Linköpings Universitet
IDA
Kristian Sandahl

Written exam for Software Engineering Theory

Course codes TDDC88, TDDC93, TXDT02

Instructions to students, please read carefully

- **Explicitly forbidden aids:** Textbooks, machine-written pages, photocopied pages, pages of different format than A4, electronic equipment.
- Try to solve as many problems as possible.
- Motivate all solutions.
- Please, write and draw clearly.
- Write solutions for different areas (fundamental part) and different problems (advanced part) on separate sheets of paper.
- Label all papers with AID-number, date of examination, course code, examination code, and page number.
- You may write solutions in either Swedish or English.
- Please, note that the problems are not necessarily written in order of difficulty.
- TIP! Read through all exercises in the beginning of the exam. This will give you the possibility to ask questions about all parts of the exam, since the examiner will visit you in the beginning of the exam time. Remember that the examiner has over 200 students, so plan for solving problems in a flexible order if you have to wait for clarification.

Grading

The exam consists of two parts: Fundamental and Advanced.

The Fundamental part has problems worth 10 credits per area. Areas are: Requirements, Design & Architecture, Testing, Planning & Processes, and Quality Factors. Thus the Fundamental part can give maximally 50 credits.

The Advanced part has problems worth 50 credits in total. Each problem typically requires a longer solution of several pages.

The maximum number of credits assigned to each problem is given within parentheses at the end of the last paragraph of the problem.

Pass condition: At least 4 credits per area in the Fundamental part **and** at least 50 credits in total. The total amount of credits includes also the bonus credits you might have got in lecture exercises autumn 2010. This gives you the mark 3 in the Swedish system and a C in ECTS. If you have at least 4 credits for 4 of the areas in the Fundamental part, then you can still pass if you have more than 60 credits in total.

Higher marks are given based on fulfilled *pass condition* **and** higher amounts of credits according to the following table:

Total credits	Mark in Swedish system	Translation to ECTS
0-49	UK	Fx
50-66	3	C
67-83	4	B
84-	5	A

Multiple choice questions

In multiple choice questions we will ask you to write down the letters A, B, C, or D for the one or two statements that you think are true. Note that you should not write down the statements that you think are false. There are exactly two true statements per question, so answering with three or four alternatives with give 0 credits.

For each statement that you select that is correct (i.e., that the statement is in fact true) you get one credit. For each statement that you select that is incorrect (i.e., that the statement is in fact false, but you believed it was true) you get minus one credit. Each multiple choice question can give maximum 2 credits and minimum 0 credits, i.e., you cannot get negative credits for one multiple choice question.

Example 1: Assume that you have written down statements A and C. If now statements A and B were true, and statements C and D were false, you would get +1 credit for writing down A, but -1 credit for writing down C. Hence, the total credits for the multiple choice question is 0.

Example 2: Assume that you have written down statement B. If now statement A and B were true, and statement and statement C and D were false, you would get +1 credit for the multiple choice question.

Example 3: Assume you correctly wrote both statement A and B. If now statement A and B were true, and statement and statement C and D were false, you would get +1 credit for writing down A, and +1 for writing down B. Hence, the total credits for the multiple choice question is 2.

Good Luck!

