
Software Engineering

<http://proglang.informatik.uni-freiburg.de/teaching/swt/2005/>

Exercise Sheet 5 & 6

Deadline: May 31st, 2005

Exercise 1 – Activity Diagramm: (4 points)

Describe the following recipe with an Activity Diagram:

Cut 400g chicken into small dices. Mix 30ml rice wine with 10ml soy sauce. Mix this sauce with starch. If no starch is available, use flour. Mix the chicken with the sauce. Cut 10g ginger and 50g field garlic. Sear 150g walnuts. Sear the meat, add ginger and field garlic and fry the meat. Add the walnuts and 10ml soy sauce.

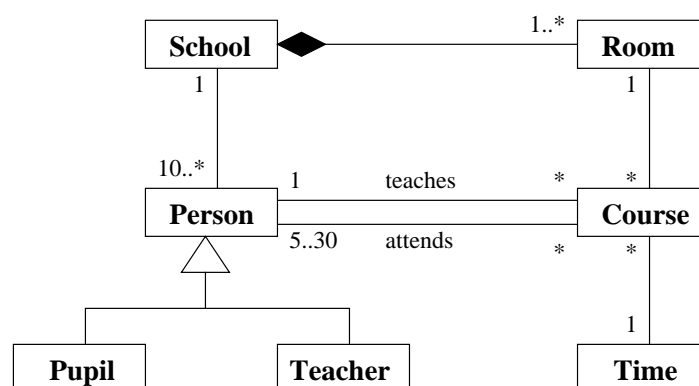
Exercise 2 – Data Dictionary: (3 points)

Write a Data Dictionary for German addresses.

Regard the different possibilities, like P.O. box or street, additional room numbers, etc.

Exercise 3 – Class Diagram: (5 points)

Analyse the following class diagram, explaining in detail what it models. Are there any flaws in this design?


Exercise 4 – Understanding Z: (4 points)

What is defined by the following (generic) axiomatic definitions and what are the missing types?

$$\frac{\text{walter} : ???}{\forall M : \mathbb{P}\mathbb{N}, x : \mathbb{N} \bullet \\ \text{walter}(\emptyset) = 0 \\ x \in M \Rightarrow \text{walter}(M) = \\ x + \text{walter}(M \setminus \{x\})}$$

$$\frac{[X]}{\frac{xaver : ???}{\forall s : \mathbb{F}X, n : \mathbb{N} \bullet \\ n = xaver \Leftrightarrow \exists f : (1..n) \twoheadrightarrow s \bullet true}}$$

$$\frac{\text{yolanda} : ???}{\forall m, n : \mathbb{N} \bullet \\ \text{yolanda}(m, n) = \{i : \mathbb{N} \mid m \leq i \leq n\}}$$

$$\frac{[X]}{\frac{zoe : ???}{\forall R : X \leftrightarrow X \bullet \\ zoe(R) = \bigcap \{T : X \leftrightarrow X \mid \\ (R \cup \text{id } X) \subseteq T \wedge \\ T \circ T \subseteq T\}}}$$

Exercise 5 – Queue in Z: (4 points)

Specify a queue with Z-Schemata. The queue stores numbers. The functions *Init*, *Add* and *Next* have to be implemented on the queue. Every function returns a variable *result!* of type *REPORT*, which may be *OK*, an error message or *Return i* where *i* is a number.

The functions work as follows:

Init instantiates an empty queue.

Add appends a given number to the end of the queue.

Next returns the first element of the queue and removes it.