

**CSCI 1100 – Fall 2016**  
**Assignment 1 – Due Sunday Oct. 23 at 11:00 pm (night time)**  
**Submit on Brightspace**

**Name: Jeremy Peters**

**Student ID: B00707976**

<b>Declaration: Please complete this declaration</b>		
1	“This document is entirely my own work.” If no, acknowledge any assistance below; outside help should only be used to help you understand the questions NOT to provide the solutions.	Yes/no.
2	I obtained help to complete this document (e.g., from a TA).	Yes/no. If Yes, give Details.
3	This document contains some guidance from the Internet or another document or file or program (e.g., Java's API).	Yes/no. <u>If Yes, give details and provide references.</u> I looked up information on the keyboard buffer from the textbook “Starting Out with Java from Control Structures through Data Structures,” written by Tony Gaddis, and Godfrey Muganda.

**Question 1.** Write a Java application that displays a triangle 4 lines high, made up of two different numbers. The program will ask a user to input two numbers: one between 0 and 4 and one between 5 and 9. The first and third line of the triangle will be filled with the first number; the second and last line of the triangle will be filled with the second number.

*Sample output:*

```
Input a number between 0-4: 2
Input a number between 5-9: 6
```

```
  2
 6 6
2 2 2
6 6 6 6
```

Provide a printout of properly formatted source code (your entire Java program).

```
/*CSCI 1100-Assignment 1-"Question 1"
This program will print a triangular formation of a series of two
different digits in alternating rows. These digits will be
inputted by the user.
<Jeremy Peters><B00707976> <Oct 23, 2016>*/
import java.util.Scanner;
public class A1Q1{
    public static void main(String[] args){
        Scanner keyboard = new Scanner(System.in);
        int x1, x2;
        System.out.print("Input a number between 0-4: ");
        x1 = keyboard.nextInt();
        System.out.print("Input a number between 5-9: ");
        x2 = keyboard.nextInt();
        /*The scanner object reads the user input and stores that
        input as x1.*/
        System.out.println("  " + x1);
        System.out.println("  " + x2 + " " + x2);
        System.out.println(" " + x1 + " " + x1 + " " + x1);
        System.out.println(x2 + " " + x2 + " " + x2 + " " + x2);
        /*Each line of the System.out.print command prints a row
        of the triangle. The spaces align the digits.*/
    }
}
```

Provide 2 example outputs/test cases.

```
└ ----jGRASP exec: java A1Q1
▶▶ Input a number between 0-4: 1
▶▶ Input a number between 5-9: 9
    1
   9 9
  1 1 1
 9 9 9 9
```

```

| ----jGRASP: operation complete.

| ----jGRASP exec: java A1Q1
▶▶ Input a number between 0-4: 4
▶▶ Input a number between 5-9: 5
    4
    5 5
    4 4 4
    5 5 5 5
| ----jGRASP: operation complete.
▶▶

```

**Question 2.** Write a Java application that displays an upside down triangle 4 lines high, made up of one of two possible characters: *asterisk* [\*] **OR** *Dollar Sign* [\$]. The program will ask a user to enter one number between 1 and 10. If the number is odd, the triangle will be made up of asterisks; if the number is even it will be made up of \$ characters. (Hint – look at the modulus operator to help you!)

**Sample outputs:**

Input a number between 1-10: 7

```

* * * *
* * *
* *
*

```

Provide a printout of properly formatted source code (your entire Java program).

```

/*CSCI 1100-Assignment 1-"Question 2"
This program will print a triangular formation of characters.
These characters are asterisks if the user inputs an odd number,
or dollar signs if the user inputs an even number.
<Jeremy Peters><B00707976> <Oct 23, 2016>*/
import java.util.Scanner;
public class A1Q2{
    public static void main(String[] args){
        Scanner keyboard = new Scanner(System.in);
        int x1;
        char symbol;
        System.out.print("Input a number between 1-10: ");
        x1 = (keyboard.nextInt())%2;
        /*The scanner object reads the user input, computes that
        input mod 2, and stores that result as x1.*/
        if (x1 == 1){/*If the user entered an odd number.*/
            symbol = '*';
        }
    }
}

