CSC 212 Final

Fall 2014

1. **(10 points) Consider the following variable declaration and values:**

*double a = 5.0; int b = 0; int y = 2;*

**Choose the correct answer for each of the following; with each question assume a, b, and y have the values shown above:**

a) What is the value of b after: *b = y + a / 2;*

A) 3 C) 4.5

B) 3.5 D) 4

E) ERROR

b) What is the value of ***y*** after: *y += ((int) (a+1)) / 2 + 5 \* y;*

A) 16 C) 18

B) 13 D) 15

E) ERROR

c) What is the value of ***a*** after: *a += (1 + b \* 10) \* y;*

A) 2.0 C) 0.0

B) 7.0 D) 25.0

E) ERROR

1. **(10 points) Consider the code below:**

*for (int i = 0; i < 5; i++) {*

*System.out.println ("i = " + i);*

*}*

1. How many times will this *for* loop iterate?
2. What would happen if we change the condition in the *for* loop to ***i > 5***? How many times, if at all, will the loop iterate in this case?
3. **(20 Points) Show the output produced by each of the two code segment:**
4. What will be the output in this case be?

*ArrayList<String> a = new ArrayList (3);*

*a.add("hello"); a.add("goodbye");a.add("next");*

*for (int i = 0; i < a.size();i++) {*

*System.out.println(a.get(i));*

*}*

**OUTPUT:**

1. What will be the output in this case be?

*ArrayList<String> a = new ArrayList (3);*

*a.add("hello"); a.add("goodbye"); a.add("next");*

*for (int i = 1; i < a.size();i++) {*

*if (a.get(i).startsWith(“g”)) System.out.println(a.get(i));*

*}*

**OUTPUT:**

1. **(20 points) Consider the following Class definition for *Song*:**

*public class Song {*

*String title; String artist; String genre;*

*int duration; //how long the song is in seconds*

*public* ***Song*** *(String t, String ar, String g, int d) {*

*title=t; artist=ar; genre=g; duration=d;*

*}*

*public int* ***duration*** *() {return duration;}*

*public String* ***toString*** *() {*

*return "Title = " + title + ", Artist = " + artist +*

*", Genre = " + genre + ", Duration = " + duration;*

*}*

*}*

1. Write the instructions that would declare two ***Song*** variables. Assign a So*ng* object to each with your own choice of values needed (i.e. use NEW to construct).
2. Write the code that compares the ***duration*** of these two song variables created in a) and display the longer of the two songs (i.e. the song with a bigger duration).
3. Assume the following declaration: ***Song [] list = new Song [100000];*** also assume that we have stored some songs in ***list*** and the variable ***size*** tracks how many songs are in it. Explain what this code segment will accomplish:

*for (int i= 0; i<=1000000;i++) {*

*int loc1 =(int) Math.random() \* size;*

*int loc2 = (int) Math.random() \* size;*

*Song temp = list[loc1];*

*list[loc1] = list [loc2];*

*list[loc2] = temp;*

*}*

1. **(20 points) Here is the definition of a Stack class.**

*public class Stack {*

*public Stack (){…}*

*public void push(Object o) {…}*

*public void pop() {…}*

*public Object peek() {…}*

*public int size() {…}*

*}*

Consider the code segment below:

|  |
| --- |
| *Stack s = new Stack();*  *For (int i = 0; i < 10; i++) s.push((int)(Math.random() \*1000);* |

1. What does this code segment do?
2. Write a code segment that will reverse the content of the stack ***s*** shown above without adding any new methods to *Stack.* You can of course use all methods available for a stack: push, pop, etc. The task here needs to be completed by only using the methods available.

You need to create another stack, let’s call it ***y***. By moving the content of ***s*** one at a time into ***y***, you will have reversed the order of the elements as they are stored in this new stack. After those steps are completed, assigning y to s (i.e. ***y = s;***), in effect, reverse the order of its content.

1. **(20 points) Consider the definition of the Dictionary class that we used for assignment 4 as well as the dictionaryEntry class in that assignment that held Word/Definitions:**

***public class dictionaryEntry {***

*public* ***dictionaryEntry*** *(String w, String d) {…}*

*public String* ***Word*** *() {…}*

*public**String* ***Definition*** *() {…}*

***}***

***public class Dictionary {***

*public static* ***load*** *(Scanner s) {…}*

*public static boolean* ***add*** *(String w, String d) {…}*

*public static boolean* ***delete*** *(String w) {…}*

*public static dictionaryEntry* ***find*** *(String w) {…}*

*public static ArrayList<dictionaryEntry>* ***browse*** *(String partialWord) {…}*

*}*

Assume that we already have performed ***load(..);*** so, our dictionary already holds word/definitions of all entries of our data file. Also assume ***keyboard*** is a Scanner object that allows us to get input from the user.

1. Complete the code segment here that prompts the user for a word, uses the method(s) in **Dictionary** to look for the word in the dictionary. If the word is found, display the **definition** of the word. If the word is not found, display a message that says that the word was not found.

System.out.println (“Enter a word to search: “);

String w = keyboard.nextLine();

1. Complete the code segment below that will display all word/definitions for words that start with the letter “h”:

*ArrayList <dictionaryEntry> a = Dictionary.browse(“h”);*