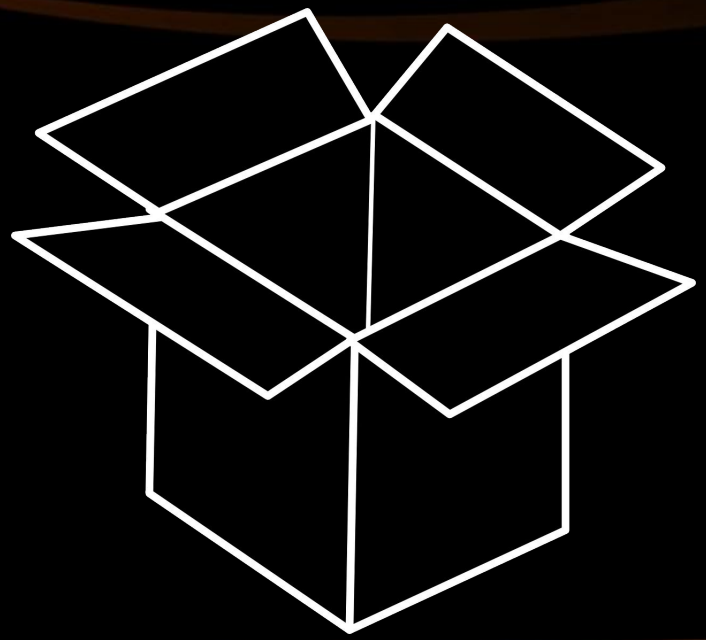
Testing Techniques

The Theory of testing

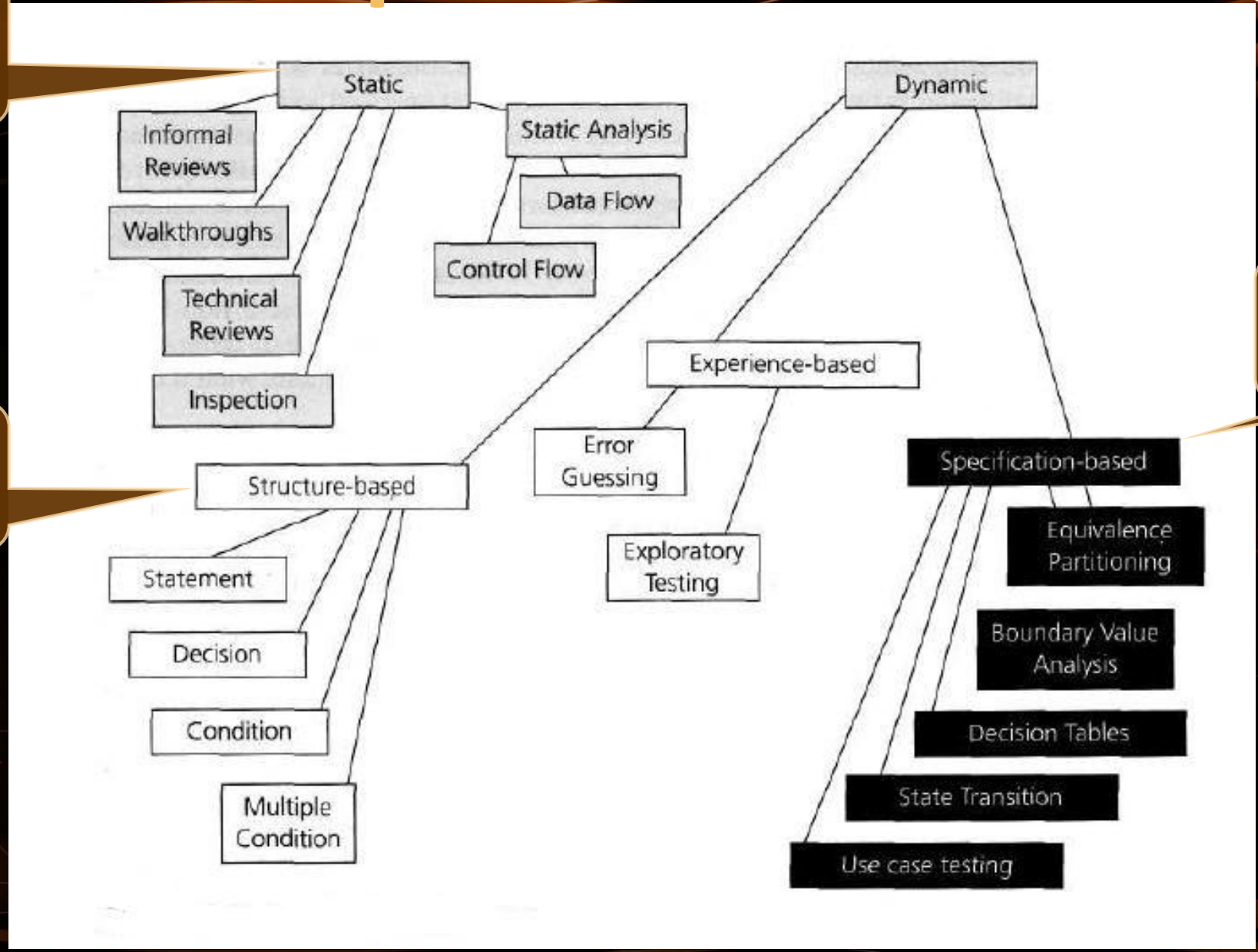
SoftUni Foundation



Yuksel Ahmedov
QA Trainer
www.qualityassuranceteam.com
Software University
http://softuni.bg

