Testing Design Techniques

Testing Design Techniques

- Categories of Test Design Techniques
- Specification based or Black Box Techniques
- Experience Based Techniques
- Identifying Test Conditions
- Designing TC
- Writing Good Test Case
- Choosing Test Techniques

Categories of Test Design Techniques

- What is black box testing?
- What is white box testing?
- Black Box Testing Techniques (in Details)

What is black box testing?

- **Black-box testing**: *Testing, either functional or non-functional, without reference to the internal structure of the component or system.*
  - testing without knowing the internal workings of the code
  - WHAT a system does, rather than HOW it does it
  - typically used at System Test phase, although can be useful throughout the test lifecycle
  - also known as specification based testing
  - applies for Functional and Non-Functional testing

What is White box testing?

- **white-box testing:** *Testing based on an analysis of the internal structure of the component or system.*
  - testing based upon the structure of the code
  - typically undertaken at Component and Component Integration Test phases by development teams
  - also known as structural or glass box testing or structure based testing

Black Box Techniques

- Based on requirements
- From the requirements, tests are created
- Specification Models can be used for systematic test case design

- **Techniques**
  - Equivalence Partitioning
  - Boundary Value Analysis
  - Decision Tables
  - Error Guessing

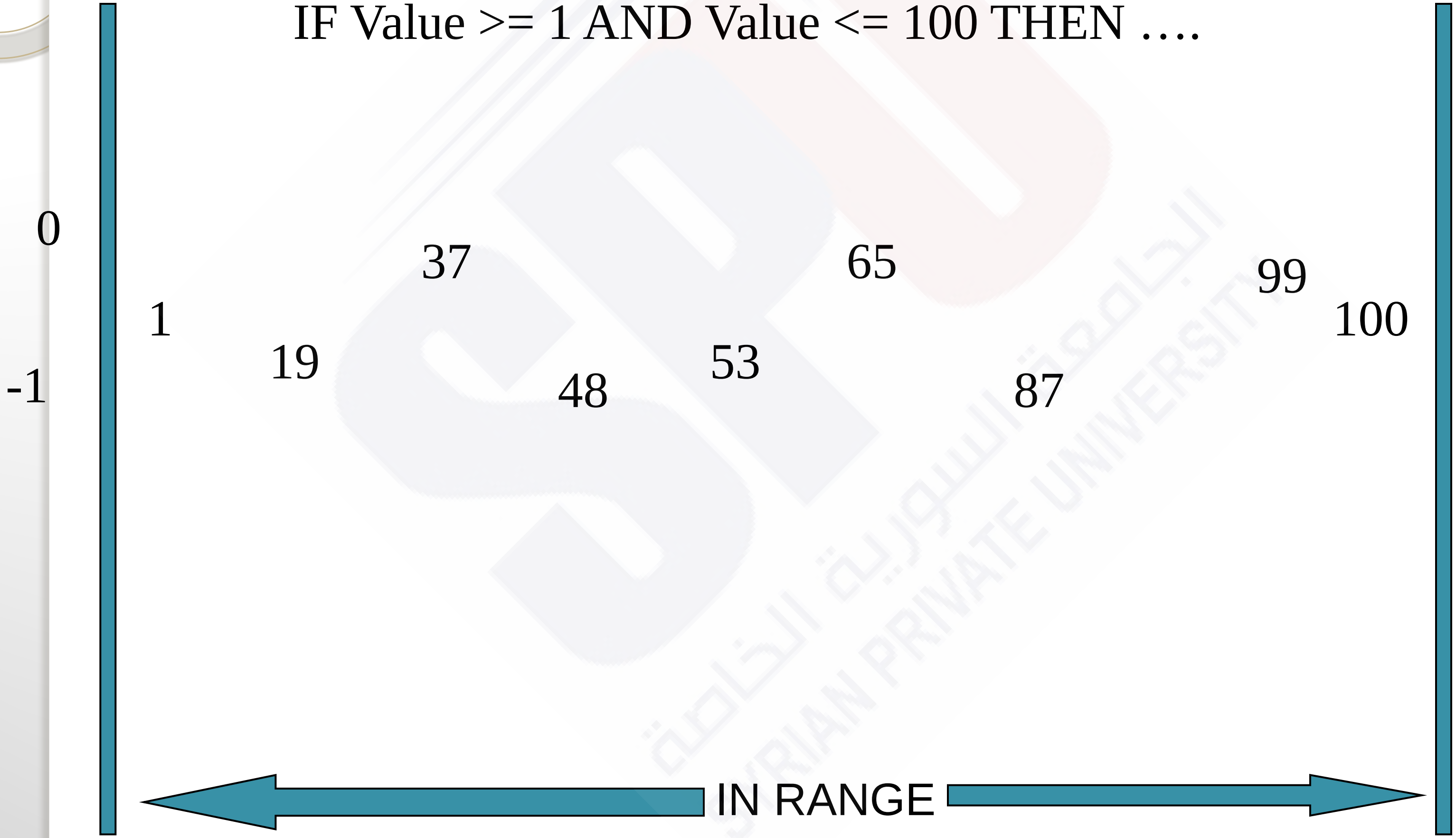
Black Box Techniques

- Equivalence Partitioning
  - aim is to treat groups of inputs as equivalent and to select one representative input to test them all
  - Best shown in the following example….
    - If we wanted to test the following IF statement:
    - ‘IF VALUE is between 1 and 100 (inclusive) (e.g. VALUE >=1 and VALUE <= 100) THEN …..’
    - We could put a range of numbers as shown in the next slide through test cases

Black Box Techniques

Equivalence Partitioning

IF Value >= 1 AND Value <= 100 THEN ….



