Software Engineering (Programvaruteknik)
Exam, 2001-12-19

Lars-Henrik Eriksson
Insitutionen för informationsteknologi
Uppsala Universitet

Duration 8:00 - 13:00.

- Start by reading all the questions to see if anything is unclear. I will come by shortly before 9 o'clock to clarify questions.
- You will be provided with a form which **must** be used as the front page for your answer sheets.
- Answers may be written in Swedish or English, or any reasonable mixture of those.
- Don't use a red pen!
- Leave a margin on each sheet for comments. It is best if you use answer sheets with preprinted margins.
- Swedish-English (or your-native-language-English) dictionaries may be used.
- **Important instructions about how to answer. Failure to comply with these items may cause delayed grading, or a points deduction, or both**.
  - Don't put the answer to more than one question on each sheet!
  - You should not need more than one page for the answer to each question (given a normal size handwriting). You must not use more than two pages.
  - The handwriting should be easy to read and the answers easy to understand.
- A checklist of common mistakes that cost points:
  - Read the question again after you have written the answer. Verify that you have actually answered the question. Verify that you answered *all* parts.
  - In particular, don't forget to give an example if that is requested, and make it a concrete one.
  - Avoid unclear or missing argumentation; risky words are "of course", "self-evident", "better", "wrong", ...
  - When a question asks you to compare two things A and B, make sure to highlight the contrasts: their differences. I do *not* want a full description of A and a full description of B, leaving it to me to find the differences.
- The maximum number of grade points you can get from the exam is 50.
- 58 (70 for non-DVP students) points are required for a pass grade, while 75 (90) points are required for the VG grade. *(These limits are estimates which could be changed.)* This includes credits from assignments and participation, so a DVP student with maximum credit (20+20+10) needs only 8 points on the exam to pass the course.
- **A note about grading the exam**: I will not be able to make a complete grading before Christmas due to the little time left. To let as many as possible of you know as soon as possible if you have to take the re-exam in January or not, I will stop grading your exam as soon as it is clear that you have passed (or not) the course (taking into account the credits from the assignment and participation). After the holidays, I will resume grading to determine the full score and a possible VG grade. Provided that you check the mark on the front page form, your score will be available via the course webpage. If you don't, you will have to ask me. The result will not be posted outside the student office until grading has been completed.

Good luck!

Lars-Henrik

1. (5p.) Describe the "V" model for validation and verification. You must *briefly* explain the various parts of the model – simply making a "V" drawing with labels is not sufficient.

2. (5p.) a) What does "traceability" mean? b) How does this help in planning the modification of a software system?

3. (5p.) What are the advantages and disadvantages of using a *formal method* for writing requirements?

4. (6p) Sommerville describes three major structure models for system design: The repository (central database) model, the client-server model and the abstract machine (layered) model. How does the choice of each of these models affect how you can distribute your program over more than one machine? What is easy and what is difficult? (Note: I do *not* want a description of the models unless you feel that it is necessary for the answer, and then not more than necessary.)

5. (6p.) Explain the four dimensions of dependability: availability, reliability, safety and security.

6. (5p.) Compare black-box with white-box testing. What are the advantages and disadvantages of each?

7. (6p.) CASE tools are important in software engineering. Outline the functions of a CASE tool for testing management.

8. (6p.) a) Describe the "Chief Programmer Team". b) What specific problem does this organisation model (attempt to) solve. What other benefits can you expect (specify at least two)? Relate the benefits to specific problems of planning and organisation.

9. (6p.) What are the principal functions of a quality management system?