Table of Contents

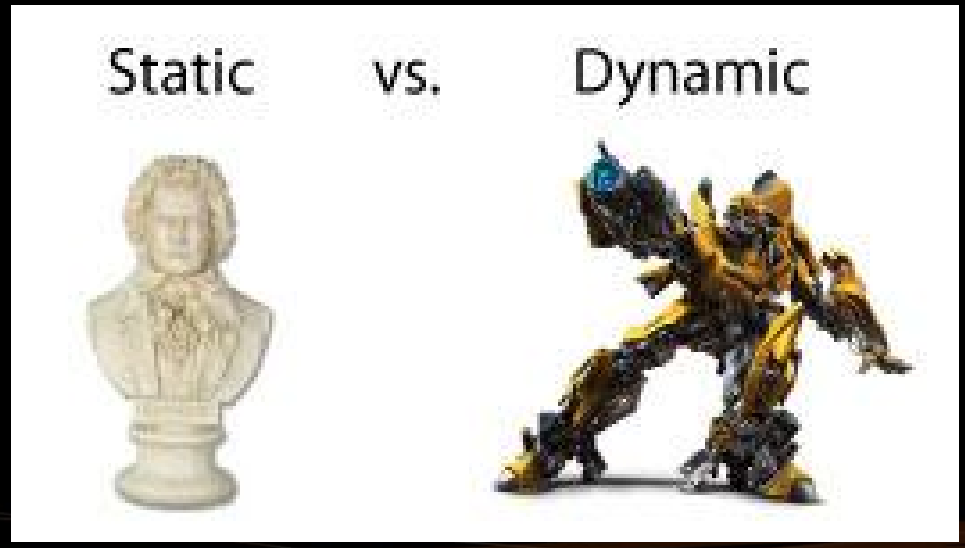
Graph of contents



Static vs. Dynamic Techniques

SoftUni Foundation

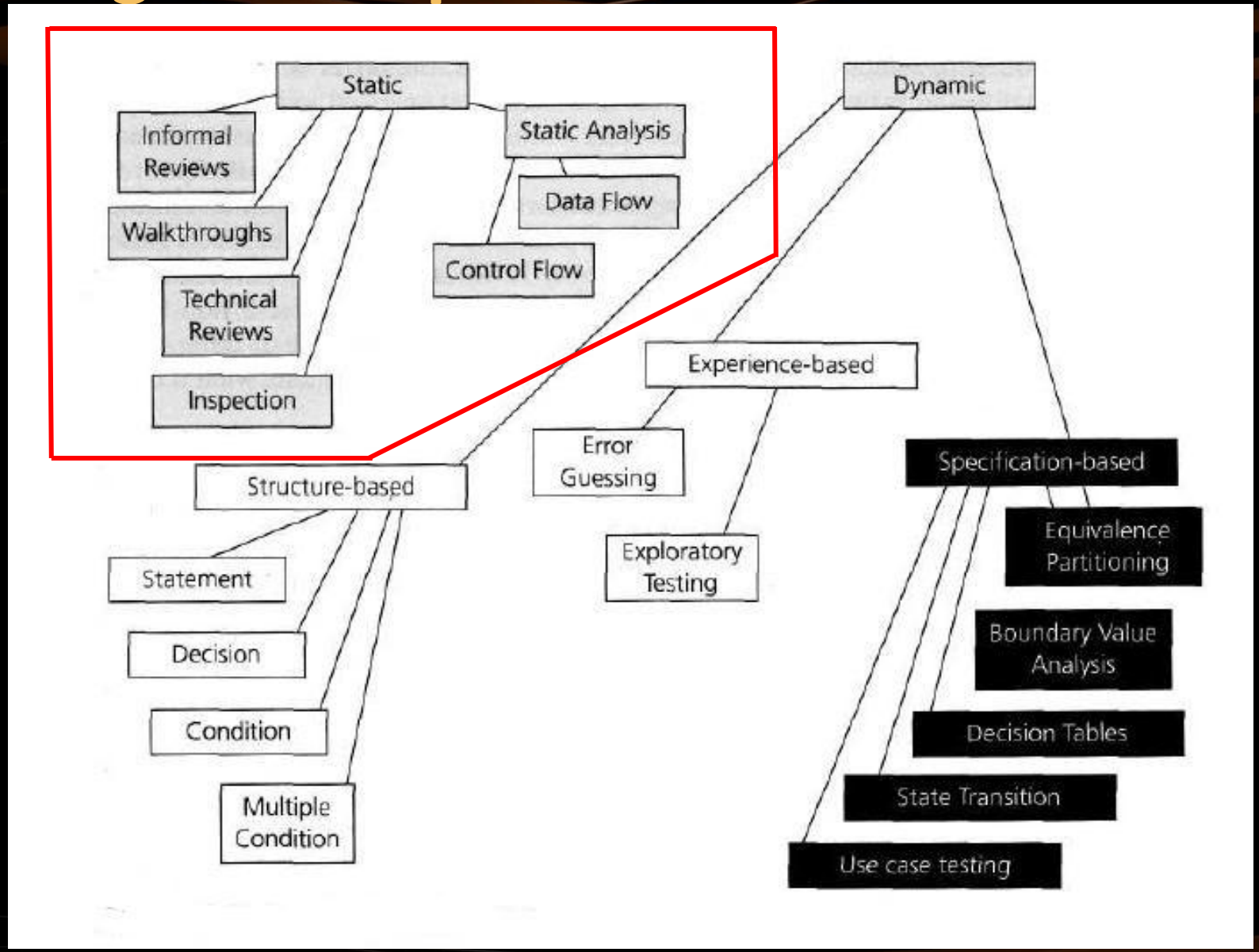
- Static tests
  - Aim to find problems in the design or concept of the prod.
  - Code is NOT being executed.
- Dynamic tests
  - Aim to find problems in an instance of the product.
  - Code execution is mandatory.



