

Test **Management**: Part I

Software Testing: IN3240 / IN 4240

Summary

Test organisation

Independence | Tasks of the test leader and testers

Test planning and estimation

Activities | Entry and exit criteria | Estimation | Strategy and approach

Test progress monitoring and control

Configuration and management

Risk and testing



Part I: Close-ended questions

Independent Testing

Question 1

Why is **independent** testing **important**?

- a. Independent testing is usually cheaper than testing your own work
- b. Independent testing is more effective at finding defects
- c. Independent testers should determine the processes and methodologies used
- d. Independent testers are dispassionate about whether the project succeeds or fails

Question 1: Clues

Why is **independent** testing **important**?

Testing software and **developing** (building) software are **not** the same

Different tasks involved

Require **different mindsets** from testers and developers

Testing is an **assessment** of **quality**

Assessments are **not always positive**

Separate the **testers** from the **developers**

Improve defect finding by using independent testers

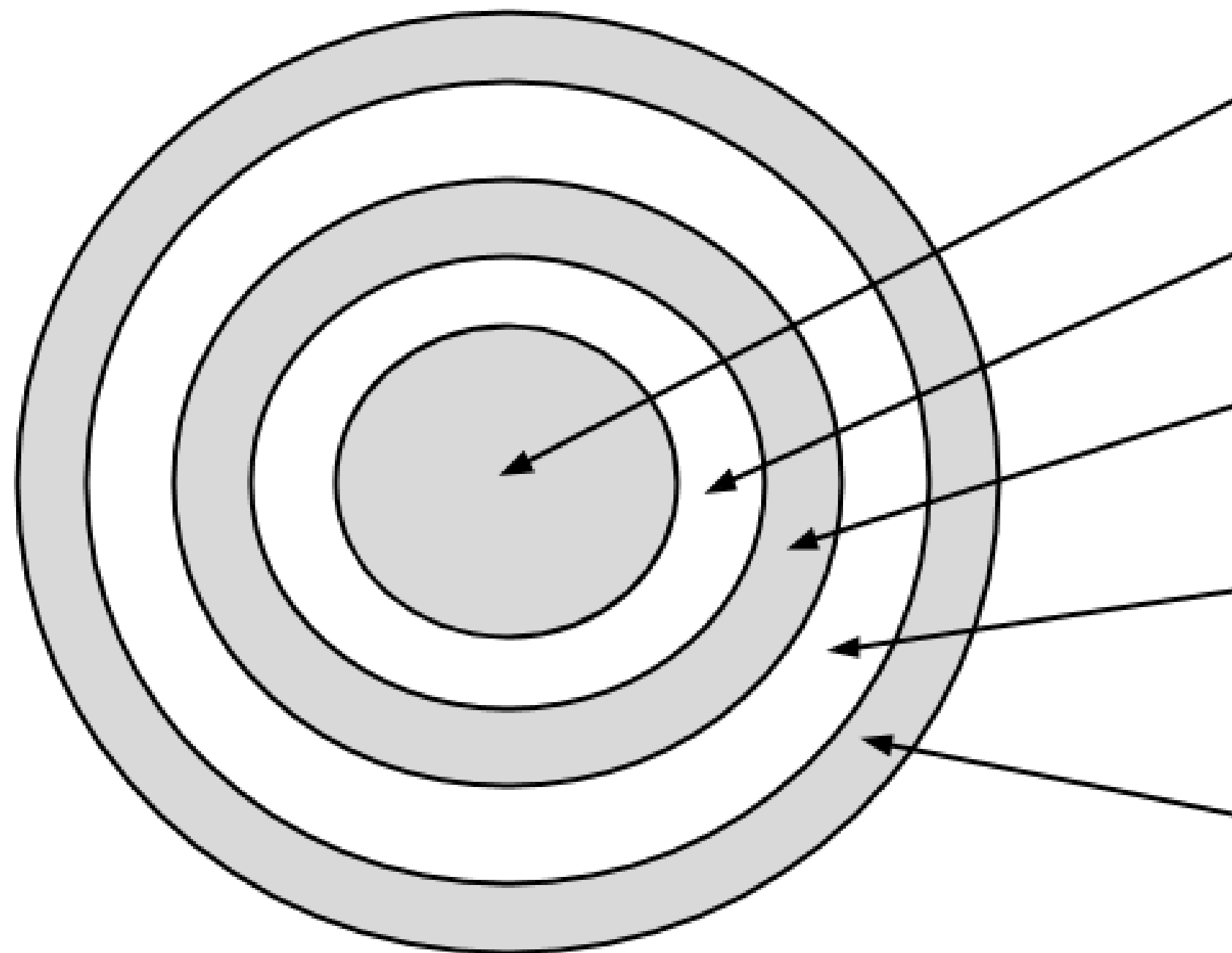
Avoid author **bias** → **Objective** assessments

Question 1: Clues

Why is **independent** testing **important**?

