Lecture 3 Clicker Questions:

1. What does the x evaluate to?

   int x = (((5/2)*3)+5);

A. 12.5
B. 11
C. 13
D. 10

Answer: B. 5/2 evaluates to 2 in Java because of how ints round down. Then 2*3 evaluates to 6 and finally 6+5 evaluates to 11.

2. Which of the following contains arguments that satisfy the parameters of the method calcChange in the BookstoreAccountant class?

A. BookstoreAccountant.calcChange(20, 14.50)
B. BookstoreAccountant.calcChange(10.00, 5.00)
C. BookstoreAccountant.calcChange(20, 10)
D. None of the above

Answer: C. Recall that calcChange takes in two ints. For A, one argument is an int but the other is a float, and for B, both are floats. C, however, has two ints for the two arguments, which will satisfy Java's parameter type checking.

3. Which of the following is not true of constructors?

A. Constructors are methods
B. Constructors always have a name as their class
C. Constructors should specify a return value(s)
D. Constructors can take in parameters

Answer: C. Recall that constructors do not specify a return value, void or otherwise, in their declaration. The declaration is usually public <name of class>(<parameters, if any>) { <body> }.

4. Using the Baker class from before, is the following method correct for creating cookie dough? Why or why not?

```
public class Baker {

      //constructor elided

      public void createDough() {
```

```
            this.combineWetIngredients();

            this.combineAllIngredients();

            this.combineDryIngredients();

        }

        //other methods elided

    }
```

A. Yes, it has all the necessary methods in proper order
B. No, it uses this instead of Baker
C. No, it has the methods in the wrong order
D. No, it is inefficient

Answer: C. The order of methods matters, and we cannot combine all ingredients before we have combined the dry ingredients. If we had done `combineWetIngredients()` then `combineDryIngredients()` then `combineAllIngredients()`, this would be valid.