SoftUni Foundation

Static Testing Techniques

Static Testing Techniques

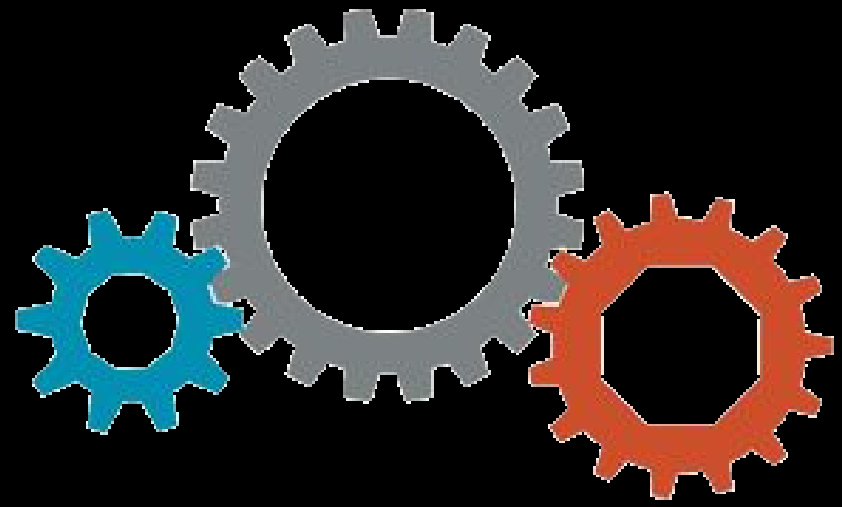


Static Testing Techniques

SoftUni Foundation

- Reviews
  - A human investigator is the primary defect finder

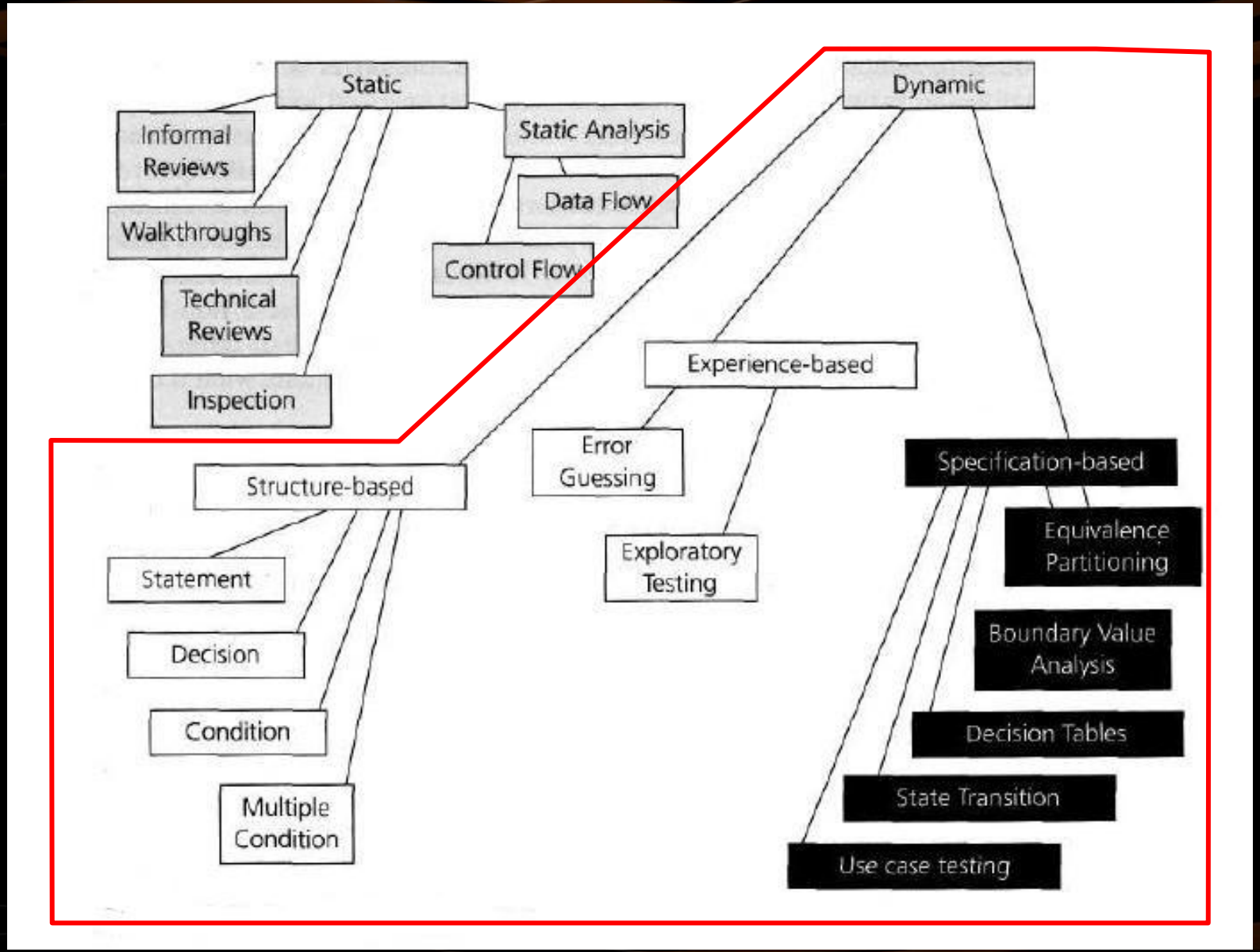
- Static Analysis
  - Testing is done by code review tools