Options for **independence**

Independence is not either or, but a **continuum**



1. No independent testers. Developers test own code.
2. Independent testers within development teams.
3. Independent test team / group within organisation report to project management.
4. Independent testers from business or user community.
5. Independent test specialists for specific test targets such as usability testers, security testers or certification testers.

Question 1: Answer

Why is **independent** testing **important**?

- a. Independent testing is usually cheaper than testing your own work
- b. Independent testing is more effective at finding defects**
- c. Independent testers should determine the processes and methodologies used
- d. Independent testers are dispassionate about whether the project succeeds or fails

Question 2

Which of the following is an **advantage** of **independent testing**?

- a. Independent testers don't have to spend time communicating with the project team
- b. Programmers can stop worrying about the quality of their work and focus on producing more code
- c. The others on the project can pressure the independent testers to accelerate testing at the end of the test schedule
- d. Independent testers sometimes question the assumptions behind the requirements, design and implementations

Question 2: Clues

Which of the following is an **advantage** of **independent testing**?

Benefits of **independent** testing

Independent testers can often see **more**, other, **different defects**

Compared to testers working within the programming team

Bring a **different** set of **assumptions** to testing / reviews

Expose hidden defects / **problems**

Not affected by business analysts, designers, programmers, etc.

Sceptical attitude

Professional **pessimism**



Question 2: Clues

Which of the following is an **advantage** of **independent testing**?

Benefits of **independent** testing

Independent test **teams** may enjoy **more credibility** in an organisation

Compared to a test leader who is part of the programming team

Can report test results **honestly** and **without concern**

Distanced from **co-workers** (programmers/managers) responsible for the defects

Often have **separate budgets**

Facilitates to ensure proper level of spending on testing resources

Training, test tools, equipment, etc.

Question 2: Clues

Which of the following is an **advantage** of **independent testing**?

Risks of **independent** testing

Testers / test team can become **isolated**

Interpersonal isolation from programmers, designers, project team

Isolation from the **broader view** of quality and business objectives

Obsessive focus on defect finding

Refusal to **accept** business **prioritisation** of defects

Communication problems

Lack of **identification** with the project goals



Question 2: Clues

Which of the following is an **advantage** of **independent testing**?

Risks of **independent** testing

Developers may **abdicate** their **responsibility** for quality

“Why bother unit testing when we already have a test team?”

Focus only on **pointing** out **flaws** / defects

Provide little (or no) solutions

Failure to **understand** the tester's **role**

Should provide a **service** to the project team

Goal is **not** to **dictate** how things should be done



Question 2: Answer

Which of the following is an **advantage** of **independent testing**?

- a. Independent testers don't have to spend time communicating with the project team
- b. Programmers can stop worrying about the quality of their work and focus on producing more code
- c. The others on the project can pressure the independent testers to accelerate testing at the end of the test schedule
- d. Independent testers sometimes question the assumptions behind the requirements, design and implementations**

Testing Roles and Tasks

Question 3

According to the **ISTQB glossary**, what do we **mean** when we call someone a **test manager**?

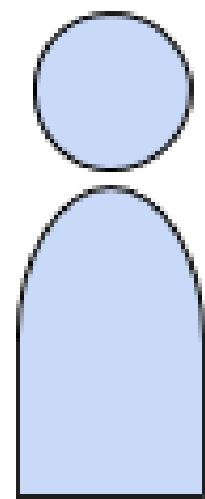
- a. A test manager manages a collection of test leaders
- b. A test manager is the leader of a test team or teams
- c. A test manager gets paid more than a test leader
- d. A test manager reports to a test leader



