MBARARA UNIVERSITY OF SCIENCE AND TECHNOLOGY
FACULTY OF COMPUTING AND INFORMATICS

COURSE UNIT: SOFTWARE ENGINEERING INDUSTRIAL MINI PROJECT II

COURSE CODE: SWE4106

Academic Year: 2024/2025

Semester: One

Student name:	**BWESIGYE TREASURE**
Regno:	**2021/BSE/145/PS**
Student number:	**2100603048**

TEST CASES

I tested the portable matrix library across different programming languages. First, I focused on unit testing.

Unit Testing

Defined Unit Tests : I started by defining tests for the main features of the library, such as creating matrices, adding them, and multiplying them.

Used Testing Frameworks : For each language, I picked the right testing framework, like JUnit for Java and pytest for Python, to run my tests.

Ran Unit Tests : I ran the unit tests in each language. I noted any failures and fixed issues as I went along.

Refined Tests : After looking at the results, I updated the tests to cover any missed scenarios and corrected errors in the library.

System Integration Testing

Set Up Environment : I made sure my test environment was ready for each language with all the right tools and libraries.

Defined Testing Scenarios : I thought of real-world scenarios where I might use the library and created tests for those situations.

Recorded Results : After running the tests, I recorded and reviewed the results to find any problems or unexpected behaviour.

Debugged and Refactored : I fixed any issues that came up during testing and refined the code to ensure everything worked smoothly.

Performance Testing : I added some basic performance tests to see how the library handled different data sets.

Documented Findings : Finally, I documented all the test cases, results, and any important lessons I learned throughout the process.

By completing these tests, I ensured that the portable matrix library functioned well across different programming languages, preparing it for the next steps in development.

The major form of testing was trying out different test cases to see whether the functions within my library can be called in these test cases.

References.

J. Pan, "Software Testing," 18-849b Dependable Embedded Systems, Carnegie Mellon University, Spring 1999. [Online]. Available: jpan@cmu.edu.