MTAT.03.159: Software Testing

Lecture 07:
Defect Estimation, Test Documentation, Organisation and Process Improvement
Exam Preparation

Spring 2018

Dietmar Pfahl
email: dietmar.pfahl@ut.ee
Structure of Lecture 07

• Defect Estimation
• Test Planning & Documentation
• Test Organisation
• Test Process Improvement (TMMi)
• Exam Preparation
Structure of Lecture 07

• Defect Estimation
• Test Planning & Documentation
• Test Organisation
• Test Process Improvement (TMMi)
• Exam Preparation
Quality Prediction

• Based on product and process properties
• Examples:
  • Quality = Function(Code Size or Complexity)
  • Quality = Function(Code Changes)
  • Quality = Function(Test Effort)
  • Quality = Function(Detected #Defects)
• Reliability Growth Models
• Capture-Recapture Models

Quality defined as: Undetected #Defects
Capture-Recapture – Defect Estimation
Capture-Recapture – Defect Estimation
Capture-Recapture – Defect Estimation

- Situation: Two inspectors are assigned to inspect the same product
  - $d_1$: #defects detected by Inspector 1
  - $d_2$: #defects detected by Inspector 2
  - $d_{12}$: #defects by both inspectors
  - $N_t$: total #defects (detected and undetected)
  - $N_r$: remaining #defects (undetected)

\[
N_t = \frac{d_1 d_2}{d_{12}} \quad \quad \quad N_r = N_t - (d_1 + d_2 - d_{12})
\]
Capture-Recapture – Example

- Situation: Two inspectors are assigned to inspect the same product
  - $d_1$: 50 defects detected by Inspector 1
  - $d_2$: 40 defects detected by Inspector 2
  - $d_{12}$: 20 defects by both inspectors
  - $N_t$: total defects (detected and undetected)
  - $N_r$: remaining defects (undetected)

\[
N_t = \frac{d_1 d_2}{d_{12}} = \frac{50 \cdot 40}{20} = 100 \quad N_r = 100 - (50 + 40 - 20) = 30
\]
Advanced Capture-Recapture Models

- Four basic models used for inspections
  - Difference: Degrees of freedom

- Prerequisites for all models
  - All reviewers work independently of each other
  - It is not allowed to inject or remove faults during inspection

Task in Lab 7
## Advanced Capture-Recapture Models

<table>
<thead>
<tr>
<th>Model</th>
<th>Probability of defect being found is equal across ...</th>
<th>Estimator</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Defect</td>
<td>Reviewer</td>
</tr>
<tr>
<td>M0</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Mt</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mh</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mth</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>
## Advanced Capture-Recapture Models

<table>
<thead>
<tr>
<th>Model</th>
<th>Probability of defect being found is equal across ...</th>
<th>Estimator</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Defect</td>
<td>Reviewer</td>
</tr>
<tr>
<td>M0</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Mt</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mh</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mth</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>
Mt Model

Maximum-likelihood:

- Mt = total marked animals (=faults) at the start of the t'th sampling interval
- Ct = total number of animals (=faults) sampled during interval t
- Rt = number of recaptures in the sample Ct
- An approximation of the maximum likelihood estimate of population size (N) is: \(\text{SUM}(Ct \times Mt) / \text{SUM}(Rt)\)

First resampling:
- M1 = 5 (first reviewer)
- C1 = 4 (second reviewer)
- R1 = 2 (duplicates)
- N = 4 \times 5 / 2 = 10

Second resampling:
- M2 = 7 (first and second reviewer)
- C2 = 4 (third reviewer)
- R2 = 3 (duplicates)
- N = (4 \times 5 + 4 \times 7) / (2 + 3) = 48 / 5 = 9.6

Third resampling:
- M3 = 8 (first, second and third reviewer)
- C3 = 3 (fourth reviewer)
- R3 = 3 (duplicates)
- N = (20 + 28 + 3 \times 8) / (2 + 3 + 3) = 72 / 8 = 9
Example:
3 Reviewers
Example:

