

FASA PENGUJIAN

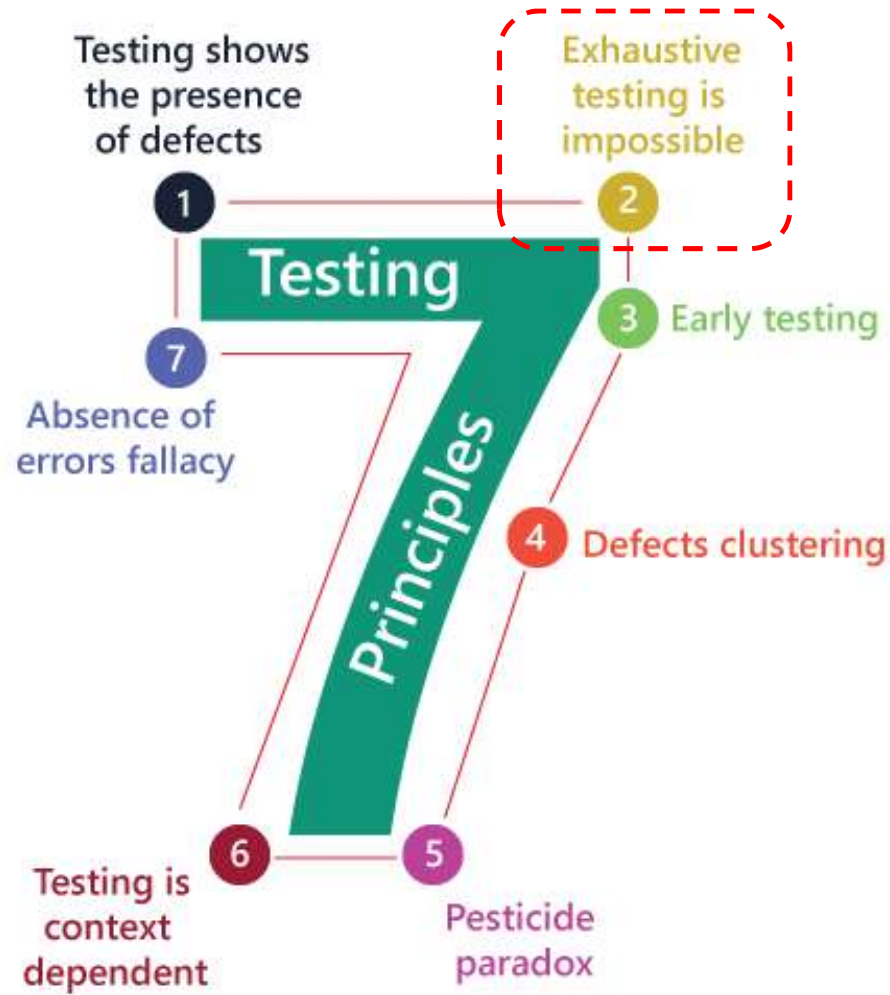
TEKNIK REKA BENTUK PENGUJIAN (TEST DESIGN TECHNIQUES)

TAKLIMAT
04



OBJEKTIF

Mengenalpasti teknik-teknik reka bentuk pengujian



Rujukan :

