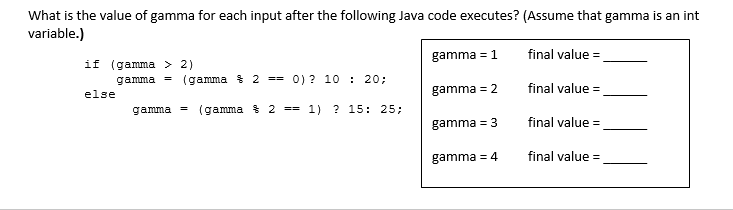
CS 170 ch.3 Lab 1

# Task 1

Question: Solve the 4 test cases in the box. Include your answers in your Word document similar to the format in the box. 

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | Initial Gamma | Step by Step Solution | Final Gamma | | Gamma = 1 | False  1 % 2 = 1  1 == 1 : true  Gamma = 15 | 15 | | Gamma = 2 | False  2 % 2 = 0  0 == 1 : false  Gamma = 25 | 25 | | Gamma = 3 | True  3 % 2 == 1  1 == 0 : false  Gamma = 20 | 20 | | Gamma = 4 | True  4 % 2 == 0  0 == 0 : true  Gamma = 10 | 10 | |

# Task 2

Question: Write a program that reports whether a number is even or odd.

**Test Case 1**

Input a number: 7

The number 7 is odd

**Test Case 2**

Input a number: 1020

The number 1020 is even

|  |
| --- |
|  |

|  |
| --- |
|  |

# Task 3

Question: Write a program that asks the user to enter two words and then reports which of the words comes first in the alphabet.

**Test case 1**

Enter your first word using all lowercase:  
hello  
Enter your second word using all lowercase:  
goodbye  
hello comes after goodbye in the alphabet

**Test case 2**

Enter your first word using all lowercase:  
apple  
Enter your second word using all lowercase:  
zebra  
apple comes before zebra in the alphabet

**Test case 3**

Enter your first word using all lowercase:  
zebra  
Enter your second word using all lowercase:  
apple  
zebra comes after apple in the alphabet

**Test case 4**

Enter your first word using all lowercase:  
java  
Enter your second word using all lowercase:  
java  
java and java are equal

|  |
| --- |
|  |

|  |
| --- |
|  |

# Task 4

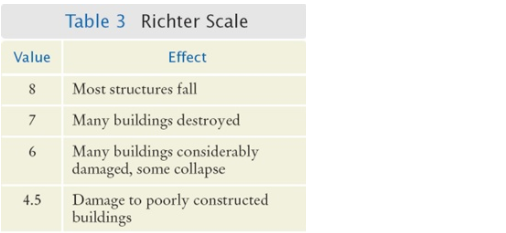
Question: **Part 1:**

Students will write two classes to represent the reading on a Richter Scale.(refer to section 3.3 pg 98 in your book)

RichterReading1.java and RichterReading2.java.

In the first class, you will use Multi-branching methods. (if -  else if -  else if -  else)

In the second class, you will only use if statements.



Include snapshots of your code and executions for both programs.

RichterReading1.java

|  |
| --- |
|  |

|  |
| --- |
| \ |

RichterReading2.java

|  |
| --- |
|  |

|  |
| --- |
|  |

**Part Two:**

Compare the results of both programs and explain the difference.

|  |
| --- |
| The results of the programs are identical, but the first one is much more efficient. In the first one, the program would stop checking the value of the reading variable after it satisfied one of the conditions, because it used if else statements instead of individual if statements. In the second one, only if statements are used, which is much less efficient since it needs to check if the value is within a specific range using operators, regardless of whether it had already satisfied one of the conditions. |