**CS 1302 – Test 2 Expectations**

**Information about the test**

1. This test covers Ch. 2 & 3.
2. You may begin the test roughly 10 minutes before class time (9:20am for section A, 3:20pm for section B). Note that there is appropriate time for the test during the regularly scheduled time for the class (75 min).
3. When you enter the room, leave all books, phones, smart watches, *etc* at the front of the room.

**Expectations**

**Ch. 2**

1. Implement one-to-many using an array, similar to what is in the text, exercises, and homework:
2. Implement: *add, get, remove*
3. Write methods that process all the items in the array (iterative methods)

Note: I am not going to test on this per se, instead, see item 4 below. In other words, the problem will involve super and sub classes. However, you can skill and confidence by studying the text, exercises, and homework.

1. Write a stack class, or part of a stack class.

**Ch. 3**

1. Write a superclass and a subclass and use them. Other details:
2. proper use of *super* and *this*
3. override a method
4. subclass introduces a new instance variable
5. subclass introduces a new behavior
6. utilize protected members
7. ~~call a superclass method~~
8. prevent extending and overriding
9. Implement one-to-many using an array, similar to what is in the text, exercises, and homework:
10. Implement: *add, get, remove*
11. Write methods that process all the items in the array
12. Write methods that use *instanceof* and casting.
13. Write or trace code that illustrates constructor chaining using *this* and *super*.
14. Trace code that use supertype references for objects and/or parameter types

**Format of Test**

You will be given a class diagram that depicts a 1-many relationship what involves a super and subclass. For example: A *Person* has many *Dogs* and *WolfDogs* (subclass of *Dog*). Here is the nature of the problems based on the class diagram:

1. (3 points) Write a single method for the *Dog* class. You can assume the rest of the class is written.
2. (26 points) Write the entire *WolfDog* class.
3. (45 points) Write the entire *Person* class.
4. (16 points) Write code to use the *Person* class.

Then, there are three additional problems:

1. (6 points) Write or trace code that illustrates constructor chaining using *this* and *super*.
2. (4 points) Trace code that use supertype references for objects and/or parameter types
3. (6-8 points). Given a description of a stack, (a) write a stack method (push, pop, or peek), (b) Create a stack object and use it to push/pop/peek.