- ISTQB
- <https://www.javatpoint.com/software-testing-principles>



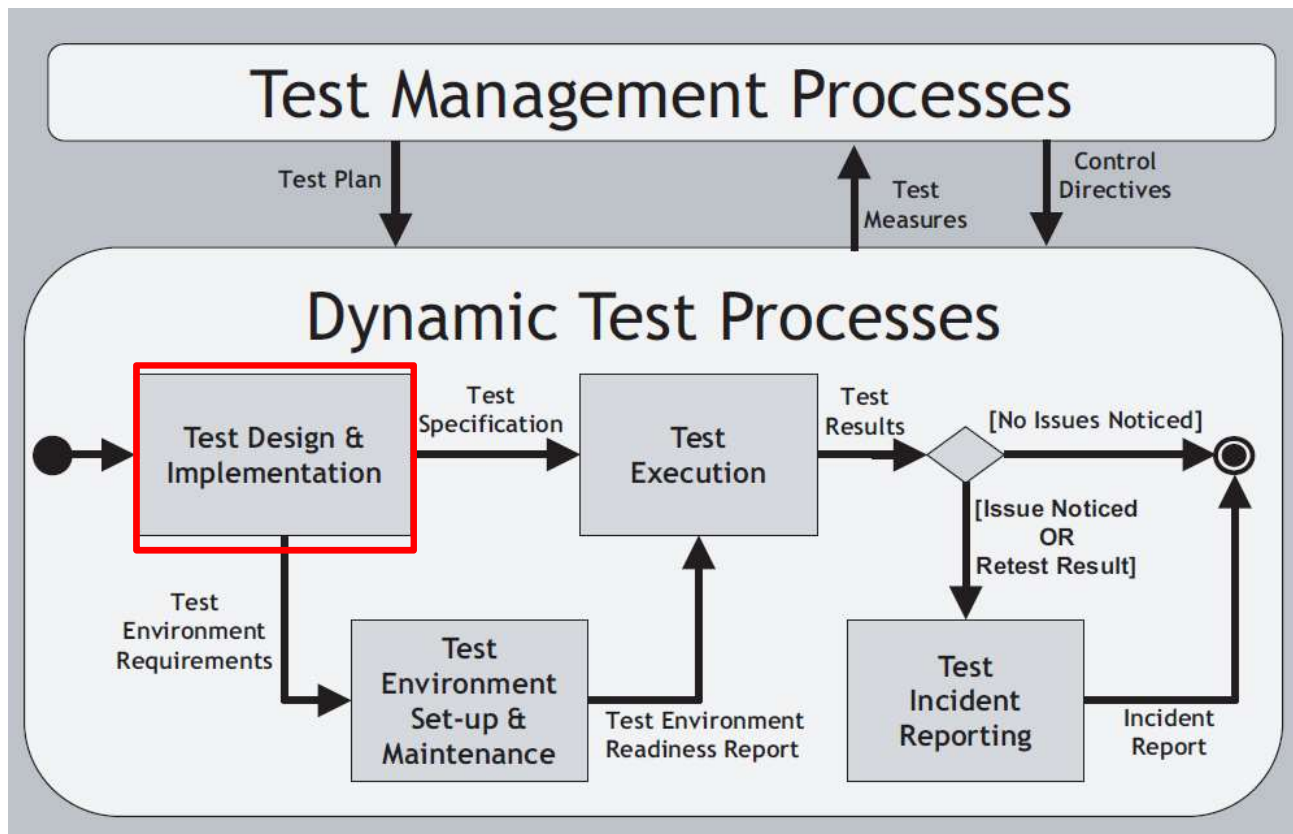
PENGENALAN: TEST PROCESS



Rujukan:
ISTQB® Certified Tester Advanced Level Syllabus Test Analyst Version 2019

