Michael Ritzschke, Klaus Bothe: HU examination questions of JCSE in English

Michael Ritzschke, Klaus Bothe
Humboldt University Berlin

16th Workshop "Software Engineering Education and Reverse Engineering
Jahorina, BiH, 22th – 26th August 2016

Bachelor at HU: Schedule Overview

1. Semester	Fundamentals of Programmierung (OO, Java)		Basics of theoretical informatics				Mathematics 1: Linear Algebra		Second subject
2. Semester	Algorithms and data structures		Compiler construction		Digital systems: basics and architectures		Informatics and Society		Second subject
3. Semester	Software Engineering		Logic in informatics		Seminar		Mathematics 2: Analysis		Second subject
4. Semester	Operating systems		Database systems		Communication systems: basics and network architectures		Mathematics 3: Numerics & Stochastics		Second subject
5. Semester	Semester Project		Elective modules						Second subject
6. Semester	Bachelor thesis		Elective modules						Second subject

Bachelor at HU: Schedule Overview

1. Semester	Fundamentals of Programmierung (OO, Java)		Basics of theoretical informatics						
2. Semester	Algorithms and data structures		Compiler construction		Digita bas arch				
3. Semester	Software Engineering		Logic in informatics		Se				
4. Semester	Operating systems		Database systems		Communication systems: basics and network architectures		Mathematics 3: Numerics & Stochastics		Second subject
5. Semester	Seme Proj				lules				Second subject
6. Semester	Bachelor thesis				Elective modules				Second subject

Organisation: JCSE
- **Lecture: 4 lecture hours / week,**
- **16 weeks (64 lh)**
- **Assignments**

Effort:
8 ECTS

Examination (in German):
- **Written**
- **120 minutes**

Why English version of exam?

- Erasmus student from England at HU (WS 2014/15)
- England ← Egypt ← Lybia
- Very good knowledge of German (for lessons, assignments!)
- Excellent knowledge of English
- Asked for English version of exam
- Her result: among the 3 best students

Written examination

- 120 minutes
- Question types:
  - knowledge
  - multiple-choice

    (special case of knowledge question)
  - application
- 42 questions, 22 pages, 190 points

English version: questions and solutions

- File questions: SE_questions_EN_Feb2015
- File questions with solutions: SE_questions+solutions_EN_Feb2015

Example of a multiple-choice question

11. (6 Points) How do you determine potential classes in the object oriented analysis?

a) Analysis of the requirements specification. □ yes □ no

b) Analysis of the software architecture. □ yes □ no

c) Consideration of design patterns. □ yes □ no

d) Separation of the problem into sub-problems. □ yes □ no

e) Consideration of quality criteria of the software system. □ yes □ no

f) In the case of technical systems: real devices or hardware. □ yes □ no

Solution: [a, d, f]

Example of a knowledge question

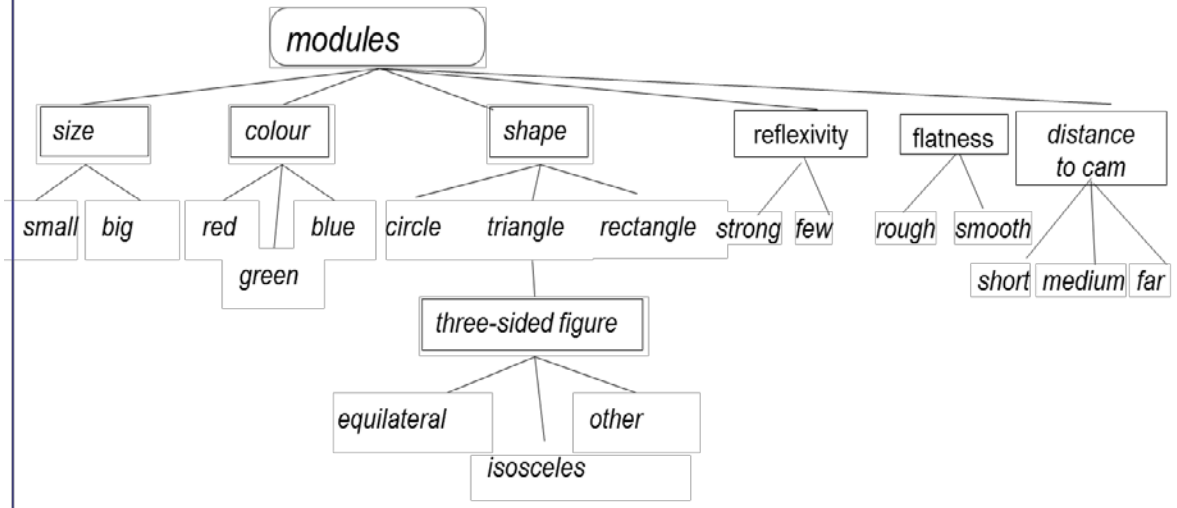
24 d) (2 points) What is the minimal value for the cyclomatic complixty? How does the corresponding program structure look like?

