

**More Challenge**

* Are you a One-Liner? Please try to complete the kata in one line(no test for it) ;-)

<https://www.codewars.com/kata/simple-fun-number-9-array-packing/csharp>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApp1

{

class Program

{

public static int ArrayPacking(int[] a)

{

//coding and coding..

string[] bin = new string[a.Length];

//List<string> bin = new List<string>();

for(int i = a.Length-1; i>=0; i--)

{

string pack = Convert.ToString(a[i], 2);

//bin.Add( new string('0', 8 - pack.Length) + pack);

bin[a.Length - i - 1] = new string('0', 8 - pack.Length) + pack;

}

return Convert.ToInt32(string.Join("", bin), 2);

}

static void Main(string[] args)

{

// For a = [24, 85, 0],

// the output should be 21784

int[] a = { 24, 85, 0 };

Console.WriteLine(ArrayPacking(a));

Console.ReadLine();

}

}

}