Black Box Techniques

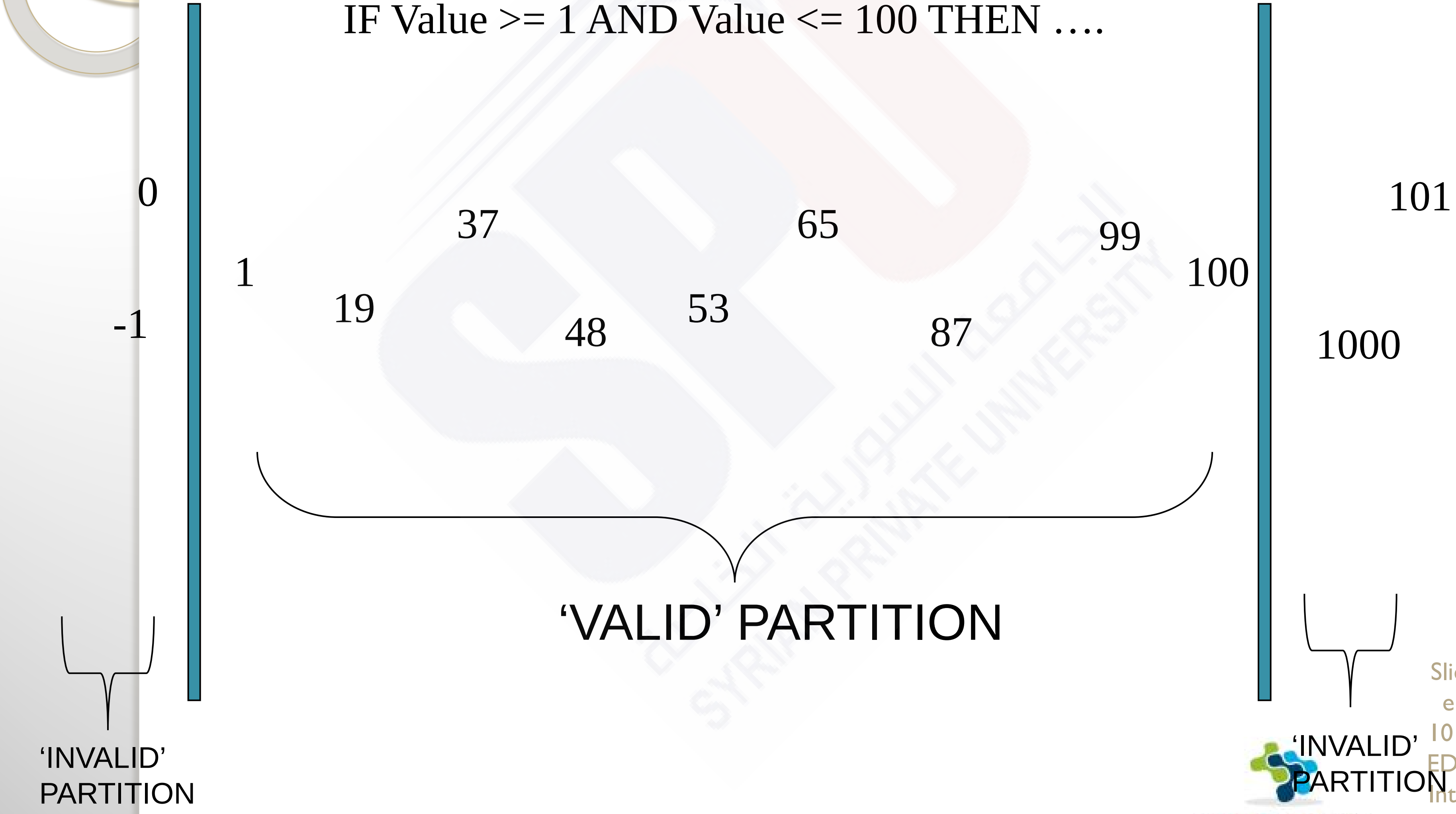
Equivalence Partitioning

- in EP we must identify Valid Equivalence partitions and Invalid Equivalence partitions where applicable (typically in range tests)
- the Valid partition is bounded by the values 1 and 100
- plus there are 2 Invalid partitions

Black Box Techniques

Equivalence Partitioning

IF Value >= 1 AND Value <= 100 THEN ….



0

-1

1

19

37

48

53

65

87

99

100

101

1000

‘VALID’ PARTITION

‘INVALID’ PARTITION

‘INVALID’ PARTITION

EgyTesters

Black Box Techniques

Equivalence Partitioning

- EP is reducing number of TCs while maintaining Coverage
- EP can be used for all Levels of Testing
- EP is used to achieve good input and output coverage, knowing exhaustive testing is often impossible
- It can be applied to human input, input via interfaces to a system, or interface parameters in integration testing

Black Box Techniques