Solution:

- minimal number = 1,
- programs without branches, i.e. straight-forward programs

Example of application question (1)

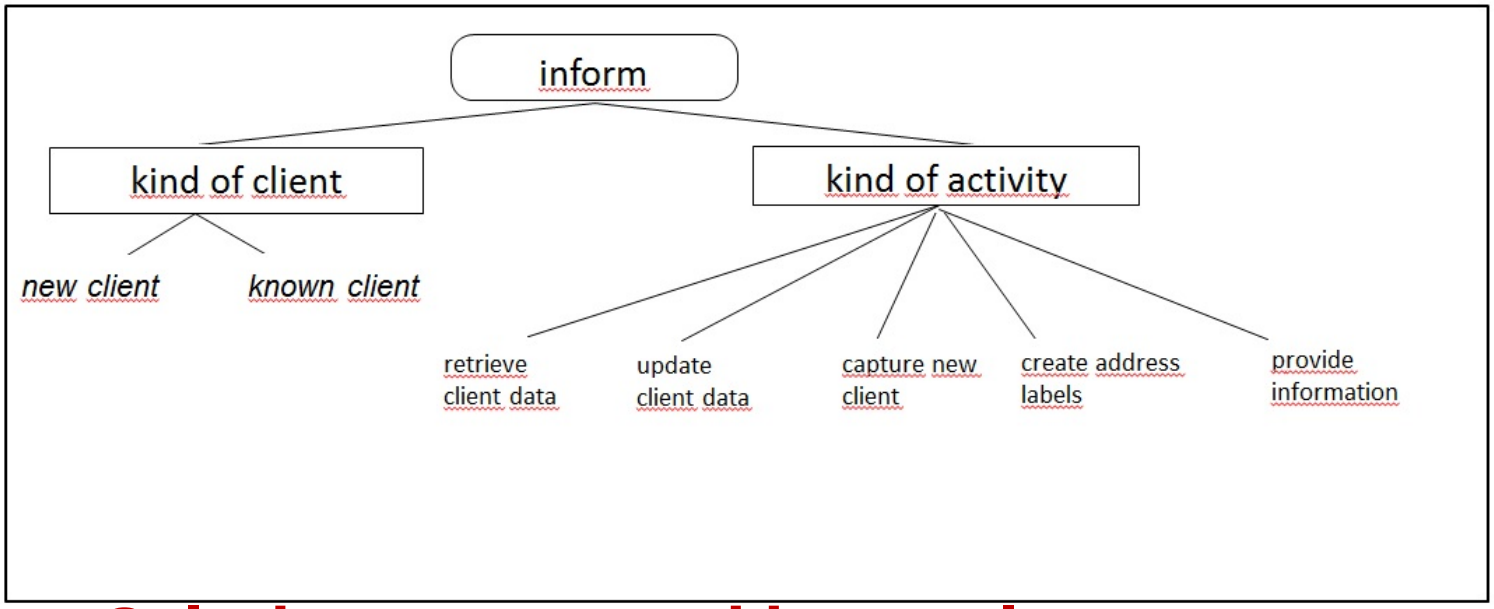
21. (6 Points = 4 + 2)) The following classification tree is given for a software to categorize the input for an image recognition system. Calculate the maximal number and the minimal number (according to the minimality criterion) of test cases for the given tree. How are these values calculated for arbitrary classification trees?