```

```

        else{/*If the user entered an even number*/
            symbol = '$';
        }
        System.out.println(symbol + " " + symbol + " " + symbol +
                            " " + symbol);
        System.out.println(" " + symbol + " " + symbol + " " +
                            symbol);
        System.out.println("  " + symbol + " " + symbol);
        System.out.println("   " + symbol);
        /*Each line of the System.out.print command prints a row
        of the triangle. The spaces align the digits.*/
    }
}

```

Provide 2 example outputs/test cases: one that shows a printout of asterisks and one that shows a printout of \$ characters.

```

    ----jGRASP exec: java A1Q2
    ▶▶ Input a number between 1-10: 3
        * * * *
          * * *
            * *
              *
                *
    ----jGRASP: operation complete.

```

```

    ----jGRASP exec: java A1Q2
    ▶▶ Input a number between 1-10: 8
        $ $ $ $
          $ $ $
            $ $
              $
                $
    ----jGRASP: operation complete.

```

▶▶

**Question 3.** Write a Java application that plays a word game with the user. The program asks the user to enter the following:

- Your name
- One number between 1 and 10
- Your favorite sport
- Your hometown
- Your favorite movie
- Your favorite animal

Once the user enters these items, the program will display one of two stories depending on what number they entered. If the number they entered is less than 5 it will print the following by replacing the inputted variables into the correct locations:

```
THIS IS THE BEST STORY EVER
My name is <name>. And I'm from <hometown>.
My favorite sport is <sport>. I have a pet <animal>
who I take to watch my favorite movie <movie>.
The End!
```

If the number they entered is 5 or greater, it will print the following:

```
THIS IS THE BEST STORY EVER
My name is <name> and I am a <animal>. I live in <hometown> and love to
watch <movie> while trying to play <sport>. It does not work out so
well though.
The End!
```

Here is a sample run. Words in orange are user input from the keyboard.

```
Enter your name: Bob
Enter a number from 1 to 10: 7
Enter your favorite sport: hockey
Enter your hometown: Halifax
Enter your favorite movie: Star Wars
Enter your favorite animal: snake
```

```
THIS IS THE BEST STORY EVER
My name is Bob and I am a snake. I live in Halifax and love to watch
Star Wars while trying to play hockey. It does not work out so well
though.
The End!
```

Provide a printout of properly formatted source code (your entire Java program).

```
/*CSCI 1100-Assignment 1-"Question 3"
This program will tell the user one of two different stories.
Both stories use personal information that the user enters.
The story the computer chooses will depend on the number that
the user enters.
<Jeremy Peters><B00707976> <Oct 23, 2016>*/

import java.util.Scanner; /*Imports scanner object.*/
public class A1Q3{
    public static void main(String[] args){
        Scanner keyboard = new Scanner(System.in);
        /*This sets up the scanner object.*/
        int number;
        String name, sport, hometown, movie, animal;
        /*The variables are ready to store values.*/
        System.out.println("Please enter the following: ");
        System.out.print("One number between 1 and 10: ");
```

```

number = keyboard.nextInt();
/*This number affects what story is told*/
keyboard.nextLine();
/*This command is necessary to remove the newline
character that the nextInt method left behind in the
keyboard buffer. Once this character is removed, the
nextLine method can read the keyboard input without
stopping prematurely at a newline character.*/
System.out.print("Your name: ");
name = keyboard.nextLine();
System.out.print("Your favourite sport: ");
sport = keyboard.nextLine();
System.out.print("Your hometown: ");
hometown = keyboard.nextLine();
System.out.print("Your favourite movie: ");
movie = keyboard.nextLine();
System.out.print("Your favourite animal: ");
animal = keyboard.nextLine();
/*The user has now entered his information.*/
if (number < 5){
/*One story will be told if the user entered a number
less than 5.*/
    System.out.println("\nTHIS IS THE BEST STORY EVER");
    System.out.println("My name is " + name +
        ". And I'm from " + hometown + ".");
    System.out.println("My favorite sport is " + sport +
        ". I have a pet " + animal);
    System.out.println("who I take to watch my favorite" +
        " movie " + movie + ".");
    System.out.println("The End!");
}
else {
/*A different story is told if the user enters a number
greater than or equal to 5.*/
    System.out.println("\nTHIS IS THE BEST STORY EVER");
    System.out.println("My name is " + name +
        " and I am a " + animal + ".");
    System.out.println("I live in " + hometown +
        " and love to watch " + movie +
        " while ");
    System.out.println("trying to play " + sport + ".");
    System.out.println("It does not work out so well " +
        "though.");
    System.out.println("The End!");
}
}
}

