Testing Introduction

IEEE Definitions

- **Software testing** is the process of analyzing a software item to detect the differences between existing and required conditions (that is, bugs) and to evaluate the features of the software item.

- **Reliability** is the ability of a system or component to perform its required functions under stated conditions for a specified period of time.
Validation vs. verification

- **Validation**: process of evaluating a system or component during or at the end of the development process to determine whether it satisfies specified requirements
  - Validation: Are we building the right product?

- **Verification**: the process of evaluating a system or component to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase
  - Verification: Are we building the product right?

- IEEE definitions
**Mistake, Fault, Failure, etc.**

- **Mistake** – a human action that produces an incorrect result.
- **Fault** or **Defect** – an incorrect step, process, or data definition in a program.
- **Failure** – the inability of a system or component to perform its required function within the specified performance requirement.
- **Error** – the difference between a computed, observed, or measured value or condition and the true, specified, or theoretically correct value or condition.

Our job as testers is to write test cases to cause failures.

But, there is no way to guarantee that all faults have been detected.

**Work smart**: write as few test cases as possible to cause failures; don’t have more than one test cause the same failure.
**Testing Models**

- **Black box testing . . . You know the functionality**
  - Given that you know what it is supposed to do, you design tests that make it do what you think that it should do
  - From the outside, you are testing its functionality against the specs
  - For software this is testing the interface
    - What is input to the system?
    - What you can do from the outside to change the system?
    - What is output from the system?
  - Tests the functionality of the system by observing its external behavior
  - No knowledge of how it goes about meeting the goals

**Testing Models - 2**

- **White box testing . . . You know the code**
  - Given knowledge of the internal workings, you thoroughly test what is happening on the inside
  - Close examination of procedural level of detail
  - Logical paths through code are tested
    - Conditionals
    - Loops
    - Branches
  - Status is examined in terms of expected values
  - Impossible to thoroughly exercise all paths
    - Exhaustive testing grows without bound
  - Can be practical if a limited number of “important” paths are evaluated
  - Can be practical to examine and test important data structures
**Types of Testing**

- **Unit Testing**
  - Testing of individual hardware or software units or groups of related units
  - Done by programmer(s)
  - Generally all white box
  - Automation desirable for repeatability

- **Integration Testing**
  - Testing in which software components, hardware components, or both are combined and tested to evaluate the interaction between them
  - Done by programmer as they integrate their code into code base
  - Generally white box, maybe some black box
  - Automation desirable for repeatability

- **Functional/System Testing**
  - Testing conducted on a complete, integrated system to evaluate the system compliance with its specified requirements
  - Stress testing, performance testing, usability testing
  - It is recommended that this be done by external test group
  - Mostly black box so that testing is not ‘corrupted’ by too much knowledge
  - Test automation desirable

- **Acceptance Testing**
  - Formal testing conducted to determine whether or not a system satisfies its acceptance criteria (the criteria the system must satisfy to be accepted by a customer) and to enable the customer to determine whether or not to accept the system
  - Generally done by customer/customer representative in their environment through the GUI . . .
  - Definitely black box

- **Regression Testing**
  - Regression testing is selective retesting of a system or component to verify that modifications have not caused unintended effects and that the system or component still complies with its specified requirements
  - Smoke test group of test cases that establish that the system is stable and all major functionality is present and works under “normal” conditions

**Fill in chart**

<table>
<thead>
<tr>
<th>Testing Type</th>
<th>Specification [what you look at to write tests]</th>
<th>Black or white? Valid. or verif?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Integration</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Functional and System</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Acceptance</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Beta</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Regression</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>