CMSC203 Assignment 2 Implementation (Documentation)

Class: CMSC203 CRN XXXX
Program: Assignment #2
Instructor: Grinberg
Summary of Description: Random number guesses game
Due Date: 09/26/2022
Integrity Pledge: I pledge that I have completed the programming assignment independently. I have not copied the code from a student or any source.

Part1: Pseudo Code: Here is a pseudo code for Assignment 2 program:

```
Import scanner
Public class RandomNumberGuesser
Public main

New scanner sc
String input
Int lowGuess
Int highGuess
Int nextGuess
Int randInt equals RNG rand method
Int count equals RNG get count method
Display “This application generates a random integer between 0 and 100 and asks the user to guess repeatedly until they guess correctly"
Display “Enter your first guess”
NextGuess equals scanner next integer

While nextGuess doesn’t equal randInt

If nextGuess is less than randInt
Display “Your guess is too low”
LowGuess equals nextGuess
Increase count by 1

If nextGuess is more than randInt
Display “Your guess is too high”
HighGuess equals nextGuess
Increase count by 1

If count is greater than 7
Display “Your ran out of guesses”
```

```
End program

If count doesn’t equal 0
Display “Number is Guesses is:”

While input validation is not true
NextGuess equals scanner next integer

If nextGuess equals randInt
Display “Congratulations, you guessed correctly. Try Again? (Yes or no)
Input equals scanner next line

If input equals yes
Reset count
Main args

If input equals no
Display “Thanks for playing”
Display “Programmer: Michael Bushman”
End program
```

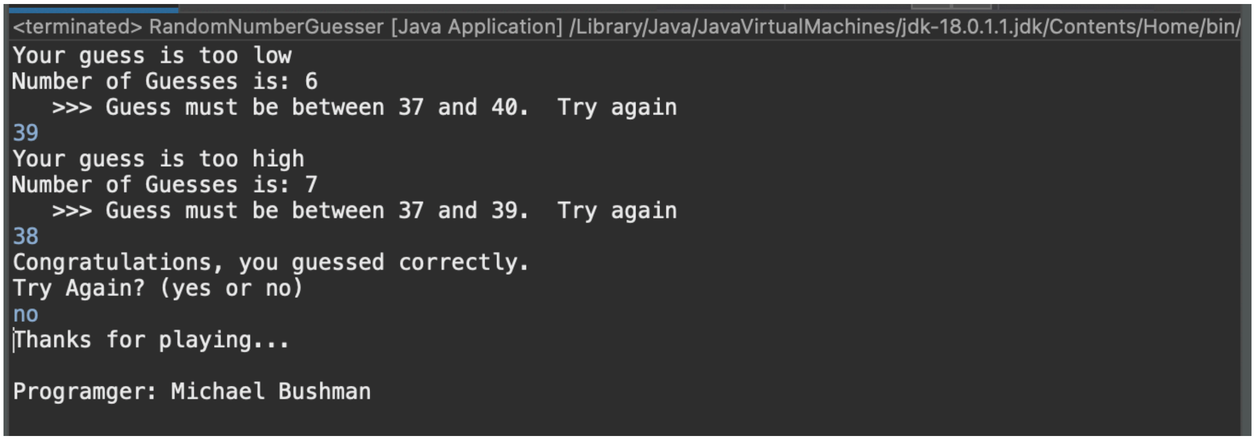
Part2: Comprehensive Test Plan

A good test plan should be comprehensive. This means you should have a few test cases that test when the input is in and out of range, division by 0, incorrect Data type, etc. (Provide valid and invalid input)

Cases	Input	Expected Output	Actual Output	Did Test Pass?
Case 1	User enters in no to end the game	Says programmers name and game ends	Says programmers name and game ends.	Yes
Case 2	User enters in more than 7 guesses	Message comes up, sayings exceeding guesses	Guess counter keeps on counting	No
Case 3	User enters in a number outside given range	Inputvalidation is used correctly and prints out a message	Inputvalidation is used correctly and prints out a message	Yes
Case 4	Number of guesses starts at one	User enters in their first guess and the message says guess #1	User enters in their first guess and the message says guess #1	Yes