FASA PENGUJIAN

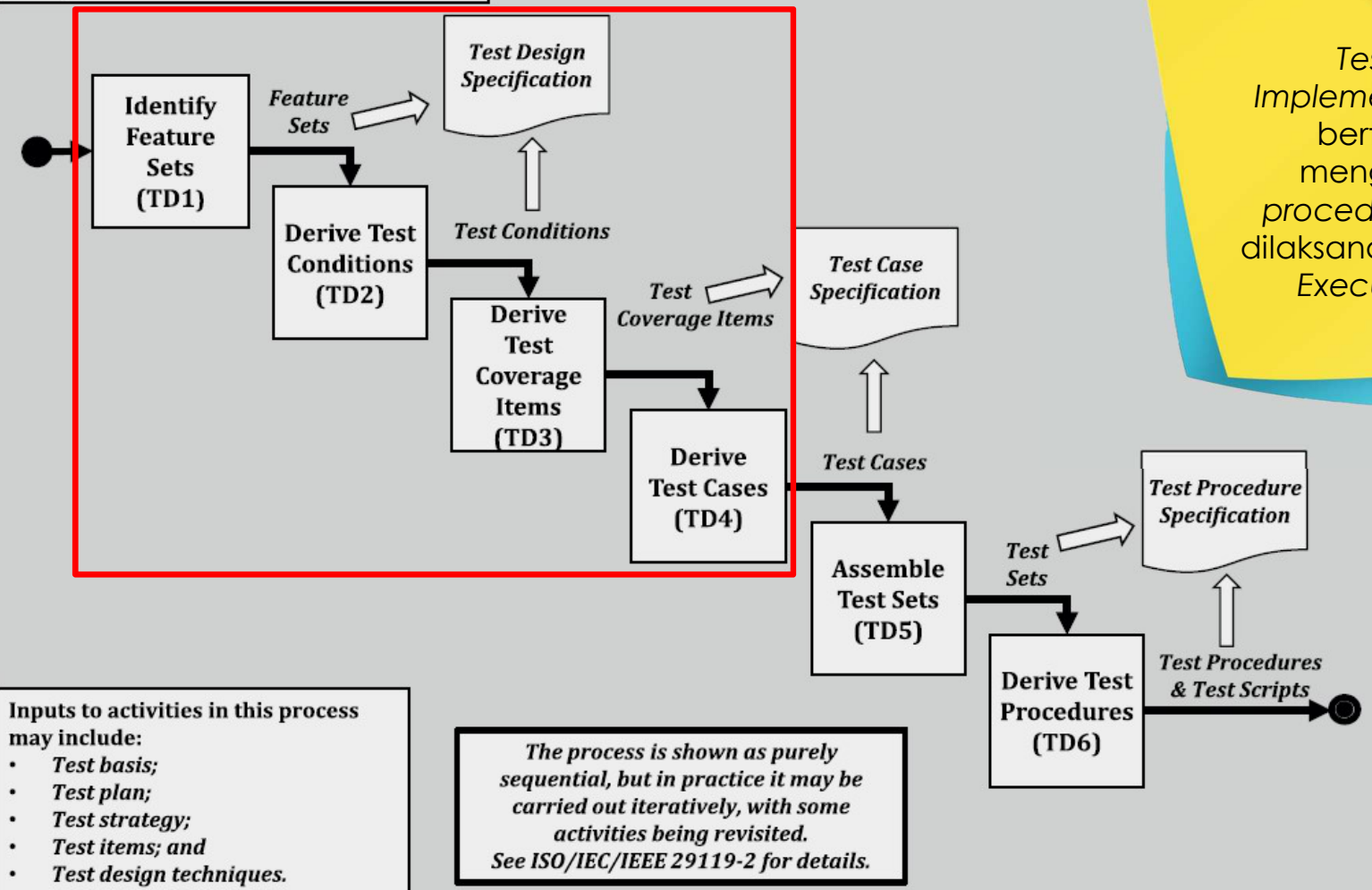
DYNAMIC TEST PROCESS



Proses untuk melaksanakan pengujian dinamik bagi pelbagai jenis ujian (contoh: pengujian unit, integrasi, sistem dan penerimaan) di pelbagai peringkat (contoh: performance testing, security testing, usability testing).

SOURCE : ISO/IEC 29119 – 4

Test Design & Implementation Process



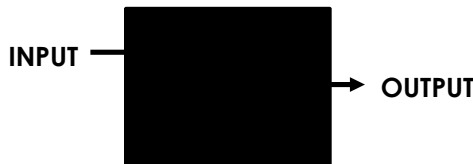
Test Design & Implementation Process bertujuan untuk menghasilkan test procedures yang akan dilaksanakan semasa Test Execution Process.

