CS 211, SECTIONS 222/226

LAB TASK 11

DUE IN LAB TODAY

The objective of this lab is to practice and demonstrate the ability to apply the concepts of inheritance, polymorphism, and abstraction when deriving new classes.

This lab task must be completed and submitted in lab. It is individual work: collaboration and use of Internet resources is not allowed.

TASK:

Write a class called `ShrinkFracArray`, which is an `AlterableArray`.

Look at the provided code for the `AlterableArray` abstract class. The `AlterableArray` provides code for storing an array and accessing its elements with appropriate getters and setters. Additionally, it provides methods to print arrays, as well as for decimating an array (dropping certain elements) in order to produce a smaller array. However, we have left out the method which creates the new, smaller array for this task - we have an abstract placeholder for it, but it is left for subclasses to define. We will extend this class to be able to fill in its undefined abstract method. We will create a `ShrinkFracArray` subclass which defines the method as one which will create a temporary array, using the input to determine the new array size: if the input is 1, the new array will be the same size as the original; if the input is 2, it will be half the size; if it is 3, it will be a third of the size, etc.

The class must have:

- A constructor which takes an array of integer argument, and uses it to initialize the superclass's array.
- Definitions for the abstract method described in the `AlterableArray` superclass. The method should initialize the `temp` array member variable (a `protected` element in the superclass) to the given fraction of the input size. So if the original array is size 50 and `n` is 5 then the new array size should be 10. If the input is zero or negative, ignore the value and create an array which is the same size as the original. A zero-length temporary array is perfectly legitimate.
- A main method, which will:
  1. Create an array of integers which contains the values 0 through 9.
  2. Create a new `ShrinkFracArray` which is initialized using the array from the previous step.
  3. Print out the initial contents of the array using an appropriate version of the `print` method from the `ShrinkFracArray` instance.
  4. Call the method which decimates the array, making the size half of the original, and print out the new result.

Note: you may not make changes to the provided `AlterableArray` class, only to your own class.

Use the Task 11 assignment under Blackboard for submission.