CSCI 145 PA __5__ Submission

Due Date:__April 3, 2023__ Late (date and time):_________________

Name(s):___Ivan Leung___ & _______________________

Exercise 1 -- need to submit source code and I/O
-- check if completely done __x__ ; otherwise, discuss issues below

Pseudocode below if applicable:

Source code below:

```
package pa5;

/*  Java Class: CSCI 145
Modified by: Ivan Leung
Class: Mon/Wed
Date: Mar 22 2023
Description:

I certify that the code below is my own work.

Exception(s): N/A

*/

//**************************************************************
//Salary.java
//
//Computes the amount of a raise and the new
//salary for an employee. The current salary
//and a performance rating (a String: "Excellent",
//"Good" or "Poor") are input.
//**************************************************************
import java.util.Scanner;
import java.text.NumberFormat;

public class Salary {
     public static void main (String[] args)
     {
           double currentSalary;
           //employee's current salary
           double raise;
           //amount of the raise
           double newSalary;
           //new salary for the employee
           String rating;
           //performance rating
           Scanner scan = new Scanner(System.in);
           System.out.print ("Enter the current salary: ");
           currentSalary = scan.nextDouble();
           System.out.print ("Enter the performance rating
(Excellent, Good, or Poor): ");
           rating = scan.next();
           scan.close();
           //Compute the raise using if ...
           if (rating.equalsIgnoreCase("Excellent"))
                raise = currentSalary * 0.06;
           else if (rating.equalsIgnoreCase("Good"))
                raise = currentSalary * 0.04;
           else
                raise = currentSalary * 0.015;

           newSalary = currentSalary + raise;
           //Print the results
           NumberFormat money =
NumberFormat.getCurrencyInstance();
           System.out.println();
           System.out.println("Current Salary:    " +
money.format(currentSalary));
           System.out.println("Amount of your raise: " +
money.format(raise));
           System.out.println( "Your new salary: " + money.
format (newSalary) );
           System.out.println();
     }

}
```

Input/output below:

```
Enter the current salary: 5489.32
Enter the performance rating (Excellent, Good, or Poor):
excellent

Current Salary:  $5,489.32
Amount of your raise: $329.36
```

```
Your new salary: $5,818.68
```

Exercise 2 -- need to submit source code and I/O
-- check if completely done __x__ ; otherwise, discuss issues below

Pseudocode below if applicable:

Source code below:

```java
package pa5;

/*  Java Class: CSCI 145
Modified by: Ivan Leung
Class: Mon/Wed
Date: Mar 22 2023
Description:

I certify that the code below is my own work.

Exception(s): N/A

 */

//**************************************************************
//Guess.java
//
//Play a game where the user guesses a number from 1 to 10
//
//**************************************************************
import java.util.Scanner;
import java.util.Random;

public class Guess {
	public static void main(String[] args)
	{
		int numToGuess; //Number the user tries to guess
		int guess;
		int totalGuess = 1;
		int tooHigh = 0;
		int tooLow = 0;
		//The user's guess
```

```
        Scanner scan = new Scanner(System.in);
        Random generator = new Random();
        //randomly generate the number to guess
        numToGuess = generator.nextInt(10) + 1;
        //print message asking user to enter a guess
        System.out.print("Enter a integer: ");
        //read in guess
        guess = scan.nextInt();
        while (guess != numToGuess) //keep going as long as
the guess is wrong
        {
            //print message saying guess is wrong
            System.out.println("You guess it wrong!\nPlease
try again!");
            if (guess > numToGuess) {
                System.out.println("Your guess is too
high!");

                ++tooHigh;
            }
            else {
                System.out.println("Your guess is too
low!");

                ++tooLow;
            }
            //get another guess from the user
            System.out.print("Enter a integer: ");
            guess = scan.nextInt();
            ++totalGuess;
        }
        scan.close();
        //print message saying guess is right
        System.out.println("Congratulation! You guessed the
correct number!");
        System.out.println("You made a total of " + totalGuess
+ " guesses");
        System.out.println("You made " + tooHigh + " guesses
too high");
        System.out.println("You made " + tooLow + " guesses
too low");
    }
}
```

Input/output below:

```
Enter a integer: 5
You guess it wrong!
Please try again!
Your guess is too low!
Enter a integer: 7
Congratulation! You guessed the correct number!
You made a total of 2 guesses
You made 0 guesses too high
You made 1 guesses too low
```

Exercise 3 -- need to submit source code and I/O
-- check if completely done __x__ ; otherwise, discuss issues below

Pseudocode below if applicable:

Source code below:

