Họ và tên: Trịnh Kiều Anh

MSSV: 21020282

**BÀI TẬP KIỂM THỬ DÒNG ĐIỀU KHIỂN**

1. **Trình bày các bước trong quy trình kiểm thử dòng dữ liệu động**

*Trả lời:*

B1: Vẽ đồ thị luồng điều khiển (CFG)

B2: Lựa chọn tiêu chí kiểm thử luồng dữ liệu

B3: Xác định các đường đi trên CFG thoả mãn tiêu chí kiểm thử đã chọn

B4: Sinh các ca kiểm thử tương ứng

1. **Cho đoạn mã nguồn sau, hãy:**

*input*(X, Y) #1

*while* (Y > 0) { #2

*if* (X > 0) #3

        Y := Y – X #4

*else*

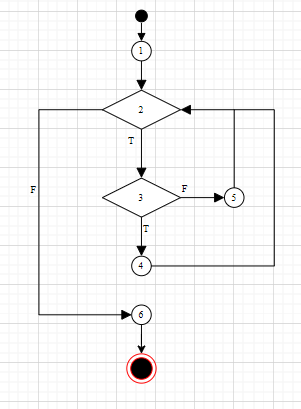
*input*(X) #5

}

*output*(X, Y) #6

* Vẽ đồ thị dòng điều khiển (CFG)
* Xác định các du-pairs cho biến X và Y
* Sinh đường đi và các ca kiểm thử với độ đo all–use

*Trả lời:*



Biến *X*:

Biến *Y*:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Variable** | **Du-pair** | **Def-clear path** | **Complete path** | **Input (X; Y)** |
| *X* | (1; 3(T)) | (1; 2(T); 3(T)) | (1; 2(T); 3(T); 4; 2(F); 6) | (3; 3) |
|  | (1; 4) | (1; 2(T); 3(T); 4) | (1; 2(T); 3(T); 4; 2(F); 6) | (2; 2) |
|  | (1; 6) | (1; 2(F); 6) | (1; 2(F); 6) | (3; -2) |
| *Y* | (1; 2(T)) | (1; 2(T)) | (1; 2(T); 3(T); 4; 2(F); 6) | (4; 4) |
|  | (1; 2(F)) | (1; 2(F)) | (1; 2(F); 6) | (4; -1) |
|  | (1; 4) | (1; 2(T); 3(T); 4) | (1; 2(T); 3(T); 4; 2(F); 6) | (5; 5) |
|  | (1; 6) | (1; 2(F); 6) | (1; 2(F); 6) | (3; -3) |
|  | (4; 4) | (4; 2(T); 3(T); 4) | (1; 2(T); 3(T); 4; 2(T); 3(T); 4; 2(F); 6) | (3; 5) |
|  | (4; 6) | (4; 2(F); 6) | (1; 2(T); 3(T); 4; 2(F); 6) | (7; 6) |

1. **Cho hàm calFactorial viết bằng ngôn ngữ C như Đoạn mã 7.7.**

* Hãy liệt kê các câu lệnh ứng với các khái niệm def, c−use, và p−use ứng với các biến được sử dụng trong hàm này.
* Hãy vẽ đồ thị dòng dữ liệu của hàm này.

int *calFactorial*(int n){ #1

    int result = 1; #2

    int i = 1; #2

*while* (i <= n){ #3

        result = result \* i; #4

        i++; #4

    }// *end while*

*return* result; #5

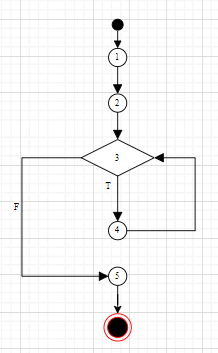
}// *the end*

*Trả lời:*

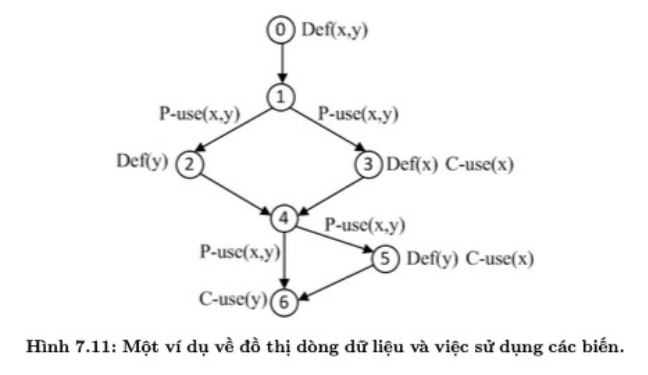
Biến *n*:

Biến *result*:

Biến *i*:



1. **Cho đồ thị dòng dữ liệu như hình 7.11.**

****

* Hãy xác định tất cả các Def-clear-path của các biến x và y.
* Hãy xác định tất cả các du-paths của các biến x và y.
* Hãy xác định tất cả các All-p-uses/Some-c-uses và All-c-uses/Some-p-uses (dựa vào các chuẩn của kiểm thử dòng dữ liệu).
* Biểu thức của các p-use(x, y) tại cạnh (1,3) và (4,5) lần lượt là x + y = 4 và x2 + y2 > 17. Đường đi (0 - 1 - 3 - 4 - 5 - 6) có thực thi được không? Giải thích.
* Tại sao tại đỉnh 3 biến x được định nghĩa và sử dụng nhưng không tồn tại mối quan hệ def-use?

*Trả lời:*

Biến *x*:

Biến *y*:

Du-paths(x): (0; 1; 2; 4; 5; 6); (0; 1; 2; 4; 6)

Du-paths(y): (0; 1; 3; 4; 6)

All-p-uses/Some-c-uses

|  |  |  |  |
| --- | --- | --- | --- |
| **Variable** | **Du-pair** | **Def-clear path** | **Complete path** |
| *x* | (0; 1) | (0; 1) | (0; 1; 2; 4; 6) |
|  | (0; 4) | (0; 1; 2; 4) | (0; 1; 2; 4; 6) |
|  | (3; 4) | (3; 4) | (0; 1; 3; 4; 6) |
| *y* | (0; 1) | (0; 1) | (0; 1; 3; 4; 6) |
|  | (0; 4) | (0; 1; 3; 4) | (0; 1; 3; 4; 6) |
|  | (2; 4) | (2; 4) | (0; 1; 2; 4; 6) |
|  | (5; 6) | (5; 6) | (0; 1; 3; 4; 5; 6) |

All-c-uses/Some-p-uses

|  |  |  |  |
| --- | --- | --- | --- |
| **Variable** | **Du-pair** | **Def-clear path** | **Complete path** |
| *x* | (0; 3) | (0; 1; 3) | (0; 1; 3; 4; 6) |
|  | (0; 5) | (0; 1; 2; 4; 5) | (0; 1; 2; 4; 6) |
|  | (3; 5) | (0; 1; 3; 4; 5) | (0; 1; 3; 4; 5; 6) |
| *y* | (0; 6) | (0; 1; 3; 4; 6) | (0; 1; 3; 4; 6) |
|  | (2; 6) | (2; 4; 6) | (0; 1; 2; 4; 6) |
|  | (5; 6) | (5; 6) | (0; 1; 3; 4; 5; 6) |

Biểu thức của các p-use(x, y) tại cạnh (1,3) và (4,5) lần lượt là x + y = 4 và x2 + y2 > 17. Đường đi (0 - 1 - 3 - 4 - 5 - 6) không thực thi được vì không tồn tại cặp giá trị (x; y) thoả mãn cả 2 p-use(x; y) như trên.

Tại đỉnh 3 biến x được định nghĩa và sử dụng nhưng không tồn tại mối quan hệ def-use vì: không tồn tại def-clear path thoả mãn từ đỉnh (3) đến đỉnh có use(x)

