



**KTH Microelectronics  
and Information Technology**

## **Exam in ID2207 Modern Methods in Software Engineering, 2012-10-18, 14:00-18:00**

### **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 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
- Sort your sheets according to the problem’s numbering
- Only one Problem must be reported on each sheet
- 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 (F)

$50 \leq n < 60$  grade E

$60 \leq n < 70$  grade D

$70 \leq n < 80$  grade C

$80 \leq n < 90$  grade B

$90 \leq n$  grade A

**GOOD LUCK!**

Mihhail Matskin, mobile 0704614269

## Problem I. General questions

**a) Briefly explain difference between functional and object oriented decompositions. What are pros and cons for each of them?**

**(5p)**

**b) List two advantages and two disadvantages of using Java programming language as sole notation throughout the software development process.**

**(4p)**

## Problem II. Software Life Cycle

**a) Give an explanation of V-Model? What are/is its main difference(s) compare to Waterfall model?**

**(5p)**

**b) We considered 3 formal software development models: inductive, deductive and transformational. Briefly (without technical details) explain main features of each of these models**

**(6p)**

**c) Explain what is Model Driven Architecture?**

**(5p)**

## Problem III. UML and OOP

**a) Explain difference between Concept and Phenomena. Give an example of Concept and Phenomena in software.**

**(4p)**

**b) Using only ONE class draw a class diagram which states that a node in a network structure has successor(s) and predecessor(s) nodes and if two nodes are in the same geographical region then they are mutually exclusive.**

**(6p)**

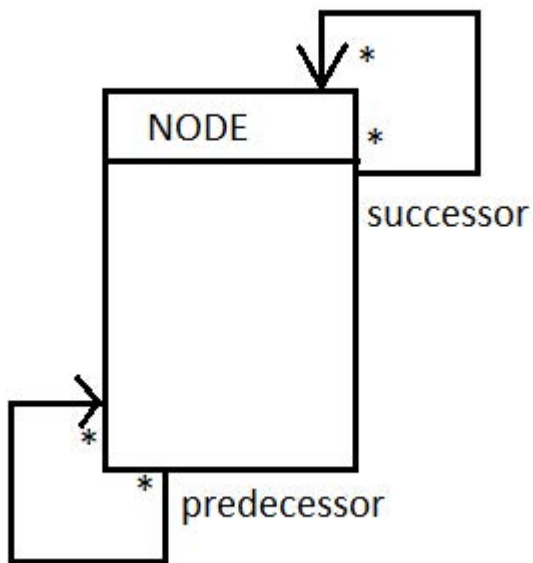
**c) Explain how <<extend>> and <<include>> are presented in the textual representation of a use case.**

**(5p)**

## Problem IV. Requirements Elicitation

**a) What is difference between traditional and modern methods for requirements elicitation? When each of them is more appropriate?**

**(5p)**



b) We identified the following types of scenario: As-Is, Visionary, Evaluation and Training scenario. Briefly explain each of them.

(4p)

## Problem V. Requirements Analysis

a) Consider a file system with a graphical user interface, such as Microsoft's Windows Explorer. The following objects were identified from a use case describing how to copy a file from a floppy disk to a hard disk: File, Icon, TrashCan, Folder, Disk, Pointer. Specify which of the identified objects are entity, boundary and control objects.

ENTITY (persistent information): File, Folder, Disk

(4p)

BOUNDARY (interaction with user): Icon, TrashCan, Pointer

CONTROL (control tasks performed by the system): -none-

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

- *"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."* system design

- *"The ticket distributor will use PowerPC processor chips."* requirements

- *"The ticket distributor provides the traveler with an on-line help."* requirements

(4p)

## Problem VI. System Design

a) Explain the following terms: coupling, cohesion, partitioning and layering.

(4p)

b) Which results from a Requirements Analysis are utilized in which particular System Design activities?

(5p)

c) Explain the Pipe and Filter architectural style. For which systems it is suitable and for which systems it is not suitable?

(5p)

## Problem VII. Object Design – Reuse

a) If you are given a choice of using implementation inheritance, delegation or specification inheritance for reuse how do you put priorities among them for usage? Briefly explain.

(5p)

b) Assume that you are developing a system that stores its data on a Unix file system. You anticipate that you will port future versions of the system to other operating systems that provide different file systems. Which design pattern would you use in order to anticipate this change? Explain your answer.

(4p)

## Problem VIII. Object Design – Interface design

a) Assume that you have the following 3 classes: parallelogram, rhombus and square. Assume that rhombus inherits from parallelogram and square inherits from rhombus. Write invariants in OCL for each of these classes taking into account the inheritance of contracts.

(5p)

## Problem IX. Moving to Code

a) Explain realization of a bidirectional, many-to-many association in a source code.

(5p)

## Problem X. Testing.

a) What is a goal of partitioning of possible inputs into equivalence classes in black-box testing? What are guidelines for selection of equivalence classes?

(5p)

## Problem XI. Agile Software Development.

a) What is metaphor in Extreme programming? Give examples of at least 3 metaphors.

(5p)

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