```

Provide 2 example output/test cases (to show both stories).

```

[ ----jGRASP exec: java A1Q3
Please enter the following:
▶▶ One number between 1 and 10: 6
▶▶ Your name: Jeremy
▶▶ Your favourite sport: basketball
▶▶ Your hometown: Nova Scotia
▶▶ Your favourite movie: Lord of the Rings
▶▶ Your favourite animal: cat

THIS IS THE BEST STORY EVER
My name is Jeremy and I am a cat.
I live in Nova Scotia and love to watch Lord of the Rings
while
    trying to play basketball.
    It does not work out so well though.
    The End!

[ ----jGRASP: operation complete.

[ ----jGRASP exec: java A1Q3
Please enter the following:
▶▶ One number between 1 and 10: 2
▶▶ Your name: Mike
▶▶ Your favourite sport: soccer
▶▶ Your hometown: Vancouver
▶▶ Your favourite movie: The Hobbit
▶▶ Your favourite animal: dog

THIS IS THE BEST STORY EVER
My name is Mike. And I'm from Vancouver.
My favorite sport is soccer. I have a pet dog
who I take to watch my favorite movie The Hobbit.
The End!

[ ----jGRASP: operation complete.
▶▶

```

**Question 4.** Write a program that will calculate the cost of airline ticket for Air Canada. There is a special sale on flights for students and senior citizens. If a person is a student there is a 20% discount; and if a person is a senior citizen there is a 10% discount. If a person is neither a student nor a senior citizen, no discount applies to their ticket price.

Use a Scanner object to read in the regular price of an airline ticket plus the sales tax rate on the ticket (e.g., the HST rate), and find out if the person flying is a student or a senior citizen. You will apply the discount (if applicable) to the cost of the ticket BEFORE adding the tax. After you apply the discount, then apply the tax and print the total. See the sample below for proper formatting of the output. *Do not concern yourself with controlling the display of the decimal point.*

**Sample 1:**

```
Enter the price of the flight: $ 300
Enter the tax rate: % 10
Enter 1 for student, 2 for senior, or 3 for neither: 1
Discount on price of ticket: $60.0
Price of ticket (after discount): $240.0
Tax: $24.0
Total: $264.0
```

**Sample 2:**

```
Enter the price of the flight: $ 300
Enter the tax rate: % 10
Enter 1 for student, 2 for senior, or 3 for neither: 3
Discount on price of ticket: $0.0
Price of ticket (after discount): $300.0
Tax: $30.0
Total: $330.0
```

