Static techniques

Software Testing: INF3121 / INF4121

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 2

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

a. True
b. False

Question 3

Which is not a type of review?

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

Question 4

Which statement about reviews is true?

a. Inspections are led by a trained 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 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 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 trained moderator

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 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 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. Too few comments

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 10

Pair the following review activities with their description:

1. Planning	A. The moderator distributes to all the participants the doc to be reviewed.
2. Kick-off	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 moderator selects who is going to attend the review activity and assigns roles in the review process
5. Rework	E. The moderator 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 moderator moderates the discussion.

Part II: Exercises and Open-ended questions

Exercise: Video

Watch video on "Clean Code"

By Robert Cecil Martin (Uncle Bob)

www.youtube.com/watch?v=QHnLmvDxGTY&feature=youtu.be&list=PL0t9k9FlHnTki4D06nfKTfO9aw0xaElwx&t=760

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!

The End

Assignments

2-4 people in each group

Alt. I: Register as an individual. We form the groups

Alt. II: Register the entire group at once.

Next week:

Work with the first compulsory assignment

The seminar slides are made by

Yulai Fjeld **ydfjeld @ uio.no**

Master student

Department of Informatics

University of Oslo

Previously taught courses

Systemutvikling (INF1050), Universitet i Oslo

Software Testing (INF3121/4121), Universitetet i Oslo

Systemutvikling (ADSE2200), Høgskolen i Oslo og Akershus