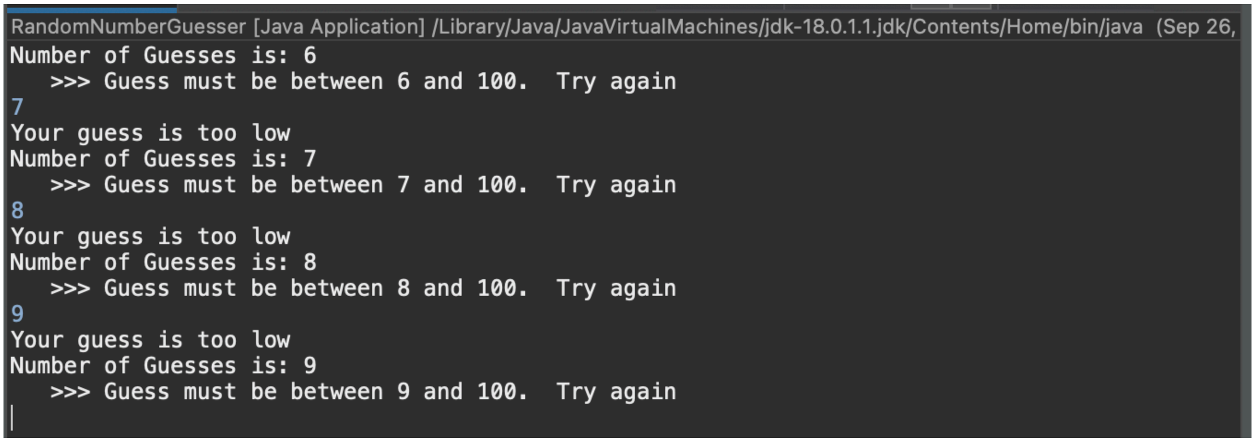
Part3: Screenshots related to the Test Plan:

Case 1



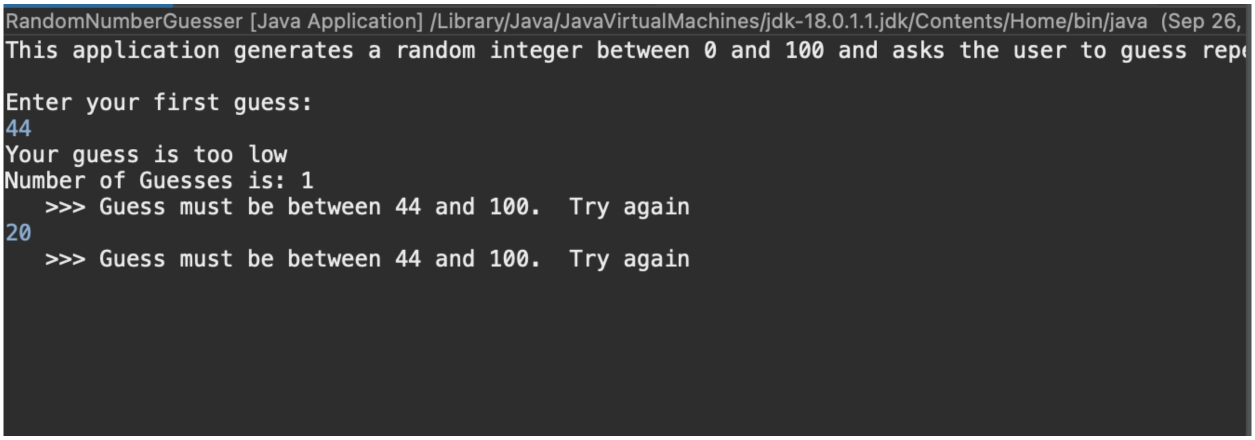
```
<terminated> RandomNumberGuesser [Java Application] /Library/Java/JavaVirtualMachines/jdk-18.0.1.1.jdk/Contents/Home/bin/
Your guess is too low
Number of Guesses is: 6
   >>> Guess must be between 37 and 40.  Try again
39
Your guess is too high
Number of Guesses is: 7
   >>> Guess must be between 37 and 39.  Try again
38
Congratulations, you guessed correctly.
Try Again? (yes or no)
no
Thanks for playing...

Programger: Michael Bushman
```

