Course 2

Testing, Inspection, Walkthrough

How to perform SQ CONTROL?

- Testing
- Inspections
- Walkthroughs
- Reviews

Testing

- Process of gathering information by making observations and comparing them to expectations [Dale Emery]

- Most used quality improvement activity

Testing

- **Black-box**:
  - tests the functionality of an application
  - Tester cannot see the inner code
- **White-box**:
  - tests internal structures of an application
  - Tester knows the inner code

Testing

- Unit testing – execution of a complete class/ routine written by a single person, tested in isolation
- Component testing – execution of a class/ package written by several persons, tested in isolation
- Integration testing – combined execution of 2 or more classes/ components/ packages/ subsystems created by teams – continuous process
- Regression testing – repetition of previously executed test cases
- System testing – execution of the final configuration, including integration with other systems

Debugging

- ≈ testing?

- When you find an error (execute test case) => 2-steps process:

  1. Determine the location and category of error
  2. Fix the error

Testing vs. Debugging

Testing	Debugging
starts with known conditions, uses predefined methods, and has predictable outcomes	starts from possibly unknown initial conditions and its end cannot be predicted
Performed by testing team	Performed by development team
Can be automated	-
Goal: find as many bugs as possible	Goal: find and remove a bug
Find bugs	Find cause of the bug

Testing tools

- Automate testing process
- Tool for generating test cases
- Tool for performing testing: unit, integration, system

Testing Tools

- xUnit
- C#, Java, Python
- Mercury
- Selenium, Winrunner, QTP
- Application Center Test (ACT) .NET
- WebSphere IBM

Testing ⟹ Software Quality

✓ Testing – important part of SQA

× Testing cannot prove error free programs

× One testing strategy (unit/ component/ integration) – finds ≤ 50% errors

× Combination of testing strategies – finds ≤ 60% errors

=> only testing does not improve SQ

Testing ⟹ Software Quality

Myers classic test:
1 program – 15 errors
Average ≈5 /15
Best ≈ 9/15

Software Inspection

- Reading or visually inspecting the code
- best industry practice for detecting software defects early and learning about software artifacts
- Include:
  - the structured review process,
  - standard of excellence product checklists,
  - defined roles of participants, and
  - the forms and reports
- Improve reliability, availability and maintainability

Steps in Software Inspection

- Systematic procedure – all life-cycle
- Steps
- System of checklists
- Roles
- Forms and reports

Steps in Software Inspection

- Planning,
- Preparation,
- Entry criteria
- Conduct
- Exit criteria
- Reporting
- Follow-up