Maximum-likelihood:

- \( Mt \) = total marked animals (=faults) at the start of the \( t \)'th sampling interval
- \( Ct \) = total number of animals (=faults) sampled during interval \( t \)
- \( Rt \) = number of recaptures in the sample \( Ct \)
- An approximation of the maximum likelihood estimate of population size \( (N) \) is: \( \frac{\text{SUM}(Ct\times Mt)}{\text{SUM}(Rt)} \)

First resampling:
- \( M1 = 11 \) (first reviewer)
- \( C1 = 9 \) (second reviewer)
- \( R1 = 6 \) (duplicates)
- \( N = \frac{11 \times 9}{6} = 16.5 \rightarrow 2.5 \) not found

Second resampling:
- \( M2 = 14 \) (first and second reviewer)
- \( C2 = 11 \) (third reviewer)
- \( R2 = 10 \) (duplicates)
- \( N = \frac{(11 \times 9 + 14 \times 11)}{6 + 10} = \frac{253}{16} = 15.8 \rightarrow 0.8 \) not found
Lab 7 – Document Inspection & Defect Prediction

Lab 7 (week 31: May 15 – May 16) - Document Inspection and Defect Prediction (10%)

Lab 7 Instructions
Lab 4 & Sample Documentation

Submission Deadlines:

- Tuesday Labs: Monday, 21 May, 23:59
- Wednesday Labs: Tuesday, 22 May, 23:59

- Penalties apply for late delivery: 50% penalty, if submitted up to 24 hours late; 100 penalty, if submitted more than 24 hours late

Documentation:
- Requirements List (User Stories)
- Specification
  - 2 Screens
  - 1 Text
Lab 7 – Document Inspection & Defect Prediction (cont’d)

**Phase A: Individual student work**

- **Instructions**
- **Requirements** (6 User Stories)
- Inspection of Specification against Requirements
  - Specification (excerpt)
    - 2 screens & Text
  - Issue List (at least 8 defects)

**Phase B: Pair work**

- **Issue List Student 1**
- **Issue List Student 2**
- **Student Pair**
  - Consolidated Issue List
  - Remaining Defects Estimation

---

**Table columns:**
- ID, Description, Location, Type, Severity
Lab 7 – Document Inspection & Defect Prediction (cont’d)

Lab 7: Must work in pairs to be able to get full marks!
Structure of Lecture 07

• Defect Estimation
• Test Planning & Documentation
• Test Organisation
• Test Process Improvement (TMMi)
• Exam Preparation
Test Planning

- Objectives
- What to test
- Who will test
- When to test
- How to test
- When to stop

Elective course (Fall’18):
Hands-on SW Testing
MTAT.03.294
FIG. 7.4

IEEE 829-2008: Standard for Software and System Test Documentation
Hierarchy of Test Plans

Test Plan Components

1. Test plan identifier
2. Introduction
3. Items to be tested
4. Features to be tested
5. Approach
6. Pass/fail criteria
7. Suspension and resumption criteria
8. Test deliverables
9. Testing Tasks
10. Test environment
11. Responsibilities
12. Staffing and training needs
13. Scheduling
14. Risks and contingencies
15. Testing costs
16. Approvals

Software quality assurance (V&V) plan

Master test plan

Review plan: Inspections and walkthroughs

Unit test plan
Integration test plan
System test plan
Acceptance test plan
Test plan according to IEEE Std 829-2008 (Appendix II)

a) Test plan identifier
b) Introduction
c) Test items
d) Features to be tested
e) Features not to be tested
f) Approach
g) Item pass/fail criteria
h) Suspension criteria and resumption requirements
i) Test deliverables
j) Testing tasks
k) Environmental needs
l) Responsibilities
m) Staffing and training needs
n) Schedule
o) Risks and contingencies
p) Approvals
Test Plan (1)

a) Test plan identifier
b) Introduction
   – Product to be tested, objectives, scope of the test plan
   – Software items and features to be tested
   – References to project authorization, project plan, QA plan, CM plan, relevant policies & standards

