Valid from 2024.HS

Module description: Theory of Computation

Module Code	t.BA.IT.THIN.19HS
ECTS Credits	4
Language of Instruction/Examination	German
Organizational Unit	CAI
Module Coordinator	Olaf Stern
Legal Framework	The module description is part of the legal basis in addition to the general academic regulations. It is binding. During the first week of the semester a written and communicated supplement can specify the module description in more detail.
Module Characteristic	Type 2a 4 consecutive lecture lessons per semester week and class
Module Description	Overview: The students will know the basic terms and concepts used in theoretical computer science. They will also understand how to build models with which to assess the performance of current and future computer systems.
Module Content	**Motivation applicable to theoretical computer science:** • - formal calculation models - ability to recognise fundamental principles, irrespective of the hardware or software being used - the limits applicable to automatic calculations **(1) Formal languages / machine theory:** • - fundamental definitions of formal languages - regular languages, finite-state machines (DFSM, NFSM, e-NFSM) - context-independent languages, pushdown automata - (recursive languages), Turing machines (TM) - Chomsky hierarchy **(2) Computability and algorithm concept:** • - computable functions. Church's thesis - Equivalence of TMs and computers - Computability and programming languages: GOTO, while and loop programmes - Algorithm concept - Primitive recursion - Non-decidability and decidability: Ld diagonal language, Rice's theorem, busy beavers - Semi-decidable problems: halting problem, game of life, Collatz numbers - Reduction **(3) Complexity theory** • - Complexity of algorithms - O-notation (omega-notation) - Polynomial and exponential functions - class P, class NP - NP-complete, NP-hard - (Polynomial time reduction)
Prerequisite Knowledge	Contents of the module Discrete Mathematics Basic programming skills (any programming language)

Module description: Theory of Computation

Learning Objectives (Competences)

Students...	Competencies	Taxonomies
(3) The students will know the most important classes of complexity – particularly P and NP – as well as the simple notations used to describe them (o notation and omega notation). They will understand the question of whether P=NP and be aware of its significance. They will be familiar with the polynomial reduction method and be able to apply it to comparatively simple examples.	M, F	K2, K3
(2) The students will be familiar with the major approaches used to formalise concepts of computability and those used to develop algorithms. They will be aware of the equivalences between these various approaches. They will know the concepts of decidable and semidecidable problems and will be aware of the relationship between these two types of problems.	M, F	K2
(1) They will acquire knowledge relating to the formal languages/grammatical systems, the types of automatas which use these and the relationships between them. They will be able to design automatas for given formal languages which will accept or define these languages, as well as design languages which given automatas can recognise and define.	F, M	K2, K3

Performance Assessment

End-of-module exam	Assessment	Length (min.)	Weighting	Form
written exam	Grade	90	80	acc. to module agreement

Performance assessment during the semester	Assessment	Length (min.)	Weighting	Form
Quizzes, group tasks and presentations	Grade	0	20	acc. to module agreement

Classroom Attendance Requirement

None

2x during the semester for the group presentations; date according to the lecturer's communication (also possible online to a limited extent by agreement)

Learning material

- Detailed set of slides, supplementary documents, series of exercises with sample solutions (available on Moodle)
- John E. Hopcroft, Rajeev Motwani, und Jeffrey D. Ullman: Einführung in die Automatentheorie, Formale Sprachen und Berechenbarkeit, Pearson Studium, Addison-Wesley Verlag, 2011

Comments