**1) The Counter class doesn't have a constructor. Why doesn't it need one?**

We do not need a constructor for anything. We are just incrementing, resetting and returning a variable between classes.

**2) The Counter class doesn't have a main. Why doesn't it need one?**

It doesn’t have a ‘main’ because it is an inherited class from ‘runner’ class

**3) How do we know ModNCounter inherits from Counter (what is the keyword)?**

We know that ModNCounter inherits from counter because of the first line in the class that saysv‘public class ModNCounter ***extends*** Counter {…}’ the keyword being ‘***extends***’.

**4) Which method is the constructor inside ModNCounter class?**

The first method ‘public ModNCounter(int n){…}’ is the constructor in the ModNCounter. You can tell due to the class not returning anything, as well as having the same name as the class itself.

**5) Is cycleLength variable visible to the parent Counter class?**

The cycleLength variable is a private variable for the class ModNCounter so it is not accessible by the parent class Counter.

**6) What happens when we call value and reset methods for ModNCounter since it is not defined in ModNCounter?**

When we call the value(){…} method from the inherited class ModNCounter it access the superclass’ Counter where value is found. It is the same situation whwn we attempt to access the method reset(){…} , it access the superclass again and accesses the method reset.

**7) What happens when we call increment and reset methods for ModNCounter2 since it is not defined in ModNCounter2?**

When the increment method is called it runs a for-loop that I coded which adds to the counter until it resets. Counter resets when it reaches a certain a certain value assigned to cycleLength.

**8) Does decrement exist inside Counter?**

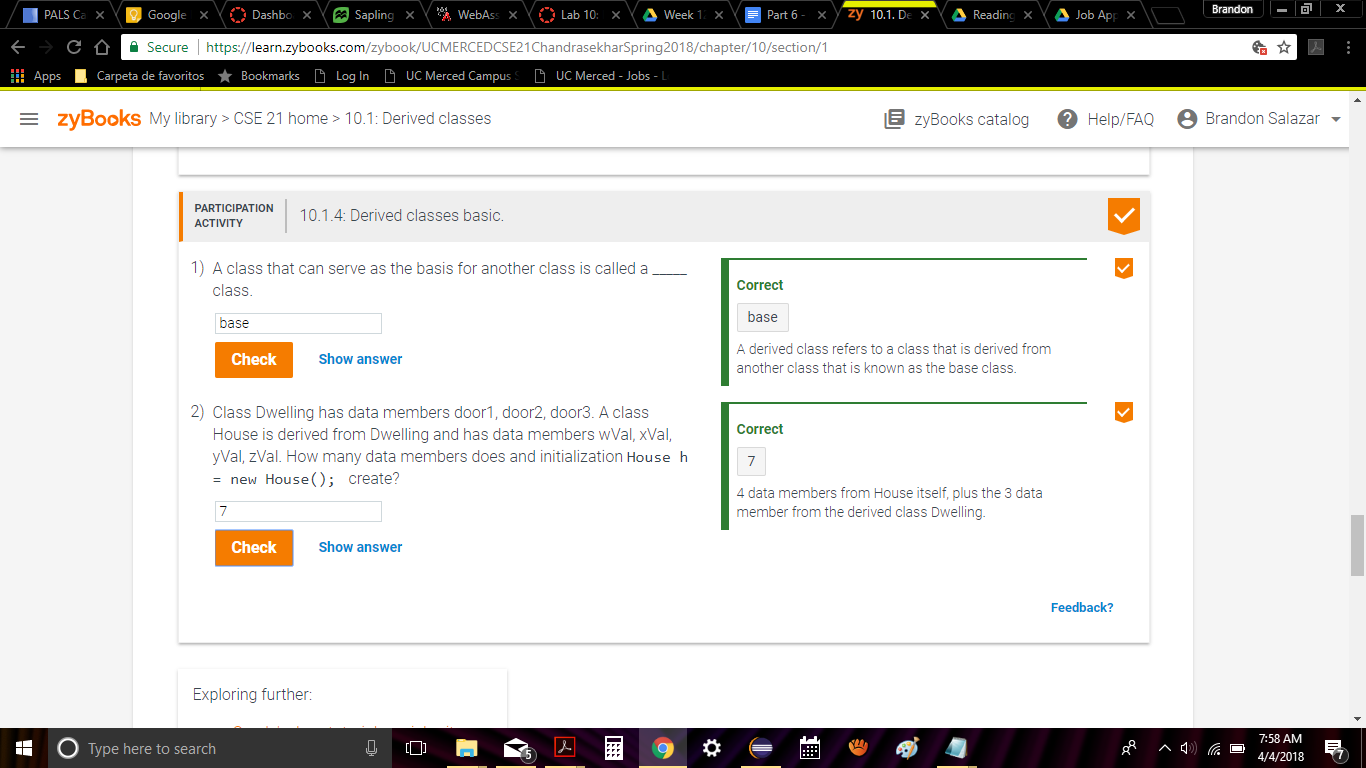
The the method decrement does not exist in the class **Counter,** it only exists in **DecrementableCounter**.