Solution:
Minimal number: 5
because:
Largest subtree: 5 (from 'shape')
Maximal number: 360
= 2 (size) * 3 (colour)
* 5 (shape)
* 2 (Reflexivity)
* 2 (flatness)
* 3 (distance to camera)

Example of application question (2)

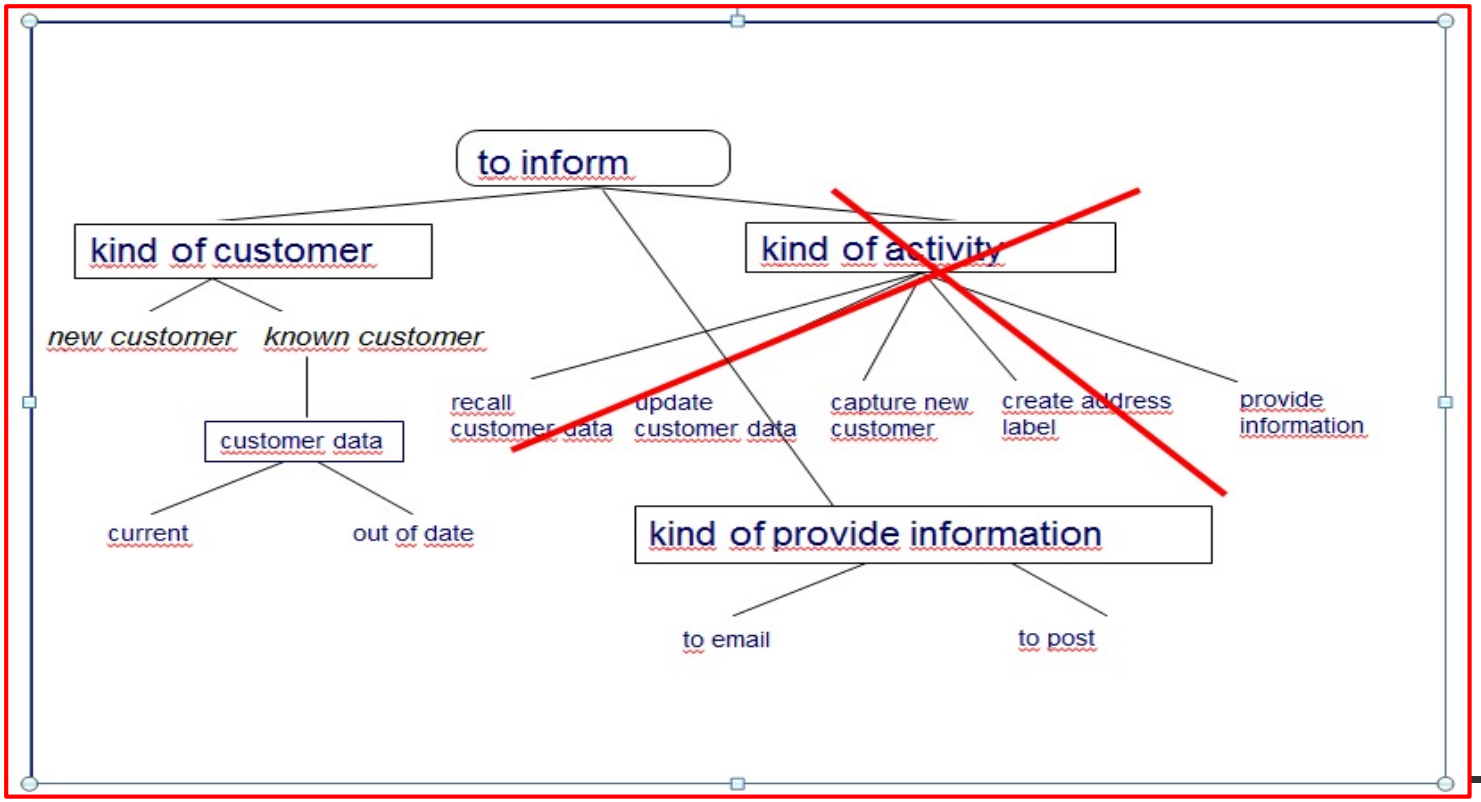
- 22. (8 Points = 4 + 4) Consider the following use case and the corresponding classification tree. Review the classification tree. a) What errors do you notice? b) Draw an improved classification tree overcoming the disadvantages of the given tree from part a)