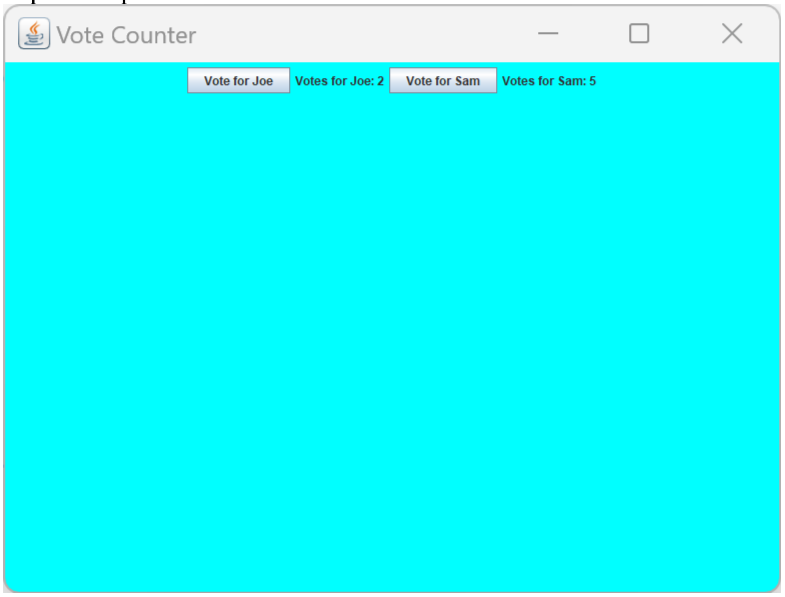
```java
package pa5;

/*  Java Class: CSCI 145
Modified by: Ivan Leung
Class: Mon/Wed
Date: Mar 22 2023
Description:

I certify that the code below is my own work.

Exception(s): N/A

*/

//***********************************************************
//VoteCounterPanel.java
//
//Panel for the GUI that tallies votes for two candidates,
//Joe and Sam.
//***********************************************************

import java.awt.*;
import java.awt.event.*;
import javax.swing. *;
```

```
public class VoteCounterPanel extends JPanel
{
    private int votesForJoe;
    private JButton joe;
    private JLabel labelJoe;

    private int votesForSam;
    private JButton sam;
    private JLabel labelSam;
    //----------------------------------------------------
    //Constructor: Sets up the GUI.
    //----------------------------------------------------
    public VoteCounterPanel()
    {
        votesForJoe = 0;
        joe = new JButton("Vote for Joe");
        joe.addActionListener(new JoeVoteButtonListener());
        labelJoe = new JLabel("Votes for Joe: " +
votesForJoe);
        add(joe);
        add(labelJoe);
        setPreferredSize(new Dimension(300, 40));

        votesForSam = 0;
        sam = new JButton("Vote for Sam");
        sam.addActionListener(new SamVoteButtonListener());
        labelSam = new JLabel("Votes for Sam: " +
votesForSam);
        add(sam);
        add(labelSam);
        setPreferredSize(new Dimension(300,40));

        setBackground(Color.cyan);
    }
    //*****************************************************
    //Represents a listener for button push (action) events
    //*****************************************************
    private class JoeVoteButtonListener implements
ActionListener
    {
        //----------------------------------------------------
        //Updates the appropriate vote counter when a
        //button is pushed for one of the candidates.
        //----------------------------------------------------
        public void actionPerformed(ActionEvent event)
        {
```

```
                 ++votesForJoe;
                 labelJoe.setText("Votes for Joe: " +
votesForJoe);

           }

     }

     private class SamVoteButtonListener implements
ActionListener
     {
           //---------------------------------------------
           //Updates the appropriate vote counter when a
           //button is pushed for one of the candidates.
           //---------------------------------------------
           public void actionPerformed(ActionEvent event)
           {
                 ++votesForSam;
                 labelSam.setText("Votes for Sam: " +
votesForSam);

           }

     }

}

package pa5;

/*  Java Class: CSCI 145
Modified by: Ivan Leung
Class: Mon/Wed
Date: Mar 22 2023
Description:

I certify that the code below is my own work.

Exception(s): N/A

*/

//*************************************************************
//VoteCounter.java
//
//Demonstrates a graphical user interface and event
//listeners to tally votes for two candidates, Joe and Sam.
```

```
//*************************************************************

import javax.swing.JFrame;
public class VoteCounter
{
	//-----------------------------------------
	//Creates the main program frame.
	//-----------------------------------------
	public static void main(String[] args)
	{
		JFrame frame = new JFrame("Vote Counter");
		frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
		frame.getContentPane().add(new VoteCounterPanel());
		frame.pack();
		frame.setSize(800, 600);
		frame.setVisible(true);
	}

}
```