**9) Does increment’s behavior change for DecrementableCounter?**

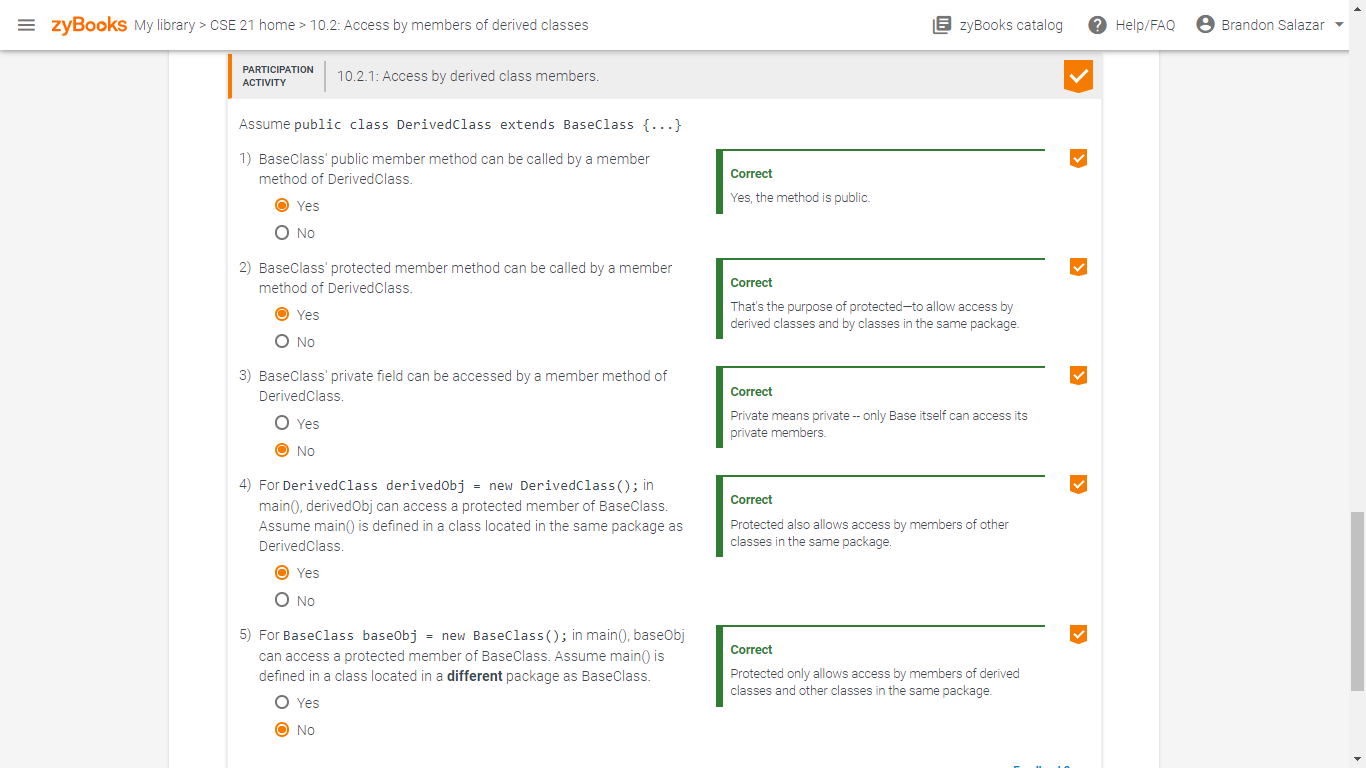
The behavior remains the same, nothing was changed in the method increment.