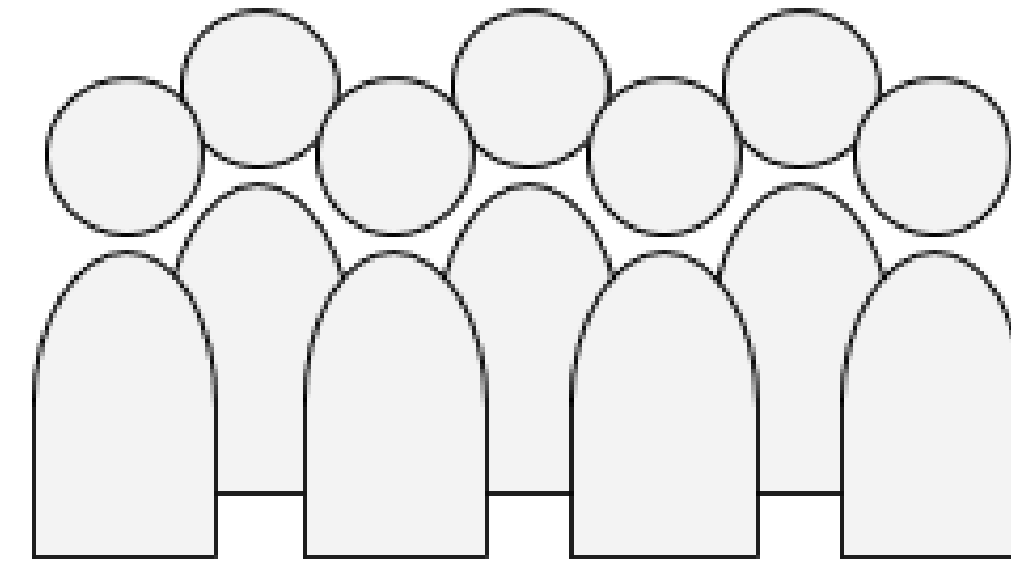
Question 3: Clues

According to the **ISTQB glossary**, what do we **mean** when we call someone a **test manager**?

Roles within a test **team**



Test Leader
(manager / coordinator)



Testers

Test **manager**

Plans / Monitors / Controls the testing **activities** and **tasks**

Leader of a test team



Question 3: Answer

According to the **ISTQB glossary**, what do we **mean** when we call someone a **test manager**?

- a. A test manager manages a collection of test leaders
- b. A test manager is the leader of a test team or teams**
- c. A test manager gets paid more than a test leader
- d. A test manager reports to a test leader



Question 4

Which of the following is among the typical **tasks** of a **test leader**?

- a. Develop system requirements, design specifications and usage models
- b. Handle all test automation duties
- c. Keep test cases and coverage hidden from programmers
- d. Gather and report test progress metrics



Question 4: Clues

Which of the following is among the typical **tasks** of a **test leader**?

Coordination

Devise test strategy and plan with project managers

Planning the tests

Understand the test objectives

Select test approaches

Estimate time, effort, and cost of testing

Define test levels

Plan incident management

Plan and Control

Analysis and Design

Implementation and Execution

Evaluation of Exit Criteria and Reporting

Test Closure Activities



Question 4: Clues

Which of the following is among the typical **tasks** of a **test leader**?

Manage test **configuration**

Set up adequate configuration management

Testware for traceability

Introduce **metrics**

For measuring test progress

Evaluating the quality of the testing and product

Automation of tests

Decide what to automate

Plan and Control

Analysis and Design

Implementation and Execution

Evaluation of Exit Criteria and Reporting

Test Closure Activities



Question 4: Clues

Which of the following is among the typical **tasks** of a **test leader**?

Test **specifications, preparation** and **execution**

Initiate specification / preparation / implementation / execution

Monitor test results

Check exit criteria

Adapt planning

Based on test results and progress

Plan and Control

Analysis and Design

Implementation and Execution

Evaluation of Exit Criteria and Reporting

Test Closure Activities

Question 4: Answer

Which of the following is among the typical **tasks** of a **test leader**?

- a. Develop system requirements, design specifications and usage models
- b. Handle all test automation duties
- c. Keep test cases and coverage hidden from programmers
- d. Gather and report test progress metrics**



Question 5

According to the **ISTQB Glossary**, what is a **test level**?

- a. A group of test activities that are organised together
- b. One or more test design specification documents
- c. A test type
- d. An ISTQB certification



Question 5: Clues

According to the **ISQTB Glossary**, what is a **test level**?

