*Họ tên: Nguyễn Văn Dũng*

*Mã sinh viên: 20020192*

*Mã học phần: INT3117\_20*

*Link github: https://github.com/nvdung1607/kiemThuDongDieuKhienC2.git*

**KIỂM THỬ LUỒNG DỮ LIỆU**

*(Áp dụng với độ phủ all-c-uses/some-p-uses)*

**BÀI TOÁN:**

Xếp loại học danh hiệu dựa trên điểm tổng kết và hạnh kiểm:

* Điểm tổng kết:
  + Học lực giỏi: Điểm trung bình tất cả các môn từ 8.0 trở lên
  + Học lực khá: Chưa đạt học lực giỏi, điểm trung bình tất cả các môn từ 6.5 trở lên
  + Học lực trung bình: Trường hợp còn lại
* Hạnh kiểm: Tốt, Khá, Trung bình, Yếu
* Công nhận đạt danh hiệu học sinh giỏi cho học sinh có học lực giỏi và hạnh kiểm tốt
* Công nhận đạt danh hiệu học sinh tiên tiến cho học sinh nếu đạt hạnh kiểm từ loại khá trở lên và học lực từ loại khá trở lên
* Không công nhận danh hiệu với các trường hợp còn lại.

**Yêu cầu bài toán:** Viết hàm; truyền vào 2 tham số: Điểm tổng kết và hạnh kiểm; đưa ra xếp loại danh hiệu học sinh.

**Mã nguồn:** (Javascript)

*main.js*

export function danhHieu(hocLuc, hanhKiem) {            //1

    let result = 'khongCoDanhHieu'                      //2

        if (hocLuc >= 8.0 && hanhKiem == 'tot')         //3

            result = 'hocSinhGioi'                      //4

        else if (hocLuc >= 6.5 && hanhKiem == 'tot')    //5

            result = 'hocSinhTienTien'                  //6

        else if (hocLuc >= 6.5 && hanhKiem == 'kha')    //7

            result = 'hocSinhTienTien'                  //8

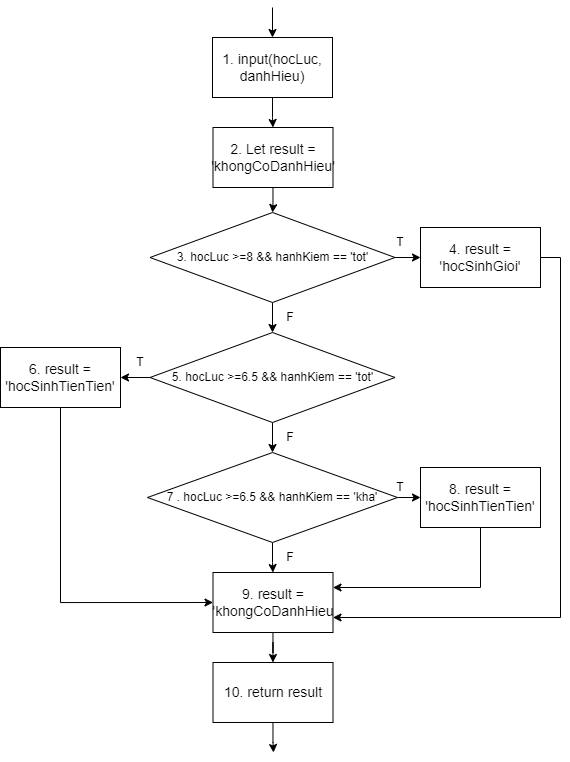
        else result =  'khongCoDanhHieu'                //9

    return result                                       //10

}

**Đồ thị dòng dữ liệu (DFG)**

Đánh số thứ tự cho các câu lệnh như phần comment trong phần source code ở trên.



