Methods and Classes Test 1

1. In baseball a player's batting average is calculated by dividing the number of hits by the number of times the player has batted. Assume the following class has been defined.

```
public class Baseball
{
   /* precondition: atBats > 0
    * postcondition: a batting average has been computed
    *  @ param hits the number of hits
    *  @ param atBats the number of atBats
    *  @ return the batting average
    */
   public static double average(int hits, int atBats)
   {
      // implementation not shown
   }

   // other methods not shown
}
```

Which of the following best demonstrates the use of method average as a static method when it is called from a method in another class?

```
A. average(100, 300);
B. player.average();
C. Baseball.average(100, 300);
D. baseball.average(100, 300);
```

2. The area of a rectangle is calculated by multiplying the length times the width. Assume the following class has been defined.

```
public class Rectangle
{
    * postcondition: the area has been computed
    *  @ return the area
    */
   public double area()
   {
      // implementation not shown
   }

   // other methods not shown
}
```

Given the following definition:

```
Rectangle rect = new Rectangle();
```

Which of the following lines of code will correctly calculate the area of a rectangle?

A. `Rectangle.area();`
B. `rect.area();`
C. `area();`
D. `rect.area(length, width);`

3. Look at the following method declaration

```
public _____________ sum(double num1, double num2)  {
    return num1 + num2;
}
```

Which of the following can be inserted into the blank so that the method will correctly calculate and return the sum of num1 and num2?

A. `void`
B. `static`
C. `int`
D. `double`

4. Look at the following method declaration

```
public void play(int a)
```

Which of the following will **not** correctly overload the method **play**?

A. `public int play(int a)`
B. `public void play()`
C. `public void play(int a, int b)`
D. `public void play(double a)`

5. Look at the following code segment.

```
public class Demo {
   public static void go(int num) {
      num = num + 100;
   }

   public static void main(String[] args) {
      int number = 25;
      go(number);
      System.out.println(number);
   }

}
```

What would be output by the code above?

A. `25`
B. `100`
C. `125`
D. `No output due to a run-time error`

Questions 6-10 involve questions related to the following class.

```
public class Fraction
{
    private int num;       // numerator
    private int denom;     // denominator

    public Fraction(int n, int d)
    {
        num = n;
        denom = d;
    }

    public int getNum()
    {
        return num;
    }

    public int getDenom()
    {
        return denom;
    }

    public void setNum(int n)
    {
        num = n;
    }

    public void setDenom(int d)
    {
        denom = d;
    }

    public String toString()
    {
        return num + "/" + denom;
    }
}
```

6. Which of the following will correctly instantiate a Fraction object?

```
A. num = 3;
   denom = 4;
B. Fraction f = new Fraction();
C. Fraction f = new Fraction(3, 4);
D. Fraction.num = 3;
   Fraction.denom = 4;
```

7. Assume a fraction object has been created and its reference is stored in a variable named `frac`. Which of the following will correctly modify the numerator value of a fraction?

```
A. frac.setNum(5);
B. num = 5;
C. Fraction.num = 5;
D. frac.getNum() = 5;
```

8. Assume a fraction object has been created and its reference is stored in a variable named `frac`. Which of the following will **not** print `frac` in the form 3/4?

```
A. System.out.println(frac.getNum() + "/" + frac.getDenom());
B. System.out.println(frac.toString());
C. System.out.println(frac);
D. System.out.println(num + "/" + denom);
```

9. Suppose the following **equals** method is added to the `Fraction` class.

```
public boolean equals(Fraction other)
{
     return /* code implementation */
}
```

Which of the following lines will **not** correctly replace `/* code implementation */` so that two fractions can be compared for equality?

```
A. this.num == other.num && this.denom == other.denom;
B. num == other.num && denom == other.denom;
C. getNum() == num && getDenom() == denom;
D. this.getNum() == other.getNum() && this.getDenom() == other.getDenom();
```

10. Which line contains a syntax error? (Assume code is not in Fraction class)

```
Line 1 : Fraction frac = new Fraction(3, 5);
Line 2 : frac.setNum(3);
Line 3 : frac.setDenom(getNum());
Line 4 : System.out.println(frac.getNum() + "/" + frac.getDenom());
```

```
A. Line 1
B. Line 2
C. Line 3
D. Line 4
```

Free Response

1. Write a complete class definition for a class that represents a rectangle. The class should include the following:
    - Two **instance variables** representing the length and width of a rectangle.
    - Two **constructors** - a no parameter (default) constructor and a two parameter constructor that will initialize the two instance variables.
    - An **accessor** method for each instance variable.
    - A **mutator** method for each instance variable.
    - A method named **area** that will calculate and return the area of a rectangle.
    - A **toString** method that will label and display the value of each instance variable.