

Static techniques

Software Testing: IN3240 / IN4240

Summary

Static techniques and the test process

What is static analysis / testing?

Review types

Informal review / Walkthrough / Technical review / Inspection

Varying degree of formality

Static analysis by tools

Typical defects detected



Part I: Close-ended questions

Question 1

Which of the following **artefacts** can be **examined** by using **review** techniques?

- a. Software code
- b. Requirements specification
- c. Test designs
- d. All of the above



Question 1: Clues

Which of the following **artefacts** can be **examined** by using **review** techniques?

Review process

Process / Meeting → **Examine software** products

From very **formal** (structured + regulated) to **informal** (no written instructions)

Objectives

Find **defects** → Defects detected earlier are usually cheaper to remove

Gain **understanding** → Find omissions in requirements / specifications

Discussion / Decision-making



Question 1: Clues

Which of the following **artefacts** can be **examined** by using **review** techniques?

Any software **product** can be **reviewed**

Requirements specification

Design specification

Source **code**

Test plans / specifications / cases / scripts

Product **manuals** / User **guides**

Web **pages**



Question 2

A **static analysis tool** gives **quality information** about **the code without executing it**.

- a. True
- b. False



Question 2: Clues

A **static analysis tool** gives **quality information** about **the code without executing it**.

Static analysis

Examination of **code without executing it**

E.g. through compiling code

Understanding code **structures** / dependencies

May help to **ensure** code **adheres** to industry **standards**

Tools for static analysis

Manual examination of work product

Automated tools to assist in examination



Question 3

Which is **not** a type of **review**?

- a. Walkthrough
- b. Inspection
- c. Informal review
- d. Management approval



Question 3: Clues

Which is **not** a type of **review**?

Types of reviews

Informal review

Inexpensive way to get some benefit

Walkthrough

Learning / Gaining understanding / Defect finding

Technical review

Discussion / Decision-making / Defect-finding / Solving technical problems / Check conformance

Inspection

Finding defects



Question 4

Which **statement** about **reviews** is **true**?

- a. Inspections are led by a facilitator or moderator, whereas technical reviews are not necessarily.
- b. Technical reviews are led by a trained leader, inspections are not
- c. In a walkthrough, the author does not attend
- d. Participants for a walkthrough always need to be thoroughly trained



Question 4: Clues

Which **statement** about **reviews** is **true**?

Reviews vary in degree of **formality**

Defines ...

Content and **focus area** of review meeting

Roles present during review

Responsibilities of each participant

Level of **documentation** / **effort** based on formality



Question 4: Clues

Which **statement** about **reviews** is **true**?

Informal review

Pair programming

Technical lead → Reviews the design / code

No formal process

Documentation optional

Walkthrough

Led by **author**

Open-ended sessions → Scenarios / Dry runs / Peer group

In practice: Varies from very informal to very formal



Question 4: Clues

Which **statement** about **reviews** is **true**?

Technical review

Peer review without management participation

Ideally led by a **facilitator or moderator**

Documented → Defined defect-detection process

Peers and technical experts present during review meeting

Requires **pre-meeting preparations**

Optional use of

Checklists / Review reports / List of findings

Management may participate



Question 4: Clues

Which **statement** about **reviews** is **true**?

Inspection

Peer examination

Always led by **facilitator or moderator** (not author)

Formal process → Checklists / Rules / Entry and exit criteria

Includes **metrics**

Pre-meeting preparations required

Defined **roles**

Produce and follows **inspection report** / list of findings

Formal **follow-up process**



Question 5

What is the main difference between a walkthrough and an inspection?

- a. An inspection is led by authors, whilst a review is led by a trained facilitator or moderator
- b. An inspection has a trained leader, whilst a walkthrough has no leader.
- c. Authors are not present during inspections, whilst they are during walkthroughs
- d. A walkthrough is led by the author, whilst an inspection is led by a facilitator or moderator

Question 5: Clues

Which of the following is **true** regarding the process of **fixing emergency changes**?

Walkthrough reviews

Objectives: Gain **understanding** / Find defects

Led by **author**

Open-ended sessions

Inspection reviews

Objectives: Find **defects**

Led by **trained facilitator or moderator**

Formal process with **follow-up** meeting



Question 6

What statement about **static analysis is true?**

- a. With static analysis, defects can be found that are difficult to find with dynamic testing
- b. Compiling is not a form of static analysis
- c. When properly performed, static analysis makes functional testing redundant
- d. Static analysis finds all faults



Question 6: Clues

What statement about **static analysis** is true?