Solution: errors and incompleteness

Example of application question (2): solution

Solution: errors and incompleteness



Example of application question (3)

27. (15 Points = **3** + 3 + 3 + 3 + 3) Structural Testing (Example)

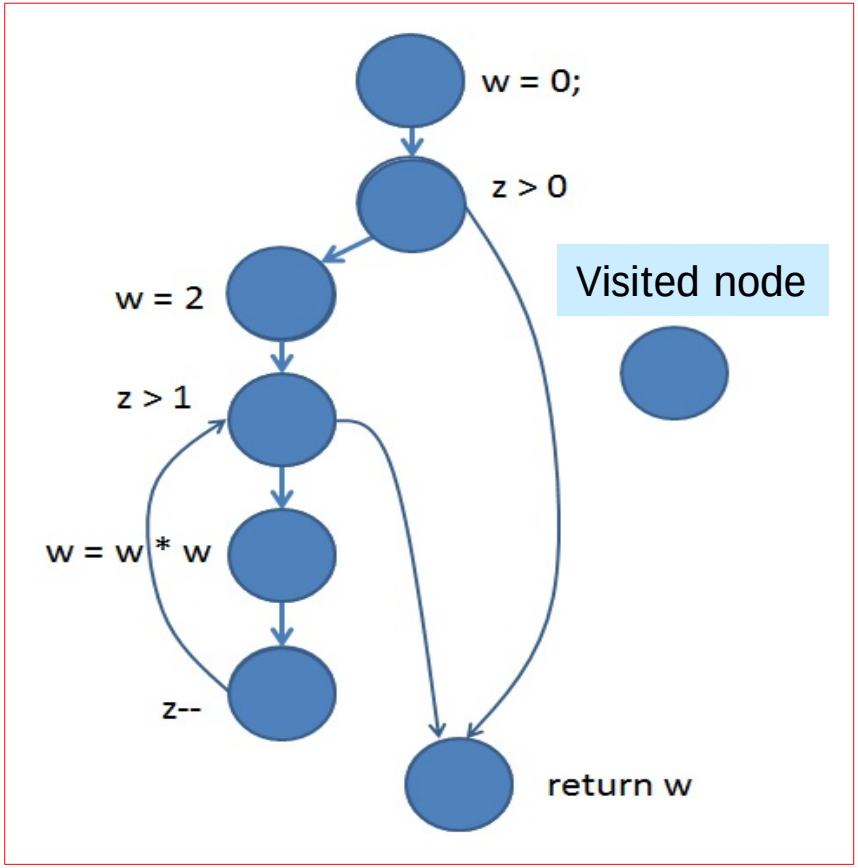
a) For the following Java source code two test cases are introduced:

```
int power(int z) {
      int w = 0;
      if (z > 0) {
            w = 2;
            while (z > 1) {
                  w = w * w;
                  z--;
            }
      }
      return w;
}
```

Variable	z
1. test case	1
2. test case	2

Calculate the statement coverage (C_0) for the given set of test cases.
Please explain your answer (mark the visited nodes in the control flow graph).

Example of application question (3): solution



Solution: 100 %
- all nodes visited

English version: questions and solutions

- File questions: SE_EN_exam-questions_Feb2015
- File questions with solutions: SE_questions+solutions_EN_Feb2015

→ **Not at workshop website**

→ **If interested: available**

Thank you for attention