

Bài thực hành số 2

KIỂM THỬ DÒNG ĐIỀU KHIỂN

Phần 1: (Làm theo nhóm; 27/9-03/10)

- Hoàn thiện về Test Plan cơ bản các mục, thông tin còn lại
- Xem kỹ hơn vào Test case template, và phân tích các unit test (tương ứng với từng usecase) của mỗi đề tài lựa chọn theo nhóm

Phần 2: (Làm theo cá nhân; hoàn thiện tại lớp)

I. Các tiện ích được dùng cho bài thực hành này :

JUnit, dJUnit hoặc CppUnit (dùng với code ở bài 1)

Địa chỉ download và tham khảo thông tin về các tiện ích :

- <http://www.junit.org/home>
- <https://www.jetbrains.com/idea/download/#section=windows> (Download IntelliJ IDEA)
- <https://www.jetbrains.com/help/idea/junit.html#54a117f6> (Ví dụ về junit)

Các tài liệu

- <http://articles.jbrains.ca/JUnitAStarterGuide.pdf>
- <http://agile.csc.ncsu.edu/SEMaterials/tutorials/djunit/>
- <http://www.eclEmma.org/index.html>

CppUnit: <https://sourceforge.net/projects/cppunit/>

II. Các yêu cầu :

1. Thiết kế Test Case
2. Dùng tiện ích JUnit/CppUnit để hiện thực các testcase.
3. Đánh giá phủ kiểm thử

II.1 Thiết kế Test Case

Xem chương trình dưới đây, tạo đồ thị dòng điều khiển cho nó, đổi ra đồ thị dòng điều khiển nhị phân, rồi đổi ra đồ thị dòng điều khiển cơ bản, xác định độ phức tạp C, xác định C đường thi hành tuyến tính cơ bản độc lập.

1. Làm với chương trình ở bài lab 1 (hoặc dùng với cppUnit, hoặc viết lại code trên java và sử dụng với JUnit)

2. Và cho đoạn mã dưới

Chú ý : s1, s2, ... là các phát biểu thực thi Java, còn c1, c2, ... miêu tả các điều kiện quyết định.

```
boolean evaluateBuySell (TickerSymbol ts) {
    s1;
    s2;
    s3;
    if (c1) {s4; s5; s6;}
        else {s7; s8;}
    while (c2) {
        s9;
        s10;
        switch (c3) {
            case-A:
                s20;
                s21;
```

```

        s22;
        break; // End of Case-A
case-B:
    s30;
    s31;
    if (c4) {
        s32;
        s33;
        s34;
    }
    else {
        s35;
    }
    break; // End of Case-B
case-C:
    s40;
    s41;
    break; // End of Case-C
case-D:
    s50;
    break; // End of Case-D
} // End Switch
s60;
s61;
s62;
if (c5) {s70; s71; }
s80;
s81;
} // End While
s90;
s91;
s92;
return result;
}

```

II.2 Dùng tiện ích JUnit để hiện thực các testcase

Dùng tiện ích JUnit để hiện thực các test cases cho Project phần mềm có tên là Triangle

II.3 Đánh giá phủ kiểm thử

Dùng tiện ích dJUnit để thi hành các testcase và đánh giá cấp phủ kiểm thử.