Provide a printout of properly formatted source code (your entire Java program).

```
/*CSCI 1100-Assignment 1-"Question 4"
This program will calculate the final cost of an airline
ticket after applying a discount (which depends on whether
the user is a student, senior, or neither), and then applying
tax.
<Jeremy Peters><B00707976> <Oct 23, 2016>*/

import java.util.Scanner;
public class A1Q4{
    public static void main(String[] args){
        Scanner keyboard = new Scanner(System.in);
        int num;
        double regPrice, taxRate, discountRate, discountPrice,
            priceAfterDiscount, tax, total;
        System.out.print("Enter the price of the flight: $");
        regPrice = keyboard.nextDouble();
        /*The user enters the initial price of the ticket.*/
        System.out.print("Enter the tax rate (as a %): ");
        taxRate = (keyboard.nextDouble())/100;
        /*The user enters the tax rate as a percentage, and
        the computer converts this percentage to a decimal,
        and stores the resulting value.*/
        System.out.print("Enter 1 for student, 2 for senior, " +
            "or 3 for neither: ");
        num = keyboard.nextInt();
        if (num == 1){
```



```

        /*For 20% student discount rate*/
        discountRate = 0.2;
    }
    else if (num == 2){
        /*For 10% senior discount rate.*/
        discountRate = 0.1;
    }
    else{
        /*For all others to whom the discount doesn't apply*/
        discountRate = 0;
    }
    discountPrice = discountRate * regPrice;
    /*Applies the relevant discount rate to the original
    price to get the discount on the ticket price.*/
    System.out.println("Discount on price of ticket: $" +
        discountPrice);
    priceAfterDiscount = regPrice - discountPrice;
    /*Subtracts the discount on the ticket price from the
    original price.*/
    System.out.println("Price of ticket (after discount): " +
        "$" + priceAfterDiscount);
    tax = taxRate * priceAfterDiscount;
    /*Applies the tax rate the user entered to the price
    after the discount was applied.*/
    System.out.println("Tax: $" + tax);
    total = priceAfterDiscount + tax;
    /*Calculates the total after discount and tax.*/
    System.out.println("Total: $" + total);
}
}

```

Provide 3 example outputs/test cases different than above: one that shows a printout of the sale of ticket for a student, a senior and a person who is neither a student nor a senior.

```

----jGRASP exec: java A1Q4
>> Enter the price of the flight: $100
>> Enter the tax rate (as a %): 15
>> Enter 1 for student, 2 for senior, or 3 for neither: 1
    Discount on price of ticket: $20.0
    Price of ticket (after discount): $80.0
    Tax: $12.0
    Total: $92.0

----jGRASP: operation complete.

```

```

----jGRASP exec: java A1Q4
>> Enter the price of the flight: $100
>> Enter the tax rate (as a %): 15

```

```

>> Enter 1 for student, 2 for senior, or 3 for neither: 2
Discount on price of ticket: $10.0
Price of ticket (after discount): $90.0
Tax: $13.5
Total: $103.5

----jGRASP: operation complete.

----jGRASP exec: java A1Q4
>> Enter the price of the flight: $100
>> Enter the tax rate (as a %): 15
>> Enter 1 for student, 2 for senior, or 3 for neither: 3
Discount on price of ticket: $0.0
Price of ticket (after discount): $100.0
Tax: $15.0
Total: $115.0

----jGRASP: operation complete.
>>

```

**Question 5.** Create a Java program that asks the user to enter four numbers between 0-9, then checks to see how many pairs of numbers there are (in any order). There can be 0 pairs, 1 pair, or 2 pairs. Once a number is assigned to a pair, it cannot be used to form a different pair. For example:

If the user enters 0 1 0 1 the output would be 2 pair.  
 If the user enters 9 9 5 5 the output would be 2 pair.  
 If the user enters 9 9 9 9 the output would be 2 pair.  
 If the user enters 9 9 9 5 the output would be 1 pair.  
 If the user enters 3 2 1 3 the output would be 1 pair.  
 If the user enters 5 6 7 1 the output would be 0 pair.  
 If the user enters 1 0 1 1 the output would be 1 pair.

See below for a sample of runs.

```

[ ----jGRASP exec: java Pairs_A1
Enter four numbers all between 0 and 9: 1 0 1 1
There are 1 pair

[ ----jGRASP: operation complete.

[ ----jGRASP exec: java Pairs_A1
Enter four numbers all between 0 and 9: 7 9 1 2
There are 0 pair

[ ----jGRASP: operation complete.

[ ----jGRASP exec: java Pairs_A1
Enter four numbers all between 0 and 9: 1 2 3 1
There are 1 pair

[ ----jGRASP: operation complete.

[ ----jGRASP exec: java Pairs_A1
Enter four numbers all between 0 and 9: 3 2 3 2
There are 2 pair

[ ----jGRASP: operation complete.