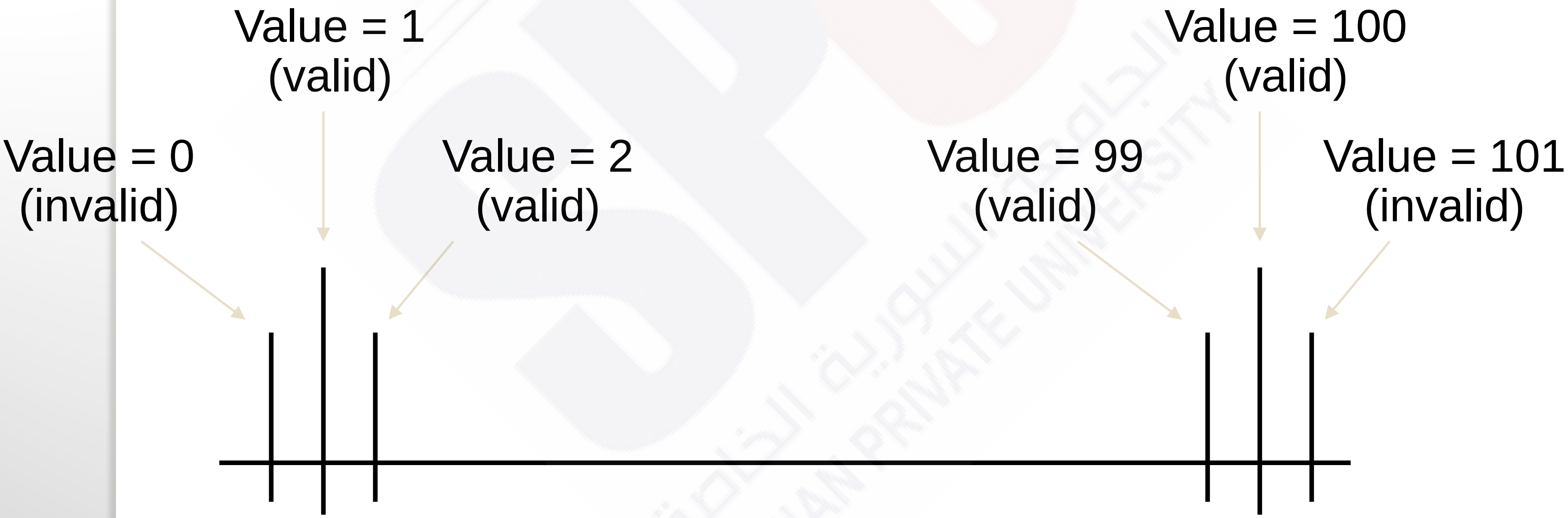
Boundary Value Analysis

- Boundary Value Analysis (BVA) uses the same analysis of partitions as EP and is usually used in conjunction with EP in test case design
- As with EP, it can be used for all Test levels
- BVA operates on the basis that experience shows us that errors are most likely to exist at the boundaries between partitions and in doing so incorporates a degree of negative testing into the test design
- BVA Test cases are designed to exercise the software on and at either side of boundary values

EgyTesters

Black Box Techniques

Boundary Value Analysis



- only applicable for numeric (and date) fields

Black Box Techniques

Decision Table Testing

	Test 1	Test 2	Test 3
> 55 yrs old	F	T	T
Smoker	F	T	F
Exercises 3 times a week +	T	F	T
History of Heart Attacks	F	T	F
Insure	Y	N	Y
Offer 10% Discount	N	N	Y
Offer 30% Discount	Y	N	N

What will be the out come of the following Scenarios?

