**CS 1302 – Test 2 Review Problems**

Here are some sample problems that may help you prepare for Test 2. These are based on the material from chapters 2 and 3. I do not have solutions for these problems available at this time. Problem 1 is based on the class diagram provided below.

1. Write the bulkUp() method for the Dog class:
* The bulkUp() method increases the weight by 1.0

Write the following methods for the WolfDog class:

* The first constructor, which calls the superclass constructor.
* The second constructor, which sets the toughness to 1 by default.
* The getPower() method, which overrides the getPower() method in the Dog class and returns the weight multiplied by the toughness.

Write the entire Person class:

* The addDog() method accepts a Dog and puts it into the first free slot in the dogs array using best practice, or does nothing if the array is full.
* The getDog() method accepts an int and returns the dog at that index, or returns null if the index is invalid.
* The removeDog() method accepts an int and removes and returns the Dog at that index if it is valid. Otherwise, it does nothing and returns null.
* The getTotalPower() method adds the power of all Dogs using the getPower() method, and returns the result.
* The getTotalToughness() method adds together the toughness of all the WolfDogs and returns the result.

Write the following lines of code:

* Create a Dog variable with any name and weight.
	+ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
* Create a WolfDog variable with toughness 2 and any name and weight.
	+ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
* Create a Person variable.
	+ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
* Add one of the Dogs created to the person variable.
	+ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
* Get the 5th Dog in the Person variable and store it in a variable (assume that many dogs have been added).
	+ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
* Get the total power of all dogs in the Person variable and store it in a variable.
	+ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
1. Trace the following constructor chains based on the classes below and provide the output:

A a1 = new B(22);

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

A a1 = new B(“Q”);

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

A a1 = new B();

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



1. Consider the following class hierarchy diagram. Determine which lines of code are valid and which are invalid.

