**Name: Huỳnh Hữu Tứ**

**SID: 3122411240**

**I – Verification and validation**

**Q: What are the problems of those two systems?**

A:

* SYSTEM 1:
  + The formula for calculating x2 is incorrect.
  + The variable DELTA used in the formula does not exist (undefined).
  + Have interface for input a,b,c and output x1,x2
  + => Verified & Invalidated
* SYSTEM 2:
  + System requires input for DELTA and has unnecessary interface.
  + => Unverified & Validated

**II – Test-cases**

**a) Q: How many test-cases we need for the following function f1. What are they?**

A: 2 test cases:

1. x>10: example: x=11, expected: 22

2. x<=10: example: x=9, expected: -9

Attached files:

a.cpp

test.ipynb

**b) Q: Check if your test-cases can detect error if f1 is implemented as follows. In this case, how many test-cases we need to test this function? What are they?**

A: 3 test cases:

1. x>10: example: x=11, expected: 22

2. 0<x<=10: example: x=5, expected: -5

3. x<=0: example: x=-1, expected: -2

Attached files:

b.cpp

test.ipynb

**c) Q: How many test-cases we need to test this function? What are they? In this case, how many test-cases we need to test this function? What are they?**

A: 2 test cases

1. x<10: example: x=9, expected: 18.

2. x>=10: example: x=11, expected: 22.

x<2 is unreachable code.

Attached files:

c.cpp

test.ipynb

**d) Q: How many test-cases we need to test this function? What are they?**

A: 2 test cases for branch coverage

1. log(x \* x \* cos(x)) < 3 \* x: example: x=1, expected: 2

2. log(x \* x \* cos(x)) >= 3 \* x: example: x=-1, expected: -2

Attached files:

d.cpp

test.ipynb

**e) Q: Check if your test-cases can detect error if findMax is implemented as follows.**

A:

Branch coverage: Every if conditional must be covered true branch and false branch.

=> 4 test cases:

1. num1=5, num2=3, num3=1 (5,3,1), expected: 5, (true,false,false)

2. num1=1, num2=5, num3=3 (1,5,3), expected: 5, (false,true,false)

3. num1=1, num2=3, num3=5 (1,3,5), expected: 5, (false,false,true)

4. num1=5, num2=5, num3=5 (5,5,5), expected: 0, (false,false,false)

(true,true,true): can't be happened

Attached files:

e.cpp

test.ipynb

**III – Practice 1**

**Q1: Describe the problem, including all possible inputs/outputs.**

**Q2: Develop a comprehensive set of test cases to verify the correctness of the program.**

**Q3: Implement an automated script to evaluate whether the given program produces correct results**.

A1: This is a problem about finding the value of x in a biquadratic equation: ax4+bx2+c=0. Introduce a substitution y = x2, then solve the quadratic equation in y, and finally take the square root to obtain the solutions for x.

All possibile inputs/outputs:

1. infinite solutions:

Input: a=0, b=0, c=0

Output: infinite solutions

2. no real solutions:

Input: a=0, b=0, c!=0

Input: a=0, b!=0, y = -c/b <0

Input: a!=0, delta<0

Input: y1,y2<0

Output: no solution

3. has real solutions:

Output: x1, x2 x … xn

* 2 solution: when has only y>=0 or y=0. Example: x=+-1
* 4 solution: when has y1 and y2 different. Example: x=+-1, +-2

A2: 7 test cases:

1. Infinite solutions:

a=0, b=0, c=0.

Example: a=0, b=0, c=0

Expected: -1 => infinite solutions.

2. No solution:

a=0, b=0, c!=0.

Example: a=0, b=0, c=1

Expected: 0 => no solution.

3. A quadratic equation in y with negative solutions for y

a=0, b!=0, y<0.

Example: a=0, b=1, c=1. => x2 + 1 = 0

Expected: 0 => no solution

4. A quadratic equation in y with solutions where y ≥ 0

a=0, b!=0, y>=0.

Example: a=0, b=1, c=-1 => x2 – 1 = 0

Expected: 2 => x1 = 1, x2 = -1

5. Delta <0

Delta = b2 – 4ac < 0

Example: a=1, b=0, c=1 => x4 + 1 = 0

Expected: 0

6. Delta >= 0 and both solutions for y are distinct and positive.

Delta = b2 – 4ac 0

Example: a=1, b=-5, c=4 = > x4 – 5x2 + 4 = 0

=> y2 – 5y + 4 = 0 => y=1, y=4

Expected: 4 => x= 1 2

7. Delta >= 0 and solution for y1 positive, y2 negative

Example: a=1, b=2, c=-2 => x4 + x2 – 2 = 0

=> y2 + y – 2 = 0 => y=1, y=-2 => y=-2 is invalid

Expected: 2 => x = 1

A3: Attached files:

solveQuartic.cpp

test.ipynb