**ĐẠI HỌC QUỐC GIA HÀ NỘI**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ**

**----------------------------------------**

A blue and white logo

Description automatically generated

**BÁO CÁO**

**BỘ MÔN: KIỂM THỬ VÀ ĐẢM BẢO CHẤT  
LƯỢNG PHẦN MỀM**

**ĐỀ TÀI: KIỂM THỬ DÒNG DỮ LIỆU**

|  |  |
| --- | --- |
| **Giảng viên:** | ThS. Nguyễn Thu Trang |
| **Sinh viên:** | Đặng Tiến Dũng |
| **Mã sinh viên:** | 21020753 |
| **Lớp:** | K66-IT3 |

**Hà Nội, 10/2024**

**Mục lục**

[1. Bài tập trong slide 2](#_Toc179402638)

[1.1. Bài 1 2](#_Toc179402639)

[1.2. Bài 2 5](#_Toc179402640)

[1.3. Bài 3 7](#_Toc179402641)

[2. Bài tập cá nhân 8](#_Toc179402642)

[2.1. Đồ thị dòng điều khiển 8](#_Toc179402643)

[2.2. Phân tích 8](#_Toc179402644)

[2.3. Thiết kế ca kiểm thử 9](#_Toc179402645)

[2.4. Kiểm thử 9](#_Toc179402646)

# Bài tập trong slide

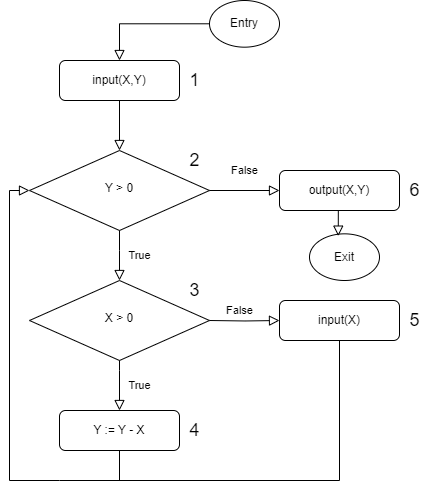
## Bài 1

Trình bày các bước trong quy trình kiểm thử dòng dữ liệu động:

1. Vẽ đồ thị luồng điều khiển
2. Lựa chọn tiêu chí kiểm thử luồng dữ liệu
3. Xác định các đường đi trên đồ thị luồng điều khiển thoả mãn tiêu chí kiểm thử đã chọn
4. Sinh các ca kiểm thử tương ứng

## Bài 2

**Đồ thị dòng điều khiển:**



**Các cặp du-pair với biến X:**

1-6, 1-3(T), 1-3(F), 1-4, 5-3(T), 5-3(F), 5-4, 5-6

**Các cặp du-pair với biến Y:**

1-2(T), 1-2(F), 1-4, 1-6, 4-2(T), 4-2(F), 4-4, 4-6

**Với độ đo all-use**

Lưu ý: Các số đằng sau biến là lần nhập khi input

Đối với biến X:

|  |  |  |  |
| --- | --- | --- | --- |
| Du-pair | Def-clear-path | Complete path | Test-case |
| 1-3(T) | 1-2(T)-3(T) | 1-2(T)-3(T)-4-2(F)-6 | X = 5, Y = 2 |
| 1-3(F) | 1-2(T)-3(F) | 1-2(T)-3(F)-5-2(T)-3(T)-4-2(F)-6 | X1 = 0, Y = 2, X2 = 5 |
| 1-4 | 1-2(T)-3(T)-4 | 1-2(T)-3(T)-4-2(F)-6 | X = 5, Y = 2 |
| 1-6 | 1-2(F)-6 | 1-2(F)-6 | X = 5, Y = 0 |
| 5-3(T) | 5-2(T)-3(T) | 1-2(T)-3(F)-5-2(T)-3(T)-4-2(F)-6 | X1 = 0, Y = 2, X2 = 5 |
| 5-3(F) | 5-2(T)-3(F) | 1-2(T)-3(F)-5-2(T)-3(F)-5-2(T)-3(T)-4-2(F)-6 | X1 = -1, Y = 2, X2 = 0, X3 = 5 |
| 5-4 | 5-2(T)-3(T)-4 | 1-2(T)-3(F)-5-2(T)-3(T)-4-2(F)-6 | X1 = 0, Y = 2, X2 = 5 |
| 5-6 | 5-2(T)-3(T)-4-2(F)-6 | 1-2(T)-3(F)-5-2(T)-3(T)-4-2(F)-6 | X1 = 0, Y = 2, X2 = 5 |

