LIMKOKWING UNIVERSITY OF CREATIVE TECHNOLOGY

BOTSWANA

DCOMP343 • Software Testing and Reliability

# CHAPTER 1

# Overview of Software Testing and Quality Assurance

# Objectives

- Type of Software Qualities
- The Objectives of Software Testing
- Testing terminologies

# Topics covered

- Software Qualities
- Dependability Properties
- Validation and Verification
- Objectives of software testing
- Testing Terminologies
- Classification of testing techniques
- Testing Tools

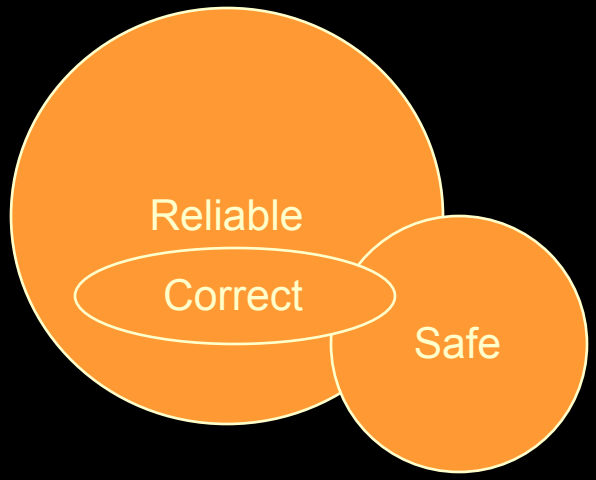
# Software Qualities

- Internal
  - Software product qualities primarily affect the software development organization.
    - Maintainability, reusability, modularity, and etc.
- External
  - Software product qualities can be directly visible by customer or client.
    - Dependability – Does the software do what is intended to do.
      - Eg: Correctness, reliability, correctness, reliability, robustness, safety and etc.
    - Usefulness.
      - Eg:User-friendliness, usability

# Dependability Properties

- Correctness
  - A programmer or system is correct if it is consistent with its specification.
- Reliability
  - A statistical approximation to correctness. Roughly speaking, reliability is a measure of the like hood of correct function for ‘some; unit of behaviour.
- Safety is concerned with preventing certain undesirable behaviours called hazards.

# Dependability Properties



# Validation and Verification

- Validation
  - Are we building the right product?
- Verification
  - Are we building the product right?

# What is Software  Testing

- Process of exercising a computer program with predetermined inputs and comparing the actual outputs with he expected outputs.

# Verification Techniques besides testing

LIMKOKWING UNIVERSITY OF CREATIVE TECHNOLOGY BOTSWANA

- Testing depends on having the system to "execute" (Dynamic).
- Inspection methods
- Static analysis methods

# What is the objective of software testing

- Phase I : Debugging
- Phase II : Show software works
- Phase III : Show software doesn't work
- Phase IV : Test for risk reduction
- Phase V : a state of mind

# Testing Terminologies

- Test cases – The collection of inputs, expected results, environment and procedural requirement for a single test.
- Test suite/test pool – a collection of test cases necessary to “adequately” test a product.
- Test plan – a document describing the scope , approach, resources and schedule testing activity.

# Fundamental Testing Questions

- How do we plan software testing?
- How do we perform testing (Test criteria)?
- How do we determine the outputs are correct? (Test oracle)
- When do we need to perform software testing?
- When do we need to perform software testing?
- When should be stop software testing ?

( Test Adequacy)

# *Classification of Testing Techniques*

- Statistical Testing
  - Used to test a program's performance and reliability.
- Defect Testing
  - Used to find areas of nonconformance to the system specification.

# *Classification of Testing Techniques*

- Functional Testing (Black box testing)
  - Test cases are generated based on software specification.
  - Views program as a black box.

- Structure Testing (White box Testing)
  - Test cases are generated based on internal structure of the software.
  - Views program as a white box.

# Classification of testing techniques

- Unit testing
  - Testing of individual component, usually requires of using test drives.
- Subsystem Testing (Integration testing)
  - Testing of interfaces between integrated components.
- System Testing
  - Testing the complete system. Validating functional as well as non-functional requirement.
- Acceptance testing (Alpha , Beta testing)

# Testing Tools

- Reviews and inspection
  - Complexity analysis
  - Code comprehension
  - Syntax and semantic analysis
- Test planning
  - Template for test plan documentation
  - Test schedule and staffing estimates.
  - Complexity and analyzer.

# Testing Tools

- Test design and development
  - Test data generator.
  - Requirement-based test design tool
  - Capture / playback
  - Coverage analysis
- Test execution and evaluation
  - Capture / playback
  - Coverage analysis
  - Memory testing
  - Simulators and performance
  - Test case management.

# Testing Tools

- Test support
  - Problem management
  - Configuration management

# Summary

- Software testing is important, expensive , difficult and stimulating.
- Restricting early testing usually increase cost.
- Design the product and process for test.
- Reality – one of most widely – used , practical faults reveling technique.
- Yet an area showing the grate gap theory and practice.

# *Quiz*

- In the year 2000 bug example , did Dave do anything wrong?
- True or False: It's important what term your company or team calls a problem in its software.
- What's wrong with just testing that a program works as expected.
- How much more does it cost to fix a bug found after the product is released than it does from the very start of the project?