SOEN 6011- SOFTWARE ENGINEERING PROCESSES
PROJECT DELIVERABLE 3

Jemish Kishor Paghadar (40080723)
https://github.com/Jemish27121997/SOEN6011_Function_Implementation
Concordia University, Montreal

August 2, 2019

Contents

0.1 Problem 7

0.1.1 Test Cases Analysis for Function $f(x) = ab^x$

The main focus of this section is to provide a brief summary of the test case analysis carried out on the function $f(x) = ab^x$.

Introduction : This function involves an exponent. This exponent is represented with a variable, and its base is represented with constant value. Let $f(x) = ab^x$ be an exponential function where "b" is a constant, the exponent is the independent variable, the coefficient "a" is called the initial value of the function , and "f(x)" represent the dependent variable.

Test Case Analysis : A test case is a set of instructions on "HOW" to validate a particular test objective/target, which when followed will tell us if the expected behavior of the system is satisfied or not. Functional testing is a type of software testing which is used to verify the functionality of the software application, whether the function is working according to the requirement specification. In functional testing, each function tested by giving the value, determining the output, and verifying the actual output with the expected value. Functional testing performed as black-box testing which is presented to confirm that the functionality of an application or system behaves as we are expecting. It is done to verify the functionality of the application. It concentrates on:

Basic Usability : Functional Testing involves the usability testing of the system. It checks whether a user can navigate freely without any difficulty through screens.

Accessibility : Functional testing test the accessibility of the function.

Mainline function : It focuses on testing the main feature.

Error Condition : Functional testing is used to check the error condition. It checks whether the error message displayed.

These are the following steps that I have followed to perform for test case analysis:

- There is a need to understand the software requirement.
- Identify test input data.
- Compute the expected outcome with the selected input values.
- Execute test cases.
- Comparison between the actual and the computed result.

For this function $f(x) = ab^x$ implementation, three test cases were created that cover general as well as common input validations. These tests cases are traceable to initial requirements. From the figure, it can be seen that there are no any test cases that directly test the function itself but all test cases are running properly without any failure.

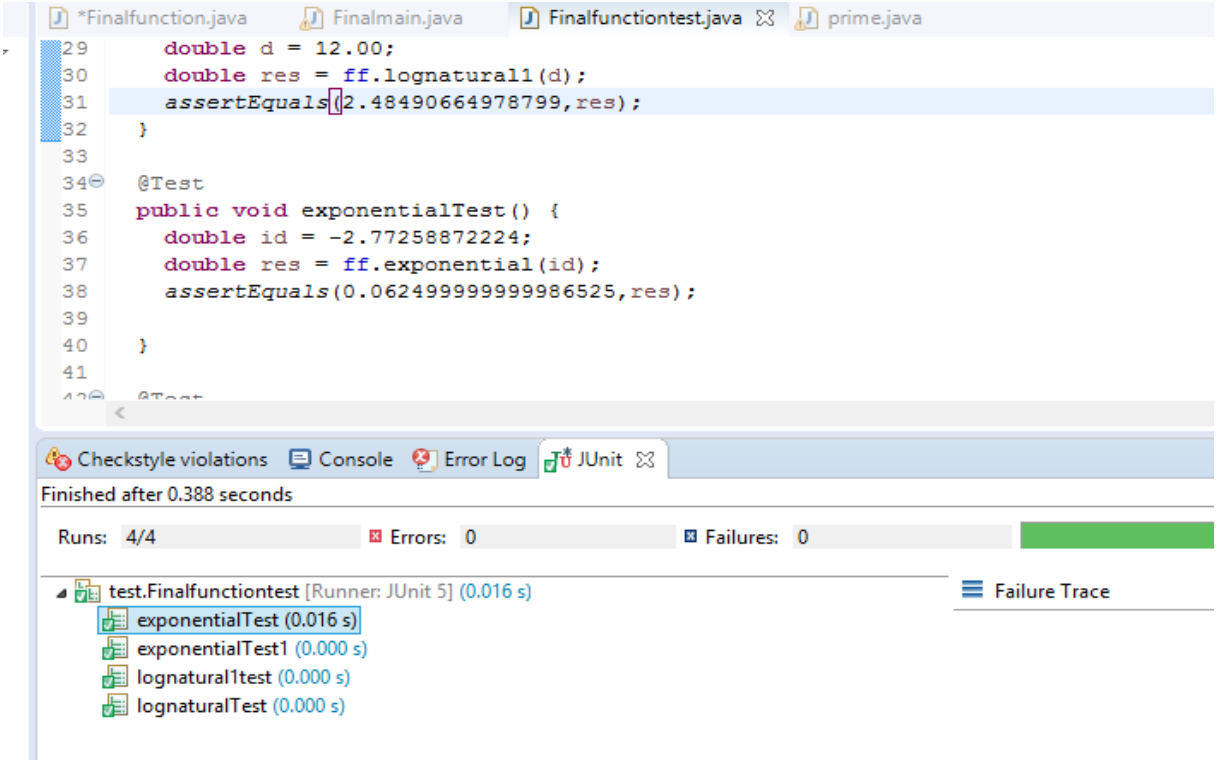

Figure 1: Successful Execution of each test cases

For validating this function a set of different inputs will be passed to each unit test, the results will be posted on a summary table.
The test cases to be analyzed are :
lognaturalTest() which tests the natural logarithm value for an input given within the range of 0 to 1.
lognatural1Test() which tests the natural logarithm value for an input given above 1.
exponentialTest() which tests the exponential value for any input.

The summary tables for the test cases are described below which compares the actual output with the expected output.

Test Case method : lognaturalTest() which validates the natural logarithm value for an input given within the range of 0 to 1. From the table, it can be seen that all the inputs are validated properly and giving the accurate results of natural logarithm.

Input	Expected Result	Actual Result	Status
0.5	-0.6931471805599445	-0.6931471805599445	Passed
0.65	-0.4307829160924539	-0.4307829160924539	Passed

Test Case method : lognatural1Test() which validates the natural logarithm value for an input above 1. From the below table, it can be seen that all the inputs are validated successfully and this method is also giving the accurate results of natural logarithm.

Input	Expected Result	Actual Result	Status
12.00	2.48490664978799	2.48490664978799	Passed
50.5	3.9219733362812597	3.9219733362812597	Passed

Test Case method : exponentialTest() which validates the exponential value for any input. All the inputs are validated successfully without any failure.

Input	Expected Result	Actual Result	Status
-2.77258872224	0.062499999999986525	0.062499999999986525	Passed
100	2.6881171418161336E43	2.6881171418161336E43	Passed
2.718	15.149991940878161	15.149991940878161	Passed

Therefore, from the test cases analysis, it is clear that all the test cases are successfully executing without any failure. However, there should be one test case to test the function itself to check final result.

0.1.2 Acknowledgments

I would like to thank Professor Pankaj Kamthan and his great group of teaching assistants for the material and guidance. I would also like to thank my teammate Koteswara Rao Panchumarthy for giving access to his code and test cases for review.

0.1.3 Version Control System

To have a common repository for all project files available and updated remotely, distributed version control system(Git) is maintained.
Clone the repository from below link
https://github.com/Jemish27121997/SOEN6011_Function_Implementation.git

Bibliography

[1] https://www.javatpoint.com/functional-testing

[2] https://github.com/Jemish27121997/SOEN6011_Function_Implementation