Đối với biến Y:

|  |  |  |  |
| --- | --- | --- | --- |
| Du-pair | Def-clear-path | Complete path | Test-case |
| 1-2(T) | 1-2(T) | 1-2(T)-3(T)-4-2(F)-6 | X = 5, Y = 2 |
| 1-2(F) | 1-2(F) | 1-2(F)-6 | X = 5, Y = 0 |
| 1-4 | 1-2(T)-3(T)-4 | 1-2(T)-3(T)-4-2(F)-6 | X = 5, Y = 2 |
| 1-6 | 1-2(F)-6 | 1-2(F)-6 | X = 5, Y = 0 |
| 4-2(T) | 4-2(T) | 1-2(T)-3(T)-4-2(T)-3(T)-4-2(F)-6 | X = 3, Y = 5 |
| 4-2(F) | 4-2(F) | 1-2(T)-3(T)-4-2(F)-6 | X = 5, Y = 2 |
| 4-4 | 4-2(T)-3(T)-4 | 1-2(T)-3(T)-4-2(T)-3(T)-4-2(F)-6 | X = 3, Y = 5 |
| 4-6 | 4-2(F)-6 | 1-2(T)-3(T)-4-2(F)-6 | X = 5, Y = 2 |

## Bài 3

**Đồ thị dòng điều khiển**

A diagram of a flowchart

Description automatically generated

**Các câu lệnh**

|  |  |  |
| --- | --- | --- |
| n | def | 1, |
| p-use | 4 |
| c-use |  |
| result | def | 2, 5 |
| p-use |  |
| c-use | 5,7 |
| i | def | 3, 6 |
| p-use | 4 |
| c-use | 5, 6 |

## Bài 4

A diagram of a algorithm

Description automatically generated

**Câu 1,2: Các def-clear-path và du-path**

|  |  |  |
| --- | --- | --- |
| Var | def-clear-path | Có là du-path không ? |
| x | 0-1 | x |
| 0-1-2 |  |
| 0-1-2-4 | x |
| 0-1-2-4-6 |  |
| 0-1-2-4-5 | x |
| 0-1-2-4-5-6 |  |
| 0-1-3 | x |
| 3-4 | x |
| 3-4-6 |  |
| 3-4-5 | x |
| 3-4-5-6 |  |
| y | 0-1 | x |
| 0-1-3 |  |
| 0-1-3-4 | x |
| 0-1-3-4-5 (c-use(x)) |  |
| 0-1-3-4-6 | x |
| 2-4 | x |
| 2-4-5 (c-use(x)) |  |
| 2-4-6 | x |
| 5-6 | x |

**Câu 3: Xác định các All-p-uses/ Some-c-uses và All-c-uses/ Some-p-uses**

All-p-uses/ Some-c-uses X:

* 0-1
* 0-1-2-4
* 0-1-3 (khi c-use(x))
* 3-4

All-c-uses / Some-p-uses X:

* 0-1-3 (khi c-use(x))
* 0-1-2-4-5
* 3-4-5

All-p-uses/ Some-c-uses Y:

* 0-1
* 0-1-3-4
* 2-4
* 2-4-6
* 5-6

All-c-uses/ Some-p-uses Y:

* 0-1-3-4-6
* 2-4-6
* 5-6

**Câu 4:**

Đường đi có được thực thi.