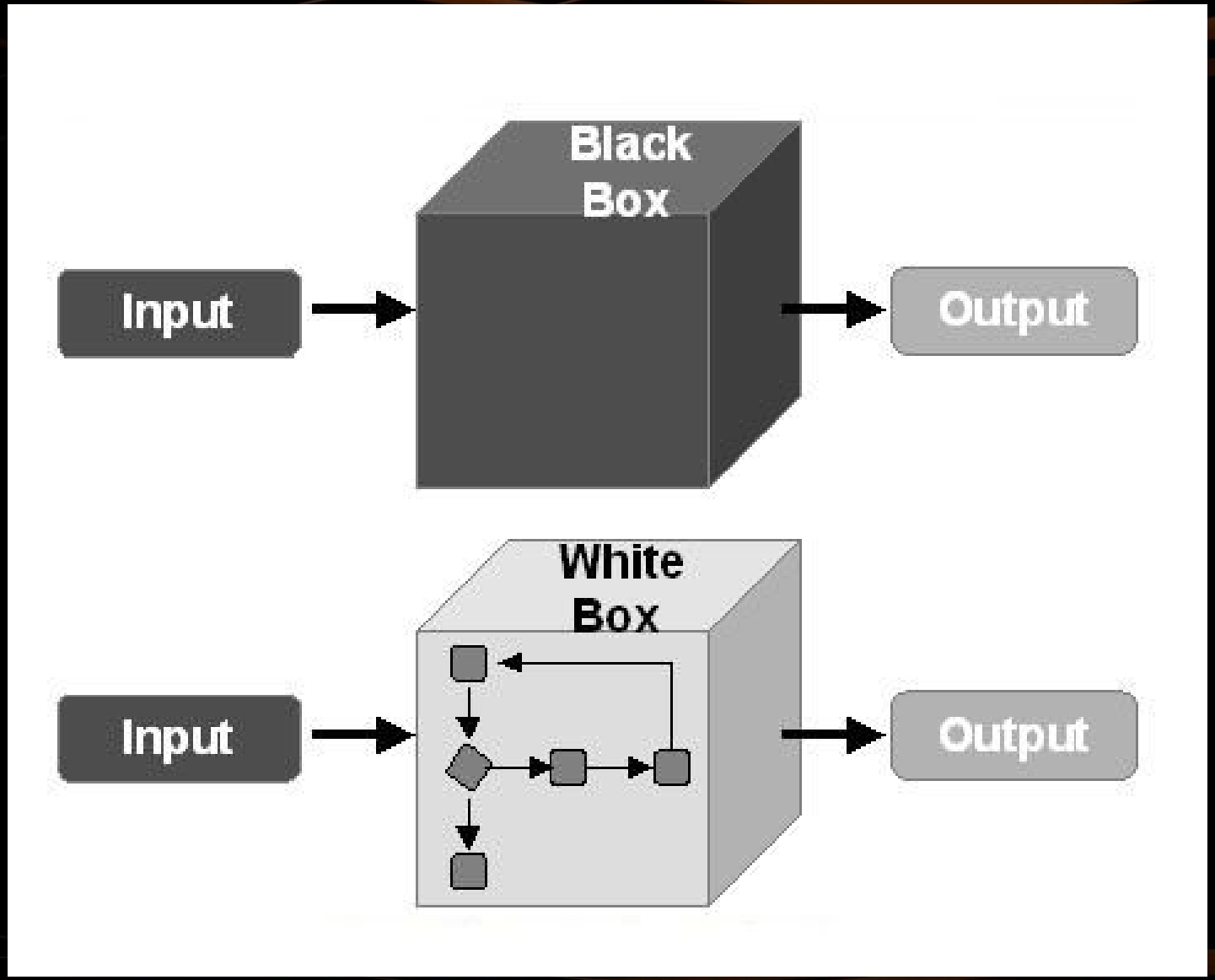
SoftUni Foundation

Dynamic Testing Techniques

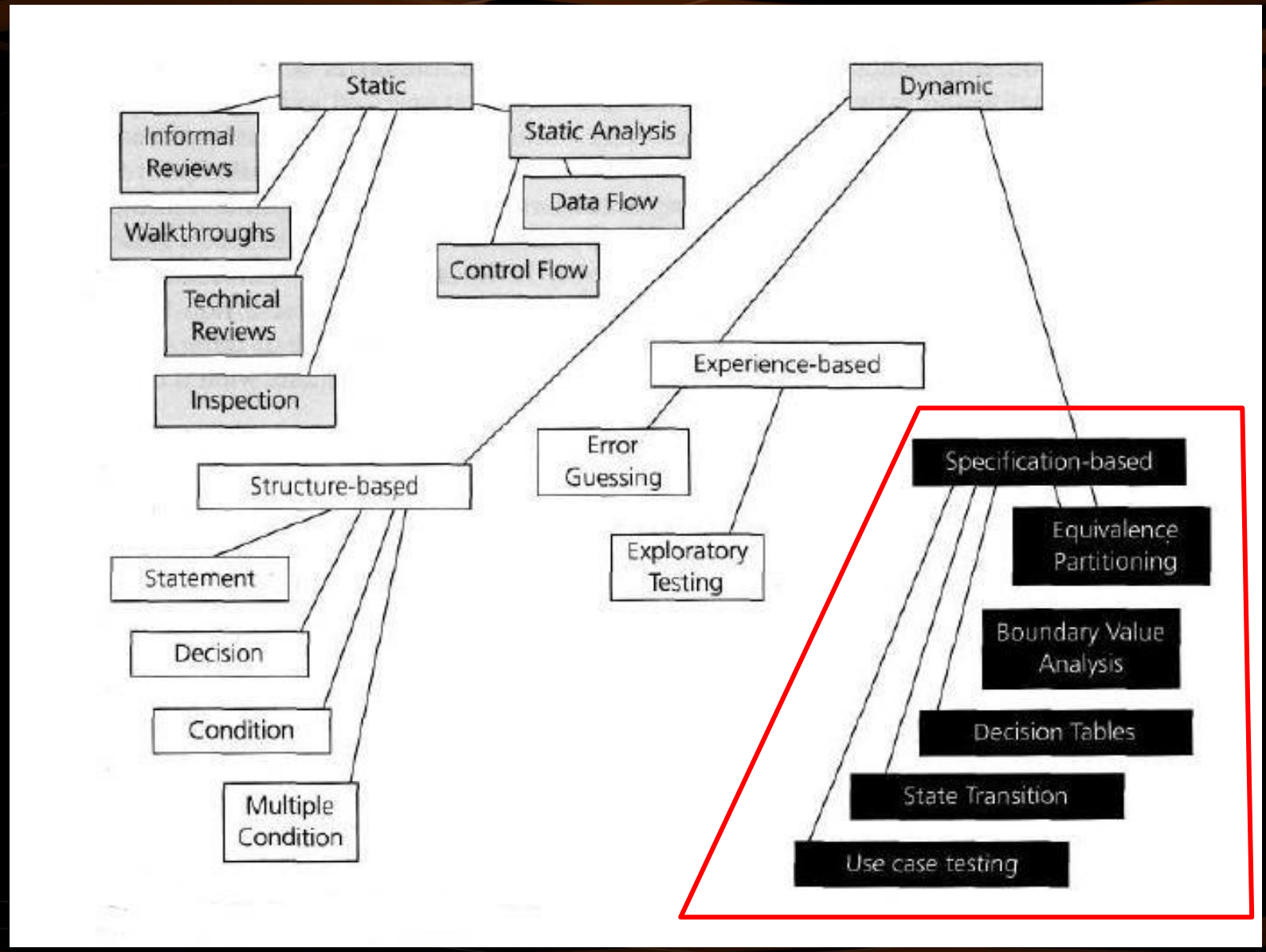
Dynamic Testing Techniques



White-Box vs. Black-Box Testing



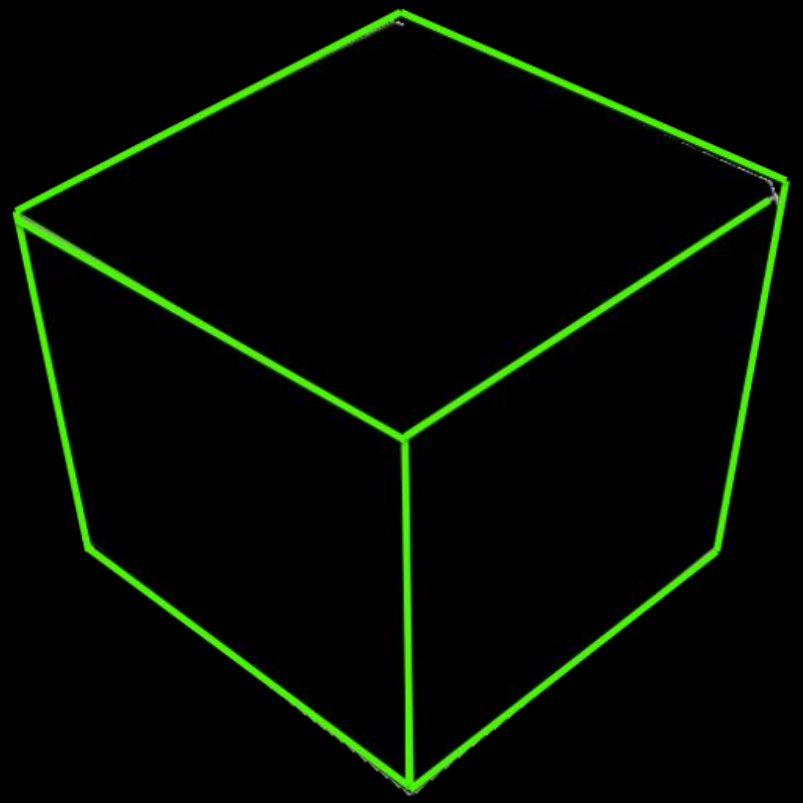
Black Box Testing Techniques



Black-Box Testing Techniques

SoftUni Foundation

- Some of the most used Black Box testing techniques are:
  - Equivalence Partitioning
  - Boundary Value Analysis
  - Decision Tables
  - State Transition Tables
  - Use-Case Testing



SoftUni Foundation

Equivalence Partitioning

What is Equivalence Partitioning?

SoftUni Foundation

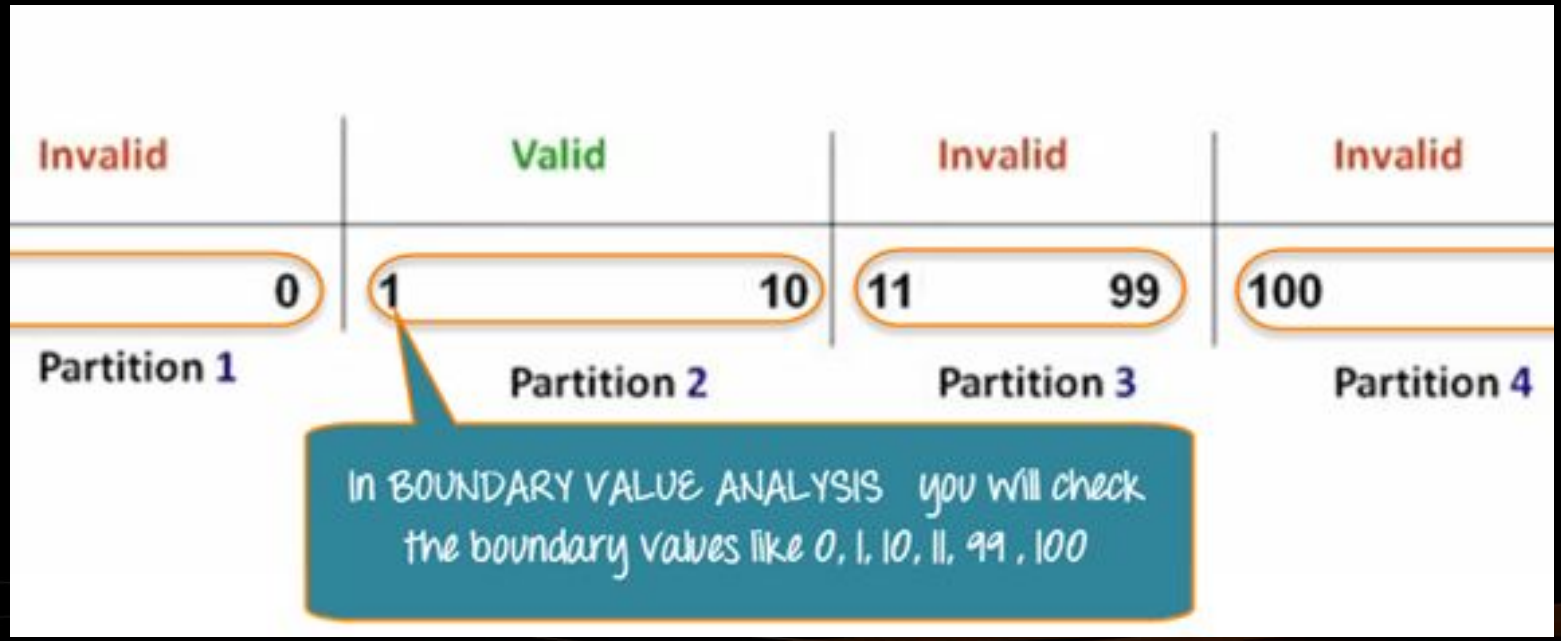
- Equivalence partitioning:

