**System Under Test (SUT)**

*p*

**Test Goal**

*e.g line coverage*

**Test Requirements** **Test Cases (One test suite)**

Redundancy on suite

TR1: Go to line 1

TR 2

…

TR n

TC 1

TC2

…

TC n

Satisfies many to many relations

Note: This course focus on imperative programming.

A; B 🡪 B will execute after A in sequence. Implication: If you hit B, you must have gone through A, can write a requirement that tackles A and B

Test Goal

Node Coverage

Edge Coverage

Line Coverage

Data Flow Coverage

Logic Coverage

Control Flow Coverage

Black-Box

Glass Box Coverage

Graph Criterion

Control Flow Coverage and Data Flow Coverage are “Paths”

Logic Coverage is not a “Path”, but “Constraints on Variable”

Data Flow Coverage

CC

PC

Logic Coverage

Control Flow Coverage

PPC

ECn

NC

Test Goal

Glass Box Testing

For ECn, when n=1, edge coverage. When n =2, edge-pair coverage.

EC ⊆ EC2 ⊆ EC3

**Control Flow:** Test requirement is a path ***p*** = n1, n2, … , nk , k ≥ 1. Test case is an input vector that covers a path ***p***.

Logic Flow

**Distributive Combination: Focus: Non-distributive Combination:**

Exponential in number of clauses Linear in number of clauses

A|B

When reach C, don’t care about A|B because when we reach there, means already go through A|B

C|D

A|B

E|F

AC, BD

AC, AD, BC, BD

E|F

C|D

**Blackbox Testing**

Var: type

**API**

Var: type

**SUT**

**V**

Use case coverage

3-wise testing

1-wise testing

pair-wise testing

n-wise testing

Test Goals

Black-box Testing

combinatorial testing

Boundary Value Testing

Random Testing

Black-box testing

Glass-box testing

**Integration Test**

**Unit Test**

**System Test**

**Test Oracle:** the black box in which we convert observations into test verdicts. Traditionally, it is through humans, nowadays is automation.

Refer to section 3.3 for J-Unit.

**Verdicts 🡪** pass, fail, warning, i.e. “anomalous”, timeout, exception

Subclasses

**3 phases of testing:**

**Test Automation**

SUT

Goals

Reactive Systems

Procedural Systems

Timed automata

(hard real time)

Linear temporal logic (Soft real time)

JML

JUnit

Modelling Languages

Test Requirement

e.g. JML

Junit

…

Other verdict

Fail verdict

Pass verdict

Test Oracle

Observations

Script

Test

Data

Test

Execution

Test Case Construction