

Formal Languages and Automata Theory

Programme course

6 credits

Formella språk och automatateori

TDDD14

Valid from: 2017 Spring semester

Determined by

Board of Studies for Computer Science
and Media Technology

Date determined

2017-01-25

Main field of study

Computer Science and Engineering, Computer Science

Course level

First cycle

Advancement level

G2F

Course offered for

- Computer Science and Engineering, M Sc in Engineering
- Industrial Engineering and Management - International, M Sc in Engineering
- Industrial Engineering and Management, M Sc in Engineering
- Mathematics, Master's programme
- Information Technology, M Sc in Engineering
- Computer Science and Software Engineering, M Sc in Engineering

Entry requirements

Note: Admission requirements for non-programme students usually also include admission requirements for the programme and threshold requirements for progression within the programme, or corresponding.

Prerequisites

Basic mathematics, for instance given by discrete mathematics courses

Intended learning outcomes

This course will give an introduction to formal languages and automata theory. Automata and formal languages appear (possibly in various disguises) in almost every branch of computer science. Having completed the course the student will be able to:

- Deal with regular and context-free languages; construct, understand and apply their formal descriptions.
- Describe relations between languages and language classes.
- Apply basic parsing methods.
- Explain the difference between decidable and undecidable problems.

Course content

Course content: Finite automata and regular expressions. Context-free grammars and pushdown automata. Deterministic context-free languages, LR parsing. Chomsky's hierarchy. Introduction to Turing Machines and undecidability.

Teaching and working methods

The theory is presented during the lectures. Problem solving is practiced during the lessons.

Examination

UPG1	Assignments	1 credits	U, G
TEN1	Written examination	5 credits	U, 3, 4, 5

Grades

Four-grade scale, LiU, U, 3, 4, 5

Other information

Supplementary courses:

Compiler Construction, Complexity Theory, Rewriting Systems, Programming Theory, Logic, advanced course

Department

Institutionen för datavetenskap

Director of Studies or equivalent

Ahmed Rezine

Examiner

Johannes Schmidt

Course website and other links

<http://www.ida.liu.se/~TDDD14>

Education components

Preliminary scheduled hours: 50 h

Recommended self-study hours: 110 h

Course literature

Additional literature

Books

D. C. Kozen, (1997) *Automata and Computability* Springer Verlag

Other

Compendium, published at the homepage.

Common rules

Regulations (apply to LiU in its entirety)

The university is a government agency whose operations are regulated by legislation and ordinances, which include the Higher Education Act and the Higher Education Ordinance. In addition to legislation and ordinances, operations are subject to several policy documents. The Linköping University rule book collects currently valid decisions of a regulatory nature taken by the university board, the vice-chancellor and faculty/department boards.

LiU's rule book for education at first-cycle and second-cycle levels is available at http://stydokument.liu.se/Regelsamling/Innehall/Utbildning_pa_grund-_och_avancerad_niva.