> Equivalence partitioning is a software testing technique that divides the input data of a software unit into partitions of equivalent data

- Test cases are designed to cover each partition at least once

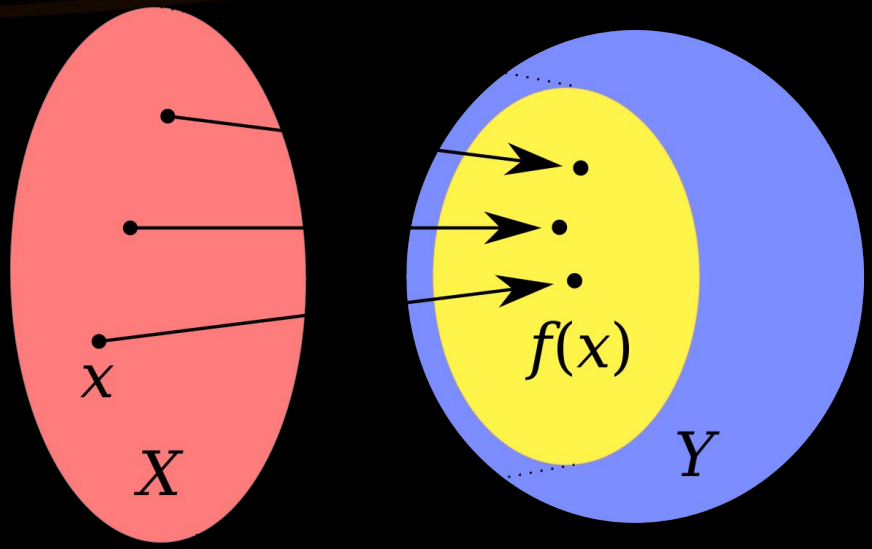
Why Equivalence Partitioning?

- Equivalence partitioning aims to reduce the total number of test cases to a feasible count
  - Exhaustive testing of all possible input / output values is usually impossible



Input / Output Domains

- A domain can be formed of:
  - Input field
  - Output field
  - Test precondition or post condition
  - Configuration
  - Etc. - anything we're interested in testing



Splitting Domains Into Partitions

SoftUni Foundation

- The operation of equivalence partitioning is performed by splitting a set (domain) into two or more subsets
  - All the members of each subset share some trait in common
  - This trait is not shared with the members of the other subsets

SoftUni Foundation

Boundary Value Analysis

What Is Boundary Value Analysis ?

- Conceptually, boundary value analysis is about testing the edges of equivalence classes
- When the members of an equivalence class are ordered Boundary value analysis can be seen as an extension of equivalence partitioning

Why Should This Work?

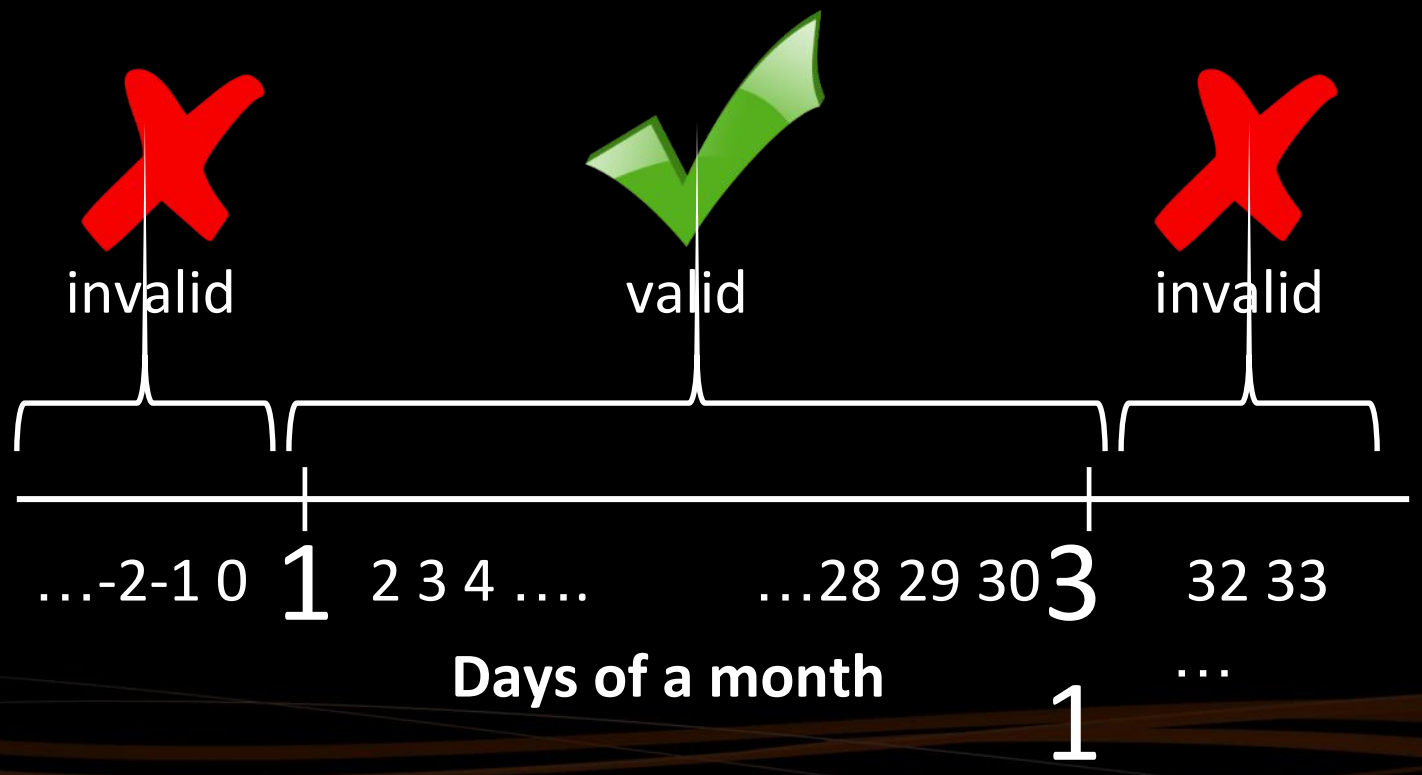
SoftUni Foundation

- Many bugs occur due to careless usage of indexes, operators
- for example < instead of <= etc.