Kristian

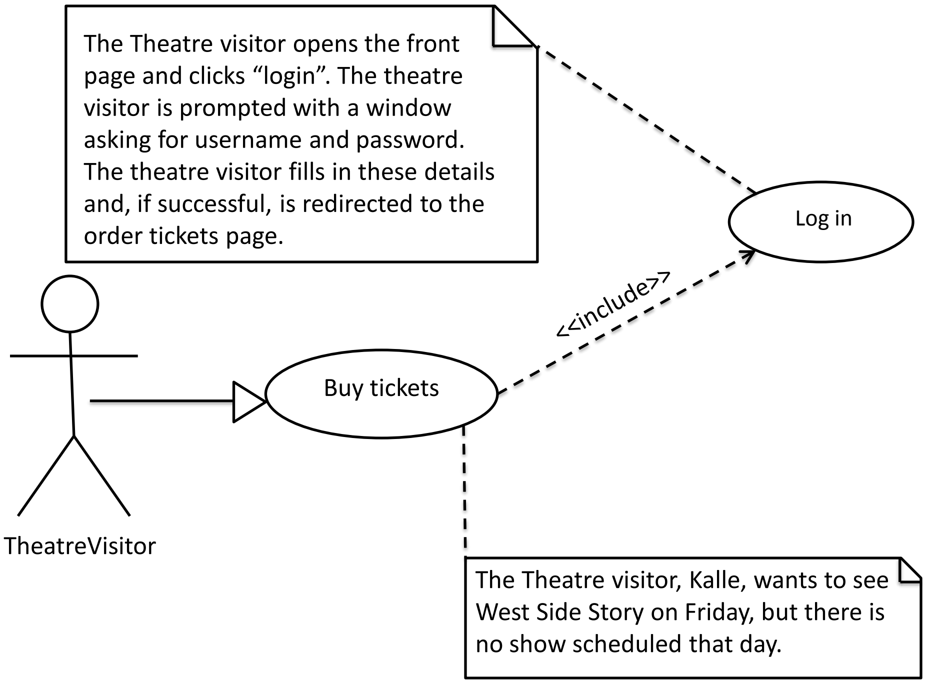
Problems

Part 1: Fundamental

Area 1: Requirements

1 a) In the use-case diagram below there are several mistakes made, but it is not completely wrong. Which of the following statements are true? Answer with the statement letter only, no motivation is needed. (2)

A. One good thing about the diagram is that the use-case names in the ovals are verbs or verb phrases, which signal that they represent some action made by the system.
B. The <<include>> relationship is correct and makes it possible for other potential use-cases to reuse the "Log in" use case.
C. The use-case text in the comment attached to "Buy tickets" is good, since it is concrete and specific.
D. The form of the arrow from the actor TheatreVisitor to the use case "Buy tickets" is correct and shows that it is the actor who starts the sequence of actions of the use-case, not the system.


The Theatre visitor opens the front page and clicks "login". The theatre visitor is prompted with a window asking for username and password. The theatre visitor fills in these details and, if successful, is redirected to the order tickets page.
Log in
<<include>>
Buy tickets
TheatreVisitor
The Theatre visitor, Kalle, wants to see West Side Story on Friday, but there is no show scheduled that day.

1 b) Write down three consistent and testable, functional requirements for a web-based time reporting system for software project members. Add a fourth requirement that is inconsistent with one or more of the first three requirements. In one sentence, explain why the fourth one is inconsistent. (4)

1 c) Describe the concepts: Requirements Elicitation, Formal methods for requirements specification, Non-Functional Requirements, and Ambiguity in the context of requirements specification. (4)

Area 2: Design and Architecture

2 a) Which of the following statements are true? Answer with the statement letter only, no motivation is needed. (2)

A. The decisions you take when the software architecture is defined, often have large impact on the quality attributes of the final software system.
B. High coupling is achieved if there are tight and many dependencies between internal parts of a module.
C. The implementation view of a software architecture documentation maps code units, such as classes and functions to the source files that contain their implementation.
D. Defining interfaces of modules shall be avoided in good design, since this only increases dependencies between modules.

2 b) Describe the principles of the layered software architecture style. Give an example of a software quality factor that is positively affected by the layered software architecture. Do not forget to motivate your example. (4)

2 c) An in-door temperature control system controls a heat pump that can either be heating or cooling the air flowing into a house. The input to the controller is: the present in-door temperature, *T*, and the user-defined set value, *ST*, for the desired in-door temperature. The output of the controller is: a signal to start the heat pump in cooling mode; a signal to start the heat pump in heating mode; and a signal to stop the heat pump.

Draw a UML statechart diagram for the class Controller that contributes to the following behaviour of the system:

1. When the system is started, the user is prompted to define *ST*.
2. When $T > ST + e$ the heat pump starts cooling. *e* is an in-built parameter that prevents unnecessary switching between modes.
3. When $T <= ST$ in cooling mode, the heat pump is turned off.
4. When $T < ST – e$ the heat pump start heating.
5. When $T >= ST$ in heating mode, the heat pump is turned off.

