# Java Program and JUnit Testing Report

Software Quality Assurance: CS469

Antoine Gaton

November 24, 2024

## 1. Sample Java Program

Below is the code for the sample Java program that calculates the grade of a student based on marks entered:

A screenshot of a computer code

Description automatically generated

## 2. JUnit Test Program

Below is the code for the JUnit test program, including 9 test cases to validate the functionality of `calculateGrade` and ensure robust exception handling:

A screen shot of text

Description automatically generated

## 3. JUnit Test Results

The following screenshot shows the results of the JUnit tests, with all 9 test cases passing successfully:

## A screenshot of a computer Description automatically generated

## 4. Explanation of JUnit Tests

1. `testCalculateGrade\_A`: Tests grade calculation for an average above 80 (Grade A).  
2. `testCalculateGrade\_B`: Tests grade calculation for an average between 60 and 80 (Grade B).  
3. `testCalculateGrade\_C`: Tests grade calculation for an average between 40 and 60 (Grade C).  
4. `testCalculateGrade\_D`: Tests grade calculation for an average below 40 (Grade D).  
5. `testCalculateGrade\_Exact80`: Edge case for an average exactly 80 (Grade A).  
6. `testCalculateGrade\_Exact60`: Edge case for an average exactly 60 (Grade B).  
7. `testCalculateGrade\_Exact40`: Edge case for an average exactly 40 (Grade C).  
8. `testCalculateGrade\_InvalidInputNegative`: Ensures an exception is thrown for negative marks.  
9. `testCalculateGrade\_InvalidInputAbove100`: Ensures an exception is thrown for marks above 100.

## 5. Reflection

This assignment helped me understand the importance of unit testing using JUnit. Writing test cases for various scenarios, including edge cases and exceptions, ensures the robustness and reliability of the code.