**EXERCISE 1**

**Name:** Phạm Minh Toàn **STUDENT CODE:** 3122411216

**I. VERIFICATION AND VALIDATION**

**Description**: The purpose is to help users to solve a 2-degree equation (ax2+bx+c).

**Spec**: Given input of a, b, and c; the system returns the outputs of x1 and x2 (extreme cases are temporarily not considered)

Two systems are developed as follows.

A white sheet of paper with text

AI-generated content may be incorrect.

What are the problems of those two systems? Write down your answer here.

System 1: Có 3 vấn đề, vấn đề đầu tiên là sai công thứ ở x2 do đặt sai (), vấn đề thứ 2 là chưa khai báo delta mà vẫn gọi delta, vấn đề 3 code vẫn chưa bao quát hết tất cả trường hợp deltal < 0 và a = 0

System 2: có 1 vấn đề code vẫn chưa bao quát hết tất cả trường hợp deltal < 0 và a = 0

**II. TEST-CASES**

Description: Some input values

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

A white rectangular object with a black border

AI-generated content may be incorrect.

Input đầu vào của hàm là: x kiểu int. Điều kiện rẻ nhánh x > 10

Trong trường hợp này quá rõ ràng ta chỉ cần 4 trường hợp để test Test case biên trên, Test case tại biên , Test case dưới biên, Test case giá trị âm.

Chọn test-case

|  |  |  |  |
| --- | --- | --- | --- |
| Class | Mô tả | Nhập | Kết quả mong dợi |
| Class 1 | x > 10 (trên biên) | 11 | 22 |
| Class 2 | x <= 10 (tại biên) | 10 | -10 |
| Class 3 | x < 10 (dưới biên) | 5 | -5 |
| Class 4 | x giá trị âm | -2 | -2 |

b) Check if your test-cases can detect error if f1 is implemented as follows

A white rectangular object with black text

AI-generated content may be incorrect.

In this case, how many test-cases we need to test this function? What are they?

Hàm có 3 nhánh

If (x > 10) return 2 \* x

else if (x > 2) return -x

else return 2 \* x

Dòng else if rất tệ không ghi rõ ràng, và cả hai trường hợp x > 10 và x <= 2 đều trả về 2\*x sự trùng lặp vô ích, có thể viết thành 2 nhánh thay vì 3 nhánh

Chọn test-case

|  |  |  |  |
| --- | --- | --- | --- |
| Class | Mô tả | Nhập | Kết quả mong dợi |
| Class 1 | x > 10 | 15 | 30 |
| Class 2 | x = 10 | 10 | -10 |
| Class 3 | 0 < x < 10 | 5 | -5 |
| Class 4 | x = 0 | 0 | 0 |
| Class 5 | x < 0 | -2 | -4 |

c) How many test-cases we need to test this function? What are they?

A white rectangular frame with black border

AI-generated content may be incorrect.

In this case, how many test-cases we need to test this function? What are they?

Hàm có 3 nhánh

x < 10 return 2 \* x

else if (x < 2) return -x;

else return 2 \* x;

nhưng ở else if (x < 2) là dead code vì nó sẽ ko bao giờ đc chạy vì if x < 10 đã bao phủ nó

Chọn test-case

|  |  |  |  |
| --- | --- | --- | --- |
| Class | Mô tả | Nhập | Kết quả mong dợi |
| Class 1 | x < 10 | 9 | 18 |
| Class 2 | x >= 10 | 10 | 20 |
| Class 3 | x < 2 nhưng x < 10 đã bao quát nó | 1 | 2 chứ ko phải là -1 |

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

A white rectangular object with black border

AI-generated content may be incorrect.

Có 2 nhánh if else nhưng return lại cùng 1 công thức, rất sai logic.ta chỉ cần 2 test case là giải quyết đc bài toàn

Ví dụ như khi cho x = 0 thì log(x \* x \* x \* cos(x)) = inf ( Vô cực) > 0, hơi kì quặc kết quả chạy trên C++, còn trên py thì yêu cầu math.log() đối số phải lớn hơn 0.

Chọn test-case

|  |  |  |  |
| --- | --- | --- | --- |
| Class | Mô tả | Nhập | Kết quả mong dợi |
| Class 1 | log(x \* x \* x \* cos(x)) < 3 \* x | 1 | 2 |
| Class 2 | log(x \* x \* x \* cos(x)) > 3 \* x | 0 | 0 với C (py thì báo lỗi |

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

A screenshot of a computer code

AI-generated content may be incorrect.

In this case, how many test-cases we need to test this function? What are they?

Hàm này sau vì chưa đưa ra trường hợp có 2 hoặc 3 số bằng nhau, nếu xảy ra 2 trường hợp trên thì đáp án luôn là 0.

|  |  |  |  |
| --- | --- | --- | --- |
| Class | Mô tả | Input | Kết quả mong đợi |
| 1 | n1 > n2 và n1 > n3 | 5,2,3 | 5 |
| 2 | n2 > n1 và n2 > n3 | 2,5,3 | 5 |
| 3 | n3 > n1 và n3 > n2 | 2,3,5 | 5 |
| 4 | n1 = n2 = n3 | 3,3,3 | 3 sai (đáp án ra 0) |
| 5 | n1 = n2 và n1,n2 > n3 | 5,5,3 | 5 sai (đáp án ra 0) |
| 6 | n1 = n3 và n1,n3 > n2 | 5,3,5 | 5 sai (đáp án ra 0) |
| 7 | n2 = n3 và n2,n3 > n3 | 3,5,5 | 5 sai (đáp án ra 0) |
| 8 | n1 > n2 và n1 > n3 | -2,-5,-6 | -2 |

