CSc 305: Assignment 1

1 Objectives

- acquire experience with object oriented programming design
- acquire experience with designing test cases
- acquire experience with C++
- practice with matrices, the "work horse" of computer graphics

2 Design

Using a language-independent object oriented approach, design objects and test cases for a Matrix and a 3D point. These objects are being designed to accomplish graphics transformations in 3D.

2.1 A Matrix object

- must have at least 2 constructors
- assume 3D transformations
- assume resultant matrices have integer components
- matrix object functionality must include:
    - transpose of a matrix
    - equality test (for 2 matrices)
    - matrix multiplication
    - matrix inverse
    - dot product
    - cross product

2.2 A 3D point object

- attributes: integer coordinates (x, y, z)
- 1 constructor
- point object functionality must include:
    - rotation
    - translation
    - scaling

2.3 Test cases for both objects

Design a set of test cases that will test both proper/expected input AND illegal/unexpected input. Your programs must gracefully handle a minimum of 5 "errors". Gracefully means that a runtime error will not be generated if the rule is violated. ie. illegal matrix multiplication

Your test cases should be in the following format

1. Test Case Name
    a. A short description of the test case
    b. Input: a description
    c. Expected Output: a description

3 Implementation

Implement your design first in Java then in C++. Proper coding conventions, commenting are expected.

3.1 Java

Implement your design in Java. You will implement a Matrix class, a 3DPoint class and a Java application(s) that supports the testing of your Matrix and 3DPoint classes. The Java test class(es) do not need to have a fancy graphical user interface, but the should allow a user to interactively run tests.

3.2 C++

"Translate" your Java classes and test application into C++.

4 Testing

Provide a report of the testing and debugging of your program using the test cases developed in the design portion of the assignment. The report should be in the following format:

1. Test Case Name:
    a. Input
    b. Expected Output
    c. Actual Output
    d. If the actual output does not match expected output, provide:
        - Fix: describe the fix that was performed
        - Retest Actual Output

5 Submission details

Submit the following files via Assignment 1 link on conneX:

- Document including
    - Your design (objects and test cases)
    - Your testing reports (one for each language implementation)
- Source files
    - All files need to compile and run your test applications (Java & C++)

6 Marking Rubric

Part	Item	Weight
Design	Design document	2
	Test cases	18
Implementation	Matrix Class – Java	10
	3D Point Class – Java	8
	Test Application – Java	5
	Handles exceptions – Java	5
	Matrix Class – C++	10
	3D Point Class – C++	8
	Test Application – C++	5
	Handle exceptions – C++	5
	Proper coding conventions and commenting	4
Testing	Report of test results – Java	5
	Report of test results – C++	5
Subjective		10
TOTAL MARKS		100