Case 2



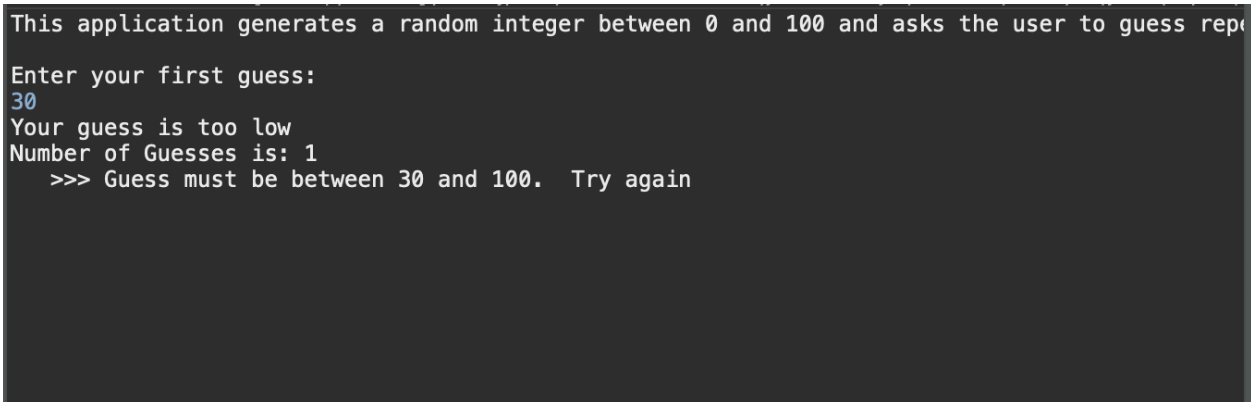
```
RandomNumberGuesser [Java Application] /Library/Java/JavaVirtualMachines/jdk-18.0.1.1.jdk/Contents/Home/bin/java  (Sep 26,
Number of Guesses is: 6
   >>> Guess must be between 6 and 100.  Try again
7
Your guess is too low
Number of Guesses is: 7
   >>> Guess must be between 7 and 100.  Try again
8
Your guess is too low
Number of Guesses is: 8
   >>> Guess must be between 8 and 100.  Try again
9
Your guess is too low
Number of Guesses is: 9
   >>> Guess must be between 9 and 100.  Try again
```

Case 3

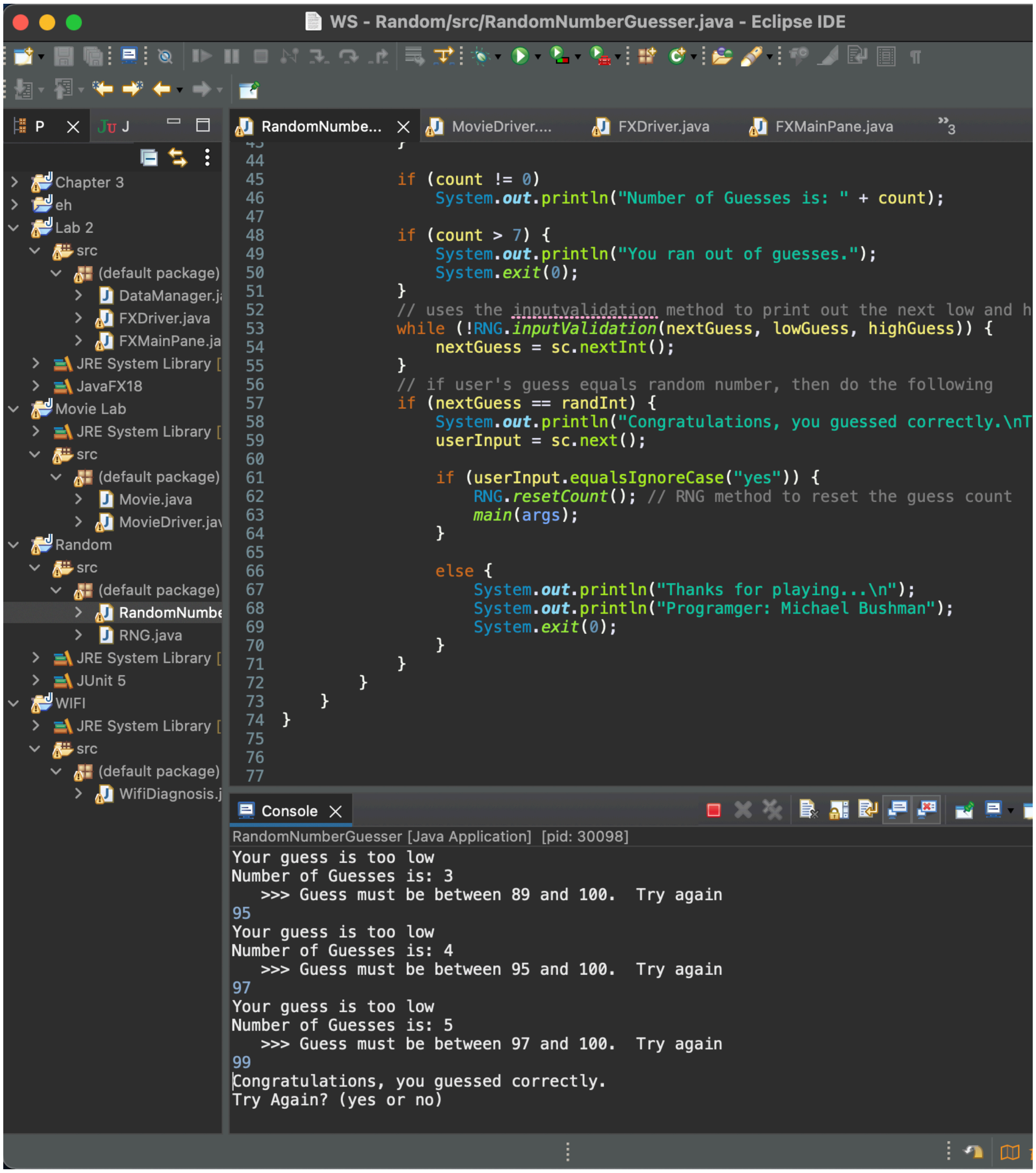


```
RandomNumberGuesser [Java Application] /Library/Java/JavaVirtualMachines/jdk-18.0.1.1.jdk/Contents/Home/bin/java  (Sep 26,
This application generates a random integer between 0 and 100 and asks the user to guess rep

Enter your first guess:
44
Your guess is too low
Number of Guesses is: 1
   >>> Guess must be between 44 and 100.  Try again
20
   >>> Guess must be between 44 and 100.  Try again
```

