

COSC-111 INTRODUCTION TO COMPUTER SCIENCE I

LAB 06: ARRAYS

Due: Friday, March 29, 8.59am

1 Setup

Make a new directory for this lab, change into that directory, and copy a java file:

```
$ mkdir lab06
$ cd lab06
$ wget -nv -O Arrays.java http://bit.ly/111s19L06Fs
```

The file `Arrays.java` currently contains two methods. Each method takes an `int` array as its parameter and fills the array with `ints`. The `fillRandom` method does this by choosing a random number between 0 and 9 (inclusive) for each array element; the `fillKeyboard` method does this by repeatedly prompting the user to enter the next element. These methods are there to help you test the additional methods that you will write in this lab.

2 Practice Using Arrays

Your job is to write six new methods that manipulate arrays in different ways. After writing each method, you will probably want to add some code to the main method that tests to make sure your methods are working properly.

1. Write a method called `print` that, given an `int` array, prints the contents of that array. The elements should all print on the same line, with spaces in between them, and with a line break after the last element prints. For example:

```
3 6 9 8 2 6 3 5
[mriondato@romulus lab06]$
```

2. Write a method called `fillFibonacci` that, given an `int` array called `myArray`, fills that array with the first `myArray.length` Fibonacci numbers (recall that the Fibonacci sequence is 1, 1, 2, 3, 5, 8, ...: the i th Fibonacci number is obtained by adding the $i - 1$ st and $i - 2$ nd Fibonacci numbers).

3. Write a method called `addAllElements` that, given an `int` array, adds up the values of all elements stored in the array and returns the total.

4. Write a method called `average` that, given a `int` array, computes the average of all elements stored in the array and returns the result. (Hint: use your `addAllElements` method!)

5. Write a method called `countElements` that, given a `int` array, counts the number of times each number from 0 to 9 appears in the array and prints the results. For example, if the input array is:

```
3 6 9 8 2 6 3 5
```

then the method should print:

```
0 0 1 2 0 1 1 0 1 1
```

6. Write a method called `reverse` that, given a `int` array, reverses the order of the elements in the array. For example, if the input array is:

```
3 6 9 13 2 6 3 5
```

then after running the method the array should contain:

```
5 3 6 2 13 9 6 3
```

Your method should *not* print the reversed array.

3 Submit your work

Submit your modified `Arrays.java` using either the submission web site or the `cssubmit` command.