Steps in Software Inspection

- Inspection Record
- Inspection Reporting Form
- Report Summary Form

Steps in Software Inspection

- Moderator
- Producer
- Reader
- Reviewer
- Recorder
- Manager
- Consumer

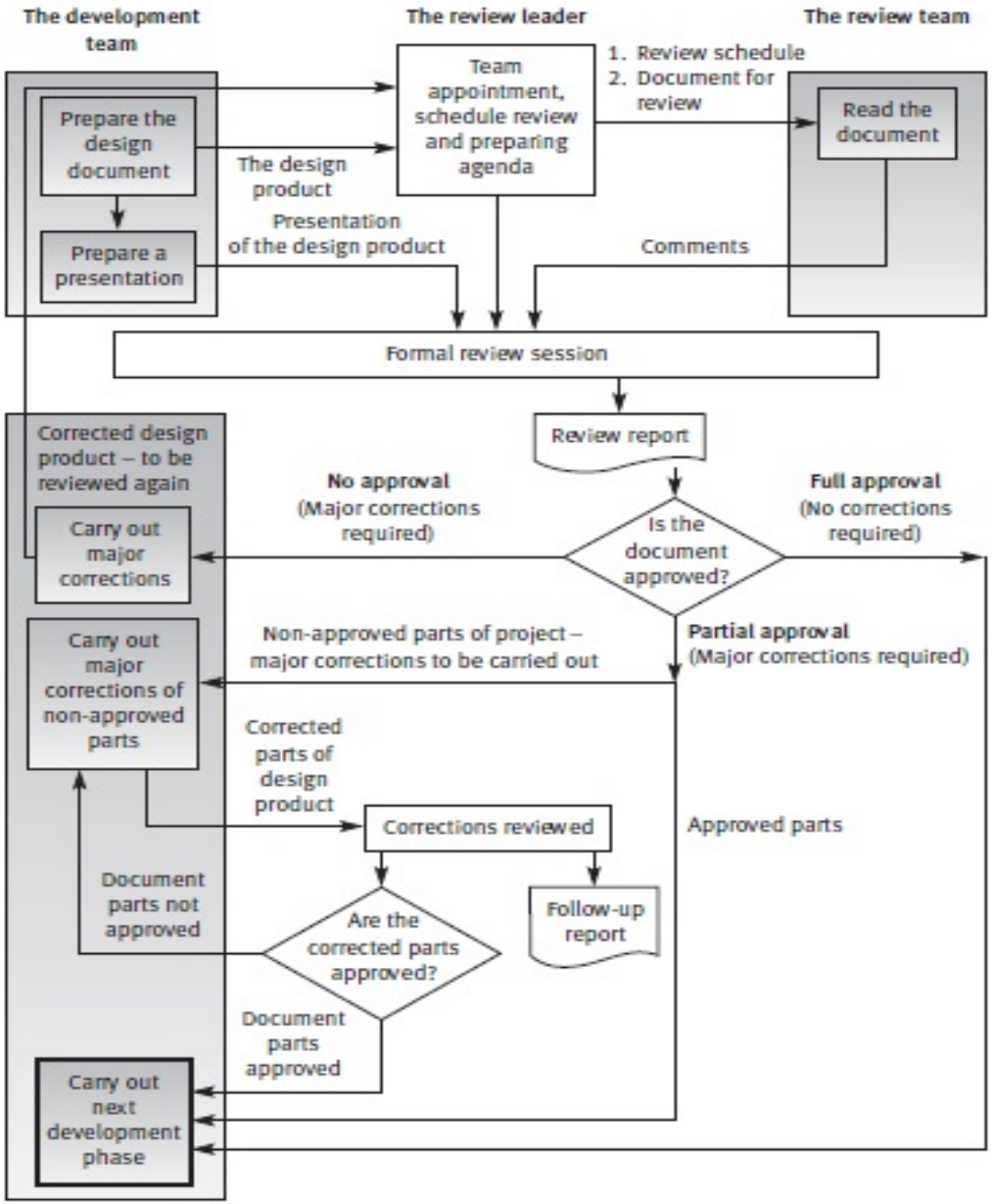
- Checklist
  - For: requirements, architecture, specification, design, code, test procedure
  - Contains: completeness, correctness, style, metrics, rules of constructions, multiple views
- example

Inspection Reporting Form

Issue no.	Line/Page	Checklist	Defect Category	Defect Type	. . .

Report Summary Form

	Major			Minor		
Defect Type	Missing	Wrong	Extra	Missing	Wrong	Extra
Interface						
Logic						
I/O						
…						
Functionality						
Maintainability						



Review process diagram (from Galin-SQA)

SQA?

- Inspections & walkthroughs — finds 30-70% of
  - logic design errors
  - coding errors
- Inspection - IBM reported an 83% defect detection rate

Inspection vs. Walkthrough

- formal review used to verify that the artifact complies with the standard of excellence
- serve the needs of quality management in verifying that the software artifact complies with the standard of excellence
- Used as an exit criterion
- Uses a structured review process
- Involves several roles

- informal review used to confirm the understanding of the producer
- serve the needs of the producer or author
- may be several walkthroughs in each life-cycle activity.
- yield open issues and action items

Inspection vs. Testing

- Issues related to non-functional properties: Maintainability, evolvability, reusability
- Properties difficult to test: Scalability, efficiency, security, integrity, robustness, reliability, exception handling
- Artifacts: requirements, architecture, design documents (cannot “execute” as tests)

Inspection

Benefits:

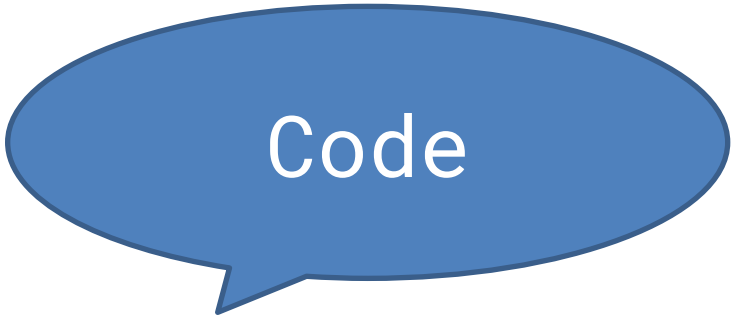
- Knowledge sharing
- Find flaws early
- Better communication: feedback

Drawbacks:

- Why fix? Why walkthrough code? / The reviewer will find it
- Used for HR evaluation

Code review

- Definition: an integral process of software development that helps identify bugs and defects before the testing phase
- Human / automated



Code review vs. inspection

- No difference – some authors
- Inspection: issues not detected by code review
- Automated code review: no human feedback
- Inspection can use code review