



**KTH Microelectronics
and Information Technology**

Exam in 2G1522 Modern Methods in Software Engineering, 2006-10-20, 15:00-19:00

Rules

This exam is “closed book” and you are not allowed to bring any material or equipment (such as laptops, PDAs, or mobile phones) with you. The only exceptions are an English to “your favorite language” dictionary and pencils.

Instructions

- Please read the entire exam first!
- Write clearly
- Each sheet of paper must contain your name, “personnummer”, Problem number and a unique sheet number
- Write only on one page of a sheet. Do not use the back side
- Only one Problem must be reported on each sheet
- Sort answer pages in your final delivery in order corresponding to the order of problems in this exam
- If more than one sheet is needed the continuation should be clearly noted on the beginning of each sheet and the sheet numbers used should be consecutive
- Always motivate your answers. Lack of clearly stated motivation can lead to a reduction in the number of points given
- The tasks are not necessarily sorted in order of difficulty. If you get stuck it might be a good idea to go on to the next task.

Grading

The grades depend on the sum of exam and bonus points n :

$n < 50$ fail (U)

$50 \leq n < 67$ grade 3

$67 \leq n < 84$ grade 4

$84 \leq n$ grade 5

GOOD LUCK!

Problem I. General questions

a) How does software differ from other products?

(4p)

b) Explain differences between a scientist, engineer and software engineer.

(3p)

c) A programming language is a notation for representing algorithms and data structures. List two advantages and two disadvantages of using a programming language as sole notation throughout the development process.

(5p)

Problem II. Software Life Cycle

a) Give an explanation of V-Model? What is its main difference with Waterfall model? What is its main weakness?

(5p)

b) Specify which of the following decisions were made during requirements or system design:

1. "The ticket distributor is composed of a user interface subsystem, a subsystem for computing tariff, and a network subsystem managing communication with the central computer."
2. "The ticket distributor will use PowerPC processor chips."
3. "The ticket distributor provides the traveler with an on-line help."

(3p)

Problem III. UML and OOP

a) Using only one class draw a class diagram which states that a task has a successor and a predecessor task and if two tasks are mutually exclusive then executing one of them may preclude from executing the other and vice versa

(5p)

b) Can the system under consideration be represented as an actor? Justify your answer.

(4p)

c) Which of the following are models?

1. a UML class diagram
2. a set of UML class diagrams describing the classes in a software system
3. a 1:100 scale clay replica of a new sports car that will be used to test its aerodynamics in a wind tunnel
4. a full-scale, working prototype of a new sports car

(4p)

Problem IV. Requirements Elicitation

a) We distinguished between traditional and modern methods of requirements elicitation. What is difference between them? When usage each of them is more appropriate?

(4p)

b) Specify which of these requirements are verifiable and which are not.

1. "The system must be usable."
2. "The system must provide visual feedback to the user within 1 second of issuing a command."
3. "The availability of the system must be above 95%."
4. "The user interface of the new system should be similar enough to the old system such that users familiar with the old system can be easily trained to use the new system."

(4p)

Problem V. Requirements Analysis

a) Consider a traffic light system at a four-way crossroads (e.g., two roads intersecting at right angles). Assume the simplest algorithm for cycling through the lights (e.g., all traffic on one road is allowed to go through the crossroad while the other traffic is stopped). Identify the states of this system and draw a statechart describing them. Remember that each individual traffic light has three states (i.e. green, yellow, and red).

(5p)

b) Both sequence diagrams and statecharts diagrams can be used in analysis. When usage each of them is more appropriate?

(4p)

Problem VI. System Design

a) Assume that we classified design goals into five categories: performance, dependability, cost, maintenance, and end user. Assign one or more categories to each of the following goals:

1. Users must be given a feedback within 1 second after they issue any command.
2. The TicketDistributor must be able to issue train tickets, even in the event of a network failure.
3. The housing of the TicketDistributor must allow for new buttons to be installed in the event the number of different fares increases.
4. The AutomatedTellerMachine must withstand dictionary attacks (i.e., users attempting to discover an identification number by systematic trial).
5. The user interface of the system should prevent users from issuing commands in the wrong order.

(4p)

b) Why are use cases that describe boundary conditions described during system design (as opposed to during requirements elicitation or analysis)?

(4p)

Problem VII. Object Design - Reuse

a) Explain how delegation, implementation and specification inheritance are shown in UML class diagrams.

(5p)

b) What is the Abstract Factory pattern? In which cases it is applicable? Give an example.

(5p)

c) Consider the following design goals. For each of them, indicate the candidate pattern(s) you would consider to satisfy each goal:

1. Given a legacy banking application, encapsulate the existing business logic component.
2. Given a chess program, enable future developers to substitute the planning algorithm that decides on the next move with a better one.
3. Given a chess program, enable a monitoring component to switch planning algorithms at runtime, based on the opposing player's style and response time.
4. Given a simulation of a mouse solving a maze, enable the path evaluation component to evaluate different paths independently of the types of moves considered by the mouse.

(4p)

Problem VIII. Object Design – Interface design

a) What are contracts and what do they include?

(4p)

b) Explain meaning of sets, sequences and bags in OCL. For which type(s) of navigation each of them is used? Explain why.

(4p)

c) Assume that we have a Square class that inherits from a Rectangle class. Write an invariant in OCL for the Square class.

(5p)

Problem IX. Moving to Code

a) Let us consider mapping of classes to relational tables. Explain how one-to-many associations are mapped in this case.

(5p)

Problem X. Testing.

a) Let us consider a path testing. What is a minimum number of tests necessary in order to exercise all possible paths through the code at least once?

(5p)

b) What is state-based testing and what are difficulties with it?

(5p)

-----End of Exam-----