- If a software can operate on the edge of its capabilities, it will almost certainly operate well under normal conditions

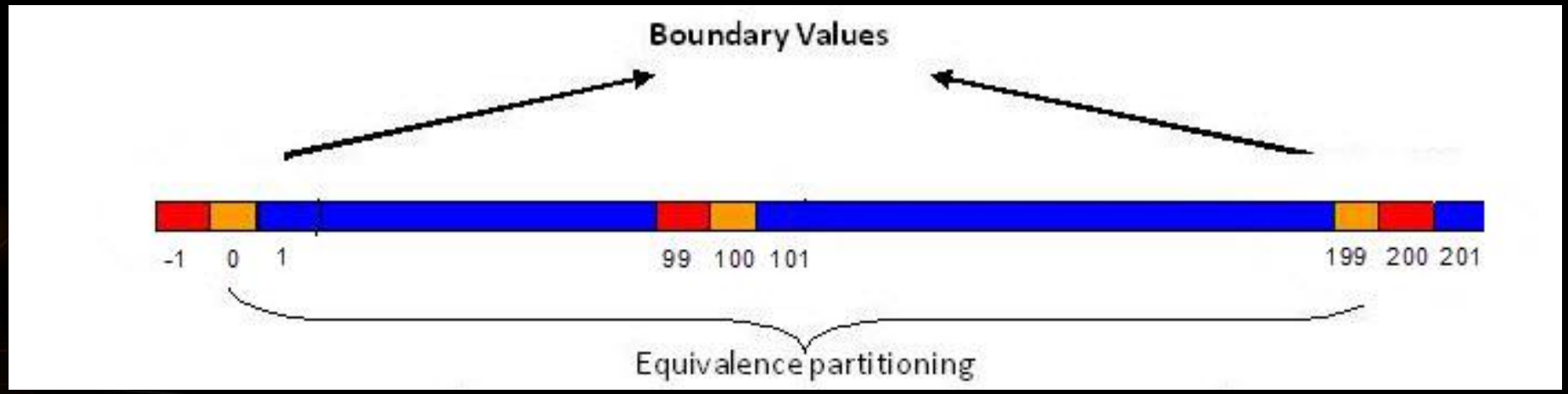
What is a Boundary Value?

- The point where the expected behavior of the system changes
- The values could be either input or output ranges of a software component



Sub-Boundary Conditions

- Some boundaries, that are internal to the software aren't necessarily apparent to an end user but still need to be checked by the software tester
- Every data structure has it's limitations



SoftUni Foundation

Decision Table Testing

Decision Table Testing

SoftUni Foundation

- Decision tables testing connects combinations of conditions with the actions that should occur
- What kind of bugs are we looking for?
  - Under some combination of conditions, a wrong action might occur

Example: Decision Table Testing

SoftUni Foundation

- Credit card example:
  - If you are a new customer and you want to open a credit card account then there are three conditions first you will get a 15% discount on all your purchases today, second if you are an existing customer and you hold a loyalty card, you get a 10% discount and third if you have a coupon, you can get 20% off today (but it can't be used with the 'new customer' discount). Discount amounts are added, if applicable.

Example: Decision Table Testing

SoftUni Foundation

- Credit card example:

Conditions	Rule 1	Rule 2	Rule 3	Rule 4	Rule 5	Rule 6	Rule 7	Rule 8
New customer (15%)	T	T	T	T	F	F	F	F
Loyalty card (10%)	T	T	F	F	T	T	F	F
Coupon (20%)	T	F	T	F	T	F	T	F
Actions								
Discount (%)	X	X	20	15	30	10	20	0

SoftUni Foundation

State Transition Testing

What is State Transition Testing?

SoftUni Foundation

- A black-box test design technique in which test cases are designed to execute valid and invalid state transitions

State Diagram Testing

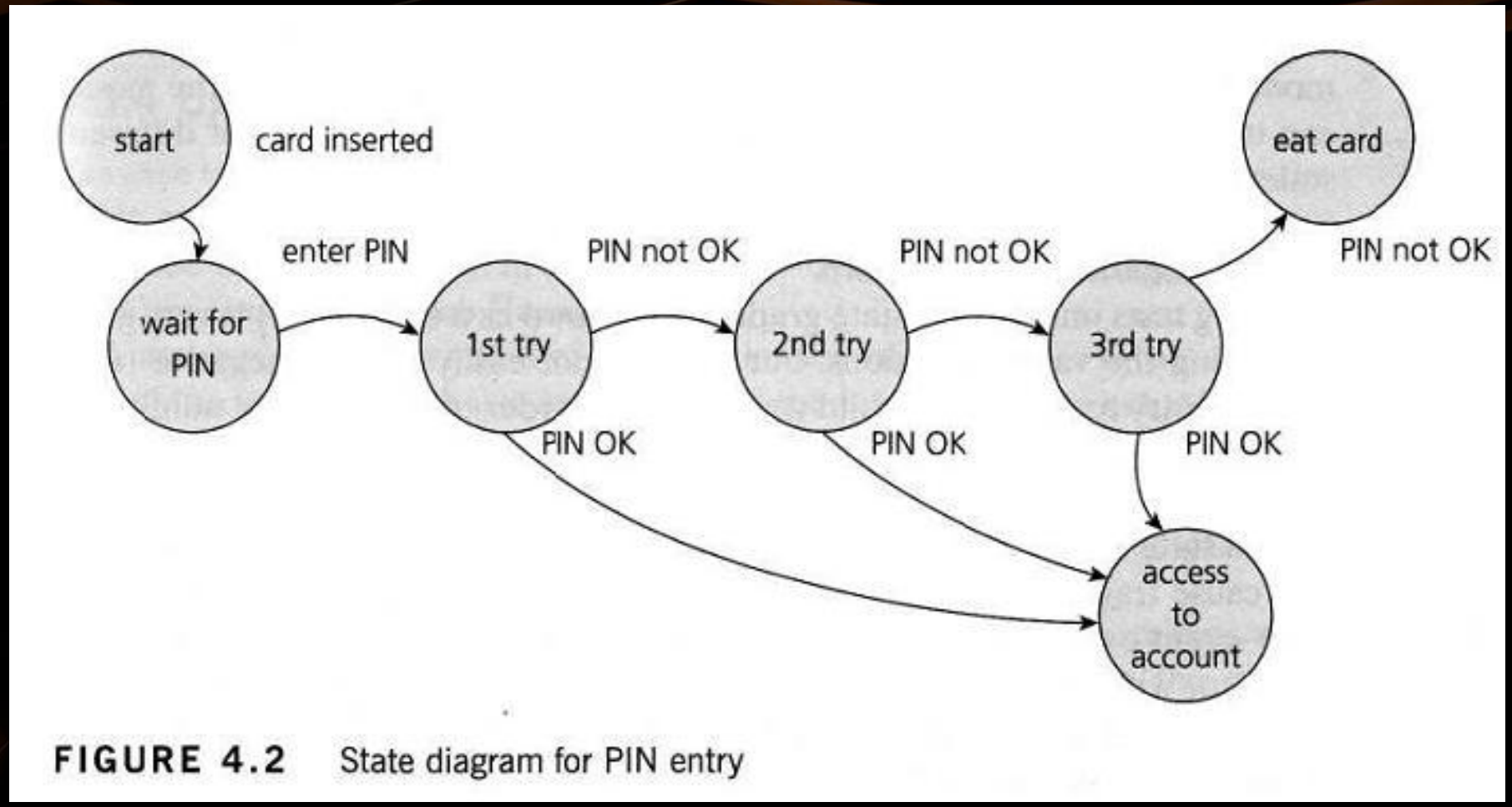


FIGURE 4.2 State diagram for PIN entry

State Transition Tables

SoftUni Foundation

- Each row in a state transition table has four fields:
  - Current state
  - Event/condition
  - Action
  - New state