```

Provide a printout of properly formatted source code (your entire Java program).

```

/*CSCI 1100-Assignment 1-"Question 5"
This program will identify the number of pairs of numbers
that the user enters.
<Jeremy Peters><B00707976> <Oct 23, 2016>*/

import java.util.Scanner;
public class A1Q5{
    public static void main(String[] args){
        Scanner keyboard = new Scanner(System.in);
        int a, b, c, d, n;
        /*a, b, c, d are the user entered numbers,
        and n is the number of pairs.*/
        System.out.println("Enter 4 numbers all from 0 and 9");
        System.out.print("1st number: ");
        a = keyboard.nextInt();
        System.out.print("2nd number: ");
        b = keyboard.nextInt();
        System.out.print("3rd number: ");
        c = keyboard.nextInt();
        System.out.print("4th number: ");
        d = keyboard.nextInt();
        /*The following statements tell the computer to determine
        whether every possible pair of numbers are equal.
        This determines the value of n, the number of pairs.*/
        if (a == b){
            if (c == d){
                n = 2; //Two pair: (a,b) and (c,d)
            }
            else{
                n = 1; //One pair: (a,b)
            }
        }
        else if (a == c){

```

```

        if (b == d) {
            n = 2; //Two pair: (a,c) and (b,d)
        }
        else{
            n = 1; //One pair: (a,c)
        }
    }
    else if (a == d) {
        if (b == c) {
            n = 2; //Two pair: (a,d) and (b,c)
        }
        else{
            n = 1; //Two pair: (a,b) and (c,d)
        }
    }
    /*At this point, "a" cannot be paired.
    There can be at most one pair.*/
    else if (b == c) {
        n = 1; //One pair: (b,c)
    }
    else if (b == d) {
        n = 1; //One pair: (b,d)
    }
    /*At this point, "b" cannot be paired.*/
    else if (c == d) {
        n = 1; //One pair: (c,d)
    }
    else{
        n = 0; //No pairs can be found.
        "a," "b," "c," and "d" are all unique.*/
    }
    System.out.print("There are " + n + " pair.");
}
}

```

Provide 4 example outputs/test cases: showing all four possibilities – no pairs, 1 pair, 2 pairs and a case with 1 pair when three numbers match.

```

    ----jGRASP exec: java A1Q5
Enter 4 numbers all from 0 and 9
▶▶▶▶ 1st number: 1
▶▶▶▶ 2nd number: 2
▶▶▶▶ 3rd number: 3
▶▶▶▶ 4th number: 4
There are 0 pair.
    ----jGRASP: operation complete.

```

```

    ----jGRASP exec: java A1Q5
Enter 4 numbers all from 0 and 9

```

```

>> 1st number: 5
>> 2nd number: 6
>> 3rd number: 7
>> 4th number: 5
    There are 1 pair.
    ----jGRASP: operation complete.

```

```

    ----jGRASP exec: java A1Q5
    Enter 4 numbers all from 0 and 9
>> 1st number: 8
>> 2nd number: 9
>> 3rd number: 9
>> 4th number: 8
    There are 2 pair.
    ----jGRASP: operation complete.

```

```

    ----jGRASP exec: java A1Q5
    Enter 4 numbers all from 0 and 9
>> 1st number: 0
>> 2nd number: 7
>> 3rd number: 7
>> 4th number: 7
    There are 1 pair.
    ----jGRASP: operation complete.

```

```
>>
```

**Question 6.** Write a Java program that uses a Scanner object to read in three numbers and determines the smallest and largest numbers without using Java's Math Class.

Ask a user to enter three numbers. Then print the sum of the three numbers, the largest of the three numbers and the smallest of the three numbers. Finally, print all three numbers from largest to smallest. See the sample below for proper formatting of the output. *Do not concern yourself with controlling the display of the decimal point. You can assume all three numbers are different.*

Sample:

```
Enter three numbers: 22 10 30
```

```
The sum of 22.0, 10.0, and 30.0 is 62.0
```

```
The largest number is 30.0
```

```
The smallest number is 10.0
```

```
The numbers from largest to smallest are: 30.0, 22.0, and 10.0
```

Provide a printout of properly formatted source code (your entire Java program).

```

/*CSCI 1100-Assignment 1-"Question 6"
This program will sort three numbers from largest to smallest and