Test **level**

“A group of test activities that are organised and managed together”

Four test levels

Component testing / **Integration** testing / **System** testing / **Acceptance** testing

Purpose of test levels

Include different **methodologies** that can be **used** when conducting the **test** effort

Has clear, **level-specific**, pre-defined **objectives**

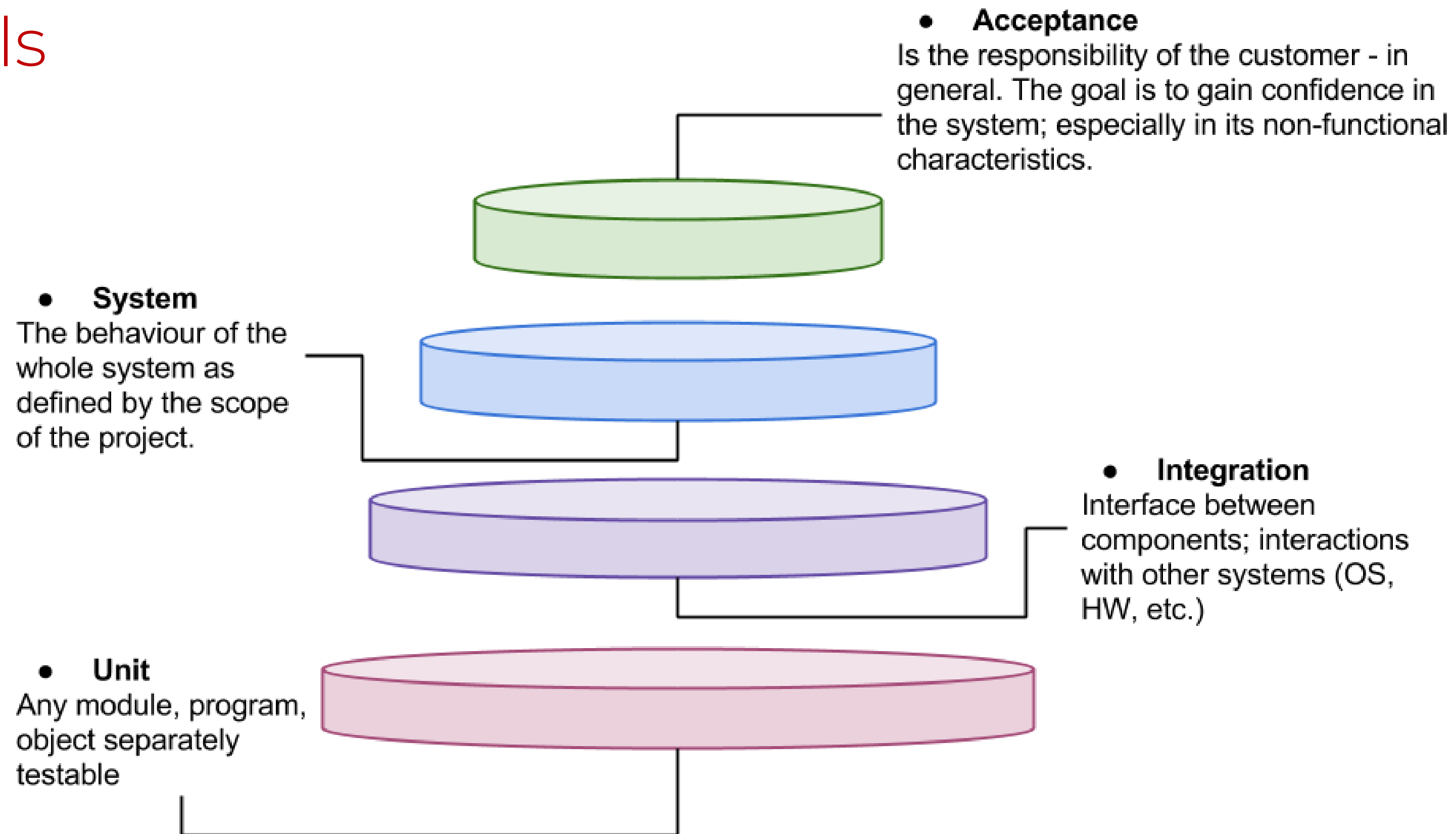
Different test **types** may be **performed** at different **levels**

A decorative graphic at the bottom of the slide consisting of a series of overlapping grey arrows pointing to the right, with a small grey circle at the far right end.

Question 5: Clues

According to the **ISQTB Glossary**, what is a **test level**?

Test **levels**



Question 5: Answer

According to the [ISQTB Glossary](#), what is a **test level**?

- a. **A group of test activities that are organised together**
- b. One or more test design specification documents
- c. A test type
- d. An ISTQB certification



Test Planning and Documents

Question 6

A **test plan** is written specifically to **describe** a **level** of **testing** where the primary goal is **establishing confidence** in the system. Which of the following is a likely **name** for the **document**?

