**Lab 29 – Waffle Irons**

Open BlueJ, and create a new BlueJ project titled **Lab29-WaffleIrons** in your CS\LABS folder.

Create a new class called **PracticeProblems**.

Like if / else if / else statements, switch statements are 'flow control' programming constructs. Essentially, a switch statement is multiple if statements in a cleaner, simpler package.

Switch statements are also faster than if statements, as the computer doesn't need to evaluate each if statement – it jumps directly to the proper case.

**Before each numbered problem, insert a COMMENT with the problem number.**

1. Write the code to let the user take this simple one-question quiz about java variables. Use a switch statement to check if they chose correctly; your switch statement should respond by saying “Correct”, “Incorrect” or “Invalid choice”.

Which of the following will declare and initialize a variable in Java?

1. int 2a = 4;

2. double circle-area = 5.78;

3. string s = hello!;

4. boolean whoa\_now = true;

1. Write the code to implement a text-based calculator. You should read in two numbers from the keyboard, then give the user the menu of options: (1) add, (2) subtract, (3) multiply, (4) divide, (5) modulus, and (6) exponentiation. Use a switch statement to handle the appropriate operation. You should have a default statement in case they enter an invalid choice.
2. Your friend started making a peanut butter and jelly sandwich, but can’t figure out what to do next. Ask your friend where they are stuck. Print that step and all steps that follow. The steps to making the sandwich are listed below. **(Be efficient. “Fall through” with a switch statement!)**
3. **Get 2 pieces of bread, a jar of peanut butter and a jar of jelly.**
4. **Open both jars.**
5. **Spread peanut butter on one piece of bread.**
6. **Spoon jelly onto the other piece of bread.**
7. **Put bread together so that the peanut butter and jelly are touching.**
8. **Give the sandwich to your dog and feast on a much better food. Salsa!**

**See the sample output to know what it should look like.**

**Waffle Irons**

Create a class called WaffleIrons. Write a program that allows you to enter orders and determine the total costs of the order. Your online startup sells five different models of single-serving USB waffle irons. The retail prices are as follows:

| **Model 1** | **Model 2** | **Model 3** | **Model 4** | **Model 5** |
| --- | --- | --- | --- | --- |
| $10.98 | $14.50 | $19.98 | $24.49 | $26.87 |

Your program should read in a series of pairs of values, modeland quantity*,* until the user enters -1. You should use a switch statement (rather than a series of if statements) to retrieve the retail price for each product entered. Your program should calculate the sum total of all products sold (until the user enters -1). Once the sentinel-controlled loop is done, print out the final total. (Java does strange things with doubles, so ensure that your final total is rounded to 2 decimal places. Remember the decimal slide trick.) Sample program run shown below (**user input shown in red**):

Enter model >>> **4**

Enter quantity >>> **3**

Enter model >>> **1**

Enter quantity >>> **6**

Enter model >>> **-1**

Total sales >>> $139.35