SOURCE :
ISO/IEC 29119 – 4

TEST DESIGN TECHNIQUES

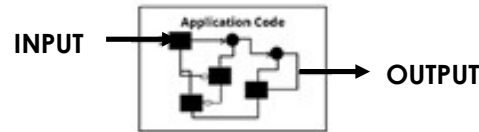
SPECIFICATION-BASED TECHNIQUES

Also known as black-box testing



STRUCTURE-BASED TECHNIQUES

Also known as white-box testing



EXPERIENCE-BASED TECHNIQUES



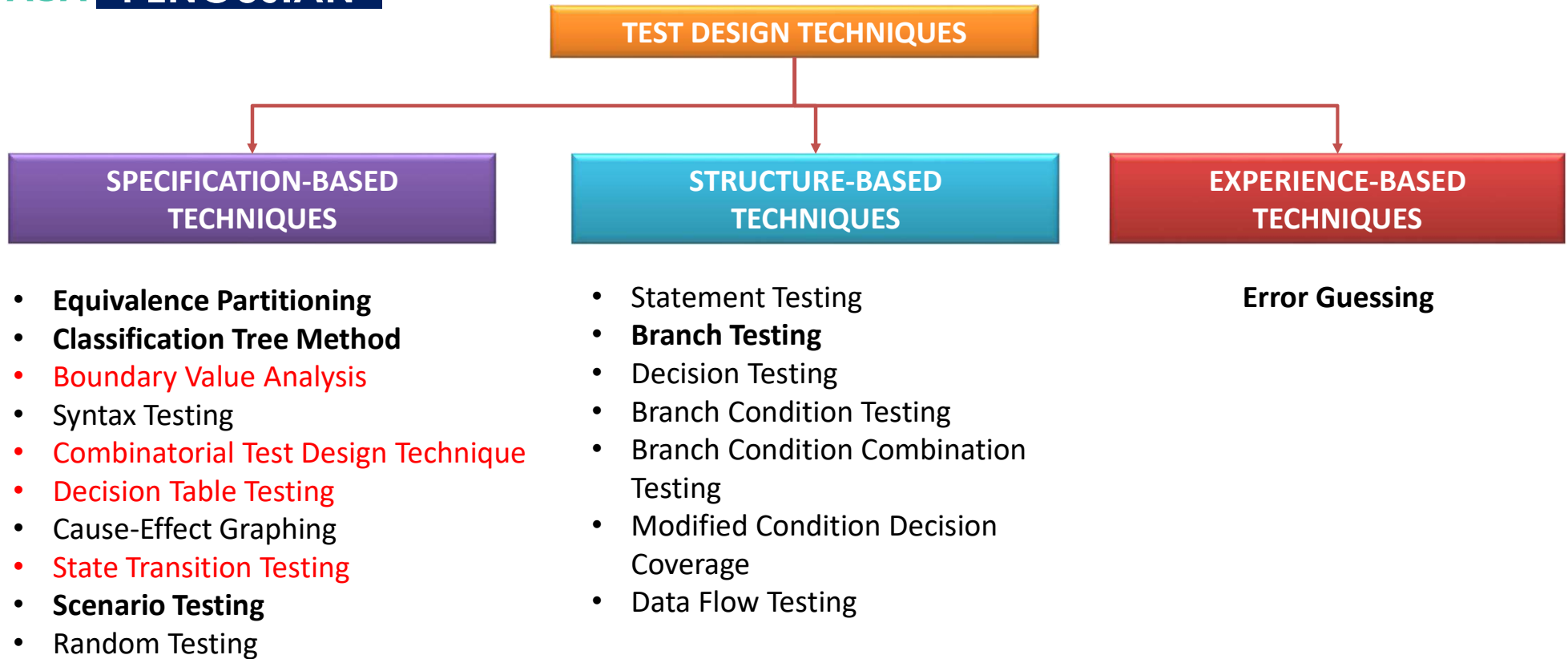
Source to design test case

- Test basis
- Requirements
- Specification
- Models

- Structure of test item
- Source code
- Structure of model

Knowledge & Experience of tester

SOURCE : ISO/IEC 29119 – 4



SOURCE : ISO/IEC 29119 – 4

SPECIFICATION BASED TECHNIQUE

NO	TECHNIQUES	DESCRIPTION	TEST CONDITION	TEST COVERAGE	TEST CASE
1	Equivalence partitioning (EP)	Partition input and output into equivalence partition (partitions/equivalence classes)	Each partition	Each partition	<ul style="list-style-type: none"> Valid input Invalid input Valid output Invalid output
2	Classification Tree	<ul style="list-style-type: none"> Partition input and output into classification tree Each classification consists of disjoint (non-overlapping set of classes/sub-classes) Hierarchical relationship modelled as tree <ul style="list-style-type: none"> domain (root node) classifications (branch nodes) classes / sub-classes (leaf nodes) 	Each classification	Each classification	<ul style="list-style-type: none"> Valid input Invalid input
3	Boundary Value Analysis (BVA)	Partition input and output into number of ordered sets and subsets (partitions/equivalence classes) with identifiable boundaries	Each boundary	For each boundary either : <ul style="list-style-type: none"> two-value boundary testing three value boundary testing 	For each boundary either : <ul style="list-style-type: none"> two-value boundary testing three value boundary testing
4	Syntax testing	<ul style="list-style-type: none"> Syntax model is represented as number of rules Each rule define format of input parameter elements in syntax(sequence of/ iteration of/selection between) Syntax can be represented in textual or diagrammatic format 	Whole or partial model of inputs	<ul style="list-style-type: none"> Positive testing (options derived from selection or iteration) Negative testing (mutations) 	<ul style="list-style-type: none"> One to one Minimized (minimum number to cover all options/ iterations/ mutations) at least once

SPECIFICATION BASED TECHNIQUE

NO	TECHNIQUES	DESCRIPTION	Test condition	Test Coverage	Test Case
5	Combinatorial test design	Combination of test item parameters and parameter's discrete values	P-V pair : Particular aspects of test item that relevant to testing (Parameter) correspond input parameter (Value)	<u>All combination testing</u> <ul style="list-style-type: none"> members of the set of all unique P-V pairs each parameter included at least once 	Each unique P-V pairs
				<u>Pair wise testing</u> <ul style="list-style-type: none"> possible pairs of selected values within total set (fewer test case) a.k.a all pairs testing 	Each pairs of P-V- pairs
				<u>Each choice (1-wise) testing</u> <ul style="list-style-type: none"> members of the set of P-V pairs. each parameter value included at least once 	Each test case exercises one or more P-V pairs that has not been included in the test case
				<u>Base choice testing</u> sets of P-V pairs for each of the input parameters, where all parameters except one are set to their "base" value and the final parameter is set to one of its other valid values	Setting all but one parameter to its base choice and then setting the final parameter to a valid value until the required level of test coverage of PV-pairs is achieved
6	Decision table testing	Logical relationship(decision rules) between conditions(causes) and actions(effects)	Conditions and actions	Each decision rule	Each decision rule

SPECIFICATION BASED TECHNIQUE

NO	TECHNIQUES	DESCRIPTION	Test condition	Test Coverage	Test Case
7	Cause effect graphing	<ul style="list-style-type: none"> Logical relationship(decision rules) between causes(inputs) and effects(outputs) may be represented in cause effect graphs 	Causes and effects	Each decision rule	Each decision rule
8	State transition testing	<ul style="list-style-type: none"> Model states of test item may occupy, transition between states, events cause the transitions and actions that may result from the transitions may be represented in state transition diagram or state table 	May be : <ul style="list-style-type: none"> all states all transition entire state model 	Depends on chosen test completion criterion :	Depends on chosen test completion criterion :
				• all states visited	• all states visited
				• single transitions (0-switch coverage)	• single transitions (0-switch coverage)
				• all transitions	• all transitions
• multiple transitions (N-switch coverage)	• multiple transitions (N-switch coverage)				