Case 4

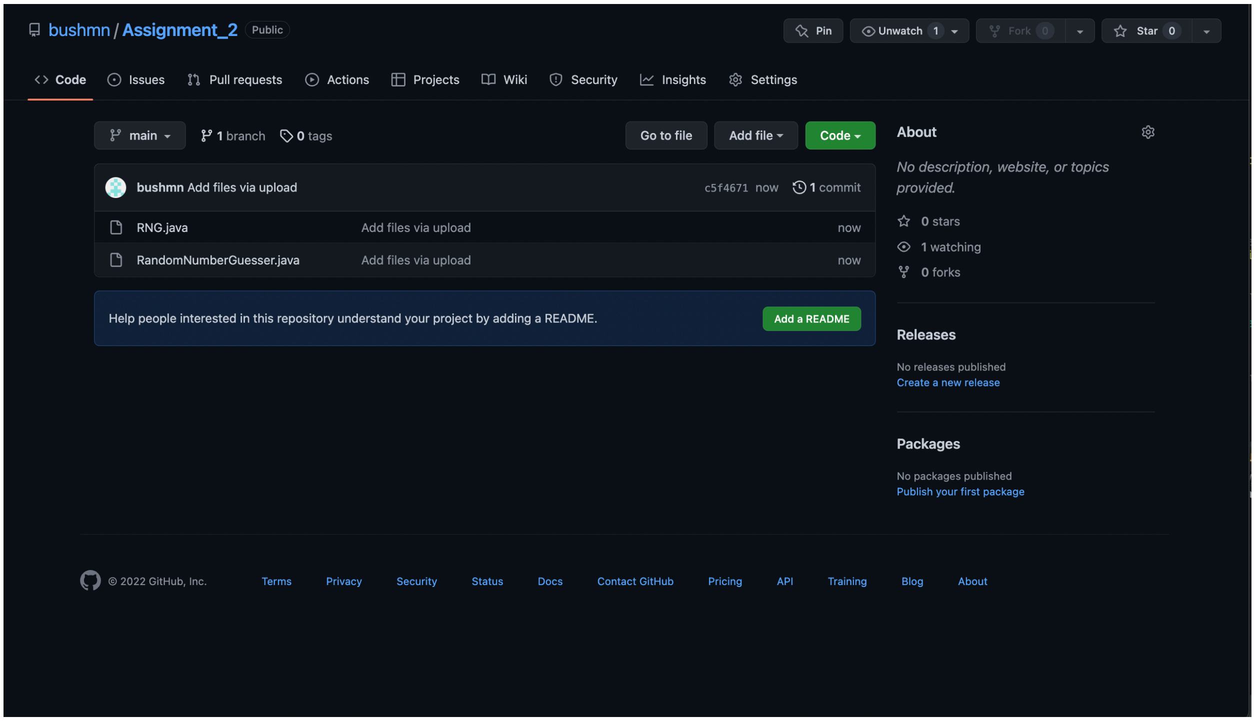


```
This application generates a random integer between 0 and 100 and asks the user to guess repe

Enter your first guess:
30
Your guess is too low
Number of Guesses is: 1
   >>> Guess must be between 30 and 100.  Try again
```



```java
            if (count != 0)
                System.out.println("Number of Guesses is: " + count);

            if (count > 7) {
                System.out.println("You ran out of guesses.");
                System.exit(0);
            }
            // uses the inputvalidation method to print out the next low and h
            while (!RNG.inputValidation(nextGuess, lowGuess, highGuess)) {
                nextGuess = sc.nextInt();
            }
            // if user's guess equals random number, then do the following
            if (nextGuess == randInt) {
                System.out.println("Congratulations, you guessed correctly.\nT
                userInput = sc.next();

                if (userInput.equalsIgnoreCase("yes")) {
                    RNG.resetCount(); // RNG method to reset the guess count
                    main(args);
                }

                else {
                    System.out.println("Thanks for playing...\n");
                    System.out.println("Programger: Michael Bushman");
                    System.exit(0);
                }
            }
        }
    }
}
```

```
RandomNumberGuesser [Java Application]  [pid: 30098]
Your guess is too low
Number of Guesses is: 3
   >>> Guess must be between 89 and 100.  Try again
95
Your guess is too low
Number of Guesses is: 4
   >>> Guess must be between 95 and 100.  Try again
97
Your guess is too low
Number of Guesses is: 5
   >>> Guess must be between 97 and 100.  Try again
99
Congratulations, you guessed correctly.
Try Again? (yes or no)
```



