Set A

We'll say that an element in an array is "alone" if there are values before and after it, and those values are different from it. Return a version of the given array where every instance of the given value which is alone is replaced by whichever value to its left or right is larger.

notAlone([1, 2, 3], 2) → [1, 3, 3]  
notAlone([1, 2, 3, 2, 5, 2], 2) → [1, 3, 3, 5, 5, 2]  
notAlone([3, 4], 3) → [3, 4]

public int[] notAlone(int[] nums, int val) {

}

Set B

Consider the series of numbers beginning at **start** and running up to but not including **end**, so for example start=1 and end=5 gives the series 1, 2, 3, 4. Return a new String[] array containing the string form of these numbers, except for multiples of 3, use "Fizz" instead of the number, for multiples of 5 use "Buzz", and for multiples of both 3 and 5 use "FizzBuzz". In Java, String.valueOf(xxx) will make the String form of an int or other type.

fizzBuzz(1, 6) → ["1", "2", "Fizz", "4", "Buzz"]  
fizzBuzz(1, 8) → ["1", "2", "Fizz", "4", "Buzz", "Fizz", "7"]  
fizzBuzz(1, 11) → ["1", "2", "Fizz", "4", "Buzz", "Fizz", "7", "8", "Fizz", "Buzz"]

public String[] fizzBuzz(int start, int end) {

}