Linnæus University

Dnr: 2022/4547-3.1.2.2

Course syllabus

Faculty of Technology

Department of Computer Science and Media Technology

4DT904 Formella metoder, 5 högskolepoäng
4DT904 Formal methods, 5 credits

Main field of study
Computer Engineering

Subject Group
Informatics/Computer and Systems Sciences

Level of classification
Second Level

Progression
A1F

Date of Ratification
Approved by Faculty of Technology 2022-12-19
The course syllabus is valid from autumn semester 2023

Prerequisites
90 credits in Computer Engineering (including a degree project at the Bachelor level). 4DT901 Systems modeling and simulation, 5 credits, Discrete Mathematics (1MA902), 7,5 credits and Algorithms (1DT907), 5 credits, or equivalent.

Objectives
After completing the course the student shall be able to:

Knowledge and understanding

- A.1 Reason about what safety and security means for a software program or system, and which properties are required,
- A.2 describe methods and challenges to formally verify safety and security properties of software systems and what limitations these methods have,
- A.3 describe the stateof the art of formal methods and verification, and
- A.4 describe how runtime-monitoring can be used to enforce security and safety requirements.

Competence and skills

- B.1 Define safety properties in concurrent systems and programs using different

types of logic,
- B.2 use different methods to verify the correctness, safety, and security of modelled and implemented systems, and
- B.3 use the most common tools to describe and verify software systems and programs.

Judgement and approach

- C.1 reason about the societal impact (including socio-economical costs) caused by faulty and in-secure software can cause and how formal methods can be used to reduce this impact.

Content

The course gives an introduction to formal verification and introduces, e.g., temporal logic.

The following topics are covered:

- Introduction to formal verification.
- Classification of different verification techniques.
- Process algebra (CCS), and how it can be extended with time delays.
- Timed automata
- Propositional and predicate logic, and temporal logic (LTL, CTL)
- Programming languages semantics
- Program verification using Hoare logic and separation logic
- LTL and CTL model checking
- Runtime monitoring

Type of Instruction

The instruction consists of lectures, seminars, and teacher-supervised laboratory sessions. The course also contains a series of guest lectures where representatives from industry and research discusses how and why they use formal verification in their work, including which methods and tools.

Examination

The examination of the course is divided as follows:

Code	Designation	Grade	Credits
2301	Formal verification of a software program (written assingments)	AF	1,00
2302	Formal verification of a systems model (written assignments)	AF	1,00
2303	Written exam	AF	3,00

The course is assessed with the grades A, B, C, D, E, Fx or F.

The grade A constitutes the highest grade on the scale and the remaining grades follow in descending order where the grade E is the lowest grade on the scale that will result in a pass. The grade F means that the student’s performance is assessed as fail (i.e. received the grade F).

Assessment of student performance is made through formal verification of a software program (written assingments), formal verification of a systems model (written assignments) and a written exam. Repeat examination is offered in accordance with Local regulations for courses and examination at the first and second-cycle level at Linnaeus University.

To pass the course, grade E or higher is required for all parts. The final grade is decided from: formal verification of a software program (written assingments) (20%), formal verification of a systems model (written assignments) (20%), and written exam(60%).

If the university has decided that a student is entitled to special pedagogical support due to a disability, the examiner has the right to give a customised exam or to have the student conduct the exam in an alternative way.

Objectives achievement

The examination elements are linked to the course objectives in the following ways:

Goal	2301	2302	2303
A.1	☑	☑	☑
A.2			☑
A.3	☑	☑	☑
A.4			☑
B.1	☑	☑	☑
B.2	☑	☑	
B.3	☑	☑	
C.1			☑

Course Evaluation

During the implementation of the course or in close conjunction with the course, a course evaluation is to be carried out. Results and analysis of the course evaluation are to be promptly presented as feedback to the students who have completed the course. Students who participate during the next course instance receive feedback at the start of the course. The course evaluation is to be carried out anonymously.

Credit Overlap

The course cannot be included in a degree along with the following course/courses of which the content fully, or partly, corresponds to the content of this course: 4DV701, 5 credits

Other

Grade criteria for the A–F scale are communicated to the student through a special document. The student is to be informed about the grade criteria for the course by the start of the course at the latest.

The course is conducted in such a way that the course participants' experiences and knowledge are made visible and developed. This means, for example, that we have an

inclusive approach and strive for no one to feel excluded. This can be expressed in different ways in a course, for example by using the gender neutral example.

Required Reading and Additional Study Material

Required reading:

Luca Aceto, Anna Ingólfsdóttir, Kim Guldstrand Larsen och Jiri Srba, *Reactive Systems: Modelling, Specification and Verification*, Cambridge University Press, latest edition. Pages: 150 av 281.

Michael Huth, och Mark Ryan, *Logic in Computer Science: Modelling and Reasoning about Systems*, Cambridge University Press, latest edition. Pages: 300 av 412.

- Compendium of Scientific Articles.