**KIỂM THỬ VỚI ĐỘ PHỦ ALL-C-USES/SOME-P-USES:**

* **Xác định def, c-use, p-use của các biến trong chương trình:** 
  + Biến hocLuc:
    - Def(hocLuc) = {1}
    - C-uses(hocLuc) = {}
    - P-uses(hocLuc) = {3, 5, 7}
  + Biến hanhKiem:
    - Def(hanhKiem) = {1}
    - C-uses(hanhKiem) = {}
    - P-uses(hanhKiem) = {3, 5, 7}
  + Biến result:
    - Def(result) = {2}
    - C-uses(result) = {2, 4, 6,8, 9, 10}
    - P-uses(result) = {}
* **Xác định các đường đi trên DFG thỏa mãn tiêu chí kiểm thử all-c-uses/ some-p-uses**

|  |  |  |  |
| --- | --- | --- | --- |
| Variable | Du-pair | Def-clear path | Complete path |
| hocLuc | (1,3(T)) | 1, 2, 3(T) | 1, 2, 3(T), 4, 9, 10 |
| (1,3(F)) | 1, 2, 3(F) | 1, 2, 3(F), 5(T), 6, 9, 10 |
| hanhKiem | (1,5(T)) | 1, 2, 3(F), 5(T) | 1, 2, 3(F), 5(T), 6, 9, 10 |
| (1,5(F)) | 1, 2, 3(F), 5(F) | 1, 2, 3(F), 5(F), 7(F), 9, 10 |
| reusult | (2,2) | 2 | 1, 2, 3(T), 4, 9, 10 |
| (2,4) | 2, 3(T), 4 | 1, 2, 3(T), 4, 9, 10 |
| (2,6) | 2, 3(F), 5(T), 6 | 1, 2, 3(F), 5(T), 6, 9, 10 |
| (2,8) | 2, 3(F),5(F), 7(T), 8 | 1, 2, 3(F),5(F), 7(T), 8, 9, 10 |
| (2,9) | 2, 3(F), 5(T), 6, 9 | 1, 2, 3(F), 5(T), 6, 9, 10 |
| (2,10) | 2, 3(F), 5(T), 6, 9, 10 | 1, 2, 3(F), 5(T), 6, 9, 10 |

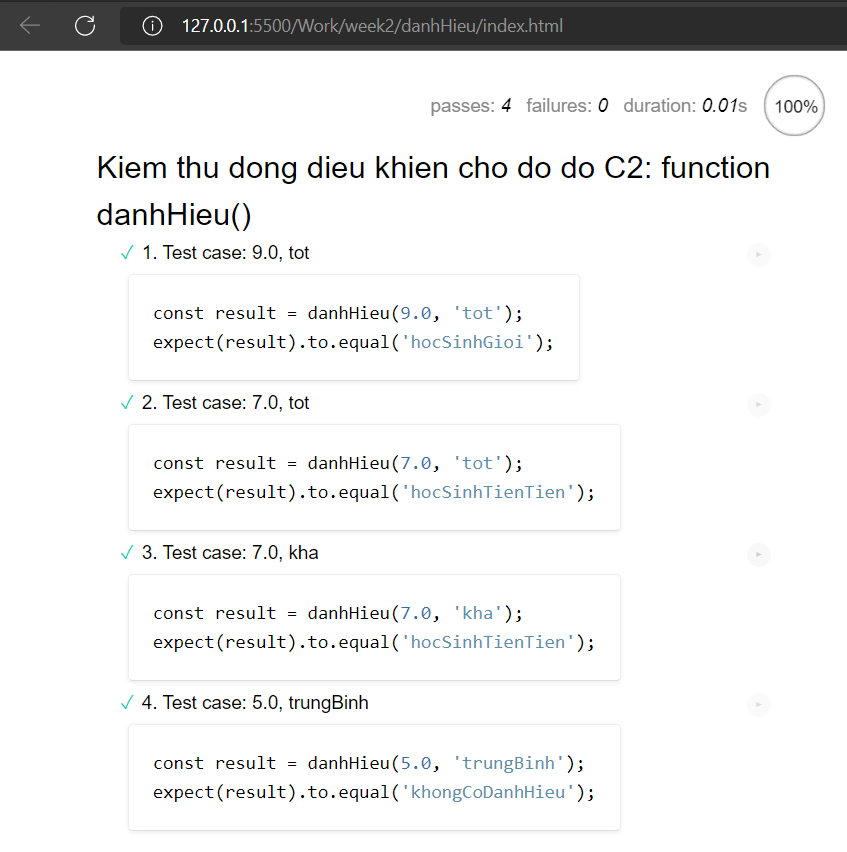
* **Loại bỏ trùng lặp, xác định được các complete paths**