c) Test items
   – Test items including version/revision level
   – Items include end-user documentation
   – Defect fixes
   – How transmitted to testing
   – References to software documentation
Test Plan (2)

d) Features to be tested
- Identify test design / specification techniques
- Reference requirements or other specs

e) Features not to be tested
- Deferred features, environment combinations, …
- Reasons for exclusion

f) Approach
- How you are going to test this system
  - Activities, techniques and tools
- Detailed enough to estimate
- Completion criteria (e.g. coverage, reliability)
- Identify constraints (environment, staff, deadlines)
Test Plan (3)

g) Item pass/fail criteria
   – What constitutes success of the testing
   – Coverage, failure count, failure rate, number of executed tests, …
   – Is NOT product release criteria

h) Suspension and resumption criteria
   – For all or parts of testing activities
   – Which activities must be repeated on resumption

i) Test deliverables
   – Test plan
   – Test design specification, Test case specification
   – Test procedure specification, Test item transmittal report
   – Test logs, Test incident reports, Test summary reports
Test Plan (4)

j) Testing tasks
   – Including inter-task dependencies & special skills
   – Estimates

k) Environment
   – Physical, hardware, software, tools
   – Mode of usage, security, office space
   – Test environment set-up

l) Responsibilities
   – To manage, design, prepare, execute, witness, check, resolve issues, providing environment, providing the software to test

m) Staffing and Training needs
Test Plan (5)

n) Schedule
   – Test milestones in project schedule
   – Item transmittal milestones
   – Additional test milestones (environment ready)
   – What resources are needed when

o) Risks and Contingencies
   – Testing project risks
   – Contingency and mitigation plan for each identified risk

p) Approvals
   – Names and when approved
Test Case Specification – Why?

• Organization
  – All testers and other project team members can review and use test cases effectively

• Repeatability
  – Know what test cases were last run and how so that you could repeat the same tests

• Tracking
  – What requirements or features are tested?
  – Tracking information’s value depends on the quality of the test cases

• Evidence of testing
  – Confidence (quality)
  – Detect failures
Defect Report (Test incidence report)

• Summary
• Incident Description
• Impact
Defect Report (Test incidence report)

Summary

• This is a summation/description of the actual incident.
  – Provides enough details to enable others to understand how the incident was discovered and any relevant supporting information

• References to:
  – Test Procedure used to discover the incident
  – Test Case Specifications that will provide the information to repeat the incident
  – Test logs showing the actual execution of the test cases and procedures
  – Any other supporting materials, trace logs, memory dumps/maps etc.
Defect Report (Test incidence report)

Incident Description

- Provides as much details on the incident as possible.
  - Especially if there are no other references to describe the incident.
- Includes all relevant information that has not already been included in the incident summary information or any additional supporting information

Information:
- Inputs
- Expected Results
- Actual Results
- Anomalies
- Date and Time
- Procedure Step
- Attempts to Repeat
- Testers
- Observers
Defect Report
(Test incidence report)

Impact

• Describe the actual/potential damage caused by the incident.
  – Severity
  – Priority

• Severity and Priority need to be defined so as to ensure consistent use and interpretation, for example:
  – Severity – The potential impact to the system
    – Mission Critical - Application will not function or system fails
    – Major - Severe problems but possible to work around
    – Minor – Does not impact the functionality or usability of the process but is not according to requirements/design specifications
  – Priority – The order in which the incidents are to be addressed
    – Immediate – Must be fixed as soon as possible
    – Delayed – System is usable but incident must be fixed prior to next level of test or shipment
    – Deferred – Defect can be left in if necessary due to time or costs
Test results report

- Test cases executed
- Versions tested
- Defects found and reported

Figure 6. Defect distributions, showing number of defects detected over time, for the three studied projects. Ship dates indicated.
Standards

- IEEE 829-2008  
  Standard for Software Test Documentation
- IEEE 1008-1993  
  Standard for Software Unit Testing
- IEEE 1012-2012  
  Standard for System and Software Verification and Validation