- a. Master test plan
- b. System test plan
- c. Acceptance test plan
- d. Project test plan



Question 6: Clues

Which of the following is a likely **name** for the **document** where the goal is establishing **confidence** in the **system**?

Test plan → Project plan for the testing work to be conducted

Elaborates on the **challenges** that await the **test effort**

Highlights **important topics**

Serve to **communicate** all **aspects** of the test effort

Project team, testers, peers, managers, other stakeholders

Helps to **manage change**

Revise test plans as we gather more information

Serves as **documentation**



Question 6: Clues

Which of the following is a likely **name** for the **document** where the goal is establishing **confidence** in the **system**?

Considerations for **writing** a test **plan**

What is **within** the **scope** of the test effort, and what is **outside** the scope?

What are the **constraints** affecting the test effort?

E.g. budget limitations, time constraints, etc.

What are the **test objectives**?

What are the important **project** and **product risks**?

Which **aspects** of the product are more (or less) **testable**?

What should be the **overall** test **execution schedule**?



Question 6: Clues

Which of the following is a likely **name** for the **document** where the goal is establishing **confidence** in the **system**?

Test plan **documentation**

Master test plan

Overall plan for the test effort

Level-specific test plans

Plans specifically **aimed** at **each** test **level**

Aims to **adhere** to the **objectives** of that level

Central **question**: What is the **purpose** of this test level?

The test plan seeks to **facilitate** the **realisation** of these **objectives**



Question 6: Answer

A **test plan** is written specifically to **describe** a **level** of **testing** where the primary goal is **establishing confidence** in the system. Which of the following is a likely **name** for the **document**?

- a. Master test plan
- b. System test plan
- c. Acceptance test plan**
- d. Project test plan



Question 7

What is the **primary difference** between a test **plan**, test **design specification**, and test **procedure specification**?

- a. The test plan describes one or more levels of testing, the test design specification identifies the associated high-level test cases and a test procedure specification describes the actions for executing a test
- b. The test plan is for managers, the test design specification is for programmers and the test procedure specification is for the testers who are automating the tests
- c. The test plan is the least thorough, the test procedure specification is the most thorough and the test design specification is midway between the two
- d. The test plan is finished in the first third of the project, the test design specification is finished in the middle third of the project and the test procedure specification is finished in the last third of the project

Question 7: Clues

What is the **primary difference** between a test **plan**, test **design specification**, and test **procedure specification**?

Before executing a test → Need to **know what** we are **trying to test**

Inputs / Expected outcomes

How to get ready and run the actual tests

Artefacts for testing → Each **specified** in its **own document**

(IEEE 829)

Test **conditions**

Test **cases**



Question 7: Clues

What is the **primary difference** between a test **plan**, test **design specification**, and test **procedure specification**?

Test **conditions** → Test **Design Specification**

Condition: Something that can be tested

Approach: “Throw the net wide”

Identify as many conditions as possible

Exhaustive testing is impossible → Select a subset of all possible tests

Examples

Measuring branch coverage → Test basis is the code itself

Requirements specification → Table of contents can be the list of test

conditions



Question 7: Clues

What is the **primary difference** between a test **plan**, test **design specification**, and test **procedure specification**?

[IEEE 829: Test **Design Specification** Template]

Test design **specification identifier**

Unique name / Version date and number / Author and contact information / Revision history

Features to be tested

Features / Appropriate level of testing / Reference to original documentation

Approach refinements

Selection of test technique / Methods for analysis / Relationship between test items and level of testing

Test **identification**

Identification of each test case / procedure

Feature **Pass/Fail** Criteria

Describe criteria for assessing the feature / Whether tests were successful or not

Question 7: Clues

What is the **primary difference** between a test **plan**, test **design specification**, and test **procedure specification**?

Test **cases** → Test **Case Specification**

Case: Set of input values, preconditions, expected results

Developed for a particular objective or test condition

Can cover a number of test conditions

E.g. Age, gender, enough credit?, place of birth, etc.

Objective

To assess that the system does what it is supposed to do

Given the conditions, does it behave correctly?



Question 7: Clues

What is the **primary difference** between a test **plan**, test **design specification**, and test **procedure specification**?

[IEEE 829: Test **Case Specification** Template]

Test case **specification identifier**

Unique name / Version date and number / Author and contact information / Revision history

Test **items**

Requirements specification / System and detail design specification / Users guide / Operations manual / etc.