**III. PRATICE 1**

1. Mô tả bài toán, các input / output có thể có của bài toán

**1.1. Mô tả bài toán**

**a·x⁴ + b·x² + c = 0**

Phương trình trùng phương là phương trình bậc 4 có dạng đặc biệt, có thể giải bằng cách đặt ẩn phụ y = x²

Chương trình nhận 3 hệ số a, b, c từ người dùng và tìm nghiệm thực của phương trình

**1.2. Các input / output có thể có của bài toán**

Có 5 output: Vô số nghiệm, vô nghiệm, 1 nghiệm thực, 2 nghiệm thực, 4 nghiệm thực

- **Input (0, 0, 0)** Phương trình: 0 = 0Kết quả mong đợi: vô số nghiệm (Infinite solutions).

- **Input (0, 0, 5)** Phương trình: 5 = 0 Đây là đẳng thức sai với mọi giá trị của x. Kết quả mong đợi: vô nghiệm (No solution).

- **Input (0, 2, -8)** Phương trình: 2x² - 8 = 0 Kết quả mong đợi: 2 nghiệm thực x = 2.0 x = -2.0

**- Input (1, 0, -1)** Phương trình: x⁴ - 1 = 0 => x² = 1.Có 2 nghiệm thự.Kết quả mong đợi: 2 nghiệm thực 1.0 -1.0.

**- Input (1, 0, 1)** Phương trình: x⁴ + 1 = 0 → x⁴ = -1 (không có nghiệm thực).Kết quả mong đợi: vô nghiệm (No solution).

**- Input (1, -2, 1)** Phương trình: x⁴ - 2x² + 1 = 0Có nghiệm kép. Kết quả mong đợi: 2 nghiệm thực 1.0 -1.0

**- Input (1, 5, 6)** Phương trình: x⁴ + 5x² + 6 = 0. Đặt y = x² → y² + 5y + 6 = 0 → nghiệm y = -2, -3 (âm, loại).Kết quả mong đợi: vô nghiệm (No solution).

**- Input (1, -5, 4)** Phương trình: x⁴ - 5x² + 4 = 0Đặt y = x² → y² - 5y + 4 = 0 → nghiệm y = 1 và y = 4.Suy ra: x = 1.0 -1.0, 2.0 -2.0.Kết quả mong đợi: 4 nghiệm thực.

**-Input (2, -3, 1)** Phương trình: 2x⁴ - 3x² + 1 = 0.Đặt y = x² → 2y² - 3y + 1 = 0 → nghiệm y = 1 và y = 0.5.Suy ra: x = ±1, ±√0.5. hay 1.618033988749895 -1.618033988749895 0.6180339887498948 -0.6180339887498948

**- Input (1, 2, 1)** Phương trình: x⁴ + 2x² + 1 = 0 → (x² + 1)² = 0.Nghiệm x² = -1 → không có nghiệm thực.Kết quả mong đợi: vô nghiệm (No solution).

|  |  |  |  |
| --- | --- | --- | --- |
| **Class** | **Mô tả** | **Input** | **Kết quả mong đợi** |
| **1** | Vô số nghiệm | (0, 0, 0) | Infinite solutions |
| **2** | Vô nghiệm | (0, 0, 5) | No solution |
| **3** | 2 nghiệm thực | (0, 2, -8) | 2 real solution(s): 2.0 -2.0 |
| **4** | 2 nghiệm thực | (1, 0, -1) | 2 real solution(s): 1.0 -1.0 |
| **5** | Vô nghiệm | (1, 0, 1) | No solution |
| **6** | 2 nghiệm thực | (1, -2, 1) | 2 real solution(s): 1.0 -1.0 |
| **7** | Vô nghiệm | (1, 5, 6) | No solution |
| **8** | 4 nghiệm thực | (1, -5, 4) | 4 real solution(s): 2.0 -2.0 1.0 -1.0 |
| **9** | 4 nghiệm thực | (2, -3, 1) | 4 real solution(s):  1.618033988749895 -1.618033988749895 0.6180339887498948 -0.6180339887498948 |
| **10** | Vô nghiệm | (1, 2, 1) | No solution |

**1.3. Viết đoạn mã tự động kiểm tra chương trình cho bên dưới đúng hay sai?**

Đúng viết kiểm tra 1 cách tự động đỡ tốn thời gian

test\_cases = [

    ((0,0,0), (-1, [])),

    ((0,0,5), (0, [])),

    ((0,2,-8), (2, [2.0,-2.0])),

    ((1,0,-1), (2, [1.0,-1.0])),

    ((1,0,1), (0, [])),

    ((1,-2,1), (2, [1.0,-1.0])),

    ((1,5,6), (0, [])),

    ((1,-5,4), (4, [1.0,-1.0,2.0,-2.0])),

    ((2,-3,1), (4, [1.0,-1.0,math.sqrt(0.5),-math.sqrt(0.5)])),

    ((1,2,1), (0, [])),

]

for i, (inp, expected) in enumerate(test\_cases, 1):

    x = [0.0] \* 4

    n = solveQuartic(\*inp, x)

    roots = x[:n]

    roots\_set = set([round(r,5) for r in roots])

    expected\_set = set([round(r,5) for r in expected[1]])

    if n == expected[0] and roots\_set == expected\_set:

        print(f"Test {i}: PASSED")

    else:

        print(f"Test {i}: FAILED (got {n}, {roots}, expected {expected})")