->

- ISO/IEC/IEEE 29119 Software Testing (5 parts)  
  – replaces most of the older standards
Structure of Lecture 07

• Defect Estimation
• Test Planning & Documentation
• Test Organisation
• Test Process Improvement (TMMi)
• Exam Preparation
7 approaches to test organisation

1. Each person’s responsibility
2. Each unit’s responsibility
3. Dedicated resource
4. Test organisation in QA
5. Test organisation in development
6. Centralized test organisation
7. Test technology centre

[Kit, Software Testing in the Real World Ch 13, 1995]
7 approaches to test organisation

1. Each person’s responsibility
2. Each unit’s responsibility
3. Dedicated resource
4. Test organisation in QA
5. Test organisation in development
6. Centralized test organisation
7. Test technology centre

[Kit, Software Testing in the Real World Ch 13, 1995]
Which organization should we choose?

• Depending on
  – size
  – maturity
  – focus

• The solution is often a mix of different approaches
Watch James Bach’s open lecture video (course wiki)!
Structure of Lecture 07

• Defect Estimation
• Test Planning & Documentation
• Test Organisation
• Test Process Improvement (TMMi)
• Exam Preparation
Process quality and product quality

- Quality in process
  Quality in product
- Project:
  - instantiated process
- ISO 25000:
  - *Process quality* contributes to improving *product quality*, which in turn contributes to improving *quality in use*
Process improvement models vs Test Process improvement models

- (Integrated) Capability maturity model (CMM, CMMI)
- Software process improvement and capability determination (SPICE)
- ISO 9001, Bootstrap, …

Test Process Improvement Models:
- Test maturity model (TMM, TMMi)
- Test process improvement model (TPI)
- Test improvement model (TIM)
- Minimal Test Practice Framework (MTPF)
- …
Test Maturity Model (TMM)

- Levels
- Maturity goals and sub-goals
  - Scope, boundaries, accomplishments
  - Activities, tasks, responsibilities
- Assessment model
  - Maturity goals
  - Assessment guidelines
  - Assessment procedure
Level 2: Phase Definition

- Institutionalize basic testing techniques and methods
- Initiate a test planning process
- Develop testing & debugging tools
Level 3: Integration

- Control and monitor the testing process
- Integrate testing into software lifecycle
- Establish a technical training program
- Establish a software test organization
Level 4: Management and Measurement

- Software quality evaluation
- Establish a test management program
- Establish an organization-wide review program
Level 5: Optimizing, Defect Prevention, and Quality Control

- Test process optimization
- Quality control
- Application of process data for defect prevention
Recommended Textbook Exercises

- Chapter 14
  - 2, 4, 5, 6, 9
- Chapter 9
  - 2, 3, 4, 5, 8, 12
- Chapter 7
  - 2, 3, 6, 8, 9, 11
- Chapter 8
  - 2, 3, 6, 7, 9
- Chapter 16
  - No exercises
Structure of Lecture 07

- Defect Estimation
- Test Planning & Documentation
- Test Organisation
- Test Process Improvement (TMMi)
- Exam Preparation
Exam Dates

• Exam 1: Fri 25-May, 14:15-15:55, room 405 – max. 49 students
• Exam 2: Mon 28-May, 12:15-13:55, room 405 – max. 49 students

You must receive

... at least 24 marks from the homework assignments (labs 1-7) to qualify for the exam and

... at least 7.5 marks in the exam to not fail the course.

In total, you need at least 50 marks to not fail the course.