SPECIFICATION BASED TECHNIQUE

NO	TECHNIQUES	DESCRIPTION	Test condition	Test Coverage	Test Case
9	Scenario testing	<ul style="list-style-type: none"> Sequences of interactions between test item and other systems for the purpose of testing usage flows involving test items. a.k.a use case testing (how the test item interacts with one or more actors for the purpose of testing sequences of interactions (i.e. scenarios)) consider main scenario and alternative scenarios 	<ul style="list-style-type: none"> One sequences of interactions (a scenario) All sequences of interactions (all scenarios) 	Each main and alternative scenarios	Each main and alternative scenarios
10	Random testing	Input domain that defines the set of all possible input values. An input distribution for the generation of random input values shall be chosen	All input domain	No recognized test coverage items	Chosen by randomly selecting input values from input domain (pseudo-randomly using tool) according to chosen input distribution

STRUCTURE BASED TECHNIQUE

NO	TECHNIQUES	DESCRIPTION	TEST CONDITION	TEST COVERAGE	TEST CASE
1	Statement Testing	A model of source code of test items which identifies executable or non-executable statements	Each executable statement	Each executable statement	
2	Branch Testing	Control flow of test item that identifies branches in control flow	Each branch either : <ul style="list-style-type: none"> • a conditional transfer of control between nodes • an explicit unconditional transfer of control between nodes • a transfer of control to entry point 	Each branch	
3	Decision Testing	<ul style="list-style-type: none"> • Control flow of test item that identifies decisions and conditions within decisions in control flow • Decisions ≥ 2 possible outcomes • Typical decisions : <ul style="list-style-type: none"> ➢ Selection (If-Else) ➢ Loops (While-loop) ➢ Case (Switch) 	Each decision	Decision outcomes	

STRUCTURE BASED TECHNIQUE

NO	TECHNIQUES	DESCRIPTION	TEST CONDITION	TEST COVERAGE	TEST CASE
4	Branch Condition Testing	<ul style="list-style-type: none"> Control flow of test item that identifies decisions and conditions within decisions in control flow Decisions ≥ 2 possible outcomes Typical decisions : <ul style="list-style-type: none"> ➤ Selection (If-Else) ➤ Loops (While-loop) ➤ Case (Switch) 	Each decision	<ul style="list-style-type: none"> all Boolean values (true/false) of the condition(s) within decisions Decision outcomes 	<ul style="list-style-type: none"> all Boolean values (true/false) of the condition(s) within decisions Decision outcomes
5	Branch Condition Combination Testing	Control flow of test item that identifies decisions and conditions	Each decision	Each unique feasible combination of Boolean values of conditions within each decision	Each unique feasible combination of Boolean values of conditions within each decision
6	Modified Condition Decision Coverage	Control flow of test item that identifies decisions and conditions	Each decision	Each unique feasible combination of individual Boolean values of conditions within a decision that allows a single Boolean condition to independently affect outcome of decision	Each unique feasible combination of individual Boolean values of conditions within a decision that allows a single Boolean condition to independently affect outcome of decision



FASA PENGUJIAN

STRUCTURE BASED TECHNIQUE

NO	TECHNIQUES	DESCRIPTION	TEST CONDITION	TEST COVERAGE	TEST CASE
7	Data Flow Testing	<ul style="list-style-type: none"> Control flow of test item that identifies control flow sub-path through test item Each definition of given variable is linked to subsequent use(s) of same variable no intervening redefinition of variable's value 	Each definition-use pair for a variable	<p><u>All definitions testing</u> The control flow sub-path from each variable definition (either p-use or c-use) of definition shall be identified at least one definition free sub-path from the definition to one of its c-uses or p-uses covered for all variable definition</p> <p><u>All-c-uses testing</u> The control flow sub-path from each variable definition to each c-use of definition shall be identified at least one of definition free sub-path from the definition to each one of its c-uses covered for all relevant variable definition</p> <p><u>All-p-uses testing</u> The control flow sub-path from each variable definition to each p-use of definition shall be identified all sub-paths from each variable definition to each of its uses that can be reached (without intervening redefinition for the variable)</p> <p><u>All-uses testing</u> The control flow sub-path from each variable definition to every use (both p-use and c-use) of definition shall be identified at least one of definition free sub-path from the definition to each one of its p-uses covered for all relevant variable definition</p> <p><u>All-du-paths testing</u> The control flow sub-path from each variable definition to every use (both p-use and c-use) of definition shall be identified all sub-paths from each variable definition to each of its uses that can be reached (without intervening redefinition for the variable)</p>	



FASA PENGUJIAN

EXPERIENCED-BASED TECHNIQUE

NO	TECHNIQUES	DESCRIPTION	TEST CONDITION	TEST COVERAGE	TEST CASE
1	Error Guessing	<ul style="list-style-type: none">• Design s checklist of defect types that may exists in test item• Allow testers to identify input that may cause failures if defects exist	Each potential defect	No recognised test item	Repeat until required testing completed



FASA PENGUJIAN

CONTOH EQUIVALENCE PARTITIONING

Sistem akan menterjemahkan markah yang diperolehi pelajar (dalam bentuk integer) kepada gred seperti berikut:

- 70 hingga 100 - 'A'
- 50 hingga 69 - 'B'
- 30 hingga 49 - 'C'
- 0 hingga 29 - 'D'

Sekiranya input yang tidak sah (invalid) dimasukkan, sistem akan memaparkan mesej “**Ralat, input tidak sah**”

