CMSC 131 Spring 2005 Quiz 2 Worksheet

The second Quiz of the course will be on Wednesday, Feb 9 during your lab (discussion) session. The following list provides more information about the quiz:

- You will have 15 minutes to complete the quiz.
- It will be a written quiz (not using any computer).
- It will be closed-book, closed-notes, and no calculator is allowed.
- Answers must be neat and legible. We recommend that you use pencil and eraser.

The following exercises cover the material to be covered in Quiz #2. Solutions to these exercises will not be provided, but you are welcome to discuss your solutions with TAs and instructors during office hours. When asked to write a Java application, you should give the entire class definition, not just the main method.

1. **From Lewis & Loftus**
   
   a. Page 199-200 (“exercises”) 3.8, 3.9, 3.12, 3.14, 3.17, 3.20
   b. Page 201 (“programming projects”) 3.4, 3.6

2. Which of the following represent valid Java identifiers?
   
   a. $5$x  
   b. _8&house  
   c. Pressure%13  
   d. while  
   e. do  
   f. theHouseFarFromHere

3. Using parentheses indicate the order in which each of the following expressions will be evaluated or whether the expression represents an invalid expression. You can assume all the variables are of type int.
   
   a. x / y * z % w  
   b. x++ + y++  
   c. x + y + z – w % p * 2  
   d. x < y || z > m && y <= 4

4. The factorial of a positive integer n is the product of all integer values from 1 up to n. (The factorial of 0 is defined to be 1.) Write a Java Program that reads a nonnegative integer value from the user (using a JOptionPane) and computes the factorial of that value.

5. Write a Java program that “simulates” the reading password process you go through while logging into a computer account. The program will ask for a password, compare that value against two possible passwords, and print “Welcome” if any of the provided passwords is valid. Otherwise, “Invalid
Password” should be printed. Your program will allow a maximum of three opportunities in order for the user to provide a correct password. Your program must end if after three attempts the user has not provided a correct password. You must use a do-while statement for this problem.

6. Implement Problem (5), but this time using a while statement.

7. Note: This problem was last semester’s quiz. The solution has been provided at the end.

Write a Java program that determines the product of a range of positive integer values. Your program will read two values representing the lower bound and upper bound of the range, compute the product of integers in the range, and print the result. For example, for the range [3,5] your program will generate 60. If the lower bound is greater than or equal to the upper bound your program should print a value of 1. The following restrictions/assumptions apply to this problem:

- The name of the class you will define is **RangeProduct**.
- Use the messages “Enter lower bound” and “Enter upper bound” to read the upper and lower bounds, respectively.
- Use the message “Range product is” followed by the computed product to display the result.
- Input and output operations must be completed using methods associated with JOptionPane.
- You do not need to provide comments, however, you must use meaningful variable names and good indentation.
- You must write any necessary import statements.
- Write the program on the next page.
Problem 7 One possible solution

```java
import javax.swing.*;

public class RangeProduct {
    public static void main(String[] args) {
        String lowerBoundStr, upperBoundStr;
        lowerBoundStr = JOptionPane.showInputDialog("Enter lower bound");
        upperBoundStr = JOptionPane.showInputDialog("Enter upper bound");

        int lowerBound = Integer.parseInt(lowerBoundStr);
        int upperBound = Integer.parseInt(upperBoundStr);

        long product = 1;
        if (lowerBound < upperBound) {
            while (lowerBound <= upperBound) {
                product *= lowerBound;
                lowerBound++;
            }
        }
        JOptionPane.showMessageDialog(null, "Range Product is " + product);
        System.exit(0);
    }
}
```