

Name _____

CISC 1115 – Introduction to Programming using Java
Fall 2023
Exam #1 Solutions

For questions asking for the output of code or final values of variables, showing your work (e.g. traces of variables) can result in partial credit

Part I: CodeLab Exercises (30 points):

Unless otherwise specified, assume all variables have been declared and appropriately initialized.

Lecture 1:

- a) Write a statement that prints `Hello, world` to the screen.

20503

- b) Use a `for` loop to print a single line consisting of 97 asterisks.

20674

Lecture 2:

- c) Calculate the average (as a double) of the values contained in the integer variables `num1`, `num2`, `num3` and assign that average to the double variable `avg`.

20758

- d) Given an integer variable `bridgePlayers`, write a statement that increases the value of that variable by 4.

20594

Lecture 3:

- e) Write a conditional that assigns 10,000 to the variable `bonus` if the value of the variable `goodsSold` is greater than 500,000.

20615

- f) Write a statement that compares the values of `score1` and `score2` and takes the following actions.
- When `score1` exceeds `score2`, the message "player1 wins" is printed to standard out.
 - When `score2` exceeds `score1`, the message "player2 wins" is printed to standard out. In each case, the variables `player1Wins`, `player1Losses`, `player2Wins`, and `player2Losses`, are incremented when appropriate.
 - Finally, in the event of a tie, the message "tie" is printed and the variable `tieCount` is incremented.

20930

Lecture 4:

- g) The area of a square is stored in a double variable named `area`. Write an expression whose value is length of one side of the square. (You should be using the appropriate `Math` class method)

21010

- h) Write the definition of a method `add`, which receives two integer parameters and returns their sum.

20656

Lecture 5:

- i) Write a sequence of statements that create a file named "greeting.text" and write a single line consisting of "Hello, World!" to that file. Make sure that the line has been flushed to the file and that any system resources used during the course of running these statements have been released (i.e., make sure you close the file). (Do not concern yourself with any possible exceptions here-- assume they are handled elsewhere.)

21200

- j) Write a statement that reads a floating point value from standard input into temperature. Assume that temperature has already been declared as a double variable. Assume also that stdin is a variable that references a Scanner object associated with standard input (System.in).

20968

Part II: Labs: (40 points)

- *With the exception of part (a), you need not supply any imports or class headers.*
- *You must create and initialize any `Scanner` objects you use.*

Lecture 1:

- a) Write a full Java program (application) that displays `Hello world` on the screen.
This should include all boilerplate.

Lab 1.2

Lecture 2:

- b) What is the value of `total` after the following code is executed. For partial credit, show your work, i.e., show the values of `i` and `total` as they are assigned and change throughout the execution of the code:

```
total = 0;
for (int i = 1; i <= 5; i = i + 1)
    total = total + i;
```

total: _____

Work

i	total

Lecture 3:

c) Given the following code:

```
int x = stdin.nextInt();
int y = stdin.nextInt();
int z = stdin.nextInt();
int total = 0;

if (x < 0) total = total + 1;
if (y < 0) total = total + 1;
if (z < 0) total = total + 1;
```

For each of the following, assume stdin contains the following values (i.e., the user at the keyboard types in the following values, one per variable). Show the value of `total` after execution of the above code . For partial credit, show the value of `total` as it changes.

a. stdin: 1 2 3

total: _____

Work

total

b. stdin: -10 2 3

total: _____

Work

total

c. stdin: -10 2 -3

total: _____

Work

total

d. In general, what does the above code do?

Assigns to total how many values of x, y, and z are less than 0

Lecture 4:

d) Write the method:

- `printOdds(int n)` - accepts an integer argument in parameter `n`, and prints the odd numbers from 2 to `n`

Lab 4.6 Odds and Evens

Lecture 5:

- e) The file `numbers.text` contains a header value (i.e., an integer specifying the number of data items in the file), followed by that number of pairs -- one pair per line. Read the numbers in as integers and print the *concatenation* of each pair — one concatenation per line. Use a `for` loop to process the two pairs.

For example, if the file `numbers.text` contains:

```
2
1 2
3 4
```

the code should produce the following output:

```
The sum of 1 and 2 is 12
The sum of 3 and 4 is 34
```

Lab 5.2 Concatenation of Pairs

Part III: New Stuff

a) (10 points) Find and correct the errors in the following code:

```
int x = scanner.nextInt();
int y = scanner.next();
int the-Result;

if x > y
    the-Result == x;
else
    the-Result = y;
system.out.println("The smaller of the two is ' + the-Result);
```

```
int x = scanner.nextInt();
int y = scanner.nextInt();
int the_Result;

if (x > y)
    the_Result = x;
else
    the_Result = y;
System.out.println("The larger of the two is " + the_Result);
```

- b) (20 points)** For this question, you are to write a full program (imports, boilerplate and everything), with proper syntax (i.e, proper parenthesization, semicolons, braces, etc). Make sure you “*dot your i’s and cross your t’s*” – proper syntax counts here. (Also, please be as neat as possible.) Don’t forget prompts.

Write a program that:

- Creates a `Scanner` object corresponding to the keyboard
- Prompt the user for the number of students to be processed
- Use a `for` loop to then process that number of students. For each student:
 - Prompt for the name and semester average (a double) of the student
 - Print out the name and letter grade of the student
 - Print out whether the student is eligible for the Deans list or needs a warning letter
 - Eligibility for the Deans list is based on an average greater than or equal to 95
 - A warning letter is issued for an average less than 70.

Optional:

- Print out the number of Deans list students
- Print out the name of the students with the highest and lowest averages

For full credit, you must define and call the method

```
char calcGrade(double average)
```

Here is a sample execution:

```
How many students? 2

name? Weiss
average? 72
Weiss gets a grade of C

name? Arnow
average? 96.4
Arnow gets a grade of A
Arnow is eligible for the Deans list
```



```

import java.util.Scanner;

public class Grader {
    public static void main(String [] atgs)
        Scanner scanner = new Scanner(System.in);

        System.out.print("How many students? ");
        int numStudents = scanner.nextInt();

        for (int i = 1; i <= numStudents; i = i + 1) {
            System.out.print("Name? ");
            String name = scanner.next();
            System.out.println("average? ");
            double average = scanner.nextDouble();

            System.out.println(name +
                " earned a " + calcGrade(average));

            if (average >= 95)
                System.out.println(name +
                    " is eligible for the Deans List");
            else if (average < 70)
                System.out.println(name +
                    " is performing below par");
        }
    }

    public static String calcGrade(double average) {
        if (average >= 90) return "A";
        else if (average >= 80) return "B";
        else if (average >= 70) return "C";
        else if (average >= 60) return "D";
        else return "F";
    }
}

```

- 6 points for boilerplate and program structure (2 methods)
- 4 points for loop logic
- 3 points for prompts and input logic
- 4 points for grade calc logic
- 3 points for deans list / warning logic