Feature Set (FS)1 – generate_grading function





FASA PENGUJIAN

CONTOH EQUIVALENCE PARTITIONING



	TC1	TC2	TC3	TC4	TC5	TC6	TC7
INPUT	15	43	52	88	-5	105	@@@
TEST COVERAGE	TCV1	TCV2	TCV3	TCV4	TCV5	TCV6	TCV7
PARTITION (TEST CONDITION)	$0 \leq \text{marks} \leq 29$	$30 \leq \text{marks} \leq 49$	$50 \leq \text{marks} \leq 69$	$70 \leq \text{marks} \leq 100$	$\text{marks} < 0$	$\text{marks} > 0$	Marks not integer
EXPECTED OUTPUT	D	C	B	A	Ralat, input tidak sah	Ralat, input tidak sah	Ralat, input tidak sah





FASA PENGUJIAN

CONTOH CLASSIFICATION TREE

Travel preference

- Destination** Domestic International
- Class** First Class Business Economy
- Seat** Aisle Window
- Meal** Vegetarian Non-vegetarian

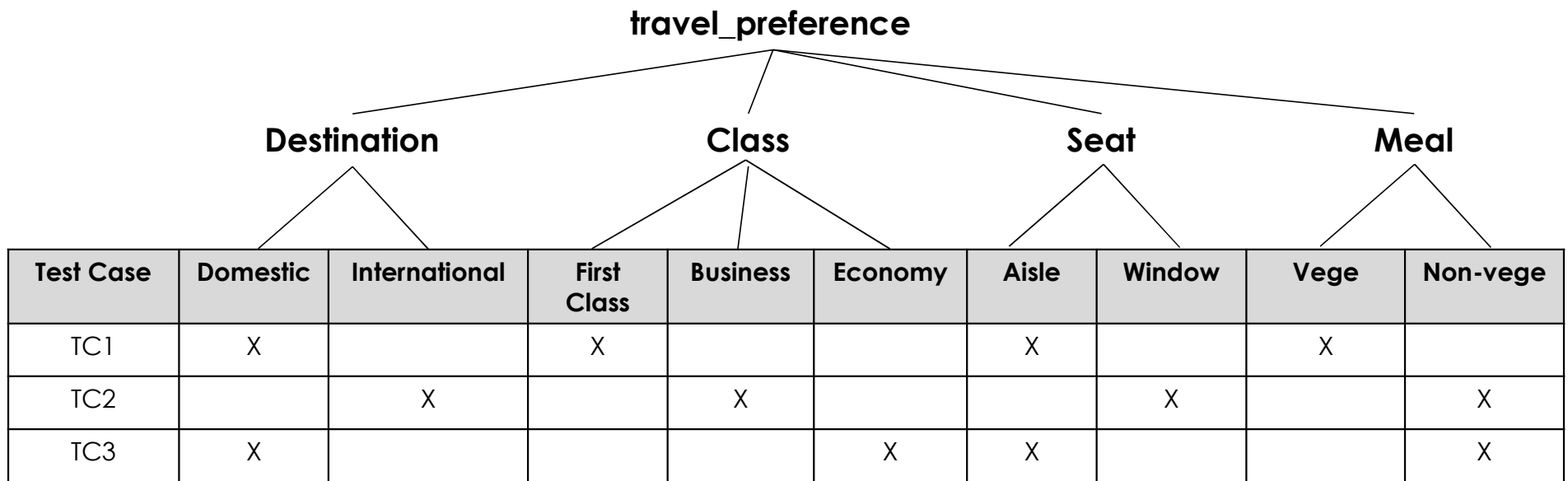
SUBMIT

CANCEL

Booking accepted



CONTOH CLASSIFICATION TREE



Feature Set (FS)1 – travel_preference function





FASA PENGUJIAN

CONTOH CLASSIFICATION TREE

TEST CASE	INPUT				EXPECTED RESULT	TEST COVERAGE ITEM
TC1	Domestic	First Class	Aisle	Vege	Booking Accepted	TCV1
TC2	International	Business	Window	Non-vege	Booking Accepted	TCV2
TC3	Domestic	Economy	Aisle	Non-vege	Booking Accepted	TCV3



CONTOH SCENARIO TESTING

Nota: Rujuk Use Case yang disertakan

Feature Set (FS)1 – change_password function

TEST CONDITION	EXPECTED RESULT	TEST COVERAGE ITEM	TEST CASE ID
TCN1	Basic Flow	TCV1	TC1
TCN2	Alternative Flow – Existing Password Incorrect	TCV2	TC2
TCN3	Alternative Flow – New Password Less Than 8 Characters	TCV3	TC3
TCN4	Alternative Flow – New Password Same as Current Password	TCV4	TC4
TCN5	Alternative Flow – New Passwords Do Not Match	TCV5	TC5



CONTOH BRANCH CONDITION TESTING

Consider the following fragment of code:

```

If A or (B and C) then
    Print "YES";
Else
    Print "Error";
End If
    
```