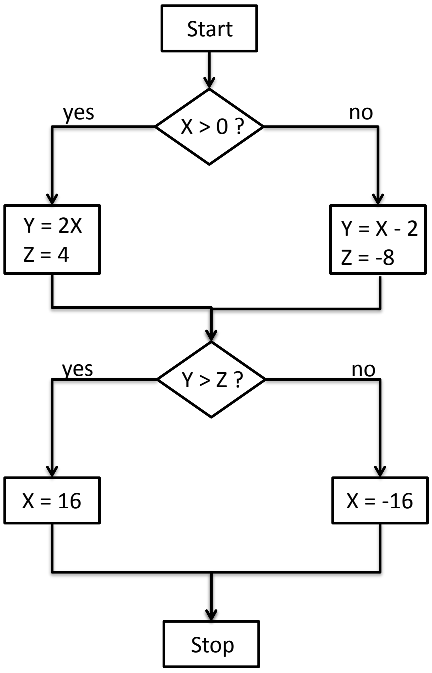
You do not need to model exceptions or turning off the system. (4)

Area 3: Testing

3 a) Which of the following statements are true? Answer with the statement letter only, no motivation is needed. (2)

A. If the customer wants to see some basic working version of a system early, it is good to use a top-down integration testing strategy.
B. A drawback of the bottom-up integration testing strategy is that you have to code many stubs.
C. In the sandwich integration testing strategy you identify a target level, and start integrating components at that level. After that you use a bottom-up strategy to reach the top node, and a top-down strategy to reach the lowest level.
D. A potential drawback of the bottom-up integration testing strategy is that the top-level components are often more important, whereas the bottom level components often perform simple, every-day tasks.

3 b) Consider the flowgraph below describing a function with an input variable X and output variables X, Y and Z. Write down two sets of test cases. The first set of test cases shall guarantee that all statements are executed at least once, using a minimum number of test cases. The second set of test cases shall guarantee that all paths[1] of the flowgraph are traversed at least once. (4)



3 c) Describe the following terms in the context of system testing: Function testing, Performance testing, Installation testing, and Pilot testing. (4)

[1] Hint: In this small example we mean testing **all** possible paths of the flowgraph. If you use cyclomatic complexity and linearly independent paths you overcomplicate this problem.

Area 4: Planning and Processes

4a) Which of the following statements are true? Answer with the statement letter only, no motivation is needed. (2)

A. Outsourcing a high-risk part of a project is an example of Risk avoidance.
B. Showing the customer different prototypes during development is a way of Risk mitigation.
C. The Risk Magnitude Indicator is calculated by the Probability of the risk multiplied with the Impact of the risk.
D. Traffic delays due to bad weather can make project meetings less effective. This is an example of a Direct risk.

4 b) Describe three advantages and one potential drawback of Iterative software development. You may assume that your reader is well acquainted with the waterfall model, so if you like to you can use the waterfall model as a contrast in your descriptions. (4)

4 c) Below you find a project plan template that is a good start, but still many important parts of a project plan are missing. Suggest four new headlines on either headline level 1 or level 2 that would improve on the plan. Do not forget to motivate your choices. (4)

---

1. Project Description
    - 1.1 Background to the project
    - 1.2 Start and expected end date.
2. Project Organization
    - 2.1 Roles
    - 2.2 Knowledge / skill
3. Time and Resource Plan
    - 3.1 Activities
    - 3.2 Resources
4. Training Plan
    - 4.1 Needed knowledge and skills.
    - 4.2 Individual assignments and training budget.
5. Change and configuration management

Area 5: Quality factors

5a) Which of the following statements are true? Answer with the statement letter only, no motivation is needed. (2)

A. A structured walk-through is a kind of a software review that can be good to improve a software architecture.
B. An inspection record contains a defect list including number; location; and description for each defect.
C. You avoid having the author of a document at the inspection meetings to allow people to speak more freely.
D. Scenario-based reading is a technique in which all inspectors follow a common, predefined scenario when inspecting the document.

