Name:____ Section: __________CM: ____________

CSSE 220---Object-Oriented Software Development

Exam 1 -- Part 1, September 25, 2019

This exam consists of two parts. Part 1 is to be solved on these pages. If you need more space, please ask your instructor for blank paper. After you finish Part 1, please turn in your Part 1 answers and then open your computers and wait quietly for the programming exam review section of class to begin.

Allowed Resources on Part 1: You are allowed one 8.5" by 11" sheet of paper with notes of your choice. This section is *not* open book or open notes; and you are not allowed to use your computer for this part.

You will have 50 minutes (the first Rose hour of class) to complete Part 1.
Part 2 will be completed tomorrow evening.

Please, begin by writing your name on every page of the exam. We encourage you to skim the entire exam before answering any questions.

Problem	**Points Possible**	**Earned**
1	9	_______
2	8	_______
3	4	_______
4	9	_______
5	5	_______
Paper Part Subtotal	35	_______
Computer Part Subtotal	65	_______
Total	100	_______

```
public class Racer {
    private String name;
    private int number;
    private int ranking;

    public Racer (String name, int number, int ranking) {
        this.name = name;
        this.number = number;
        this.ranking = ranking;
    }

    public void setName (String name) {
        this.name = name;
    }

}
public class RaceTeam {
    private String teamName;
    private Racer [] racers;
    private int wins;
    private int racerCount;

    public RaceTeam (String teamName) {
        this.teamName = teamName;
        this.racers = new Racer [3];
        this.wins = 0;
        this.racerCount = 0;
    }

    public void addRacer (Racer r) {
        racers[racerCount] = r;
        racerCount++;
    }

}
```

The next question refers to the classes on this page. The Javadocs are omitted to save space.

1. (9 points, 3 points each) Below are several code snippets that use the `Racer` and `RaceTeam` classes. For each snippet, first *draw a box-and-pointer diagram* (in the blank area below the snippet) showing the *final* result of executing the code. Changes to pointers or values for variables defined in the code snippets below CAN be shown but are NOT required if you produce the exactly correct final answer.

```
Racer r1 = new Racer("Billy-Bob", 15, 3);
```

a) **Diagram**:

```
Racer r1 = new Racer("Maria", 20, 2);
Racer r2 = r1;
RaceTeam team1 = new RaceTeam("Hot Rods");
team1.addRacer(r1);
r1 = new Racer("Tim", 12, 3);
```

b) **Diagram**:

```
RaceTeam team2 = new RaceTeam("Speed Demons");
Racer r1 = new Racer("Jennifer", 50, 1);
String [] alts = {"Dan", "Tiffany", "Claire"};
for (String alt : alts) {
      r1.setName(alt);
      team2.addRacer(r1);
}
r1 = new Racer(alts[1], 75, 4);
```

c) **Diagram**:

2. (8 points, 2 points Each) Predict the output for each code snippet below.

- Each code snippet has no errors.
- You do not need to draw a diagram, but you may if it might help you.
- DO NOT TYPE THE CODE SNIPPETS FOR THIS QUESTION INTO ECLIPSE.
- If output spans multiple lines, write additional lines below the Output: line.

```
int a = 10;
int b = 4;
double d = 4.0;
System.out.println(a/b);
System.out.println(a/d);
```

(a) Output:______________________

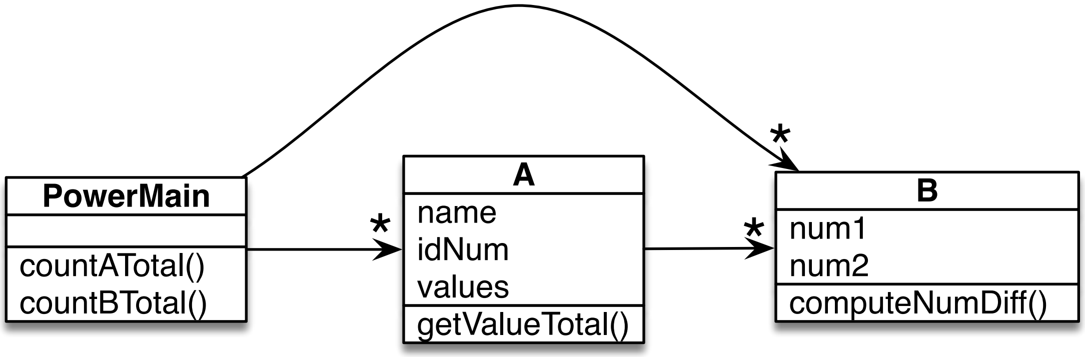
```
HashMap<String, Integer> stuff = new HashMap<>();
HashMap<Integer, String> other = new HashMap<>();
String[] str = {"thing", "diff", "another", "try"};
stuff.put(str[0], 1);
stuff.put(str[1], 2);
stuff.put(str[2], 3);
stuff.put(str[3], 2);
for(int i = 0; i < str.length; i++) {
  other.put(stuff.get(str[i]), str[i]);
}
for(int i = 1; i <= 3; i++) {
  System.out.println(i + " : " + other.get(i));
}
```