STATE	EVENT	ACTION	NEXT STATE
Wait For Dollar	Bill Detected	Load Bill	Verify Dollar
Verify Dollar	Verification Failed	Reject Bill	Wait For Dollar
Verify Dollar	Verification Passed	Dispense Coins	Dispensing Coins
Dispensing Coins	Sufficient Funds Remain	Accept Another Dollar	Wait For Dollar
Dispensing Coins	Insufficient Funds Remain	Turn On Out Of Money Light	Out Of Money
Out Of Money	Money Refill	Accept Another Dollar	Wait For Dollar

What do we expect to find ?

- We're looking for situations where the wrong action or the wrong new state occurs in response to a particular event
- State transition tables force us to consider combinations of states with event/condition combinations that we might have forgotten

Use-Case Testing

SoftUni Foundation

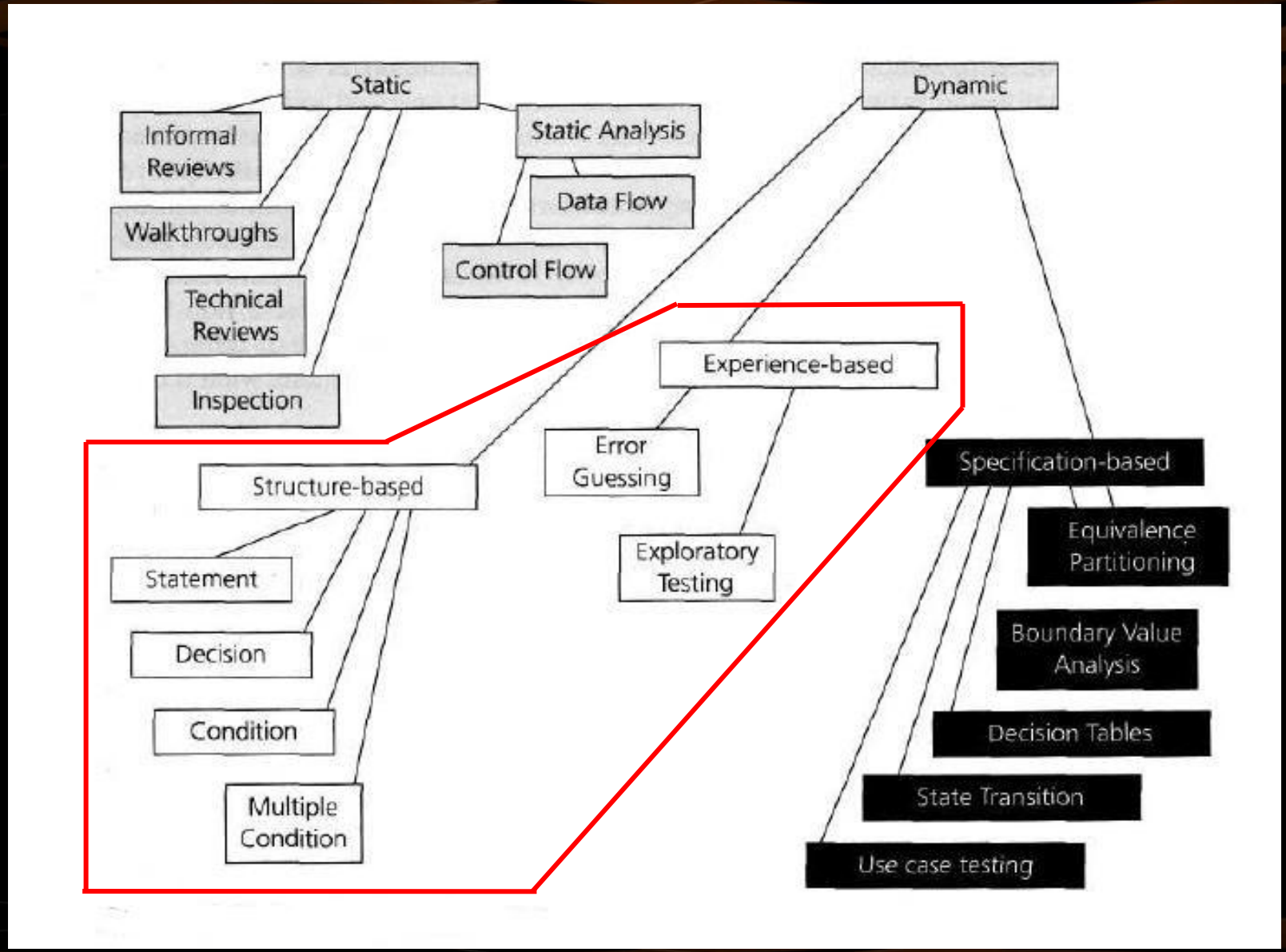
- A use case is a description of a particular use of the system by an actor (a user of the system).
- Each use case describes the interactions the actor has with the system in order to achieve a specific task (or, at least, produce something of value to the user).
- Actors are generally people but they may also be other systems.