Input/output below:



Add more exercises as needed

Exercise 4 -- need to submit source code and I/O
  -- check if completely done __x__ ; otherwise, discuss issues below

Pseudocode below if applicable:

Source code below:

```
package pa5;

//Class Player for CSCI 145 PA 4 Spring 2023
//Modified by: Ivan Leung

/*  Java Class: CSCI 145
Modified by: Ivan Leung
Class: Mon/Wed
Date: Mar 22 2023
Description:

I certify that the code below is my own work.

Exception(s): N/A

*/

import java.util.*;

//Class Player represents one roulette player.
class Player {
        private static final int RELOAD_AMOUNT = 100;
        private int bet, money, betType, number;
        private String name;

        // The Player constructor sets up name and initial available
money.
        public Player(String playerName, int initialMoney) {
                name = playerName;
                money = initialMoney;
        }

        // Returns this player's name.
        public String getName() {
                return name;
        }

        // Returns this player's current available money.
        public int getMoney() {
                return money;
        }

        // Prompts the user and reads betting information.
```

```
    public void makeBet(Scanner scan) {
        System.out.print("Enter a bet option, " + name + " (1,
2, or 3): ");
        betType = scan.nextInt();
        while (betType < 1 || betType > 3) {
            System.out.print("Invalid betting option.Try
again.\nEnter a bet option, " + name + " (1, 2, or 3): ");
            betType = scan.nextInt();
        }
        if (betType == 3) {
            System.out.print("Enter a number between 1 and
36: ");
            number = scan.nextInt();
            while (number < Roulette.MIN_NUM || number >
Roulette.MAX_NUM) {
                System.out.print("Invalid number. Try
again.\nEnter a number between 1 and 36: ");
                number = scan.nextInt();
            }
        }
        System.out.print("How much to bet: ");
        bet = scan.nextInt();
        while (bet < Roulette.MIN_BET || bet > money) {
            System.out.print("Invalid betting amount. Try
again.\nHow much to bet: ");
            bet = scan.nextInt();
        }
        System.out.print("You chose to bet $" + bet + " on ");
        if (betType == 1)
            System.out.println("Black color");
        else if (betType == 2)
            System.out.println("Red color");
        else
            System.out.println("number " + number + ".");

        money = money - bet;
    }

    // Determines if the player wants to play again.
    public boolean playAgain(Scanner scan) {
        String answer;

        System.out.print("Play again, " + name + "? [y/n] ");
        answer = scan.next();
        return (answer.equals("y") || answer.equals("Y"));
    }
    // payment method (determines winnings)
    public int payment() {
        return 0;
    }

}
package pa5;

//Class Roulette for CSCI 145 PA 4 Spring 2023
//Modified by: Ivan Leung

/*  Java Class: CSCI 145
Modified by: Ivan Leung
Class: Mon/Wed
Date: Mar 22 2023
Description:

I certify that the code below is my own work.

Exception(s): N/A

*/

import java.util.*;

//Class Roulette represents a roulette betting game.
class Roulette {
    // public name constants -- accessible to others
    public final static int BLACK = 0; // even numbers
    public final static int RED = 1; // odd numbers
    public final static int GREEN = 2; // 00 OR 0
    public final static int NUMBER = 3; // number bet
    public final static int MIN_NUM = 1; // smallest number to
bet
    public final static int MAX_NUM = 36; // largest number to
bet
    public final static int MIN_BET = 1; // minimum amount to
bet

    // private name constants -- internal use only
    private final static int MAX_POSITIONS = MAX_NUM + 2; //
number of positions on wheel
    private final static int NUMBER_PAYOFF = MAX_NUM - 1; //
payoff for number bet
```

```
    private final static int COLOR_PAYOFF = 2; // payoff for
color bet

    // private variables -- internal use only
    private static int ballPosition = 0; // 00, 0, 1 .. MAX_NUM
    private static int color = GREEN; // GREEN, RED, OR BLACK

    // private variables -- testing only
    private static int next = 0; // next value in the list
    private static int[] randValues = { 20, 5, 0, 1, 36 }; // 5
values

    // Contains the main processing loop for the roulette game.
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        Player player = new Player("Jane", 100); // $100 to
start for Jane
        boolean done = false;
        int currentSpin;

        System.out.println("Author: [Your Name]\n");
        welcomeMessage();

        while (!done) {
            System.out.println("Money available for " +
player.getName() + ": " + player.getMoney());
            betOptions();

            // Add code so player can make a bet
            player.makeBet(scan);
            System.out.println();
            // spin() and display value
            currentSpin = spin();
            System.out.println();
            // Assume player lost a bet so no payment at this
point

            done = !player.playAgain(scan);

            System.out.println();
        }

        System.out.println("Game over!  Thanks for playing.");
        scan.close();
    }
    //
=================================================================
====
    // Presents welcome message
    //
=================================================================
====
    public static void welcomeMessage() {
        System.out.println("Welcome to a simple version of
roulette game.");
        System.out.println("You can place a bet on black, red,
or a number.");
        System.out.println("A color bet is paid " +
COLOR_PAYOFF + " times the bet amount.");
        System.out.println("A number bet is paid " +
NUMBER_PAYOFF + " times the bet amount.");
        System.out.println("You can bet on a number from " +
MIN_NUM + " to " + MAX_NUM + ".");
        System.out.println("Gamble responsibly.  Have fun and
good luck!\n");
    }

    //
=================================================================
====
    // Presents betting options
    //
=================================================================
====
    public static void betOptions() {
        System.out.println("Betting Options:");
        System.out.println("    1. Bet on black (even
numbers)");
        System.out.println("    2. Bet on red (odd numbers)");
        System.out.println("    3. Bet on a number between " +
MIN_NUM + " and " + MAX_NUM);
        System.out.println();
    }

    // Spins the wheel
    public static int spin() {
        int result;

        // use nextRandom() for testing now
        result = nextRandom();
        Roulette.ballPosition = result;
```

```
            if (result == 0 || result == 37)
                  Roulette.color = Roulette.GREEN;
            else if ((result & 1) == 0)
                  Roulette.color = Roulette.BLACK;
            else
                  Roulette.color = Roulette.RED;

            System.out.println("Spinning ...");
            System.out.print("Current number: ");

            if (result == 37)
                  System.out.print("00, color: ");
            else
                  System.out.print(result + ", color: ");

            if (Roulette.color == Roulette.GREEN)
                  System.out.println("Green");
            else if (Roulette.color == Roulette.BLACK)
                  System.out.println("Black");
            else
                  System.out.println("Red");

            return result;

            // comment above code and add your code to spin
      }

      // Payoff method for number bet
      public static int payoff(int betAmt, int betType, int
numberBet) {
            int pay = 0;


            return pay;
      }

      // Returns a simulated "random" value for testing
      // Assume a value between 0 and 36
      public static int nextRandom() {
//          int num = randValues[next];
//          next++;
//          next = next % randValues.length; // back to 0 if
needed
            Random rand = new Random();
            int num = rand.nextInt(Roulette.MAX_POSITIONS);
            return num;
        }

}
```