Input and **Output** specifications

Data (values, ranges, sets) / Tables / Human actions / Conditions (states) / Files / Relationships

Environmental needs

Special procedural requirements

Intercase dependencies

Any prerequisite test cases



Question 7: Clues

What is the **primary difference** between a test **plan**, test **design specification**, and test **procedure specification**?

Test **procedures** → Test **Procedure Specification**

Procedure: Sequence of actions for the execution of a test

Test script

Manual test script

Formed into a **test execution schedule**

Description of the **order** of the test **procedures**

By **whom** do the **tests** need to be **run**?

Useful for **prioritising** tests → Ensure best testing is done in the time

available

A decorative graphic at the bottom of the slide consisting of a series of overlapping grey arrows pointing to the right, with a small grey circle at the far right end.

Question 7: Clues

What is the **primary difference** between a test **plan**, test **design specification**, and test **procedure specification**?

[IEEE 829: Test **Procedure Specification** Template]

Test procedure **specification identifier**

Unique name / Version date and number / Author and contact information / Revision history

Purpose

List all test cases covered by the procedure / Description of the procedure

Special requirements

Manual or automated / Stages in which the test is to be used (pre-testing, regression, etc.)

Test environment / Skills required / Prerequisite procedures

Steps

Log / Setup / Start / Proceed / Measure / Shutdown / Restart / Stop / Wrap-up

Contingencies



Question 7: Answer

What is the **primary difference** between a test **plan**, test **design specification**, and test **procedure specification**?

- a. The test plan describes one or more levels of testing, the test design specification identifies the associated high-level test cases and a test procedure specification describes the actions for executing a test
- b. The test plan is for managers, the test design specification is for programmers and the test procedure specification is for the testers who are automating the tests
- c. The test plan is the least thorough, the test procedure specification is the most thorough and the test design specification is midway between the two
- d. The test plan is finished in the first third of the project, the test design specification is finished in the middle third of the project and the test procedure specification is finished in the last third of the project

Entry and Exit Criteria

Question 8

Entry criteria for testing means that the company **management** gave their **OK** to the development team to **start the test activities**

- a. True
- b. False



Question 8: Clues

Entry criteria

Defines **when** to **start testing**

Central **question**: Do we have **enough** to conduct a meaningful test effort?

Typical **considerations**

Test **environment** availability and readiness

Test **tool** readiness in the test environment

Testable **code** availability

Test **data** availability



Question 8: Answer

Entry criteria for testing means that the company management gave their OK to the development team to start the test activities

- a. True
- b. False



Question 9

The ISTQB Foundation Syllabus established a fundamental **test process** where test **planning** occurs **early** in the project, while test **execution** occurs **later**. Which of the following **elements** of the test **plan**, while **specified during** test **planning**, are **assessed** during test **execution**?

- a. Test tasks
- b. Environmental needs
- c. Exit criteria
- d. Test team training



Question 9: Clues

Which of the following **elements** of a test **plan**, while **specified during test planning**, are **assessed** during test **execution**?

Purpose of testing

Verify some **aspect** of the **system** / Reveal **faults** in the implementation

What do we define **prior** to test **execution**?

Test **tasks**: What is to be done and when?

Environmental needs: What requirements do we have for the test environment?

Entry and **exit criteria**: When to start and when to end the test effort?

Question 9: Clues

Which of the following **elements** of a test **plan**, while **specified during** test **planning**, are **assessed** during test **execution**?

What do we assess *during* to test *execution*?

The results of the test against the pre-defined objectives

We are interested in **answering** the following **questions**

How did the test **go**?

Did it go according to **plan**? Why, why not?

Do we **need** to run **more** tests?



Question 9: Clues

Which of the following **elements** of a test **plan**, while **specified during test planning**, are **assessed** during test **execution**?

Exit criteria → When to stop testing? (Definition of *enough*)

Typical **considerations**

Thoroughness measures: Code coverage / Functionality coverage / Risk

Estimates: Defect density / Reliability measures

Cost

Residual risk: Defects not fixed / Lack of test coverage in some areas

Schedule: Time to market



Question 9: Answer

The ISTQB Foundation Syllabus established a fundamental **test process** where test **planning** occurs **early** in the project, while test **execution** occurs **later**. Which of the following **elements** of the test **plan**, while **specified during** test **planning**, are **assessed** during test **execution**?