Ta có x + y = 4

⬄ x2 + y2 + 2\*x\*y = 16

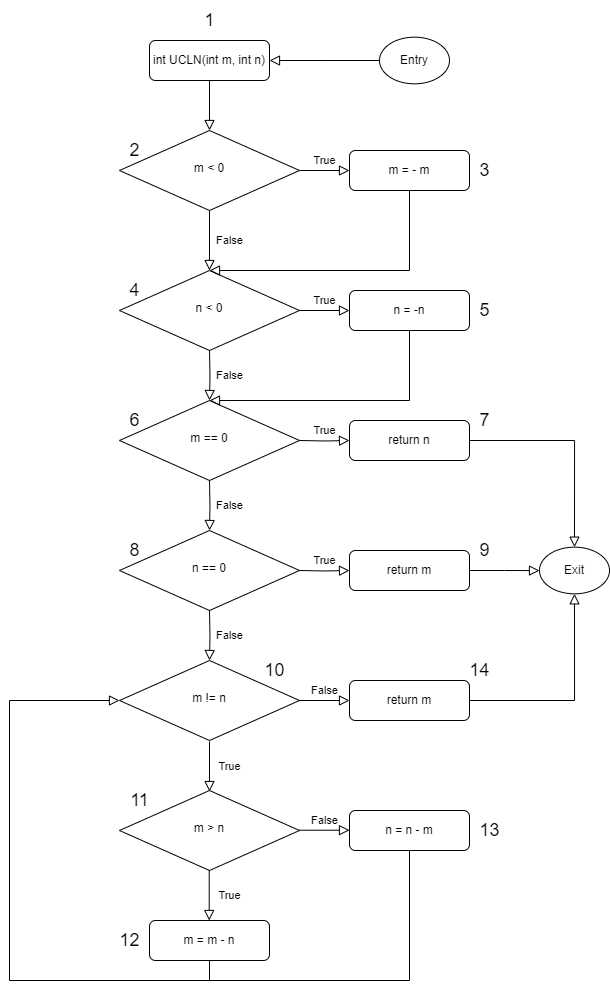
⬄ x2 + y2 = 16 – 2\*x\*y

Vậy để x2 + y2 > 17 thì 16 – 2\*x\*y > 17. Vậy x\*y < -1. Điều này hoàn toàn có thể vì không còn các điều kiện khác của x và y.

**Câu 5:**

Tại đỉnh 3, vừa được định nghĩa và sử dụng nhưng không tồn tại mối quan hệ def-use do thông thường các biến được sử dụng trước khi được định nghĩa lại. Do vậy việc sử dụng được thực thi trước khi định nghĩa lại. Sẽ không thể có luồng ngược lại vậy nên không tồn tại cặp d-u ở đỉnh 3.

## Bài 5



# Bài tập cá nhân

## Đồ thị dòng điều khiển

A diagram of a function

Description automatically generated

## Phân tích

Trong bài toán, 2 điều kiện rẽ nhánh, vậy nên sẽ có 4 nhánh đường đi khác nhau từ 2 điều kiện đó.

P1: 1-2T-3

P2: 1-2F-4T-5

P3: 1-2F-4F-6

## Thiết kế ca kiểm thử

**T1:**

Để có thể đạt hết đường đi P1, ta cần đạt được điều kiện 2 true. Vậy ta có ca kiểm thử với input weight = -1, distance = 0.

**T2:**

Để có ca kiểm thử đi hết đường đi P2, ta cần đạt điều kiện 2 false và 4 true. Vậy ta có hệ

* weight >= 0 AND distance >= 0
* weight > 10 OR distance > 100

Vậy ta có ca kiểm thử với input weight = 4, distance = 160.

**T3:**

Để có ca kiểm thử đi hết đường đi P3, ta cần đạt điều kiện 2 false và 4 false. Vậy ta có hệ

* weight >= 0 AND distance >= 0
* weight <= 10 AND distance <= 100

Vậy ta có ca kiểm thử với input weight = 5, distance = 50.

## Kiểm thử

Các test case được thực hiện trong file Week2ServiceTest.java trong: [Link github](https://github.com/TDungx2k3/Testing_Assignment/tree/master/code/src/test/java/org/example)

|  |  |  |  |
| --- | --- | --- | --- |
| Input | Expected Output | Output | Status |
| weight = -1, distance = 0 | “Invalid input” => -1 | “Invalid input” => -1 | Pass |
| weight = 4, distance = 160 | “Overload” => -1 | “Overload” => -1 | Pass |
| weight = 5, distance = 50 | => 120000 | => 120000 | Pass |

Kết quả kiểm thử cho thấy tất cả các test case đều pass và có thể đạt được các trường hợp đầu ra mong muốn.

Tuy nhiên, với mỗi đầu ra chỉ có 1 ca kiểm thử. Điều này có thể không phát hiện lỗi trong nhiều trường hợp đầu vào khác.