**CS 145 Practice Midterm Solutions**

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_­­­­­­­­­­­­­­­­­­­­­­\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |  |
| --- | --- | --- |
| Problem | Points | Points Earned |
| Multiple Choice, True/False | 20 |  |
| Reading Code #1 | 20 |  |
| Reading Code #2 | 20 |  |
| Writing Code #1 | 20 |  |
| Writing Code #2 | 20 |  |
| Extra Credit | 3 |  |
| **Total** | **100** |  |

**Multiple Choice, True False (2 pts each)**

1. What is wrong with the following initialization of an ArrayList of integers stylistically and/or syntactically? ArrayList<int> list = new ArrayList<int>()
   1. Nothing is wrong
   2. The left hand side should be using the interface type
   3. **int is not a valid type for an ArrayList**
   4. Both b and c
2. What would the following print to the console?

System.out.println(4 - 6 \* 2 + "1 + 3" + (7 % 4) + 6 / 4);

* 1. **-81 + 331**
  2. -81 + 331.5
  3. -81 + 34
  4. None of the above

1. Which of the following is the correct way with good style to set up a field in your class before your constructor?
   1. int x = 5;
   2. **private double cash;**
   3. public String word;
   4. private boolean amPm = false;
2. Duplicate **values** are not allowed in a Map.
   1. True
   2. **False**
3. When passing in an array as a parameter, a reference to the actual array is passed, so if any changes are made to the array in the method, they change the array outside of the method as well.
   1. **True**
   2. False
4. There are two types of methods in objects that we write: accessors which access information about the object and mutators which change the state of the object.
   1. **True**
   2. False
5. Which of the following uses the proper naming convention for a method name?
   1. PrintVerseOne
   2. ADD\_NUMBERS
   3. **findDistance**
   4. calculate exponent
6. Which is the proper way stylistically to test multiple conditions?
   1. if(x < 5 OR x > 10)
   2. if (y == 3, 7)
   3. if(count != 2)

if(count > 9)

* 1. **if(val == 8 || val == 9)**

1. What information should you include in your method comments? (select all that are correct)
   1. How it is performing its tasks, such as using for loops or if statements
   2. **A brief description of what your method does**
   3. **Description of what parameters (if any) are needed and what they represent**
   4. **Description of what the method returns (if it returns) and what it represents**
2. If you’re throwing an Exception, when should it be thrown to have good Java style?
   1. **As soon as possible (usually at the beginning of a method)**
   2. At some point before the data it would check is being used, but it doesn’t matter when
   3. What are Exceptions?
   4. At any point in the method of your own choosing

**Reading Code #1 (20 pts)**

What is the output of the following code?

1 import java.util.\*;  
 2   
 3 public class ReadingCode {  
 4   
 5 public static void main(String[] args) {  
 6 Map<Integer, List<String>> map = new TreeMap<Integer, List<String>>();  
 7 List<String> list1 = new ArrayList<String>();  
 8 list1.add("hello");  
 9 list1.add("hi");  
10 map.put(3, list1);  
11   
12 List<String> list2 = new ArrayList<String>();  
13 list2.add("howdy");  
14 list2.add("hello");  
15 list2.add("hola");  
16 map.put(2, list2);  
17   
18 List<String> list3 = new ArrayList<String>();  
19 list3.add("hi");  
20 map.put(4, list3);  
21   
22 readMaps(map);  
23 }  
24   
25 public static void readMaps(Map<Integer, List<String>> map) {  
26 for(int k : map.keySet()) {  
27 for(String s : map.get(k)) {  
28 for(int i = 0; i < k; i++) {  
29 System.out.println(s);  
30 }  
31 }  
32 System.out.println();  
33 }  
34 }  
35   
36 }

**Reading Code #1 continued (20 pts)**

howdy  
howdy  
hello  
hello  
hola  
hola  
  
hello  
hello  
hello  
hi  
hi  
hi  
  
hi  
hi  
hi  
hi

**Reading Code #2 (20 pts)**

What is the output of the following code with the provided classes?

public class Polymorphism {  
 public static void main(String[] args) {  
 Mars[] arr = {new Mercury(), new Jupiter(), new Mars(), new Saturn()};  
 for(Mars element : arr) {  
 element.method2();  
 element.method1();  
 System.out.println(element);  
 System.out.println();  
 }  
 }  
}

public class Jupiter extends Saturn {  
 public void method1() {  
 System.out.println("Saturn1");  
 }  
}

public class Saturn extends Mars {  
 public String toString() {  
 return "Saturn";  
 }  
}

public class Mars {  
 public void method1() {  
 System.out.println("Mars1");  
 }  
   
 public void method2() {  
 System.out.println("4th Planet");  
 }  
   
 public String toString() {  
 return "Mars";  
 }  
}

public class Mercury extends Saturn {  
 public void method2() {  
 System.out.println("6th planet");  
 }  
   
 public String toString() {  
 return "Saturn";  
 }  
}

**Reading Code #2 continued (20 pts)**

6th planet  
Mars1  
Saturn  
  
4th Planet  
Saturn1  
Saturn  
  
4th Planet  
Mars1  
Mars  
  
4th Planet  
Mars1  
Saturn

**Writing Code #1 (20 pts)**

Write a method called removeEvens as part of the LinkedIntList class. Your method should remove all nodes in the list that contain an even number as their data. For example, given a list such as:

2 -> 4 -> 3 -> 6 -> 7 -> 8, the list after removeEvens is run would be 3 -> 7.

public void removeEvens() {

while(front != null && front.data % 2 == 0) {

front = front.next;

}

ListNode current = front;

while(current.next != null) {

if(current.next.data % 2 == 0) {

current.next = current.next.next;

} else {

current = current.next;

}

}

}

**Writing Code #2 (20 pts)**

Write a method called subset that takes in two Sets of integers and determines if the second Set is a subset of the first Set. A subset means that the contents of the set are found entirely in the other set. It returns true if it is a subset, and false otherwise. For example, if set one contains {15, 2, 9, 4, 6, 12, 3} and set two contains {3, 9, 4}, this method would return true as the second set is a subset of the first.

public static boolean subset(Set<Integer> s1, Set<Integer> s2) {  
 for(int x : s2) {  
 if(!s1.contains(x)) {  
 return false;  
 }  
 }  
 return true;  
 }

**Extra Credit (3 pts)**

For 3 points of extra credit, take this page and draw a picture or write a poem about your favorite thing about Computer Science.