TCV1	A = TRUE	TCN1
TCV2	A = FALSE	
TCV3	B = TRUE	
TCV4	B = FALSE	
TCV5	C = TRUE	
TCV6	C = FALSE	
TCV7	A or (B and C) = TRUE	
TCV8	A or (B and C) = FALSE	

Feature Set (FS)1 – condition code fragment

TCN1 : A or (B and C)

TEST CASE	A	B	C	A or (B and C)	TEST COVERAGE
TC1	FALSE	FALSE	FALSE	FALSE	TCV2, TCV4, TCV6
TC2	TRUE	TRUE	TRUE	TRUE	TCV1, TCV3, TCV5





FASA PENGUJIAN

CONTOH ERROR GUESSING

Sistem akan menterjemahkan markah yang diperolehi pelajar (dalam bentuk integer) kepada gred seperti berikut:

- 70 hingga 100 - 'A'
- 50 hingga 69 - 'B'
- 30 hingga 49 - 'C'
- 0 hingga 29 - 'D'

Sekiranya input yang tidak sah (invalid) dimasukkan, sistem akan memaparkan mesej “**Ralat, input tidak sah**”. Input yang diterima hanya dalam format integer.

Feature Set (FS)1 – generate_grading function





FASA PENGUJIAN

CONTOH ERROR GUESSING

TCN1	Enter NULL	TCV1
TCN2	Enter 0	TCV2
TCN3	Enter negative number	TCV3
TCN4	Enter very large number (10 digit)	TCV4
TCN5	Enter very large string (10 chars)	TCV5

	TC1	TC2	TC3	TC4	TC5
INPUT	NULL	0	-5	1234567910	abcdefghij
TEST COVERAGE	TCV1	TCV2	TCV3	TCV4	TCV5
OUTPUT	Ralat, input tidak sah	Ralat, input tidak sah	Ralat, input tidak sah	Ralat, input tidak sah	Ralat, input tidak sah



FASA PENGUJIAN

1. LATIHAN BOUNDARY VALUE ANALYSIS (BVA) TECHNIQUE

Sistem akan menterjemahkan markah yang diperolehi pelajar (dalam bentuk integer) kepada gred seperti berikut:

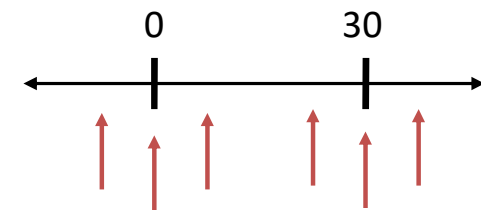
- ❑ 70 hingga 100 - 'A'
- ❑ 50 hingga 69 - 'B'
- ❑ 30 hingga 49 - 'C'
- ❑ 0 hingga 29 - 'D'

Sekiranya input yang tidak sah (invalid) dimasukkan, sistem akan memaparkan mesej “**Ralat, input tidak sah**”

Feature Set (FS)1 – generate_grading function

Apakah *test condition*, *test coverage* dan *test case* bagi spesifikasi keperluan di atas menggunakan *three-value boundary testing with minimum test case*?

Nota : Nilai terkecil bagi markah adalah 0.5



three-value boundary testing



FASA PENGUJIAN

1. LATIHAN BOUNDARY VALUE ANALYSIS (BVA) TECHNIQUE



EQUIVALENCE PARTITION	$0 \leq \text{marks} \leq 29$	$30 \leq \text{marks} \leq 49$	$50 \leq \text{marks} \leq 69$	$70 \leq \text{marks} \leq 100$	$\text{marks} < 0$	$\text{marks} > 100$
TEST CONDITION		Each boundary is a test condition				
TEST COVERAGE		Three value for each boundary				
TEST CASE		One TC can fulfil more than one TCV which is similar				
OUTPUT	D	C	B	A	Ralat, input tidak sah	Ralat, input tidak sah

2. LATIHAN COMBINATORIAL TEST DESIGN TECHNIQUES

Travel preference

Destination Domestic International
Class First Class Business Economy
Seat Aisle Window

SUBMIT

CANCEL

Booking accepted

Apakah test condition, test coverage dan test case bagi spesifikasi berikut menggunakan all combination testing?

Feature Set (FS)1 – travel_preference function



FASA PENGUJIAN

2. LATIHAN COMBINATORIAL TEST DESIGN TECHNIQUES

Each P-V is a test condition

TCN1	Destination – Domestic

Identify test coverage using unique combinations of P-V pairs, made up of one P-V pair for each test item parameter

TCV1	Destination – Domestic Class – First Class Seat - Aisle



FASA PENGUJIAN

2. LATIHAN COMBINATORIAL TEST DESIGN TECHNIQUES

TEST CASE	INPUT			EXPECTED RESULT	TEST COVERAGE ITEM
TC1	Domestic	First Class	Aisle	Booking Accepted	TCV1

Identify TC for each test coverage

3. LATIHAN DECISION TABLE TEST DESIGN TECHNIQUES

The check out function for MyShop application will check for membership status, total purchase (initially inclusive with delivery fee) and payment type. If the customer is a member, he/she will get 10% discount from total purchased item (excluding delivery fee). If the customer total purchase exceed RM 100 , he or she will get free delivery and free gift.

