

Design Patterns

- ◉ Automated Testing
- ◉ JUnit
- ◉ Test-Driven Development
- ◉ Test Coverage
- ◉ **Integration Tests** 

Integration Tests

- An integration test crosses unit boundaries
 - When more than one class is involved
 - Interactions with external systems
- Simple examples:
 - Testing that a database works with a software system
 - Testing that a frontend works with a backend
 - Testing a system component works with another system component
 - Testing that the interface for an external data source works with a software system

Limits on Testing

Testing can find the presence of faults, not absence

Testing is difficult for certain domains (games, graphics, ...)

Testing assumes that methods terminate (the halting problem)

- If they don't, testing must freeze or be occasionally wrong

Testing is not a substitute for code review.

Mutation Testing

Basic idea: test the tests

Take some code that passes all the tests

Mutate that code (an operation, a constant, etc.)

One of the tests should now fail

If not, you need more tests!