Asg 4 Guidance

It is not sufficient simply to eliminate the reported bugs. You need to record the debugging process. This means that you need to record the sequence of hypotheses (guesses) and tests (checks) that lead you from the point of failure (where the bug manifests as output) BACKWARDS to the origin of the defect.

You should record each debug log in this form:

Bug X

Point of Failure: At line XX in main, <varY> is incorrect.

H0: var Y is reported incorrectly

T0: set a breakpoint at line XX, visually inspect varY value before output.

Result0: H0 false: varY is reported correctly.

H1: at line XY (some earlier line that affects the value of varY), varZ (some var used to affect varY) is incorrect.

T1: set a breakpoint at line XY, visually inspect varZ value

Result1: H1 true: varZ is incorrect

(H0 etc represent your hypotheses (or guesses) and T0 etc represent the test (or check) you make to validate or falsify your hypotheses) Continue to record hypotheses and tests until you find the bug.

Note that you are not required to write a Junit test for each step – a test can be a prediction on what you will observe at a particular break point in a debugger.

You will need to modify the format suggested above to reflect the exact hypotheses and tests that you perform. The point is that you need to document the logical progression of your debugging process.

Remember – you need to either work back up the dependence chain from the point of failure (externally observable fault), or progressively narrow down the area of code where the bug could be located. Just jumping straight to where you think the bug must be will not get you maximum marks. Maximum marks are for demonstrating you understand the process of scientific debugging and the steps involved in simplifying, isolating, testing, and rectifying, the bug then validating your fix.

If you are looking for higher grades (DI and HD) you should also write an automated test using JUnit (and Mockito as appropriate) that simplifies and replicates the bug. It is usually best to make the bug test fail when the bug is expressed and for the test to pass when correct behaviour is demonstrated.

Simplify where you think the bug should be down to a single function call (perhaps a pretty high level function’, set up the initial conditions for the bug to be expressed (perhaps by setting up some patrons/items and loans, then incrementing the date as appropriate)