1. **Cho đoạn mã nguồn như hình bên,**

int *UCLN* (int m , int n){ #1

*if* (m < 0) #2

        m = -m; #3

*if* (n < 0) #4

        n = -n; #5

*if* (m == 0) #6

*return* n; #7

*if* (n == 0) #8

*return* m; #9

*while* (m != n){ #10

*if* (m > n) #11

            m = m - n; #12

*else*

            n = n - m; #13

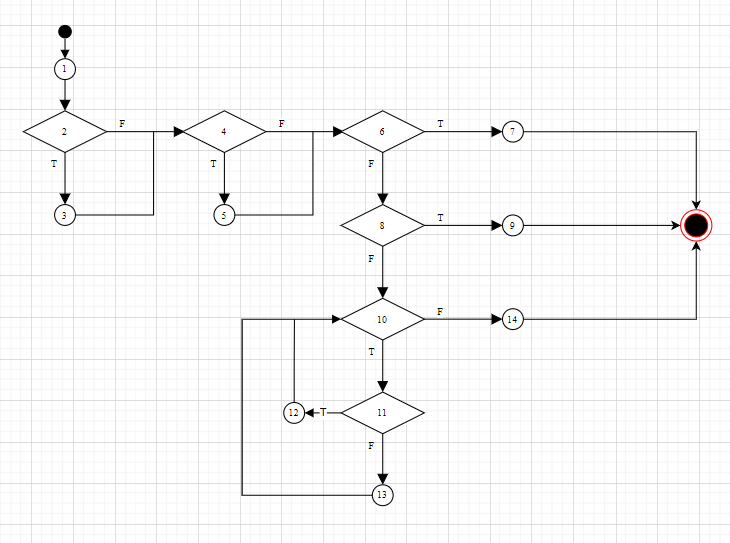
    }// *end while*

*return* m; #14

}

* Xây dựng CFG cho hàm UCLN với đồ thị C2
* Sinh đường đi và các ca kiểm thử với độ đo C2
* Sinh đường đi và các ca kiểm thử với độ đo all-def coverage

*Trả lời:*



Sinh đường đi và các ca kiểm thử với độ đo C2

|  |  |  |  |
| --- | --- | --- | --- |
| **Đường đi** | **Input (m; n)** | **E.O** | **Output** |
| 1 – 2(F) – 4(T) – 5 – 6(T) – 7 | (0; -1) | 1 | 1 |
| 1 – 2(T) – 3 – 4(F) – 6(F) – 8(T) – 9 | (-1; 0) | 1 | 1 |
| 1 – 2(F) – 4(F) – 6(F) – 8(F) – 10(F) – 14 | (1; 1) | 1 | 1 |
| 1 – 2(F) – 4(F) – 6(F) – 8(F) – 10(T) – 11(T) – 10(F) – 14 | (6; 3) | 3 | 3 |
| 1 – 2(F) – 4(F) – 6(F) – 8(F) – 10(T) – 11(F) – 13 – 10(F) – 14 | (2; 4) | 2 | 2 |

Sinh đường đi và các ca kiểm thử với độ đo all-def coverage

Biến *m*:

Biến *n*:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Variable** | **Du-pair** | **Def-clear path** | **Complete path** | **Input (m; n)** |
| *m* | (1; 3) | (1; 2(T); 3) | (1; 2(T); 3; 4(F); 6(F); 8(T); 9) | (-3; 0) |
|  | (3; 9) | (3; 4(F); 6(F); 8(T); 9 | (1; 2(T); 3; 4(F); 6(F); 8(T); 9) | (-5; 0) |
|  | (12; 12) | (12; 10(T); 11(T); 12) | (1; 2(F); 3(F); 6(F); 8(F); 10(T); 11(T); 12; 10(T); 11(T); 12; 10(F); 14) | (9; 3) |
| *n* | (1; 4(F)) | (1; 2(F); 4(F)) | (1; 2(F); 4(F); 6(T); 7) | (0; 3) |
|  | (5; 7) | (5; 6(T); 7) | (1; 2(F); 4(T); 5; 6(T); 7) | (0; -3) |
|  | (13; 13) | (13; 10(T); 11(F); 13) | (1; 2(F); 3(F); 6(F); 8(F); 10(T); 11(F); 13; 10(T); 11(F); 13; 10(F); 14) | (3; 9) |

