CSCI 207 Midterm - Fall 2020 **Name** Frederick Uy

Directions: The midterm will be posted at the beginning of class on Moodle (9:30am). You will have the entire class time to complete the exam. You must submit your midterm before class ends (10:45am). Everyone must take the exam at the same time, no exceptions. The exam is a total of 20 points. I will be online, and you can join the class link at any time if you have questions.

Important note: Be sure that you do not upload a blank copy of your exam! You will not receive credit for a blank exam. Check that you have properly saved the contents of your new file. As a precaution, consider **also** uploading a word document with your answers.

Question 1 [10 points] Choose a programming problem from the list. For each problem, do not hardcode your answers. Your code should work if I change the values of any of the items that are indicated in **bold, orange** font.

[1] Create **2 int arrays, a and b**, each of length 7. Calculate the sum of the values in each array. Print the name of the array with the largest sum. If the sums are equal, print *a*.

Example 1
Sample array

a	3	6	4	7	8	9	10
b	6	8	4	75	2	31	1

Sample output
b

Example 2
Sample array

a	4	6	4	10	8	2	10
b	20	3	7	5	1	2	6

Sample output
a

Example 3
Sample array

a	32	6	48	70	8	9	10
b	6	18	4	5	2	3	1

Sample output
a

[2] Create a **string, myString** and print *Same* if the first 2 chars in the string also appear at the end of the string. If the first 2 chars in the string do not appear at the end of the string, print *different*.

Example 1
Sample string
edited

Sample output
Same

Example 2
Sample string
course

Sample output
Different

[3] Create an **int array of length 5**. Determine if the first or last element in the array is larger and set all the other elements to be that value. Print the updated array.

Example 1
Sample array

4	18	3	71	8

Sample output
8 8 8 8 8

Example 2
Sample array

3	1	35	17	2

Sample output
3 3 3 3 3

- Write out the code in the space provided below. Be sure to indicate the problem that you chose.
  Rubric:

Excellent	Good	Below Average	Poor
Demonstrates excellent understanding of programming: ▪ 100% of requirements met ▪ no syntax errors ▪ no logic errors	Demonstrates good understanding of programming: ▪ 79-60% of requirements met ▪ 1-2 syntax errors ▪ 1-2 logic errors	Demonstrates fair understanding of programming: ▪ 59-30% of requirements met ▪ 3-4 syntax errors ▪ 3-4 logic errors	Demonstrates poor understanding of programming: ▪ < 30% of requirements met ▪ > 5 syntax errors ▪ > 5 logic errors
10 points	9 – 7 points	6 - 4 points	3 - 0 points

Question 2 [10 points] Write a one-page analysis of your program in the space provided below. See Question 2. Do not exceed one page. Be sure that the analysis is thought provoking and intellectual. You should speak as a computer scientist. Include the following:

- Describe your algorithm. How does your code work? Why did you choose this approach?
- Give direct examples (at least 2) of similarities in your past lab assignments in this course.
- Bonus (+5): Discuss a real-world application of how your program can be used.

Rubric:

Excellent	Good	Below Average	Poor
Demonstrates excellent understanding of programming: ▪ 100% of requirements met ▪ Very informative and well-organized ▪ Virtually no spelling, punctuation or grammatical errors	Demonstrates good understanding of programming: ▪ 79-60% of requirements met ▪ Somewhat informative and organized ▪ Few spelling and punctuation errors, minor grammatical errors	Demonstrates fair understanding of programming: ▪ 59-30% of requirements met ▪ Gives good information but poorly organized ▪ A number of spelling, punctuation or grammatical errors	Demonstrates poor understanding of programming: ▪ < 30% of requirements met ▪ Gives no information and very poorly organized ▪ So many spelling, punctuation and grammatical errors that it interferes with the meaning
10 points	9 – 7 points	6 - 4 points	3 - 0 points

Question 1 [10 points] **Problem #** 1

```
/*
This program will ask users for values and it will calculate the sum of values in each array and print the name of the array
with the largest sum. If answer is equal, it will print a
Midterms Exam
CSCI 207
Frederick Uy
*/


import java.util.Scanner;

public class MidtermsExam {
    public static void main(String[] args) {

     Scanner scan = new Scanner (System.in);

    int a[] = new int[7];  //int array a of length 7
    int b[] = new int[7];  //int array b of length 7

//calculates sum of a
    int suma = 0;
   for (int i=0; i < a.length; i++){
   a[i] = scan.nextInt();
    suma = suma + a[i];
    }

 //calculates sum of b
    int sumb = 0;
   for (int j=0; j <b.length; j++){
   b[j] = scan.nextInt();
   sumb = sumb + b[j];
   }

   //if sum of a is bigger, it will print a
   if (suma > sumb){
   System.out.println("a");
   }

   //if sum of b is bigger, it will print b
   else if (sumb > suma){
   System.out.println("b");
   }

   //if equal, it will print a
   else{
   System.out.println("a");
   }

}
}
```

Question 2 [10 points]

This program will take seven numbers from the user and it will calculate the sum, of the values in each array. Then, it will print the name of the array with the largest sum. In addition, if the sums are equal, it will also print a. The name of my first array is "a" and the name of the second array is "b". For this program. I utilized a scanner to get input from the user. Then, I also created int values for "suma" and "sumb" to calculate the sum of the arrays. Then, I used if statements so that if "suma" is bigger than "sumb" it would print the name of the array (a) instead, and vice versa. Lastly, I used else if statement that if it does not meet the following commands above, it will print "a" which means that they both got the same sum.

Looking back in our previous lab and lab assignments, I think that this lab can be directly related to the lab 4 which is about calculating and printing the largest, smallest, and the sum of 6 values given by the user. The main difference is that for lab4, it utilizes a command line argument, and for this program, it utilizes a scanner. Another lab assignment that this is related with is the Wheel of fortune game. Although this homework is a lot simpler, it still utilizes the idea of asking user for a value and calculating values to produce a value and replaces it if it satisfies a certain argument. In this example, it prints the name of the array with the largest sum, and if it is equal, it prints A.

Lastly, in real life application, this can be applied in various applications such as in a competition with various players. For example, in a spelling bee competition, each player is asked to spell 10 words. Each correct answer has a specific value attached to it. The harder the word to spell, the higher the attached value. If the player gets the answer right, it will be recorded in an array. At the end, the player who got the most answer will show the name of the winner (the name of the array) instead of the total points the player got. At the same time, if the scores are equal, it will say equal.