Reviewer: Nayana Raj Cheluvaraju
Course: SOEN 6011
Team: Team B
Professor: P.Kamthan

Problem 7- Review of function tan(x)

Testing of the above assigned function is done manually in comparison with requirements mentioned in problem 2 and also reviewed the test cases written using Junit. All the observations and variation are noted and recorded.

The following are the Manual Testing result:

Requirement Number	Input given	Expected Output	Actual Output	Test Result (Pass/Fail)	Comments
2.1 Hardware	Calculator.java (including jar)	Must run on both Mac OS and Windows	As Expected	Pass	NA
2.2. Software	Calculator.java	The application shall use only standard Java functions.	As Expected.	Pass	NA
2.3. Operating System	Calculator.java	Must run on both Mac OS and Windows	-	-	Repeated requirement(same as requirement 2.1)
2.4. Human Interfaces	Calculator.java	The application shall function in a similar way as a regular calculator	As Expected.	Pass	NA
2.4.1. Input	Calculator.java	The application shall allow the user to enter the numbers manually through his/her keyboard.	As Expected.	Pass	Application is not using whole text box while inputting data. Input is going out of text box. (Attached screenshot, Figure1)
2.4.2. Output	Calculator.java	The application shall display the results on the appropriate output area.	As Expected.	Pass	NA

3.1 Input(major function requirements)	Calculator.java (GUI)	The input function shall accept the input from the user via the terminal, validate it and send it to the appropriate function to calculate the output.	As Expected.	Pass	NA
3.2 Calculate	Calculator.java	This function will accept the correct input from the input function(through arguments) and will perform the operations requested by the user.	As Expected.	Pass	NA
3.3 Output	Calculator.java (GUI)	Function will display the result, when the equals(=) sign button is pressed(GUI).	As Expected.	Pass	NA
4.1. Illegal Input Sequence	Calculator.java	Appropriate error message shall be printed	As Expected	pass	It is displayed in console, user has to clear message manually. If user forgets to clear input is take along with message (Figure 2)
4.2. Division By zero	Calculator.java	shall be detected by the program and an appropriate error message is printed.	Not as Expected.	Fail	Message not clear enough. (Figure 3)
4.3 Overflow/Under flow	Calculator.java	Overflow and underflow do not need to be detected.	As Expected	pass	NA

Screenshots for above Commented requirements :

Figure 1 : Screenshot for Requirement 2.4.1(Human Interface)

