Inspection and Code Review
Inspections & Reviews

→ Types of Inspection

→ Benefits of Inspection
  - Inspection is more cost effective than testing

→ How to conduct an inspection
  - who to invite
  - how to structure it

→ Some tips
Reviews, Inspections, Walkthroughs...

Note: these terms are not widely agreed

- **Formality**
  - **informal**: from meetings over coffee, to regular team meetings
  - **formal**: scheduled meetings, prepared participants, defined agenda, specific format, documented output

- **“Management reviews”**
  - E.g. preliminary design review (PDR), critical design review (CDR), ...
  - Used to provide confidence that the design is sound
  - Attended by management and sponsors (customers)
  - Usually a “dog-and-pony show”

- **“Walkthroughs”**
  - developer technique (usually informal)
  - used by development teams to improve quality of product
  - focus is on finding defects

- **“(Fagan) Inspections”**
  - a process management tool (always formal)
  - used to improve quality of the development process
  - collect defect data to analyze the quality of the process
  - written output is important
  - major role in training junior staff and transferring expertise
Benefits of formal inspection

*Source: Adapted from Blum, 1992, Freedman and Weinberg, 1990, & notes from Philip Johnson.*

→ For applications programming:
  - most reviewed programs run correctly first time
    - compare: 10-50 attempts for test/debug approach

→ Data from large projects
  - Data from Bell-Northern Research:
    - Inspection cost: 1 hour per defect.
    - Testing cost: 2-4 hours per defect.
    - Post-release cost: 33 hours per defect.
  - error reduction by a factor of 5; (10 in some reported cases)
  - improvement in productivity: 14% to 25%
  - percentage of errors found by inspection: 58% to 82%
  - cost reduction of 50%-80% for V&V (even including cost of inspection)

→ Effects on staff competence:
  - increased morale, reduced turnover
  - better estimation and scheduling (more knowledge about defect profiles)
  - better management recognition of staff ability
Constraints


→ Size
- “enough people so that all the relevant expertise is available”
- min: 3 (4 if author is present)
- max: 7 (less if leader is inexperienced)

→ Scope
- focus on small part of a design, not the whole thing
- Fagan recommends rates:
  - 130-150 SLOC per hour

→ Duration
- never more than 2 hours
  - concentration will flag if longer

→ Outputs
- all reviewers must agree on the result
  - accept or re-work or re-inspect
- all findings should be documented
  - summary report (for management)
  - detailed list of issues

→ Timing
- Examines a product once its author has finished it
- not too soon
  - product not ready - find problems the author is already aware of
- not too late
  - product in use - errors are now very costly to fix
Choosing Reviewers

Source: Adapted from Freedman and Weinberg, 1990.

→ Possibilities

✧ specialists in reviewing (e.g. QA people)
✧ people from the same team as the author
✧ people invited for specialist expertise
✧ people with an interest in the product
✧ visitors who have something to contribute
✧ people from other parts of the organization

→ Exclude

✧ anyone responsible for reviewing the author
  ➢ i.e. line manager, appraiser, etc.
✧ anyone with known personality clashes with other reviewers
✧ anyone who is not qualified to contribute
✧ all management
✧ anyone whose presence creates a conflict of interest
Roles

Source: Adapted from Blum, 1992, pp369-373

Formal Walkthrough

→ Review Leader
  - chairs the meeting
  - ensures preparation is done
  - keeps review focussed
  - reports the results

→ Recorder
  - keeps track of issues raised

→ Reader
  - summarizes the product piece by piece during the review

→ Author
  - should actively participate (may be the reader)

→ Other Reviewers
  - task is to find and report issues

Fagan Inspection

→ Moderator
  - must be a competent programmer
  - should be specially trained
  - could be from another project

→ Designer
  - programmer who produced the design being inspected

→ Coder/Implementor
  - programmer responsible for translating the design to code

→ Tester
  - person responsible for writing/executing test cases
Guidelines

*Source: Adapted from Freedman and Weinberg, 1990.*

→ Prior to the review
  - schedule Formal Reviews into the project planning
  - train all reviewers
  - ensure all attendees prepare in advance

→ During the review
  - review the product, not its author
    - keep comments constructive, professional and task-focused
  - stick to the agenda
    - leader must prevent drift
  - limit debate and rebuttal
    - record issues for later discussion/resolution
  - identify problems but don’t try to solve them
  - take written notes

→ After the review
  - review the review process
Structuring the inspection

→ Checklist
  - uses a checklist of questions/issues
  - review structured by issue on the list

→ Walkthrough
  - one person presents the product step-by-step
  - review is structured by the product

→ Round Robin
  - each reviewer in turn gets to raise an issue
  - review is structured by the review team

→ Speed Review
  - each reviewer gets 3 minutes to review a chunk, then passes to the next person
  - good for assessing comprehensibility!
Tactics for problematic review meetings

→ Devil’s advocate
  - deliberate attempt to adopt a contrary position

→ Bebugging
  - put some deliberate errors in before the review
    - with prizes for finding them!

→ Money bowl
  - if a reviewer speaks out of turn, he/she puts 25c into the drinks kitty

→ Alarm
  - use a timer to limit ‘speechifying’

→ Issues blackboard
  - appoint someone to keep an issues list, to be written up after the review

→ Stand-up review
  - no tables or chairs!