Joe is a 22 year old non smoker who goes to the gym 4 times / week and has no history of heart attacks in his family

Kevin is 62 year old non smoker who swims twice a week and plays tennis. He has no history of heart attacks in his family

EgyTesters

Black Box Techniques

Decision Table Testing

- Very useful for complex scenarios
- Combining multiple combinations

- Real Example
  - Requirement was “ “
  - Decision Table as

Microsoft Office
Excel Worksheet

EgyTesters

Experience (Black box)

- Based on the knowledge of the tester
- Using past experienced use & intuition to "guess" where errors may occur

- **Techniques**
- Error Guessing
- Exploratory Testing

Experienced Based Techniques

Error Guessing

- Using experience to postulate errors
- Use Error Guessing to complement test design techniques
- Use as a "mopping up" approach to supplement systematic techniques
- Can be useful to identify special tests not easily captured by formal techniques, especially when applied after more formal approaches
- So don't use as a first choice technique!
- Structured approach to error guessing
  - Create a list of all possible errors
  - Then create tests to attack these errors
  - Remember these defect attack lists are built on experience, previous defects and from common knowledge as to why systems fail

Experienced Based Techniques

•Error Guessing

- Error Guessing tests may include
  - ‘Enter 00000 or 99999 in to a field’
  - Creating surnames with quotes in, such as O’Donnell
  - Nulls in mandatory fields
  - Reserved characters ($%& for web systems)

EgyTesters

Experienced Based Techniques

Exploratory Testing

- Exploratory testing is a concurrent process where
  - Test design, execution and logging happen simultaneously
  - Testing is often not recorded
  - Makes use of experience, heuristics and test patterns
- More structured than Error guessing

Identifying Test Conditions

- As simple as **What & How** to Test
  - **What** means the Scope , Item , Function , System
  - **How** means the Condition , Statement , State

Identifying Test Conditions

- Test Conditions should :
  - Test Conditions is based on analysis of Req Doc
  - Test Conditions are then cross referenced to one or more test cases for execution
  - Not all Test Conditions are as important as others so each Test Condition is assigned a risk (Priority)
  - Test Conditions should be linked back to their source documents from which they are derived.
    - This helps for two reasons:
      - Impact Analysis
      - Traceability

Designing TC

- ***Test Cases*** are the implementation of a *test case* design that helps the tester to detect defects in the application
- Test Cases judge if Condition(s) is met
- TC typically contains :
  - pre conditions
  - Input actions / values
  - Expected results (output, changes in state etc)
  - Post conditions
  - Cross referenced test conditions