**10) Where is toString being inherited from?**

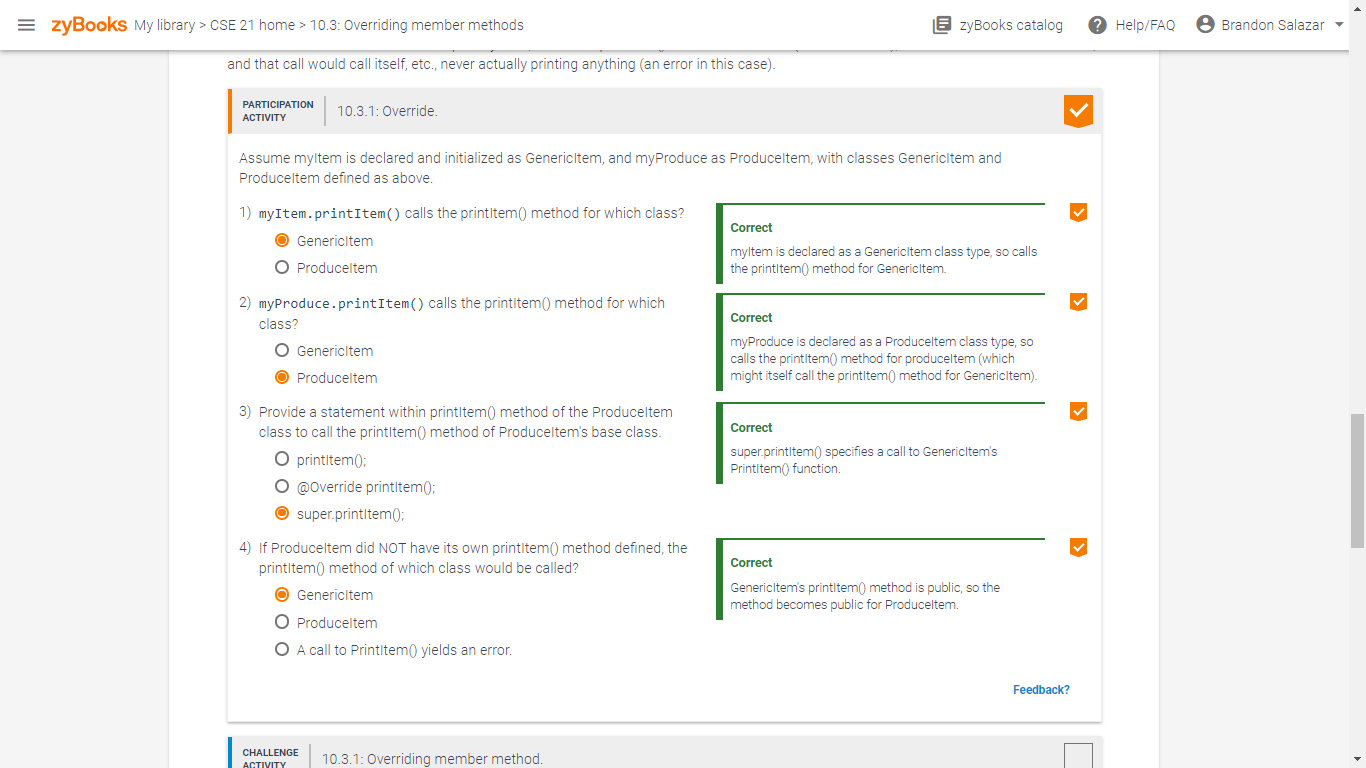
**10.1.4**

****

**10.2.1**

****

**10.3.1**

****

**10.4.2**