**PHÂN TÍCH, THIẾT KẾ CÁC CA KIỂM THỬ VÀ KIỂM THỬ CHƯƠNG TRÌNH CỦA BẠN VỚI ĐỘ PHỦ ALL–USES**

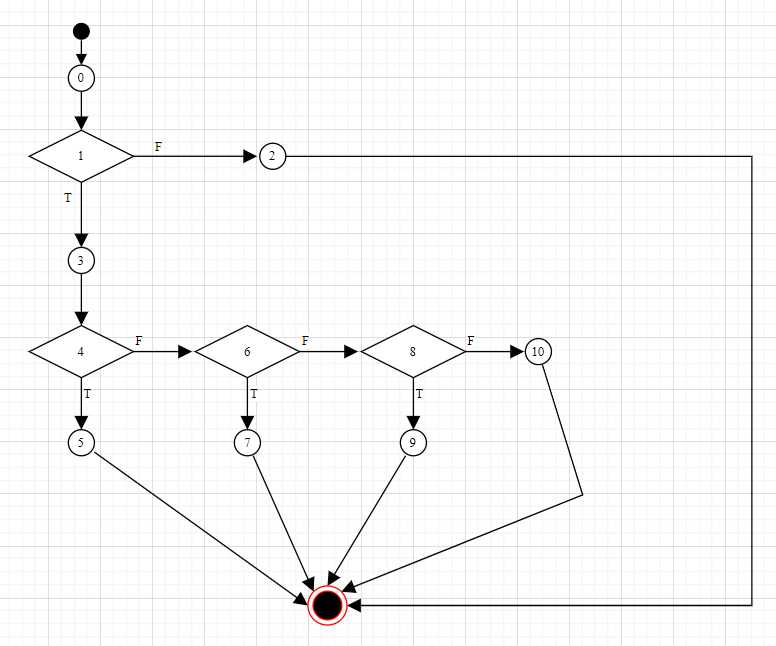
**Bài toán:** Đánh giá chỉ số BMI của người châu Á.

**Link github:** <https://github.com/tkanhhh/kiemthudongdieukhien>

**Mô tả bài toán:**

* Sản phẩm đánh giá: chỉ số BMT của người châu Á
* Đầu vào:
  + Cân nặng: weight(kg)
  + Chiều cao: height(m)
* Công thức tính:
* Đầu ra:
  + Đầu vào không hợp lệ: weight <= 0 || height <= 0
  + Nhẹ cân: BMI < 18.5
  + Bình thường:18,5 <= BMI < 23
  + Thừa cân: 23 <= BMI < 30
  + Béo phì: BMI >= 30





Biến *weight*:

Biến *height*:

Biến *BMI*:

|  |  |  |  |
| --- | --- | --- | --- |
| **Variable** | **Du-pair** | **Def-clear path** | **Complete path** |
| *weight* | (0; 1(T)) | (0; 1(T)) | (0; 1(T); 2) |
|  | (0; 1(F)) | (0; 1(F)) | (0; 1(F); 3; 4(T); 5) |
|  | (0; 3) | (0; 1(F); 3) | (0; 1(F); 3; 4(T); 5) |
| *height* | (0; 1(T)) | (0; 1(T)) | (0; 1(T); 2) |
|  | (0; 1(F)) | (0; 1(F)) | (0; 1(F); 3; 4(T); 5) |
|  | (0; 3) | (0; 1(F); 3) | (0; 1(F); 3; 4(T); 5) |
| *BMI* | (3; 4(T)) | (3; 4(T); 5) | (0; 1(F); 3; 4(T); 5) |
|  | (3; 4(F)) | (3; 4(F)) | (0; 1(F); 3; 4(F); 6(T); 7) |
|  | (3; 6(T)) | (3; 4(F); 6(T)) | (0; 1(F); 3; 4(F); 6(T); 7) |
|  | (3; 6(F)) | (3; 4(F); 6(F)) | (0; 1(F); 3; 4(F); 6(F); 8(T); 9) |
|  | (3; 8(T)) | (3; 4(F); 6(F); 8(T)) | (0; 1(F); 3; 4(F); 6(F); 8(T); 9) |
|  | (3; 8(F)) | (3; 4(F); 6(F); 8(F)) | (0; 1(F); 3; 4(F); 6(F); 8(F); 10) |