Designing TC

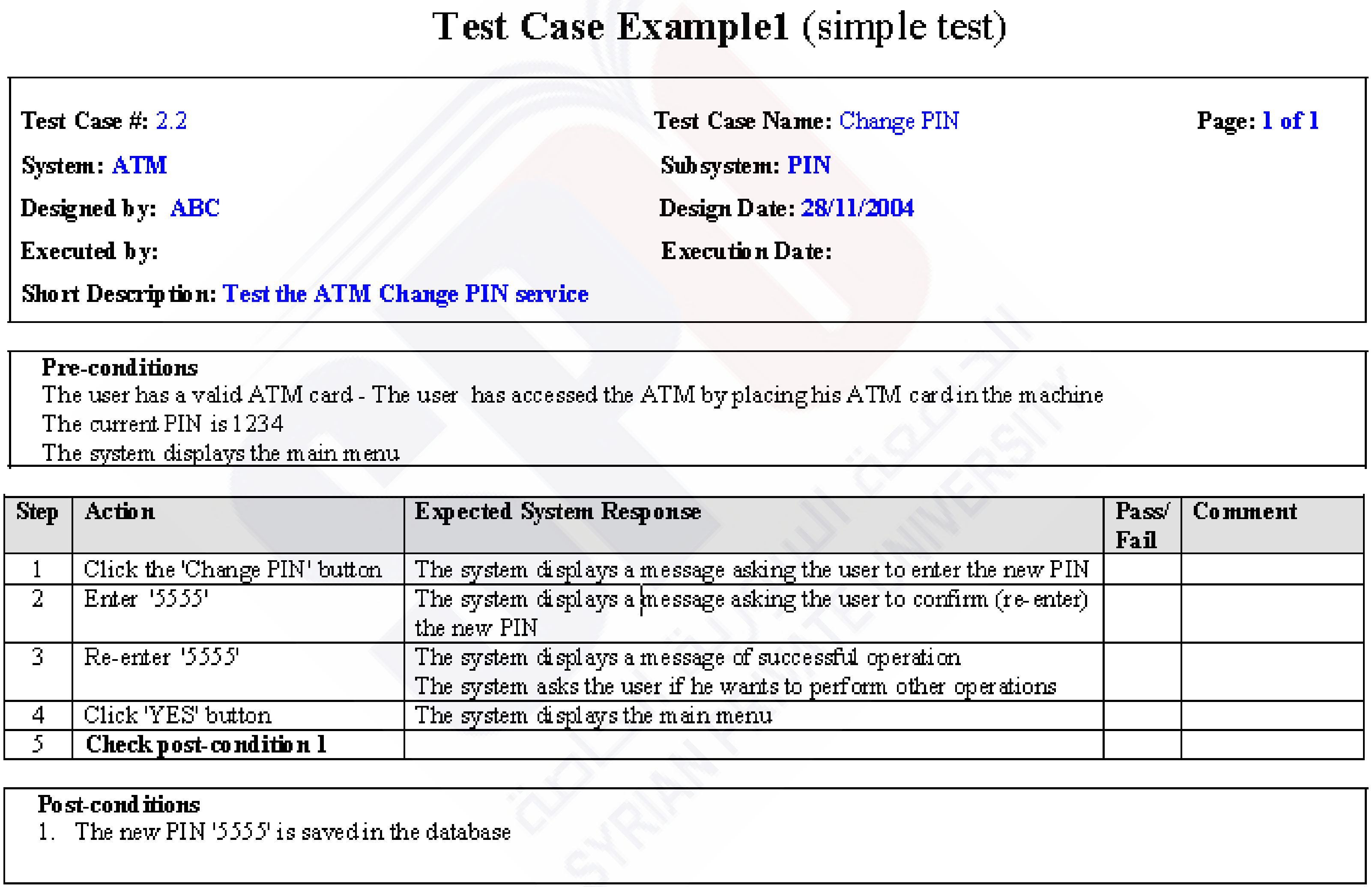
- Test Procedure / Script
  - Can be manual or automated
  - Specifies the sequence of actions for a test, i.e. one or more Test Cases

Writing Good Test Case

- Factors of Good Test case (Basics)
  - TC #
  - TC Name
  - Description
  - Designed by
  - System
  - Subsystem (function)
  - Pre Condition
  - Steps (Clear , Detailed)
  - Expected Result
  - Post Condition
  - Status (Pass / Fail)

  - Multi TC Step status **VS** One TC Status

Sample of Good TC



Test Case Example1 (simple test)

Test Case #: 2.2	**Test Case Name:** Change PIN	**Page:** **1 of 1**
System: **ATM**	**Subsystem:** **PIN**	
Designed by: **ABC**	**Design Date:** **28/11/2004**	
Executed by:	**Execution Date:**	
Short Description: **Test the ATM Change PIN service**		

Pre-conditions
The user has a valid ATM card - The user has accessed the ATM by placing his ATM card in the machine
The current PIN is 1234
The system displays the main menu

Step	Action	Expected System Response	Pass/ Fail	Comment
1	Click the 'Change PIN' button	The system displays a message asking the user to enter the new PIN		
2	Enter '5555'	The system displays a message asking the user to confirm (re-enter) the new PIN		
3	Re-enter '5555'	The system displays a message of successful operation The system asks the user if he wants to perform other operations		
4	Click 'YES' button	The system displays the main menu		
5	**Check post-condition 1**			

Post-conditions
1. The new PIN '5555' is saved in the database

Choosing Test Techniques

- How do you chose the right technique?
  - Type of system
  - Standards
  - Customer or contractual requirements
  - Level of risk
  - Type of risk
  - Testing objectives
  - Documentation available
  - Knowledge / skills of the testers
  - Time and budget
  - Development processes
- Pick the right techniques for the right situation