Model-Based Testing

(DIT848 / DAT260)

Spring 2014

Gerardo Schneider
Dept. of Computer Science and Engineering
Chalmers | University of Gothenburg

gerardo@cse.gu.se
http://www.cse.chalmers.se/~gersch/

Model-Based Testing

- What is testing?
  - The process of systematically experimenting with an object in order to establish its quality
  - Object "=" software -> software testing
- Why software testing?
  - Most used technique in industry to increase confidence in Sw quality
  - Job possibilities ☺
- What is model-based testing?
  - Generate tests (semi-)automatically from the model of the system under test
- Why model-based testing?
  - Cost saving, systematic approach to testing, automated traceability, early detection of flaws, etc.

Overview course content

- Overview on verification and validation
- Testing in general
  - Black box testing (Junit)
  - White box testing (Coverage analysis)
- Property-Based testing
  - QuickCheck
- FSM / EFSM
- Model-based testing
  - How to select your tests
  - Graph theory in MBT
  - ModelJUnit
  - Making your tests executably

How much of each topic?

- We will discuss today

Guest lectures:

- QuickCheck
- …

Theory and practice

Learning Outcomes

- Describe the distinction between software verification and validation;
- Describe the connection between software development phases and kinds of testing;
- Describe and explain (a number of) different test methods, and use them in practical situations;
- Describe and explain what model-based testing is;
- Construct models in the modeling and specification languages learned in the course;
- Construct appropriate and meaningful test cases, and interpret and explain (to stakeholders) the results of the application of such test cases (using appropriate tools) to practical examples;
- Apply model-based testing on realistic examples;
- Exemplify and describe tools for testing software, and use them and interpret their output;
- Identify and hypothesize about sources of program failures, and reflect on how to better verify the correctness of such programs.

Staff

- Gerardo Schneider – gerardo@cse.gu.se

- Guest Lecturers (on QuickCheck)
  - Koen Claessen (Chalmers)
  - John Hughes (Chalmers / QuviQ) - TBC

- Course assistant
  - Grégoire Détrez – gregoire.detrez@cse.gu.se

Student representatives

- TBA

Course organization

- 15 lectures including guest lectures
  - In modules of approx. 45 min
- 4 assignments
  - **Mandatory!**
  - Meeting with the assistant on a (bi-) weekly basis
- All the information on the course page
  - http://www.cse.chalmers.se/edu/year/2014/course/DAT260/
  - Considered official! (Any message will be written in the News section under Home)
    - Students from Chalmers and GU – different systems
- Written exam

Schedule lectures

- "Theoretical" interactive lectures and exercise (assignments) supervision / consultation

- The Monday slot is 08.15-12.00
  - Each lecture will in general be approx. 2hs
  - Some days the lectures will start a bit later (9.00, 9.15?) – Check the homepage regularly!

Literature

- M. Utting and B. Legeard, **Practical Model-Based Testing**. Elsevier (Morgan Kaufmann Publishers, 2007)
  - An electronic version is available at http://bit.ly/wGlT94 (you must be logged in Chalmers' network to get access)
- Other interesting books
  - P.C. Jorgensen. **Software Testing: A Craftsman's Approach** (Auerbach Publications, 3rd edition, 2008)
  - Sommerville…
- Papers on QuickCheck
  - See course homepage
- Other books and references
  - See list in course homepage

"Weekly" assignments

- Not fixed days – Check homepage ("Lectures" tab)
  - Assignment consultation meetings after the topic has been covered in the theoretical lectures
- **Mandatory!**
- The assistant will give you feedback on your assignments
  - You will get information on how to submit
  - The assistant will let you know about submission "deadlines"
- Take feedback seriously
  - It's part of your learning
- If recurrent questions/problems with assignments - discussion during the lectures
  - If you have questions/doubts about the exercises be sure you ask the assistant during the consultation meetings

Exam

- Written exam: May 28, 2014 (morning)
- Re-exam: August 25, 2014 (afternoon)

Important!

- The exam is designed to increase the confidence that a student passing the course achieve the Intended Learning Outcomes
- **Strongly recommended to learn when you work on the assignments!**
- So, *most probably* the exam will consist in 5 tasks widely covering the content of the course
- You will need to have at least **50**/100 points for getting ***G* (3)** (and at least **65** points for **4**) **and at least a minimum of correct answers for each task** (*e.g.*, **8 points** per task)
- To get ***VG* (5)** you will need to have at least **80**/100 points **and at least a minimum of correct answers for each task** (*e.g.*, **12 points** per task)
- *Open book* exam modality

Changes wrt last year
Result of course evaluation

- 7 assignments - explicitly asked to have less and more complex, and more time to work on them
  - 4 assignments now – some complexity added
  - Lectures have been reorganized so all the theory is there before starting working on assignments
- Doing assignments were essential for passing the exam but were not mandatory
  - Assignments are mandatory now
- Concerns with learning modeling – (E)FSM
  - One more (interactive) lecture on EFSM
- No time to work on QuickCheck for students without background on FP
  - QuickCheck has been moved to second week of lecturing
- Slides using Comic Sans SF (strong concern by 1 student)
  - Have not changed them yet – Something against that?

What is your background?

Number of students: 26 (Chalmers) + YY (GU)

- Knowledge on logic?
  - Propositional (classical): 17 (FOL: 5)
  - Other: 1 (Hoare)
- Which functional prog. lang. do you know?
  - Haskell: 8
  - Erlang: 8
  - Other: 0
- Which imperative/OO prog. lang. do you know?
  - Java: 24
  - C (C++): 21
  - Other: 5 (C# / Python)
- Knowledge on Testing? 13
- Knowledge on automata theory or FSM (Finite State Machines)? 10
- Knowledge on QuickCheck? 5

Preliminary schedule

- Is the content "appropriate" according to your background?
- Remember requirements for the course:
  - General programming knowledge in both *imperative/object oriented* and *functional* programming
  - Knowledge of *propositional logic*
  - Have some experience in *testing/debugging* your own programs

Wish list…

- What are your expectations?
- Something you would like to learn on testing (or verification in general) not covered in the programme?
- Are there topics you already know and don't want to see again?

About Registration…

- If you are a **GU student**
  - You need to register through the *Studieportalen* at GU
  - An exception: students from the *SEM kandidatprogrammet*: You need to ask the Studieexpeditionen (student office) to register you
- For **Chalmer** students (late registration)
  - Contact the *Studieexpeditionen* (student office) student_office.cse@chalmers.se

Questions?

Check the course page regularly

Hope you enjoy the course!