Item	Stainless steel shoe rack	
Quantity	<input type="text" value="1"/>	<input type="button" value="▼"/>
Delivery fee	RM 8.00	
Price	RM 59.00	
Member ID	<input type="text" value="S3289"/>	
	<i>- RM 5.90 (10% discount for member)</i>	
Payment type	<input checked="" type="radio"/> Online <input type="radio"/> COD	
Total	RM 61.10	<input type="button" value="CHECKOUT"/>

The total purchase will be re-calculated automatically if customer entitled for member's discount or exemption of delivery fee. If customer select for COD, the item will be delivered immediately after checkout. Else, customer will be directed to payment gateway.

Apakah test condition, test coverage dan test case bagi spesifikasi di atas menggunakan decision table testing?

Feature Set (FS)1 – checkout function



FASA PENGUJIAN

3. LATIHAN DECISION TABLE TEST DESIGN TECHNIQUES

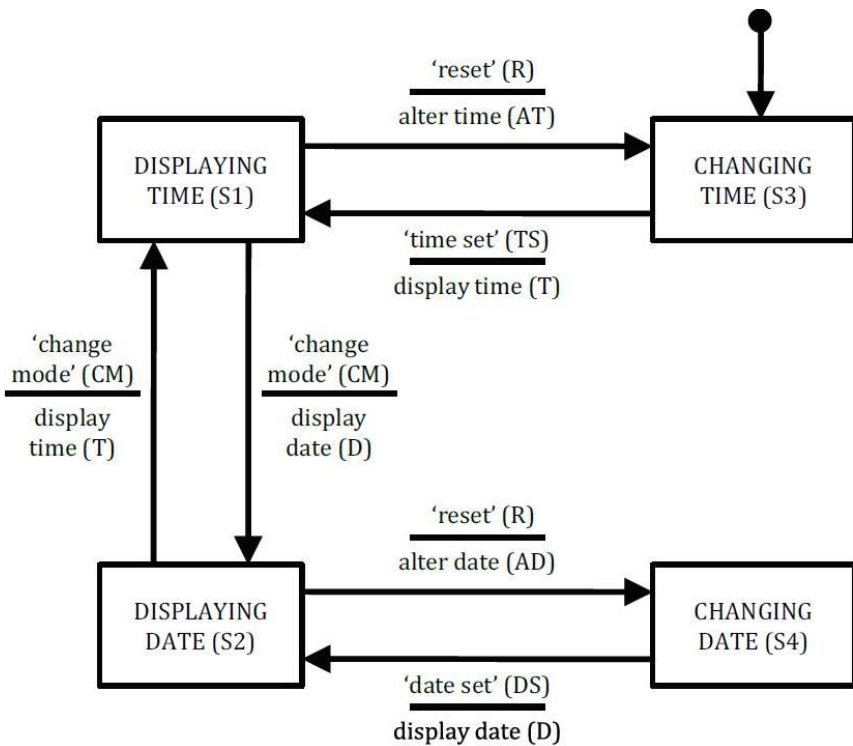
INPUT/CONDITION					
OUTPUT/ACTION					

Identify test condition

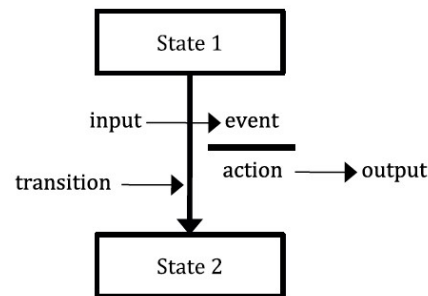
INPUT/CONDITION					
OUTPUT/ACTION					

Identify test coverage
(each decision rule)

4. LATIHAN STATE TRANSITION TEST DESIGN TECHNIQUES



Apakah test condition, test coverage dan test case bagi state transition (all-transition) diagram berikut ?



Generic state model

	Input 1	Input 2	etc
Start State 1	Entry A	Entry B	etc
Start State 1	Entry C	Entry D	etc
etc	etc	etc	etc

State table

Entry [] = Finish state/Output or Action for the given start state or input

Feature Set (FS)1 – manage_display_changes



FASA PENGUJIAN

4. LATIHAN STATE TRANSITION TEST DESIGN TECHNIQUES

	CM	R	TS	DS
S1				
S2				
S3				
S4				

Identify test coverage for all transition

Test Case			
Start State			
Input			
Expected Output			
Finish State			
Test Coverage			

Identify test case for each test coverage



MAMPU

Unit Pemodenan Tadbiran dan Perancangan Pengurusan Malaysia

All information incorporated within this slide is created for
Malaysian Administrative Management and Planning Unit (MAMPU),
Prime Minister's Department, Malaysia.

All information is the property of MAMPU and any unauthorized
reproduction is prohibited

**TERIMA
KASIH**