(b) Output:______________________

```
ArrayList<Integer> list1 = new ArrayList<Integer>();
list1.add(1);
list1.add(2);

ArrayList<Integer> list2 = list1;

if(list2.equals(list1))
    System.out.println("equals1");
if(list2 == list1)
    System.out.println("equals2");

list2 = new ArrayList<Integer>();
list2.add(1);
list2.add(2);

if(list2.equals(list1))
    System.out.println("equals3");
if(list2 == list1)
    System.out.println("equals4");
```

(c) Output:______________________

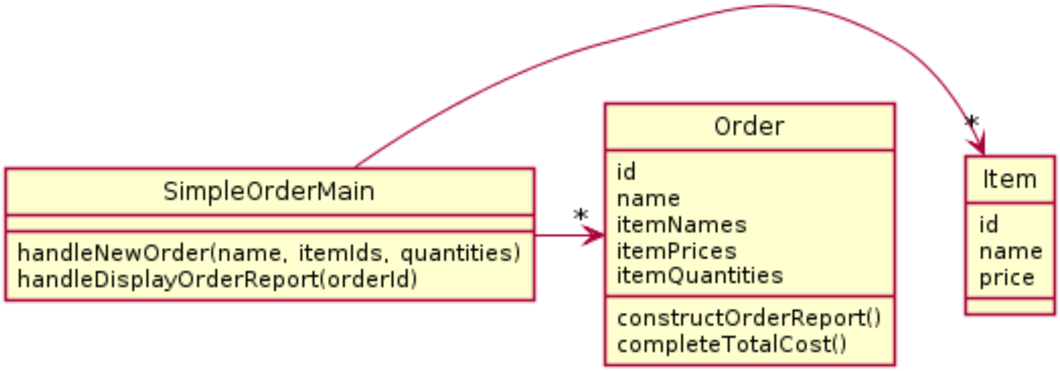
3. (4 points) Write Java code that corresponds to the given UML diagram.



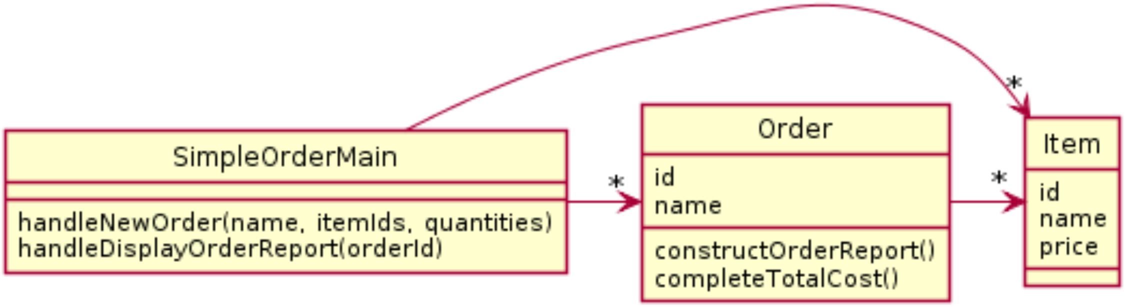
4. (9 points, 3 points each) For the following, you are given a problem statement and then some possible solutions. Answer each of the two questions in regards to each solution provided and then provide your own solution to the problem.

A particular company tracks orders for items. When an order is placed it consists of a customer name, a list of item ids, and a list of quantities. For example, an order "Steve", [30012,30044], [1,2] would mean Steve has ordered one of item 30012 and 2 of item 30044. The ordering system keeps track of all the items that can be ordered - each item has id, name, and (per item) price. When an order is placed the system gives the order id that can be used to identify the order later. Using that order id, the user can at any time display a report containing the names and quantities of all items ordered, the price of each item, and the total price.

(a) This solution does not function correctly. Explain why.



(b) This solution does not function correctly. Explain why.



(c) Draw a UML diagram of your own proposed solution here.

5. (5 points) Write T next to the statements that are true in Java, F next to the statements that are false.

1. _____ The `get` function of the class HashMap is definitely not static.

2. _____ The following is the correct way to check if a HashMap named `myMap` is null:
```
if(myMap.isEmpty()) { /* do something */}
```

3. _____ A static function can access field variables that are not static as long as the programmer has a good reason.

4. _____ With a String named `str`, calling `str.toUpperCase();` will change the String pointed to by `str`.

5. _____ A class can have many constructors.