```

sum them. These three numbers are entered by the user.  
<Jeremy Peters><B00707976> <Oct 23, 2016>\*/

```
import java.util.Scanner;
public class AlQ6{
    public static void main(String[] args){
        Scanner keyboard = new Scanner(System.in);
        int a, b, c, max, med, min, sum;
        /*All variables are declared.
        a, b, c will store the user input, max, med, min could each
        be any one of a, b, or c depending on the relative values
        of a, b, and c. sum is the sum of a, b, and c.*/
        System.out.println("Enter three different numbers: ");
        System.out.print("First number: ");
        a = keyboard.nextInt();
        System.out.print("Second number: ");
        b = keyboard.nextInt();
        System.out.print("Third number: ");
        c = keyboard.nextInt();
        /*The following statements will compare each number to each
        other number and store the largest number as the variable
        "max," the second largest number as the variable "med," and
        the smallest number as the variable "min."*/
        if (a>b && a>c){//Checks if a is the largest number.
            max = a;
            if (b>c){//Compares the lower two numbers b and c.
                med = b;
                min = c;
            }
            else{//c<b.
                med = c;
                min = b;
            }
        }
        else if (b>a && b>c){//Checks if b is the largest number.
            max = b;
            if (a>c){//Compares the lower two numbers a and c.
                med = a;
                min = c;
            }
            else{//c<a.
                med = c;
                min = a;
            }
        }
        else{//The final possibility: c is the largest number.
            max = c;
            if (a>b){//Compares the lower two numbers a and b.
                med = a;
```

```

        min = b;
    }
    else{ //b<a
        med = b;
        min = a;
    }
}
sum = a + b + c;
/*The results of comparison are now printed.*/
System.out.println("The sum of "+a+", "+b+", and " +
                    +c+" is "+sum);
System.out.println("The largest number is "+max);
System.out.println("The smallest number is "+min);
System.out.print("The numbers from largest to smallest "+
                  "are: "+max+", "+med+", "+min+".");
}
}

```

Provide 3 example outputs/test cases (different from above that shows all possible cases).

```

----jGRASP exec: java A1Q6
Enter three different numbers:
>> First number: 10
>> Second number: 5
>> Third number: 1
The sum of 10, 5, and 1 is 16
The largest number is 10
The smallest number is 1
The numbers from largest to smallest are: 10, 5, 1.
----jGRASP: operation complete.

```

```

----jGRASP exec: java A1Q6
Enter three different numbers:
>> First number: 10
>> Second number: 5
>> Third number: 7
The sum of 10, 5, and 7 is 22
The largest number is 10
The smallest number is 5
The numbers from largest to smallest are: 10, 7, 5.
----jGRASP: operation complete.

```

```

----jGRASP exec: java A1Q6
Enter three different numbers:
>> First number: 10
>> Second number: 15

```

```

▶▶ Third number: 5
    The sum of 10, 15, and 5 is 30
    The largest number is 15
    The smallest number is 5
    The numbers from largest to smallest are: 15, 10, 5.
    ----jGRASP: operation complete.

    ----jGRASP exec: java A1Q6
    Enter three different numbers:
▶▶ First number: 10
▶▶ Second number: 30
▶▶ Third number: 23
    The sum of 10, 30, and 23 is 63
    The largest number is 30
    The smallest number is 10
    The numbers from largest to smallest are: 30, 23, 10.
    ----jGRASP: operation complete.

    ----jGRASP exec: java A1Q6
    Enter three different numbers:
▶▶ First number: 10
▶▶ Second number: 5
▶▶ Third number: 20
    The sum of 10, 5, and 20 is 35
    The largest number is 20
    The smallest number is 5
    The numbers from largest to smallest are: 20, 10, 5.
    ----jGRASP: operation complete.

    ----jGRASP exec: java A1Q6
    Enter three different numbers:
▶▶ First number: 5
▶▶ Second number: 10
▶▶ Third number: 20
    The sum of 5, 10, and 20 is 35
    The largest number is 20
    The smallest number is 5
    The numbers from largest to smallest are: 20, 10, 5.
    ----jGRASP: operation complete.
▶▶

```