Input/output below:

```
Author: [Your Name]

Welcome to a simple version of roulette game.
You can place a bet on black, red, or a number.
A color bet is paid 2 times the bet amount.
A number bet is paid 35 times the bet amount.
You can bet on a number from 1 to 36.
Gamble responsibly.  Have fun and good luck!

Money available for Jane: 100
Betting Options:
    1. Bet on black (even numbers)
    2. Bet on red (odd numbers)
    3. Bet on a number between 1 and 36

Enter a bet option, Jane (1, 2, or 3): 0
Invalid betting option.Try again.
Enter a bet option, Jane (1, 2, or 3): 4
Invalid betting option.Try again.
Enter a bet option, Jane (1, 2, or 3): 1
How much to bet: 0
Invalid betting amount. Try again.
How much to bet: 5
You chose to bet $5 on Black color

Spinning ...
Current number: 32, color: Black

Play again, Jane? [y/n] y

Money available for Jane: 95
Betting Options:
    1. Bet on black (even numbers)
    2. Bet on red (odd numbers)
    3. Bet on a number between 1 and 36

Enter a bet option, Jane (1, 2, or 3): 2
```

```
How much to bet: 96
Invalid betting amount. Try again.
How much to bet: 10
You chose to bet $10 on Red color

Spinning ...
Current number: 17, color: Red

Play again, Jane? [y/n] y

Money available for Jane: 85
Betting Options:
    1. Bet on black (even numbers)
    2. Bet on red (odd numbers)
    3. Bet on a number between 1 and 36

Enter a bet option, Jane (1, 2, or 3): 3
Enter a number between 1 and 36: 0
Invalid number. Try again.
Enter a number between 1 and 36: 37
Invalid number. Try again.
Enter a number between 1 and 36: 18
How much to bet: 20
You chose to bet $20 on number 18.

Spinning ...
Current number: 11, color: Red

Play again, Jane? [y/n] n

Game over!  Thanks for playing.
```

Answer for Question 1

We may use either -1 or 37 to represent 00. I used 37 in my code to represent 37. I tested it by setting the result to 37 to see if I get 00 and green color on the ouput.