- a. Test tasks
- b. Environmental needs
- c. Exit criteria**
- d. Test team training



Question 10

Consider the following **exit criteria** which might be found in a **test plan**. Which of these **belong** in an **acceptance test plan**?

1. No known customer-critical defects
 2. All interfaces between components tested
 3. 100 % code coverage of all items
 4. All specified requirements satisfied
 5. System functionality matches legacy system for all business rules
- a. All statements belong in an acceptance test plan
 - b. Only statement 1 belongs in an acceptance test plan
 - c. Only statements 1, 2 and 5 belong in an acceptance test plan
 - d. Only statements 1, 4 and 5 belong in an acceptance test plan

Question 10: Clues

Which of the given **exit criteria belong** in an **acceptance test plan**?

Acceptance testing

Validation testing with respect to the users

Requirements, business processes

Typically **assesses** aspects such as

The system's **adherence** to the **requirements** specification

Defects that may be **critical** to the customer

Functionality in accordance with the **business rules**?

Whether or not the system is **fit for use**

A decorative graphic at the bottom of the slide consisting of a series of overlapping, right-pointing chevrons in shades of gray, ending with a small gray circle.

Question 10: Answer

Consider the following **exit criteria** which might be found in a **test plan**. Which of these **belong** in an **acceptance test plan**?

1. No known **customer-critical defects**
 2. All interfaces between components tested
 3. 100 % code coverage of all items
 4. All **specified requirements** satisfied
 5. System **functionality** matches **legacy** system for all **business rules**
- a. All statements belong in an acceptance test plan
 - b. Only statement 1 belongs in an acceptance test plan
 - c. Only statements 1, 2 and 5 belong in an acceptance test plan
 - d. Only statements 1, 4 and 5 belong in an acceptance test plan**

Question 11

During **test execution**, the test **manager describes** the following **situation** to the project team:

- 90 % of the test cases have been run.
- 20 % of the test cases have identified defects.
- 110 defects have been found.
- 100 defects have been fixed and have passed confirmation testing.
- Of the remaining 10 defects, project management has decided that they do not need to be fixed prior to release.

Question 11

Which of the following is the most **reasonable interpretation** of this **test status report**?

- a. The remaining 10 defects should be confirmation tested prior to release
- b. The remaining 10 % of test cases should be run prior to release
- c. The system is ready for release with no further testing or development effort
- d. The programmers should focus their attention on fixing the remaining known defects prior to release

Question 11: Answer

Which of the following is the most **reasonable interpretation** of this **test status report**?

- a. The remaining 10 defects should be confirmation tested prior to release
- b. The remaining 10 % of test cases should be run prior to release**
- c. The system is ready for release with no further testing or development effort
- d. The programmers should focus their attention on fixing the remaining known defects prior to release

Question 12

The purpose of _____ criteria is to define **when to stop** testing, such as at the **end** of a **test level** or when a **set of tests** has a **specific goal**.



Question 12: Answer

The purpose of _____ criteria is to define **when to stop testing**, such as at the **end** of a **test level** or when a **set of tests** has a **specific goal**.

EXIT (criteria)



Question 13

The **metrics** for **test progress monitoring** can be **collected** both **manually** and **automatically**

- a. True
- b. False



Question 13: Clues

The **metrics** for **test progress monitoring** can be collected both **manually** and **automatically**

Test progress monitoring

Provide feedback on how the test effort is going

Visible information about the test results

Measure the status of testing

Gathering data to be used for future estimation of test effort

IEEE 829: Test Log Template



Question 13: Answer

The **metrics** for **test progress monitoring** can be collected both **manually** and **automatically**

- a. **True**
- b. False



Question 14

Pair the following **test design techniques** with the typical **problems** they address:

Decision tables	Applied when the inputs or outputs can be grouped in a way that exhibits similar behaviour
Use case testing	Used to test sequences of states or sequences of transitions
State transition testing	Used when the problem can be described as an interaction between an actor and the system
Boundary value analysis	Used when the inputs and actions can be expressed as Boolean values
Equivalence partitioning	Applied when the inputs and outputs can be grouped in equivalent partitions. The technique tests the edges of each equivalence partition

Question 14: Answer

Pair the following **test design techniques** with the typical **problems** they address:

Decision tables	Applied when the inputs or outputs can be grouped in a way that exhibits similar behaviour
Use case testing	Used to test sequences of states or sequences of transitions
State transition testing	Used when the problem can be described as an interaction between an actor and the system
Boundary value analysis	Used when the inputs and actions can be expressed as Boolean values
Equivalence partitioning	Applied when the inputs and outputs can be grouped in equivalent partitions. The technique tests the edges of each equivalence partition

Part II: Exercises and Open-ended questions

Exercise 1

Describe briefly what is **meant** by the following **test approaches** (strategies)

- a. Analytical approach
- b. Model-based approach
- c. Methodical approach
- d. Process- or standard-compliant approach
- e. Dynamic and heuristic approach
- f. Consultative approach
- g. Regression-averse approach

Is one approach better than the other? Why, why not?
Which do you prefer?



