

Name:

Answer the questions below. Be clear and concise in your answers, writing in complete sentences and providing examples for support as appropriate. Use as many sentences as you want.

Consider stubs (also known as mock objects) used in unit and integration testing.

1. (5 pts) What is the difference between a stub and an ordinary piece of code?

A stub is a minimal version of a module used to simply provide an interface during testing. Stubs include the same methods as their ordinary counterparts, but hard-code values of the required type instead of actually processing input and producing output. The hard-coded values can be checked and manipulated by the test.

In your project, you developed stubs for unit and integration tests.

2 point for consistent definition/use of term stub

3 points for mentioning hard-coded values and/or minimal version

2. (5 pts) Why should unit tests use stubs of other units instead of the “real” code?

If we use the real code instead of stubs during unit tests, issues with other modules that we are linking against could cause failures of the unit tests. We would like for unit tests to indicate problems with module they were designed to test. Therefore, the unit tests should use stubs for all other modules so that we know that a failed test usually indicates a problem with the unit itself rather than a problem elsewhere. This is a principle called “fault localization,” and is sometimes more important than knowing the mere presence of a fault.

5 points for correlation of unit tests failures and unit itself