Answer for Question 2

1. Boolean expression

2. Test for the Boolean expression

3. Statements of the loop

4. Update for the Boolean expression

Extra Credit – provide if applicable

Pseudocode below if applicable:

Source code below:

```
package pa5;

/*  Java Class: CSCI 145
Modified by: Ivan Leung
Class: Mon/Wed
Date: Mar 22 2023
Description:

I certify that the code below is my own work.

Exception(s): N/A

*/

//***************************************************************
*
//BaseballStats.java
//
//Reads baseball data in from a comma delimited file. Each line
//of the file contains a name followed by a list of symbols
//indicating the result of each at bat: h for hit, o for out,
//w for walk, s for sacrifice. Statistics are computed and
//printed for each player.
//***************************************************************
*

import java.util.Scanner;
import java.io.*;
import java.text.DecimalFormat;

public class BaseballStats {
      // -------------------------------------------------
      // Reads baseball stats from a file and counts
      // total hits, outs, walks, and sacrifice flies
      // for each player.
```

```
    // ----------------------------------------------------
    public static void main(String[] args) throws IOException {
        Scanner fileScan, lineScan;
        String fileName;
        String nextLine;
        int hits = 0;
        int outs = 0;
        int walks = 0;
        int sacrifices = 0;
        double averageBatting;
        String batCode;
        DecimalFormat decimal = new DecimalFormat("#.###");
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter the name of the input file: 
");
        fileName = scan.nextLine();
        fileScan = new Scanner(new File(fileName));
        // Read and process each line of the file
        while (fileScan.hasNext()) {
            nextLine = fileScan.nextLine();

            lineScan = new Scanner(nextLine);
            lineScan.useDelimiter(",");
            System.out.println(lineScan.next() + ":");
            while (lineScan.hasNext()) {
                batCode = lineScan.next();
                if (batCode.equalsIgnoreCase("h"))
                    ++hits;
                else if (batCode.equalsIgnoreCase("o"))
                    ++outs;
                else if (batCode.equalsIgnoreCase("w"))
                    ++walks;
                else if (batCode.equalsIgnoreCase("s"))
                    ++sacrifices;
            }
            averageBatting = hits / ((double) hits + outs);
            System.out.println("Total hits:" + hits);
            System.out.println("Total outs:" + outs);
            System.out.println("Total walks:" + walks);
            System.out.println("Total sacrifices:" + 
sacrifices);
            System.out.println("Average batting: " + 
decimal.format(averageBatting));
            lineScan.close();
//              System.out.print(" " + lineScan.next());
```

```
            System.out.println();
        }
        fileScan.close();
        scan.close();
    }

}
```

Input/output below:

```
Enter the name of the input file:
C:\Users\ivanl\OneDrive\Desktop\tmp\stats.dat
Willy Wonk:
Total hits:4
Total outs:11
Total walks:1
Total sacrifices:1
Average batting: 0.267

Shari Jones:
Total hits:7
Total outs:20
Total walks:1
Total sacrifices:3
Average batting: 0.259

Barry Bands:
Total hits:13
Total outs:29
Total walks:6
Total sacrifices:3
Average batting: 0.31

Sally Slugger:
Total hits:17
Total outs:32
Total walks:7
Total sacrifices:3
Average batting: 0.347

Missy Lots:
Total hits:17
Total outs:39
Total walks:8
Total sacrifices:4
Average batting: 0.304
```

```
Joe Jones:
Total hits:23
Total outs:52
Total walks:9
Total sacrifices:4
Average batting: 0.307

Larry Loop:
Total hits:27
Total outs:60
Total walks:10
Total sacrifices:6
Average batting: 0.31

Sarah Swift:
Total hits:29
Total outs:67
Total walks:11
Total sacrifices:6
Average batting: 0.302

Bill Bird:
Total hits:34
Total outs:78
Total walks:12
Total sacrifices:8
Average batting: 0.304

Don Daring:
Total hits:39
Total outs:89
Total walks:12
Total sacrifices:8
Average batting: 0.305

Jill Jet:
Total hits:46
Total outs:99
Total walks:13
Total sacrifices:10
Average batting: 0.317


Enter the name of the input file:
C:\Users\ivanl\OneDrive\Desktop\tmp\stats2.dat
```

```
Barry Bands:
Total hits:6
Total outs:9
Total walks:5
Total sacrifices:0
Average batting: 0.4
```