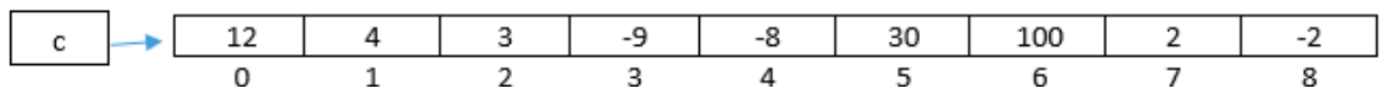
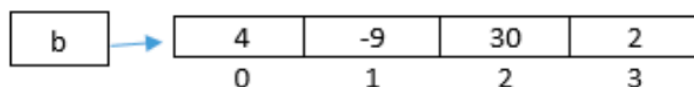
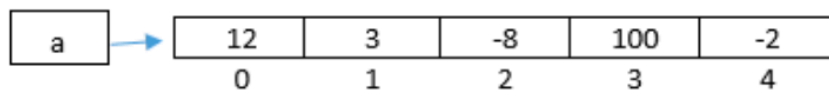


CPS 210 — Lab 10

PROBLEMS:

1. Declare an array of size 10. Fill the array with the values 1.7, 2.25, 3.45, -5.0, 8.3, -13.77, 21.2, 34, 55.67, -89.9.
 - a. Display the array by creating a method, **printArray**.
 - b. Create a method that finds the sum of an array. Try to first do this without looking at my notes.
 - c. Create a method that finds the average of an array. Utilize the sum method you just created. Try to first do this without looking at my notes.
2. Create a method that counts how many numbers are divisible by 3 in an **int** array.
 - a. Create an overloaded method that finds how many numbers are divisible by 3 in a **double** array.
 - b. In the main method, test out both methods by creating an integer array and a double array with numbers of your choosing.
3. Create a method that creates a **copy** of an int array. Use one of the previous int arrays you created to call and test your method.
4. Create method that returns a reversed copy of an int array.
 - **Hint:** Use the process for reversing the array (remember when we printed an array backwards in class?) combined with the process for creating a copy of an array.

5. Write a program that reads in an integer value for the size of an array, say n . Now have the program read in a value for each space in the new integer array of size n . Your choice if you want to use a method or not.
6. Create the array: `double[] grades = {98.6, 78.2, 56.9};`
Suppose these represent test grades for a student. The professor decides to give an extra exam. Write code to create a larger array to hold the old grades plus the new exam grade of 88.8.
 - **Hint:** This is similar to making a copy of an array, but you need to make the 'copy' larger by one.
7. Create an array: `int[] a = {56, 9, 17, 6, 2, 0, 1, 199, 256, 94};`
 - a. Create an array to hold the even values of a .
 - **Hint:** First count how many even numbers there are so you know what size to make the new array.
 - b. Create an array to hold the odd values of a .
8. This one is a bit challenging so don't get discouraged if it takes a few attempts. Try figuring out the loops and logic of the program on paper first. Think about what index in each array you are going from and to.
 - a. Given the following two arrays a , and b , create one array, c , by alternating between the two. You may assume that array a will always be one longer than b .

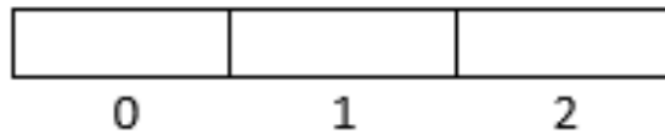


9. Write a program to count:

- How many numbers are even
- How many numbers are divisible by 3 and 7
- How many numbers are divisible by 5 or divisible by 8

in the interval $[0,1000]$.

- **Hint:** you can use an array of size 3 to hold the counts for each bullet point:



Where index 0 holds the count for the number of evens, index 1 holds the count of how many numbers are divisible by 3 and 7 and index 2 holds the count of how many numbers are divisible by 5 or 8.

10. Create an account at <https://leetcode.com/>. Complete the following two questions in LeetCode's text editor, not your own. If you completed the question correctly, you should be able to submit it on LeetCode without errors.

a. <https://leetcode.com/problems/build-array-from-permutation/>

b. <https://leetcode.com/problems/concatenation-of-array/>

c. When you have time, try the other questions on Arrays. Remember, this is good practice for understanding Arrays as well as good practice for potential technical interviews. Be sure to check out the discussion tab to see how other programmers solved the problem: <https://leetcode.com/tag/array/>