bushmn / Assignment_2 Public

1 branch 0 tags

bushmn Add files via upload c5f4671 now 1 commit

RNG.java	Add files via upload	now
RandomNumberGuesser.java	Add files via upload	now

Help people interested in this repository understand your project by adding a README.

About

No description, website, or topics provided.

0 stars

1 watching

0 forks

Releases

No releases published

Create a new release

Packages

No packages published

Publish your first package

© 2022 GitHub, Inc.

Lessons Learned <Provide answers to the questions listed above>**:**

Write about your Learning Experience, highlighting your lessons learned and learning experience from working on this project.

What have you learned? I have learned to get comfortable with the different types of loops and when to and when to not use them. I have gotten comfortable with nested if and while loops inside other while loops. I have also learned to write code that is easier to follow and read including the indentations and layout of the code.

What did you struggle with? I didn’t struggle too much. It took some time to make sure I was using the correct loops for the overall program. But once that was figured out, the rest was pretty straight forward.

What would you do differently on your next project? The one thing I would have done differently is making sure to read all user inputs for whether or not they would like to continue playing. With the current program, if the user enters anything other than upper or lower case yes, it will continue. If the user enter in something that isn't NO or no, it should display a message saying invalid response, try again.

What parts of this assignment were you successful with, and what parts (if any) were you not successful with? Successful with all parts of the assignment.

Provide any additional resources/links/videos you used to while working on this assignment/ project.

Check List: <Provide answers to the column Y/N or N/A >**:**

#		Y/N	Comments
	Assignment files:		
	• FirstInitialLastName_ Assignment#_Moss.zip	**Yes**	
	• FirstInitialLastName_Assignment#.docx/.pdf	**Yes**	
	• Source java files	**Yes**	
	Program compiles	**Yes**	
	Program runs with desired outputs related to a Test Plan	**Yes**	
	Documentation file:	**Yes**	
	• Comprehensive Test Plan	**Yes**	
	• Screenshots related to the Test Plan	**Yes**	
	• Screenshots of your GitHub account with submitted Assignment# (if required)	**Yes**	
	• UML Diagram (if required)	**No**	
	• Algorithms/Pseudocode (if required)	**Yes**	
	• Flowchart (if required)	**No**	
	• Lessons Learned	**Yes**	
	• Checklist is completed and included in the Documentation	**Yes**	