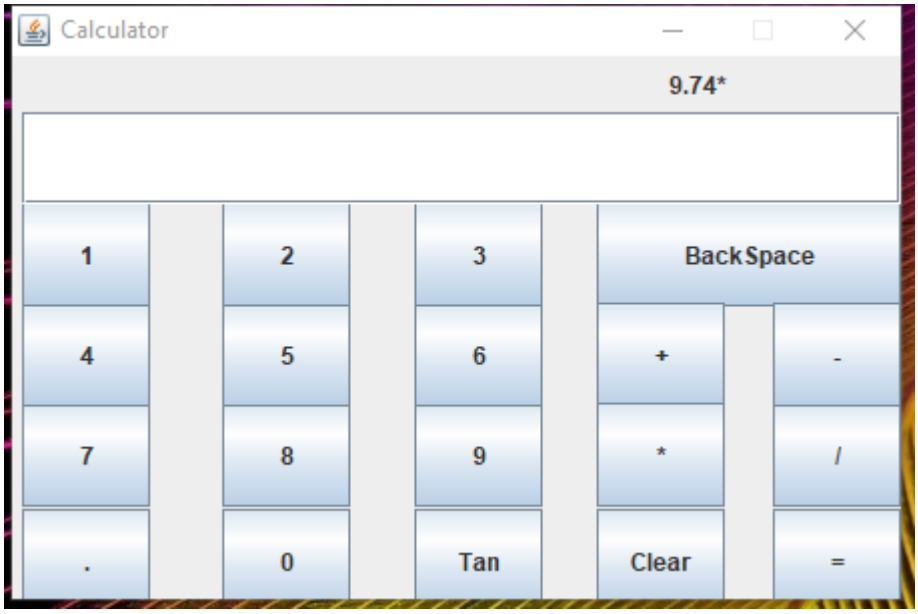


Figure 2: Screenshot for 4.1. Illegal Input Sequence

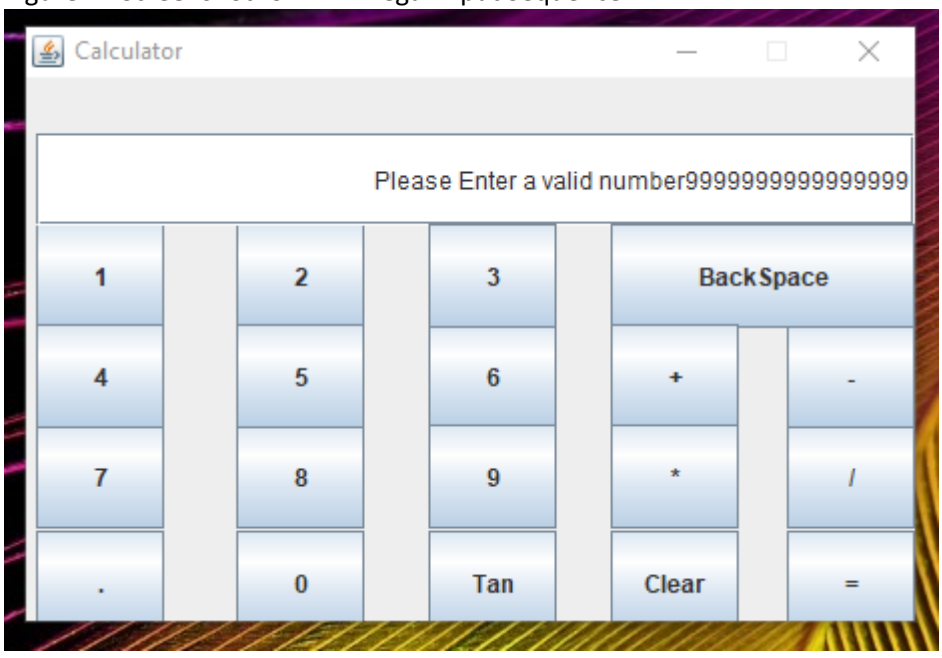
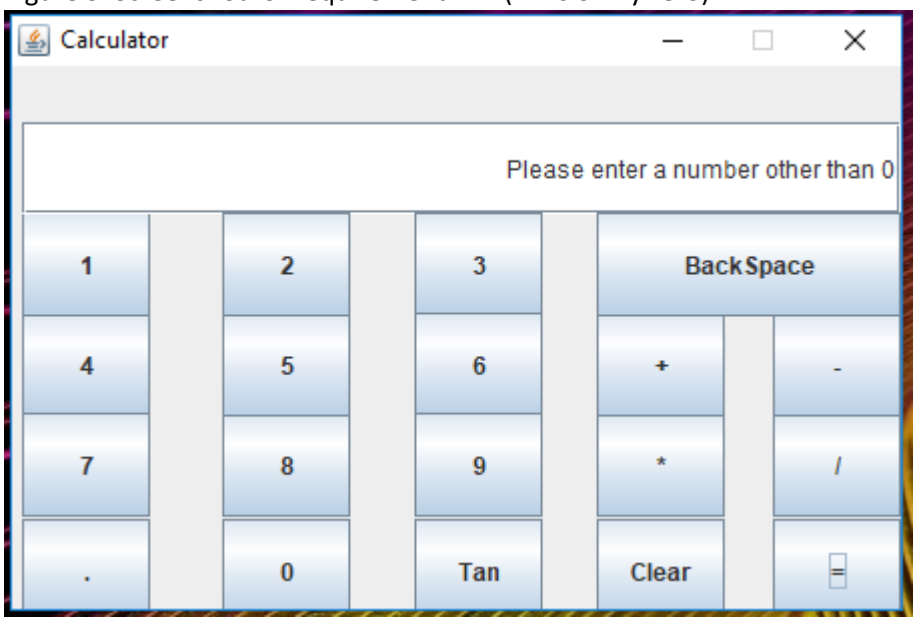


Figure 3: Screenshot for requirement 4.2. (Division By zero)



Junit Review Results:

Summary of work exist:

1. Test cases are noted for all functions and are running successfully.
2. Extra functions such as add, multiply, subtract and divide are implemented along with assigned tan() function. Same are tested.
3. There are ten testcase written for tan(), one each for add, subtract, multiply and two for division.

My observations:

1. For Calculating tan() function test cases are not written Negative values.
   But verified for negative values and application is working as expected.
2. Not tested for null value and zero .
   Verified testing and is working as expected.
3. Not tested for rational and irrationals numbers.
   Verified testing and is working as expected.