• Retake Exam (resit): 13-June, 10:15-11:55 (J. Liivi 2-611)
  – Please note that you must register for the retake exam at the latest 3 days before the exam date
Important

• You are registered for either exam 1 or exam 2
• You must go to the exam for which you are registered – otherwise you will not be admitted

• Exam 1: Friday, May 25
• Exam 2: Monday, May 28

Check it out in the SIS
Important

• The exam duration is 100 min!
• In the SIS, the reserved time slots are 120 min to accommodate administrative tasks such as distribution and collection of exam materials.
• In order to get started in time, please arrive a few minutes before the exam starts.
Final Exam

Admission Rule:
• *At least 24 marks from labs 1-7*

Minimum required to not fail the course:
• *At least 24 marks from labs AND*
• *At least 7.5 marks in exam AND*
• *At least 50 marks overall*
Questions ?
Final Exam – Format

• Written exam (30% = 30 marks)
  – Based on textbook, lectures and lab sessions
  – Open book
  – 100 min
  – 2 Parts:
    • Part 1 – Multiple-Choice (8 marks)
    • Part 2 – Constructive Tasks (22 marks)
      – Answers might require some technical work & calculation
Questions ?
Final Exam – Content/Topics Overview

• Introduction to Software Testing
• Black-Box Testing Techniques
• White-Box Testing Techniques
• Mutation Testing
• Static Testing (Inspection/Review) and Defect Estimation
• Test Tools (and Measures)
• Test Documentation, Organisation, Improvement
Final Exam – Content/Topics Overview

Introduction to Software Testing:
• Know the basic terminology
  – Software Testing & Software Quality
  – Verification & Validation
  – Error – Fault – Failure
    • NB: Two competing definitions of ’Error’
  – Test Case – Test Suite – Test Oracle ... 
  – Test Levels
  – Issue Reporting
  – Debugging
Final Exam – Content/Topics Overview

Black-Box Testing Techniques:

- Difference between Black-Box and White-Box Testing
  - Strengths & Weaknesses of each
- Know various BBT Techniques (Methods):
  - Equivalence Class Partitioning
  - Boundary Value Testing
  - Combinatorial Testing
  - State-Transition-Testing
  - Exploratory Testing
  - Usability Testing
Final Exam – Content/Topics Overview

White-Box Testing Techniques:
• Difference between Black-Box and White-Box Testing
  – Strengths & Weaknesses of each
• Control-Flow Testing
  – Know how to construct a Control-Flow-Graph
  – Know different coverage criteria:
    • Statement/Block, Decision/Branch, Condition, Linearly Independent Paths, etc.
• Data-Flow Testing
• Mutation Testing
Final Exam – Content/Topics Overview

Static Testing (Inspection) and Defect Estimation:

• Document Reviews (Inspections)
  – Why needed?
  – What variants exist?
• Static Code Analysis
  – What are false positives?
• Defect Estimation:
  – Capture-Recapture Model
Final Exam – Content/Topics Overview

Test Lifecycle:
• Agile Testing
• Specifics of Testing OO Code
  – Intra-Class Testing (’Stack’ Example)
  – Inter-Class Testing
  – State-Transition-Testing
• System versus Unit Testing
• Regression Testing
Final Exam – Content/Topics Overview

Tools (and Measures):
• Different Tools for Different Purposes!
• Test Automation
  – Capture-Replay (Web App Testing)
  – Automated Regression Testing
• Test Measures
  – Test Coverage, Test Effectiveness, Test Efficiency, etc.
Questions ?
Next Weeks

• Submit Lab 6 (Tuesday Lab Groups) on time
• Attend and do Lab 7
  – Must work in Pairs to be able to get full marks

• Prepare for Exam
• Take Exam
Fall 2018: Course “Hands-on Software Testing” (MTAT.03.294)

Schedule:

• 21.09.2018 Session 1. Introduction to testing. Oracles, heuristics and their limitations.
  – Attendance of this session is compulsory.
• 12.10.2018 Session 3. Test Design.
• 09.11.2018 Session 5. Bug Advocacy and Reporting.
• 07.12.2018 Session 7. Presentations of work done. Conclusive exercise which covers all topics of previous sessions.
  – Attendance of this session is compulsory.
Fall 2018: Course “Hands-on Software Testing” (MTAT.03.294)

Teachers are experts from industry (Estonian companies)
Limit: 20 students (first come first serve)

Coordinator: Oliver Vilson
(oliver.vilson@hannas.ee)
Thank You!