Name _____ ID _____ Final Score _____
Extra Credit _____

Section (circle one): MW 8:30-9:50 TTh 9:30-10:50 TTh 11:00-12:20

CSCI 201L Final – Written
Spring 2016
10% of course grade

1. Anonymous Inner Classes – In lecture we walked through the following:

1. Having two classes in different file.
2. Having two classes in the same file.
3. Having one class with the second class inside of it (an inner class).
4. Having one class with the second class inside of one of the methods.
5. Having one class with the second class inside the parameter call to a method with no name (anonymous inner class).

a. Give two reasons why a programmer would choose to create an inner class (#3 above) instead of having classes in different files (#1 above). **(0.5% + 0.5%)**

Reason #1

Reason #2

b. Give two reasons why a programmer would choose to create an anonymous inner class (#5 above) instead of a class inside of another class (#3 above). **(0.5% + 0.5%)**

Reason #1

Reason #2

2. **Software Engineering** – In the book The Mythical Man-Month, software engineering professor Fred Brooks states that it takes approximately nine times as long to create a software product system over just writing a program. Give two reasons why it takes more time to create a professional software product instead of just writing code to make a program. **(0.5% + 0.5%)**

Reason #1

Reason #2

3. **Networking Theory** – You have started a small company and want to host a few servers out of your home. Since you have taken CSCI 201, you know that the best way to do this is to have a static IP address on each of those servers. Assume that you have a separate server for each of the following – mail server, FTP server, web server, SSH server, and DNS server. When you call your ISP, there is a UCLA alumnus working there. You tell him what you want, and he says, "Your starting IP address is 121.156.99.192," but he doesn't tell you anything else. Because you went to USC, you can hopefully answer the following questions though.

IP – `0111 1001  1001 1100  0110 0011  1100 0000`

   a. What is the network address? Provide this in the dotted IP notation, not in binary. **(0.5%)**

   b. So that you are given the fewest number of IP addresses for your desired purpose, what is the subnet mask? Provide this in the dotted IP notation AND slash notation, not in binary. **(0.5% + 0.5%)**

   c. What are five possible IP addresses you could assign to your servers? **(0.5%)**

4. Databases and SQL – Answer the following questions concerning the database below.

Here is the `Book` table.

bookID	title	author	isbn	numCopies
1	Tonight on the Titanic	Mary Pope Osborne	978-0-606-16894-6	3
2	Afternoon on the Amazon	Mary Pope Osborne	978-0-679-86372-9	1
3	Balto of the Blue Dawn	Mary Pope Osborne	978-0-553-51085-0	2
4	Happy Birthday, Bad Kitty	Nick Bruel	978-0-545-29863-6	2
5	Bad Kitty Does Not Like Candy	Nick Bruel	978-1-62672-230-9	1
6	Bad Kitty Drawn to Trouble	Nick Bruel	978-1-62672-117-3	2

Here is the `User` table.

userID	username
1	jimmy
2	joannie
3	johnny
4	jenny

Here is the `CheckedOut` table.

checkedOutID	bookID	userID	numCheckedOut
1	3	4	1
2	3	3	1
3	1	1	2
4	2	2	1

a. Write the SQL code to create the `CheckedOut` table. Don't forget to include the foreign keys. **(0.5%)**

b. Draw the table that is returned from the following query. **(0.5%)**

```
SELECT u.username, b.title, co.numCheckedOut
  FROM Book b, User u, CheckedOut co
  WHERE b.bookID=co.bookID
    AND u.userID=co.userID;
```

5. **JDBC –** Give two advantages to using prepared statements with JDBC instead of just using statements. **(0.5% + 0.5%)**

Advantage #1

Advantage #2

6. **Distributed Computing –** What is the major difference between distributed computing and parallel computing? **(0.5%)**

7. **Multi-Threading and Parallel Programming** – Explain why a program written using parallel computing could run more slowly than a program written using multi-threading. **(0.5%)**

8. Concurrent Programming – Answer the two questions that follow the code below. **(2.0%)**

```
1  public class Problem8 {
2     public static void main(String [] args) {
3           for (int i=1; i <= 10; i++) {
4                 int arr[] = new int[i];
5                 for (int j=0; j < i; j++) {
6                       arr[j] = j;
7                 }
8                 new P8Thread(arr);
9           }
10    }
11 }
12 class P8Thread extends Thread {
13    private int[] arr;
14    private int num;
15    public P8Thread(int [] arr) {
16          this.arr = arr;
17          this.num = arr.length;
18          start();
19    }
20    public void run() {
21          System.out.print("Thread " + num + ":");
22          for (int i=0; i < arr.length; i++) {
23                System.out.print(arr[i]);
24                if (i != arr.length - 1) {
25                      System.out.print(",");
26                }
27          }
28          System.out.println();
29    }
30 }
```

a. The above code does not always print out all of the numbers for a specific thread before printing the values for other threads. Explain why this is the case. **(0.5%)**

b. Modify the above code *using concurrency topics* to make the code print all of the numbers for a specific thread before it prints any numbers from a different thread. **(1.5%)**

Extra Credit Questions

Extra credit is applied after the curve so does not affect other students.

☐

9. Extra Credit – With group projects in an academic setting, there are at least two camps of people with regards to grading – those who think the group should be graded as a whole, and those who think the members should be graded individually. For the group project in CSCI 201, do you think that:

- **a.** The individuals in the group should all receive the same grade?
- **b.** The individuals in the group should be graded independently of the other members of the group?
- **c.** Some hybrid of the above two options (i.e. some parts of the project graded as a group and others graded individually)?

Explain your answer and provide a way that your solution could be performed. **(0.25% + 0.25%)**

☐

10. Extra Credit – We covered six major topics in this class. Rank these six topics in order of what you thought was most useful to least useful. Provide one sentence explaining why you thought the most useful topic was the most useful and one sentence explaining why you thought the least useful topic was the least useful. The topics were: Java porting from C++, GUIs, Software Engineering, Networking, Databases, and Concurrent Programming. **(0.25% + 0.25%)**

#1 (Most Useful) –
Explanation –

#2 –

#3 –

#4 –

#5 –

#6 (Least Useful) –
Explanation –