Static analysis

Testing **code without executing** it

E.g. Compiling code

Checks code / requirement and design **documents**

Objective: **Improve** quality / **Prevent** defects / **Verify** software product

Verification process → Have we built the *correct* software?

Dynamic testing

Testing done by **executing** source **code**

Validation process → Have we built the software *correctly*?



Question 7

Which of the following statements about **early test design are **true** and which are **false**?**

1. Defects found during early test design are more expensive to fix
2. Early test design can find defects
3. Early test design can cause changes to the requirements
4. Early test design takes more effort

- a. 1 and 3 are true. 2 and 4 are false.
- b. 2 is true. 1, 3 and 4 are false.
- c. 2 and 3 are true. 1 and 4 are false.
- d. 2, 3 and 4 are true. 1 is false.

Question 7: Clues

Which of the following statements about **early test design** are **true** and which are **false**?

Early test design

Preventive action → Avoid defects being introduced

Find defects

Less expensive to fix defects during earlier stages → *Less to fix*

Less effort involved → *Less to do*

Reveals faults in requirements

Can change the requirements specification



Question 8

Static code analysis typically identifies all but one of the following problems. Which is it?

- a. Unreachable code
- b. Undeclared variables
- c. Faults in the requirements
- d. Redundant code



Question 8: Clues

Static code analysis typically identifies **all but one** of the following **problems**. Which is it?

Static code analysis

Examination of code without executing it

Finds **defects** rather than **failures**

Typical **defects** discovered

Undefined / unused **variables**

Inconsistent interface between modules and components

Unreachable code / **Deadlocks** / **Duplicates**

Programming standard **violations** / **Syntax** violations

Question 9

The _____ of a **review** process is related to the following factors:

- The **maturity** of the development process
- Any **legal requirements** for the software product/project
- The **need** for an **audit trail**



Question 9: Clues

The ____ of a **review** process is related to the following factors:

Review process

Objectives: Find **defects** / Gain **understanding** / **Decision-making**

Different **types** of reviews

Informal review / Walkthrough / Technical review / Inspection

Varying degree of **formality**

What is the **main objective** of a **specific** review (meeting)?

How **far** we have come (**maturity**)

Jurisprudence and other **regulations**

Documentation and **audit** trails needed?



Question 10

Pair the following **review activities** with their **description**:

1. Planning	A. The facilitator distributes to all the participants the doc to be reviewed.
2. Initiate review	B. Each participant reads their part of the document and notes the defects found
3. Individual preparation	C. The author of the reviewed doc fixes the defects found and reported in the review meeting
4. Review meeting	D. A facilitator selects who is going to attend the review activity and assigns roles in the review process
5. Rework	E. The facilitator checks if the defects have been fixed
6. Follow-up	F. Meeting in which each participant lists the defects they have found. The author takes notes. The facilitator moderates the discussion.

Part II: Exercises and Open-ended questions

Exercise: Video

Watch video on “**Clean Code**”

By Robert Cecil Martin (Uncle Bob)

<https://www.youtube.com/watch?v=7EmboKQH8IM>



Open-Ended Questions

Why do you think it is **important** to have **clean code**?

Why is it important to **keep it clean**?

Do you think it is **good** to **impose coding conventions** to a team?

For example: Naming conventions, tabs, complexity of methods, interfaces, API, etc.



Importance of Clean Code

Clean Code: **Aspects** to consider

Rigidity / Dependencies

Coupling

Maintainability / Portability

Robustness

Is **clean code** more **important** than **efficient** code?

Back in the day → Important to write efficient code

Maximise functionality packed into each kilobyte of storage

How tightly it compiled / How much RAM it used

Perhaps no longer such marginal restrictions?



Coding Conventions

Guidelines for **specific** programming **language**

Improve software **quality**

Readability / Maintainability of source code

Limit **complexity**

Recommendations for ...

Programming **style**

Such as comment conventions / Indentation / Line length / Naming conventions

Practices and **methods**

Not enforced by **compilers!**

