MIS 768: Java Programming
Spring 2019

Individual Assignment 2

Due Date: **5:29 PM, February 19 2019** (Submit via WebCampus).
Weights: 5% of total grades.

Purpose

- Design a program that saves the output to a file.
- Use **ArrayList** objects in a program, such as adding elements, traversing the elements, and print the entire **ArrayList** object.
- Design a class (with proper attributes and methods) to complete a task.

NOTE:

1. Please submit the **.java** files for each question. Zip all the.java files and upload the zip file to WebCampus for submission.
2. Please provide proper comments to document your code, including the following:
   a. Author's name;
   b. Purpose of the program;
   c. In-line comments for the statements, including the variable/constant declaration and initialization.

Questions 1: Polling System (50 points)

Write a simple polling program that allows users to rate the presidential candidates from 1 (least favorable) to 7 (most favorable). When the program starts up, you can specify the numbers of candidates in your poll and then enter the names of the candidates. You can also specify how many respondents you plan to get. Then have your friends and family to take the survey.

Once the poll is closed, generate a summary of the result and write it to a file, including: (1) a tabular report with the candidates down the left side and the 7 ratings across the top, listing in each column the number of ratings received for each candidate. (2) To the right of each row, show the average of the ratings for that candidate. (3) Which candidate received the highest point total? Display both the candidate and the point total. (4) Which candidate received the lowest point total? Display both the candidate and the point total.

Grading criteria:

(1) Correctness
   (a) The code can be compiled without any syntax error.

(b) The code can generate the requested results. Please make up some numbers to test the program and validate the results
(c) The program is documented using comments.
(d) The program is properly documented using comments.

(2) Technique used
(a) Variable and constant declaration and initialization is done following the naming convention of Java programs. (Note: Constant must be used in this program).
(b) Getting user input is done. Proper messages are shown to the user to give proper instruction.
(c) Arithmetic operators are used.
(d) **`ArrayList`** or **`Array`** is used.
(a) The file is properly opened/created, written, and closed.

Question 2: Shipping Cost Calculation (50 points)

Write a program that computes the shipping cost of a package. The program should prompt the user to enter the weight and the package dimension in inches. However, each package's should not exceed 150 pounds (actual weight) and 110 outside linear inches (length plus width plus height). Overweight or oversize package will not be accepted. If the user enters zero or a negative number, the program should show an error message and request input again.

If the package is accepted, the program then calculates the dimensional weight. Dimensional weight reflects the amount of the space a package occupies in relation to its actual weight; dimensional weight is calculated by the following steps: (1) Determine the package dimensions in inches. For each dimension, any fraction of the measurement should be increased to the nearest whole number (for example, 1.3 will be considered 2). (2) Multiply the package length by the width by the height. The result is the cubic size in inches. (3) Divide the cubic size in inches by 166 to determine dimensional weight in pounds. (4) Increase any fraction to the next whole pound.

Compare the package's actual weight to its dimensional weight. The greater of the two is the billable weight and should be used to calculate the rate. The final cost of the shipment is calculated by multiplying the price per pound ($2.5) by the billable weight. The program shows the shipping costs to the user.

Grading criteria:

(2) Correctness
(a) The code can be compiled without any syntax error.
(b) The code can generate the requested results. Please make up some numbers to test the program and validate the results
(c) The program is properly documented using comments (/**…..*/ and //.).

(3) Technique used
(a) A class **Package** should be designed and implemented, including (but not limited to) the methods determining whether the package is oversize or overweight, calculating the billing weight, calculating the shipping cost. The object(s) of Package should be used in the program to complete the task.
(b) Variable and constant declaration and initialization is done following the naming convention of Java programs. (Note: Constant must be used in this program).
(c) Getting user input and showing result are proper done, including the error messages.
(d) Arithmetic operators are used.