Recitation 7 Quiz 2

Follow the directions below.

1. Log into cs.bridgewater.edu.
2. Change your working directory to your **csci101/quizzes** directory.
3. In your **csci101/quizzes** directory, create a program in a file named **R7Quiz2.java** that satisfies the Program Requirements shown below.

Program Requirements

1. Print to the screen the string **Recitation 7 Quiz 2**.
2. Print to the screen the string **----------**.

3. Write a method named **printSeason** which simply prints **Fall** to the screen.
4. Call **printSeason** from main.

5. Write a method named **longerString** which takes two Strings as arguments and returns the string which has a longer length. If the strings have equal lengths, return either string.
6. Invoke **longerString** from main, passing to it the string **Hello** and the string **Bye**. Store the value returned by the method in a variable named **longer**. Print to the screen **LongerString:** followed by the value in the variable named **longer**.

7. Write a method named **isOdd** that takes an integer as an argument and returns **true** if the value of the argument is odd, otherwise it returns **false**.
8. In main, ask the user to enter an integer and read the value into a variable named **input**. Pass the value in **input** to the method **isOdd** and store the result in a variable named **result**. Print to the screen **Is odd:** followed by the value in the variable **result**.

9. Write a method named **numRows** that takes a 2D array as an argument and returns the number of rows in the 2D array.
10. In main, allocate a 2D array named **m1** that holds the following values:
    - row 1: 2, 4, 6, 4
    - row 2: 5, 7, 8, 9
11. In main, pass **m1** to the **numRows** method and store the return value in a variable named **numRows**. Print to the screen **Number of rows:** followed by the value in **numRows**.

12. Write a method named **sameRowLengths** that takes a 2D array of integers as an argument and returns **true** if *all of the rows* in the 2D array have the same number of elements, otherwise the method returns **false.**
13. Call **sameRowLengths**, passing to it **m1** and store the result in a variable named **sameLengths**. Print to the screen **Same row lengths:** followed by the value in **sameLengths**.

14. In main, allocate a 2D array named **m2** that holds the following values:
    - row 1: 2, 4, 6, 4
    - row 3: 5, 7, 3, 1
15. Call **sameRowLengths**, passing to it **m2** and store the result in a variable named **sameLengths**. Print to the screen **Same row lengths:** followed by the value in **sameLengths**.