## Setting up a class for searching, sorting algorithms

**The goals for this exercise** are to

1. Prepare a 'utility method' that will be used in next week's lesson, including (potentially) in future in-class work time on searching and sorting.
2. Review how to pass an array as a parameter to a method
3. *For this exercise, you will do neither searching, nor sorting.*

 This has been covered in previous lessons, it's possible to pass an array as an argument to a method. You should practice that by creating a method that will print out all the elements within an array.

Create a class named **SearchingAndSorting**, and within it, create a method that does this printing. (Go back and review the work from prior lessons, if you need help remembering how to do this.)

class SearchingAndSorting

{

 public void printArray(int[] nums)

 {

 Console.WriteLine("The array is " + nums.Length + "integers long!");

 // Your code goes here, to print out the array!

 }

}

Also remember that in C#, unlike C++ or C, arrays know how long they are. For example, in the above example code, you can get the length by using the Length property. If you need to write a loop (or anything else) that needs to know the length of the array, use this property (DON'T simply write the number into the code, and don't pass the length as a separate parameter).

Add code to the Main method of the program, and try printing out a couple of arrays (which you will also need to created) so that you’ve tested out your method, and you feel confident that it works well.

**What you need to do for this exercise:**

1. Write a method to print out the contents of an integer array, as described above.
	1. This should be a really really easy exercise ☺
2. The following is NOT required (which is why it’s crossed out).
I may want to add it back in a future quarter, though, which is why it’s still here. ~~Test your method by adding code to Test\_Print\_Array.RunExercise(), so that you can ‘try out’ your method a couple of times.~~
	1. ~~Note that this should be a very easy method to implement. The purpose of making you do this is to have it ready for the rest of the and out-of-class exercises in this week’s PCEs.~~
	2. ~~Test\_Print\_Array is located in the~~ **~~PCE\_StudentCode~~** ~~project, in the Student\_Answers.cs file. In order to run the RunExericse method, you may need to uncomment some of the lines in the Program.cs file, and/or set the~~ **~~03\_PCE\_StudentCode~~** ~~project to be the startup project by right-clicking on that other project, and selecting the “Set As Startup Project” menu option.~~
	3. ~~Note that the printArray routine will be used by some of the unit tests, so if you don’t implement this, those other unit tests won’t tell you as much information as they might.~~