# Assignment 7.1 (Common Defects - 1h)

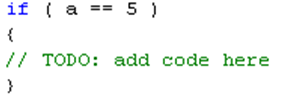
Học viên tìm các lỗi common defect trong thư mục **CodingStandard** và fix các lỗi common defects

1. **Coding Standard**
2. **Hard Code**
3. **Logic**
4. **Performance**

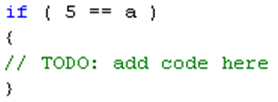
**Rule**: học viên sẽ nộp bài bằng cách: nộp file source đã fix và kèm theo 1 file doc. Trong file doc sẽ chụp lại màn hình hoặc copy các loại loại lổi mà học viên tìm được và nội dung sau khi đã fix.

VD:

**Coding standard:**



**Fixed:**



Tạo thành 1 thư mục và đặt kết quả Assignment 13.1 vào trong thư mục này.

# Assignment 7.2 (Unit Testing – 2h)

Học viên sử dụng kết quả đã làm ở Assignment 13.1 và viết tiếp code theo yêu cầu sau:

1. Xử lý trường hợp số a > b
2. Xử lý trường hợp số b > a

Dùng CPPUnit, viết test scripts để test chương trình cộng hai số nguyên “big” trong 3 trường hợp như sau:

1. a = b
2. a > b
3. a < b

( trong trường hợp a, b > 0)

**Bài tập cộng điểm:**

1. Học viên có thể xử lý trường hợp a,b < 0 ( cộng điểm )
2. Và có thể tạo ra cộng hai số bất kỳ và tạo Unit Test đầy đủ( có thể áp dụng **template** vào ) (cộng điểm)

**Rule**: học viên sẽ nộp bài bằng 2 cách: nộp file source đã fix và kèm theo 1 file doc. Trong file doc sẽ chụp lại màn hình hoặc copy test case mà học viên tạo được. Chụp lại kết quả màn hình sau khi chạy CPPUnit vào file doc.

VD:

/\* Test case 01 : Positive + Positive \*/

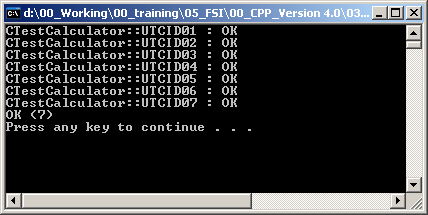
void CTestCalculator::UTCID01()

{

CCalculator cal;

CPPUNIT\_ASSERT(cal.Addition(1,2) == 3);

}



Tạo thành 1 thư mục và đặt kết quả Assignment 2.2 vào trong thư mục này.