Bug 000004 –compare() function may return negative number

Bug 000004 –compare() function may return negative number

Status: solved Reported: 2013-04-12 11:21 AM by Bill Yeung

Product: Code-Similarity Assigned to: Thomas Chan

Component:

Version: unspecified

Platform: PC Microsoft Windows 7

Importance: High

Bill Yeung 2013-04-12 11:21 HKT

When I run the test cases named testCompare\_8 in FuncByFuncCompararTest, it return a negative number, which should be impossible in any test case.

The error message as follow:

=======================================================================================

org.junit.ComparisonFailure: expected:<[]0.25> but was:<[-]0.5>

at org.junit.Assert.assertEquals(Assert.java:123)

at org.junit.Assert.assertEquals(Assert.java:145)

at edu.hkcity.cs.unittest.FuncByFuncCompararTest.testCompare\_8(FuncByFuncCompararTest.java:87)

=======================================================================================

[Solved] Thomas Chan 2013-04-13 08:40 HKT

The bug may occur when the number of function in original string is less than the number of function in target string. In the compare function, it will split the string by function. After that, it loop all the target functions to try to match will the original functions. When all the original functions are paired up (no more original function can pair), all the target function’s similarity will be the initial value of the variable maxSim. Noticed that the maxSim is wrong initialized as -1 and that causes all unmatched target function have a negative similarity. This bug is solved by changing the initial value of maxSim to 0.

<< Before

for (int i = 0; i < targetFuncList.size(); i++) {

double maxSim = -1;

int jMax = -1;

<<After

for (int i = 0; i < targetFuncList.size(); i++) {

double maxSim = 0;

int jMax = -1;

EOF