1. Start, 1, 2, 3(T), 4, 9, 10, End
2. Start, 1, 2, 3(F), 5(T), 6, 9, 10, End
3. Start, 1, 2, 3(F),5(F), 7(T), 8, 9, 10, End
4. Start, 1, 2, 3(F), 5(F), 7(F), 9, 10, End

* **Sinh các ca kiểm thử từ các đường đi tương ứng**

1. danhHieu(9.0, ‘tot’), return ‘hocSinhGioi’
2. danhHieu(7.0, ‘tot’), return ‘hocSinhTienTien’
3. danhHieu(7.0, ‘kha’), return ‘hocSinhTienTien’
4. danhHieu(5.0, ‘trungBinh’), return ‘khongCoDanhHieu’

* **Thực thi test cases**
  + Javascript Test Framework: Mocha
  + Assertion Library: Chai
  + Mã nguồn:
    - *Index.html*
* <!DOCTYPE html>
* <html lang="en">
* <head>
* <meta charset="utf-8" />
* <title>Mocha Tests</title>
* <meta name="viewport" content="width=device-width, initial-scale=1.0" />
* <link rel="stylesheet" href="https://unpkg.com/mocha/mocha.css" />
* </head>
* <body>
* <div id="mocha"></div>
* <script src="https://unpkg.com/chai/chai.js"></script>
* <script src="https://unpkg.com/mocha/mocha.js"></script>
* <script class="mocha-init">
* mocha.setup('bdd');
* mocha.checkLeaks();
* </script>
* <script type="module" src="./main.test.js"></script>
* <script type="module" class="mocha-exec">
* mocha.run();
* </script>
* </body>
* </html>
  + - *Main.test.js*
* import { danhHieu } from './main.js'
* const expect = chai.expect;
* describe('Kiem thu dong dieu khien cho do do C2: function danhHieu()', () => {
* it('1. Test case: 9.0, tot', () => {
* const result = danhHieu(9.0, 'tot');
* expect(result).to.equal('hocSinhGioi');
* });
* it('2. Test case: 7.0, tot', () => {
* const result = danhHieu(7.0, 'tot');
* expect(result).to.equal('hocSinhTienTien');
* });
* it('3. Test case: 7.0, kha', () => {
* const result = danhHieu(7.0, 'kha');
* expect(result).to.equal('hocSinhTienTien');
* });
* it('4. Test case: 5.0, trungBinh', () => {
* const result = danhHieu(5.0, 'trungBinh');
* expect(result).to.equal('khongCoDanhHieu');
* });
* });
* **Kết quả thực thi test:**



* **Bảng tổng hợp kết quả:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | Input | | Output | | Result |
| hocLuc | hanhKiem | Expected output | Actually output |
| 1 | 9.0 | ‘tot’ | 'hocSinhGioi' | 'hocSinhGioi' | Pass |
| 2 | 7.0 | ‘kha’ | 'hocSinhTienTien' | 'hocSinhTienTien' | Pass |
| 3 | 7.0 | ‘kha’ | 'hocSinhTienTien' | 'hocSinhTienTien' | Pass |
| 4 | 5.0 | ‘trungBinh’ | 'khongCoDanhHieu' | 'khongCoDanhHieu' | Pass |