Exercise 1(a): Answer

Analytical approach

Use of formal / informal **analytical technique**

Factors strongly affecting the testing environment

Risk-based strategy

Perform risk analysis using project documents and stakeholder input

Planning, estimation, designing, and prioritising tests based on risk

Requirements-based strategy

Analysis of the requirements specification

Basis for planning, estimation, design



Exercise 1(b): Answer

Model-based approach

Tests designed based on models of the object functionality

Critical system behaviour

Emphasis on identification and selection of the appropriate model

Preventive test approach

Example

Can build mathematical models for loading/response for e-commerce servers

Design tests based on the models

Exercise 1(c): Answer

Methodical approach

Adhere to a **pre-planned**, systematised **approach**

Developed in-house, assembled from various concepts

Following a specific **method**

Tests are designed, executed and implemented in accordance

Examples

Adherence to certain checklists

Failure-based (error checking, fault-attacks)

Experience-based



Exercise 1(d): Answer

Process- or standard-compliant approach

“Go by the book” as opposed to *“do it your own way”*

Uses externally developed **industry standards**

Design and implement test assets based on these

Little (or no) customisation

Examples

Adhering to IEEE 829 standards

Adhering to agile methodologies

E.g. Extreme programming (XP)



Exercise 1(e): Answer

Dynamic and heuristic approach

Heuristic (techniques)

Approach to problem solving that employs a practical methods

Not guaranteed to be optimal, but sufficient for intermediate goals

Typically applied during the later stages of testing

Problem: Exhaustive testing is impossible

Solution: Finding as many defects as possible

Examples

Exploratory testing



Exercise 1(f): Answer

Consultative approach

Rely on a group of non-testers to guide / perform test effort

Seek advice and guidance from externals

- Business domain experts

- Technical experts (e.g. security experts)

- Outside the test team

Examples

- Asking users for what to test

- Asking developers to develop the tests



Exercise 1(g): Answer

Regression-averse approach

Techniques to manage the risk of regression

Goal: Avoid (re-)occurrence of defects

Create effective regression tests to run when anything changes

Re-run every test to ensure nothing has been broken

Practices

Automating functional tests prior to release / Re-use existing test material

Example

Having a standard set of test data for which a screen needs to pass



Exercise 1: Answer

Is one **approach better** than the **other**? **Why**, why not?

Some strategies are **preventive** (prior)

Analytical test strategies involve upfront analysis of the problem area

Tend to identify test basis prior to test execution

Some strategies are **reactive** (during)

Dynamic test strategies focus on the test execution period

Enable the identification of defect (clusters) that may be hard to anticipate

Strategies **complement** each other

Testing is context-dependent: There is no *best way* to test



Exercise 1: Answer

Factors to consider when choosing test strategies

Risks

Testing is about risk-management

Consider the risks and level of risk

Well-established applications that evolve slowly → Regression is an important risk

Skills

Strategies must not only be chosen → They must also be executed

Consider which skills the test team possesses / lacks

Standard-compliant strategy can make up for lack of skill within a test team



Exercise 1: Answer

Factors to consider when choosing test strategies

Objectives

Testing must satisfy the needs of the stakeholders to be successful

Example

If the objective is to find as many defects as possible, with limited resources

Dynamic testing may prove beneficial

Regulations

Sometimes you must satisfy not only to stakeholder, but also to existing regulations

Devise a methodical test strategy that adheres to these regulations

A decorative graphic at the bottom of the slide consisting of a series of overlapping grey arrows pointing to the right, with a small grey circle at the end.

Exercise 1: Answer

Factors to consider when choosing test strategies

Product

Some products tend to have well-specified requirements

E.g. weapons and contract-development systems

Synergy with a requirements-based analytical strategy

Business

Business considerations and continuity are important

Example: Model-based approach

When you can use a legacy system as a model for a new system



The slides are made by

Yulai Fjeld, revised by Eva H. Vihovde

