

Software Testing

- Dynamic analysis of a software system
- Main components
 - Input
 - System Execution
 - Assessment

Test Input

- Test case generation
 - Granularity
 - Quantity
 - Characteristics

Test Execution

- Test case selection
- Instrumentation
- Scaffolding

Assessment

- Develop hypotheses
- Determine whether observed behavior is (in)consistent with hypotheses

Goodenough & Gerhart

- Exhaustive testing is infeasible
- Must sample input space
- How should that sampling be done?
- Testing criterion
 - Expresses the properties of a program that must be exercised to constitute a thorough test
 - i.e., one whose successful execution implies no errors in a tested program

Goodenough & Gerhart ['75] (cont.)

- Reliability
 - Criterion always produces consistent test results
 - All test sets that satisfy the criterion behave the same
- Validity
 - Test sets that satisfy the criterion always produces meaningful results
 - For every error in a program, there exists a test set satisfying the criterion that is capable of revealing the error
- There is no computable criterion that satisfies the above requirements

Test Coverage & Adequacy

- Test criterion used as objective measurement of test quality
 - Drives test data selection
 - Measures degree of completeness

Some Criteria Types

- Program-based
- Specification-based
- Random testing
- Statistical testing

Classification by Underlying Testing Approach

- Structural testing
 - Coverage of a particular set of elements in the structure of the program
- Fault-based testing
 - Measurement of the test set's fault detecting ability
- Profile-driven
 - Modeling of expected usage patterns

Program-based Structural Testing

- Control-flow based adequacy criteria
 - Statement coverage
 - Branch coverage
 - Path coverage
 - Multiple condition coverage
 - All possible combinations of truth values of predicates

Structural Testing (cont.)

- Data-flow based adequacy criteria
 - All definitions criterion
 - Each definition to some reachable use
 - All uses criterion
 - Some definition to each reachable use
 - All def-use criterion
 - Each definition to each reachable use

Structural Testing (cont.)

- All DU-paths criterion
- A set P of execution paths satisfies the all-DU paths criterion iff
 - for all definitions of a variable x and all paths q through which that definition reaches a use of x,
 - there is at least one path p in P such that q is a subpath of p and q is cycle-free

Fault-based Testing

- Program mutation testing
- Error seeding
 - Introduce artificial faults to estimate the actual number of faults
 - Distinguish between original and mutants
- Competent programmer assumption
 - Mutants are close to the program
- Coupling effect assumption
 - Simple and complex errors are coupled

Statistical Testing

- Develop operational profile capturing expected usage patterns
- Measure time between failures (e.g., MTBF)
 - Idea: for an improving system MTBF should keep increasing
- Can be used as stopping condition/release criteria

Subsumption

- Criteria C1 subsumes criteria C2, iff
 - For all programs p tested with specification s and all test sets t
 - t is C1 adequate implies that t is C2 adequate
- Examples
 - Path coverage subsumes branch coverage
 - Path coverage subsumes statement coverage