1. **Assignment Description:**

On sometimes, you may be required to improve, update, or fix a program that has been written by someone else. You will begin this project with a pre-existing version of the classify triangle software that you will be given. Additionally, you will receive a beginning test program that partially tests the classify triangle program.

You must alter the list of test cases in the test program in order to assess whether the software is correctly implemented. Until you believe that your tests appropriately test all of the conditions, you will need to update the test program. Then, to determine how accurate the original triangle program is, you should run every test against it. Observe the results, and then document them in the official test report that is described below. You shouldn't alter the categories triangle program at all for this initial stage. Simply changing the test software is sufficient.

The classify triangle program will then be updated to address all issues based on the outcomes of your initial tests. Run the test cases again and again until all of the errors have been corrected. Run the test program one last time, record the results, and then report on them in the official test report outlined below.

2. **Author:** Amrutha Malavalli Srinivas

3. **Summary:**

GitHub Repo: https://github.com/malavalliamrutha/Triangle567

Erroneous code can be fixed quite effectively with test-driven debugging. As I performed the tests and addressed bugs in the code, more flaws were discovered. Nevertheless, I believe that building tests first and then developing all the code is a more efficient approach to error-check than the other way around.

4. **Honor pledge:**

I pledge my honor that I have abided by the Stevens Honor System.

5. **Detailed results:**

**Initial test results:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test ID | Input | Expected Results | Actual Results | Pass or Fail |
| testRightTriangleA | (3,4,5) | Right | InvalidInput | Fail |
| testRightTriangleB | (5,3,4) | Right | InvalidInput | Fail |
| testRightTriangleC | (13, 12, 5) | Right | InvalidInput | Fail |
| testRightTriangleD | (8, 6, 10) | Right | InvalidInput | Fail |
| testEquilateralTriangleA | (1,1,1) | Equilateral | InvalidInput | Fail |
| testEquilateralTriangleB | (10,10,10) | Equilateral | InvalidInput | Fail |
| testIsocelesTriangleA | (3, 3, 2) | Isoceles | InvalidInput | Fail |
| testIsocelesTriangleB | (4, 6, 6) | Isoceles | InvalidInput | Fail |
| testIsocelesTriangleC | (6, 4, 6) | Isoceles | InvalidInput | Fail |
| testIsocelesTriangleD | (6, 6, 4) | Isoceles | InvalidInput | Fail |
| testScaleneTriangleA | (10, 15, 12) | Scalene | InvalidInput | Fail |
| testInvalidInputA | (-1, -1, -1) | InvalidInput | InvalidInput | Pass |
| testInvalidInputB | (201, 201, 201) | InvalidInput | InvalidInput | Pass |
| testInvalidInputC | ("200", "0", "0") | InvalidInput | TypeError | Fail |
| testNotATriangleA | (5, 1, 1) | NotATriangle | InvalidInput | Fail |
| testNotATriangleB | (1, 5, 1) | NotATriangle | InvalidInput | Fail |
| testNotATriangleC | (1, 1, 5) | NotATriangle | InvalidInput | Fail |

**Test Run Matrix:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Test Run 1 | Test Run 2 | Test Run 3 | Test Run 4 |
| Tests Planned | 17 | 17 | 17 | 17 |
| Tests Executed | 17 | 17 | 17 | 17 |
| Tests Passed | 2 | 6 | 10 | 17 |
| Defects Found | 2 | 1 | 3 | 0 |
| Defects Fixed | 0 | 2 | 1 | 3 |

**Final test results:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test ID | Input | Expected Results | Actual Results | Pass or Fail |
| testRightTriangleA | (3,4,5) | Right | Right | Pass |
| testRightTriangleB | (5,3,4) | Right | Right | Pass |
| testRightTriangleC | (13, 12, 5) | Right | Right | Pass |
| testRightTriangleD | (8, 6, 10) | Right | Right | Pass |
| testEquilateralTriangleA | (1,1,1) | Equilateral | Equilateral | Pass |
| testEquilateralTriangleB | (10,10,10) | Equilateral | Equilateral | Pass |
| testIsocelesTriangleA | (3, 3, 2) | Isoceles | Isoceles | Pass |
| testIsocelesTriangleB | (4, 6, 6) | Isoceles | Isoceles | Pass |
| testIsocelesTriangleC | (6, 4, 6) | Isoceles | Isoceles | Pass |
| testIsocelesTriangleD | (6, 6, 4) | Isoceles | Isoceles | Pass |
| testScaleneTriangleA | (10, 15, 12) | Scalene | Scalene | Pass |
| testInvalidInputA | (-1, -1, -1) | InvalidInput | InvalidInput | Pass |
| testInvalidInputB | (201, 201, 201) | InvalidInput | InvalidInput | Pass |
| testInvalidInputC | ("200", "0", "0") | InvalidInput | InvalidInput | Pass |
| testNotATriangleA | (5, 1, 1) | NotATriangle | NotATriangle | Pass |
| testNotATriangleB | (1, 5, 1) | NotATriangle | NotATriangle | Pass |
| testNotATriangleC | (1, 1, 5) | NotATriangle | NotATriangle | Pass |