

1. Top Down integration testing is a method of performing integration testing from the top to the bottom of a software system's control flow. To ensure software functionality, higher level modules are tested first, followed by lower level modules that are tested and integrated. If some modules aren't ready, stubs are used to test them.

Ex: Imagine testing a car. In top down testing we would first test the car itself, then each important component such as the engine or the light system, and eventually the bulb itself.

2. The Bottom - up strategy of integration testing, is to test the lower level modules first. The tested modules are then utilized to help with the testing of higher - level modules. This method is repeated until all top-level modules have been tested. The next level of modules is generated when the lower level modules have been tested and merged.

Ex: Using the same car from before, in bottom up testing we first break down the bulb and work back testing the whole lighting system, and each component of the engine before testing the engine, and every other major piece before finally testing the car itself.

Similarities:

- Both are incremental testing techniques.
- all modules are tested
- easy to find problematic modules.

Differences:

- Modules testing order is different
- In top-down we may get an early prototype, but for bottom up we may not
- Lower levels aren't tested as thoroughly in top-down testing, being tested last, while in bottom up it's the main modules that isn't tested thoroughly enough, being tested last.