5 b) Describe two Process Areas required for CMMI level 3. Besides a general description of the Process Areas, also describe how a customer of a company implementing the Process Areas will benefit from them. (4)

5 c) Describe a way to measure each of the Quality factors: Reliability, Availability, Learnability, and Understandability. The answer shall thus contain four metrics. It is important that you describe how you obtain data and how to calculate numerical values. (4)

Part 2: Advanced

6. Your company is developing a command and control system for emergency services. An important class of input to the system is *alarms* that can be received automatically by direct lines, or in the form of telephone calls to the operator. Each alarm generates a *case*, which consists of one or more *actions*. The cases are dynamically prioritized in lists visible to the operators. The lists also contain additional information which can be accessed at various levels of detail. It is also possible to view the cases on a large map in the command room. The map can, for instance, display the location of an alarm and the position of a vehicle sent to the location of the alarm.

The system also monitors the available resources, such as vehicles, specialized equipment, and personnel with different areas of competence. The system also contains efficient communication solutions both internally between co-workers, but also externally, for instance, different authorities. All system events are recorded. Your task is now to model, plan, develop and test different parts of the system.

a) Draw a UML class diagram involving the classes: Automatic alarm, Operator called alarm, Case, Action, Priority list. Each class shall contain at least one attribute and one operation. Relations between classes shall be drawn so that no class becomes isolated. You may add more classes if their purposes are described. (5)

b) Draw a UML sequence diagram covering the following parts of an accident with potential environmental damage. The customer of your system is a private security company, PrivSec, which in turn have chemical industries as customers. You may add more system events.
   1. A person observing the accident calls the operator at PrivSec.
   2. Within1 minute the operator shall call the fire brigade and direct a patrol of PrivSec to the location.
   3. The patrol of PrivSec gives information about the accident and its own resources to the incident manager of the fire brigade.
   4. The incident manager of the fire brigade might decide to order the patrol of PrivSec to participate in immediate work.
   5. The fire brigade sends information to and receives orders from responsible authorities.
   6. Within two hours the environmental damage expert of PrivSec shall be given information about the accident and actions taken.
   7. When the patrol of PrivSec is relieved from immediate services by the incident manager of the fire brigade, it sends information to the operator and receives orders from the operator, but also from the environmental damage expert of PrivSec.
   8. The environmental damage expert of PrivSec has information exchange with both the fire brigade and responsible authorities. (5)

c) Draw a use-case diagram of two actors and three use-cases. One of the actors shall be the Operator. Do not forget a use-case text for each individual use-case. (5)

d) Suppose that your customer has an alarm system at a factory, and that the automatic alarms are triggered when an associated sensor records different critical levels. For instance, a sensor measuring temperature at an ice-cream vendor might generate an alarm if the temperature goes above 0 centigrades. Some of the alarms are *minor automatic alarms* that can be acknowledged and reset by the operator, without creating a case. This can happen under the following circumstances:

   - Only one minor automatic alarm is activated; **and**
   - There are no high priority cases in the priority list of the command central from the factory; **and**
   - The operator calls the factory and gets the information that everything is OK within 2 minutes from receiving the minor alarm.

   If this is not fulfilled a case must be created and actions identified. Draw a UML activity diagram of the management of minor automatic alarms. (5)

e) Write down five functional test cases, that test the management of minor automatic alarms as described in sub-problem **6 d)** above. (5)

7. You are still employed at the company described in problem 6, but you have been transferred from the design department to the processes and tools management department.

a) Since many parts of the system are critical for safety and security, some special means have to be used to fulfill the very strict requirements. Suggest five different methods, tools or techniques you can use to achieve critical requirements. Write a short description of each of your suggestions together with a motivation. (5)

b) Many parts of the system also have high requirements on usability, and for these parts it is often good to work with agile methods in close collaboration with customer and user representatives. One agile method is eXtreme Programming. Describe 5 rules or practices of eXtreme Programming and how they can contribute to better usability. (5)

c) Your company is divided in several quite independent special teams, and one way to organize and manage team work is SCRUM. Explain the concepts of Sprint, Backlog, Product Owner, Daily SCRUM, and Burn-down chart. (5)

d) SCRUM is quite general and has many positive effects, but there are also potential problems and introduction costs in your organization. Describe five different problems or introduction costs that might occur if your company starts using SCRUM for all the teams. (5)

e) Your success in the company has inspired other companies to take up competition with you. To keep your position you must show the benefits of your product. Suggest five different quality metrics that can be used to describe a command and control system for emergency services. Do not forget to motivate your choice, and describe how the measurement can be calculated. (5)