Use-Case Testing

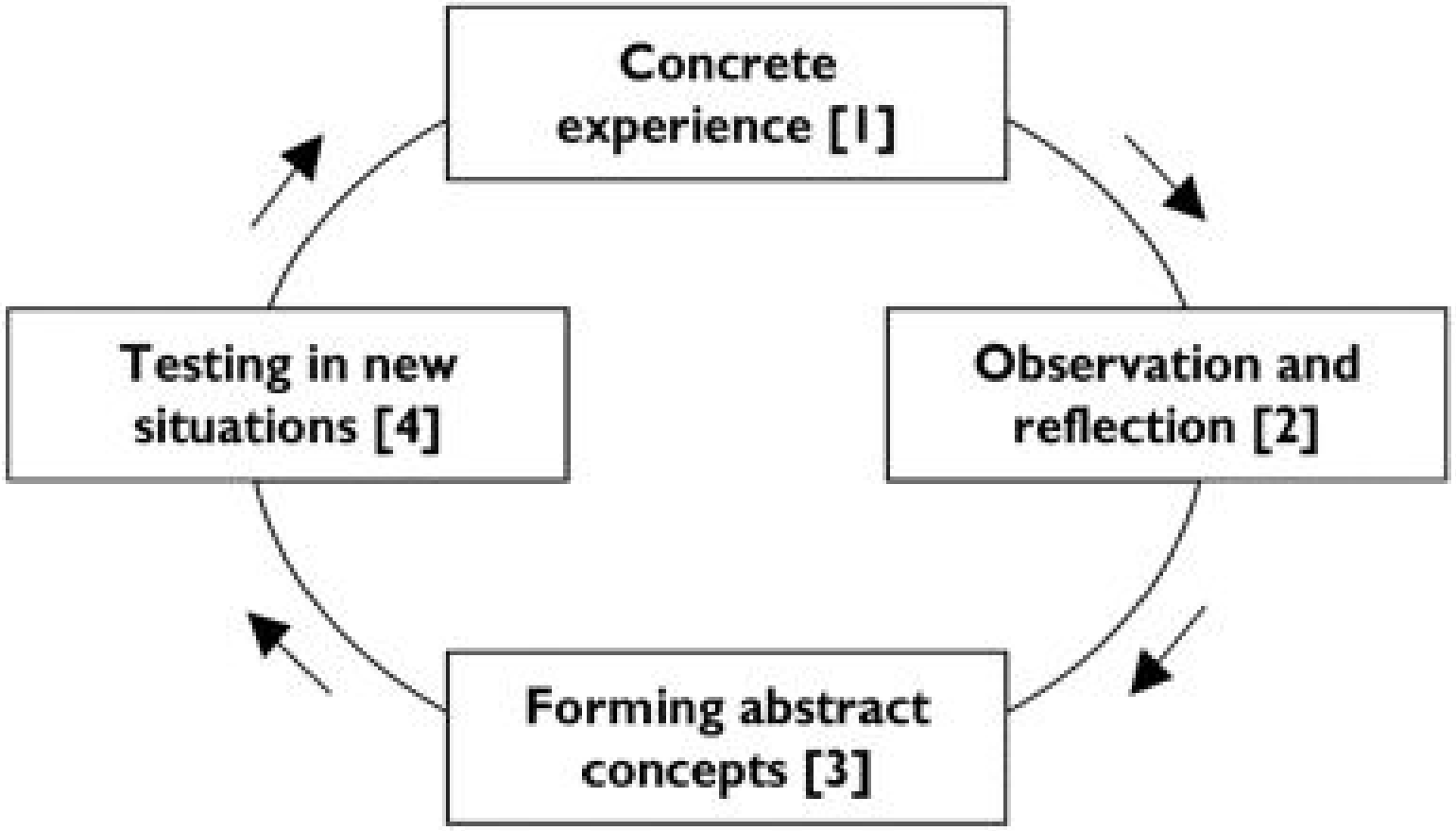
SoftUni Foundation

	Step	Description
Main Success Scenario **A: Actor** **S: System**	1	A: Inserts card
	2	S: Validates card and asks for PIN
	3	A: Enters PIN
	4	S: Validates PIN
	5	S: Allows access to account
Extensions	2a	Card not valid S: Display message and reject card
	4a	PIN not valid S: Display message and ask for re-try (twice)
	4b	PIN invalid 3 times S: Eat card and exit

White-Box Testing Techniques

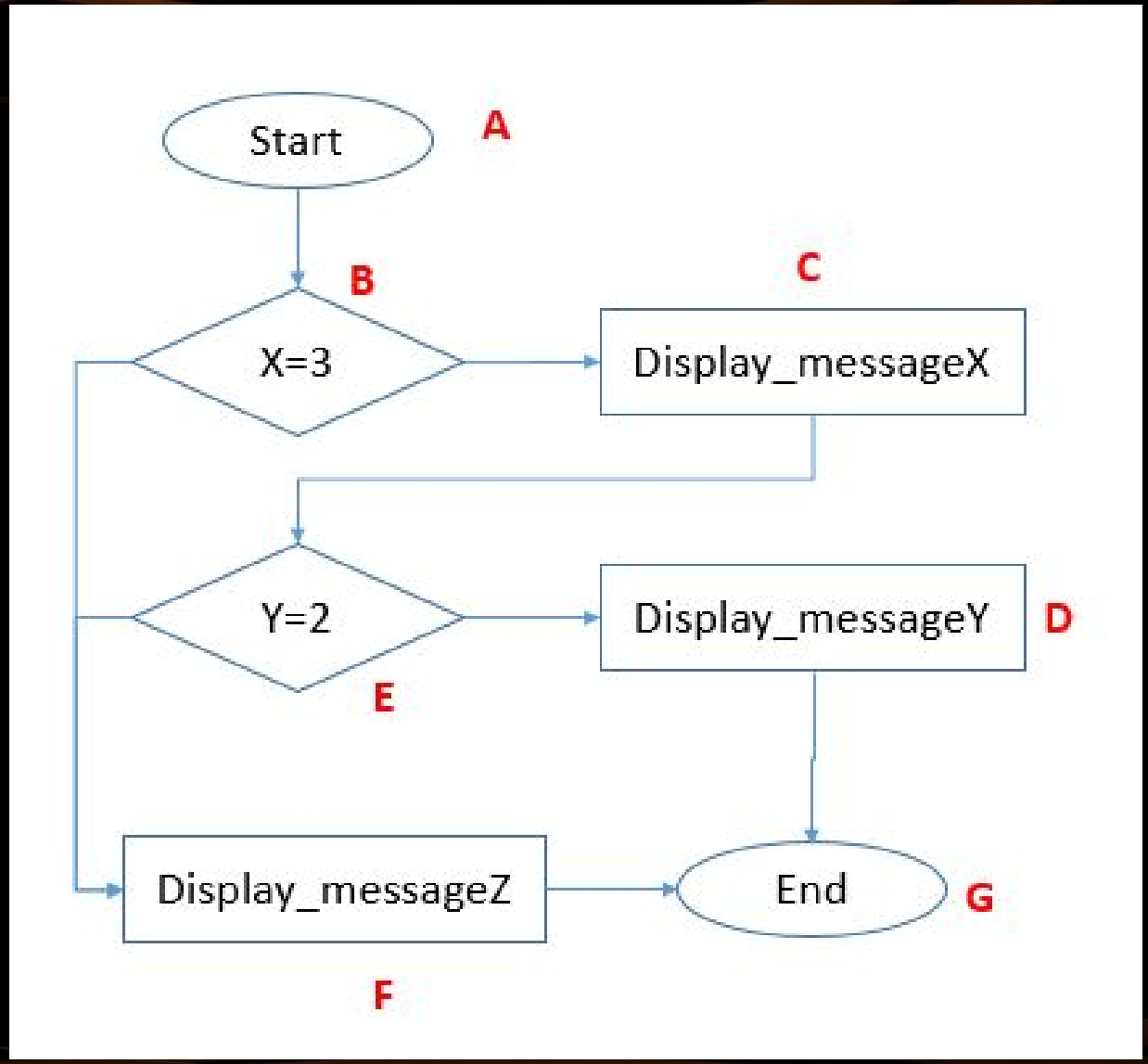


Experience Based White-Box Testing



Structure Based White-Box Testing Metrics

- Statement Coverage
- Branch/Decision Coverage
- Path Coverage



Useful Links

- Static testing tool (validator): https://validator.w3.org/
- Top 40 Static Analysis tools: http://www.softwaretestinghelp.com/tools/top-40-static-code-analysis-tools/
- Demo test environments:
  - https://lab.uk.qateam.eu/
  - https://box.uk.qateam.eu/04/epbva/

Testing Techniques

SoftUni Foundation

Questions?

License

SoftUni Foundation

- This course (slides, examples, demos, videos, homework, etc.) is licensed under the "Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International" license

Trainings @ Software University (SoftUni)

- Software University – High-Quality Education, Profession and Job for Software Developers
  - softuni.bg
- Software University Foundation
  - http://softuni.foundation/
- Software University @ Facebook
  - facebook.com/SoftwareUniversity
- Software University Forums
  - forum.softuni.bg

Software University

SoftUni Foundation

FORUM