**Tribonacci Sequence**

3422685% of 1,599912 of12,256[GiacomoSorbi](http://www.codewars.com/users/GiacomoSorbi)[12 Issues Reported](http://www.codewars.com/kata/556deca17c58da83c00002db/discuss#label-issue)

* C#
* Mono 4.2.3
  + VIM
  + EMACS

Instructions

Output

* Well met with Fibonacci bigger brother, AKA Tribonacci.

As the name may already reveal, it works basically like a Fibonacci, but summing the last 3 (instead of 2) numbers of the sequence to generate the next. And, worse part of it, regrettably I won't get to hear non-native Italian speakers trying to pronounce it :(

So, if we are to start our Tribonacci sequence with [1,1,1] as a starting input (AKA *signature*), we have this sequence:

[1,1,1,3,5,9,17,31,...]

But what if we started with [0,0,1] as a signature? As starting with [0,1] instead of [1,1] basically *shifts* the common Fibonacci sequence by once place, you may be tempted to think that we would get the same sequence shifted by 2 places, but that is not the case and we would get:

[0,0,1,1,2,4,7,13,24,...]

Well, you may have guessed it by now, but to be clear: you need to create a fibonacci function that given a **signature** array/list, returns **the first n elements - signature included** of the so seeded sequence.

Signature will always contain 3 numbers; n will always be a non-negative number; if n==0, then return an empty array and be ready for anything else which is not clearly specified ;)

If you enjoyed this kata more advanced and generalized version of it can be found in the [Xbonacci kata](http://www.codewars.com/kata/fibonacci-tribonacci-and-friends" \o "Xbonacci sequence" \t "_blank)

*[Personal thanks to Professor [Jim Fowler](https://www.coursera.org/instructor/jimfowler" \o "Jim Fowler" \t "_blank) on Coursera for his awesome classes that I really recommend to any math enthusiast and for showing me this mathematical curiosity too with his usual contagious passion :)]*

FUNDAMENTALS

SEQUENCES

ARRAYS

ARITHMETIC

MATHEMATICS

ALGORITHMS

NUMBERS

LISTS

DATA STRUCTURES

<http://www.codewars.com/kata/tribonacci-sequence/train/csharp>

public static double[] Tribonacci(double[] signature, int n)

{

// hackonacci me

if (n == 0) return new double[] {0.0 };

//Array.Sort(signature);

double[] fib = new double[n];

for (int i = 0; i < fib.Length && i < signature.Length; i++)

{

fib[i] = signature[i];

}

for (int i = signature.Length; i < n; i++)

{

double sum = 0;

if (i >= 1)

{

sum += fib[i - 1];

}

if (i >= 2)

{

sum += fib[i - 2];

}

if (i >= 3)

{

sum += fib[i - 3];

}

fib[i] = (sum);

//fib.Add(fib[i - 1] + fib[i - 2] + fib[i - 3]);

}

return fib;

}