

Theoretische Informatik 2: Berechenbarkeitsmodelle und Komplexität								Modulnummer:	
<i>Theoretical Computer Science 2</i>									
Bachelor Pflicht/Wahl <input checked="" type="checkbox"/> Wahlpflicht <input type="checkbox"/> Wahl <input type="checkbox"/> Sonderfall <input type="checkbox"/>				Modulbereich: Pflicht					
Anzahl der SWS	V	UE	K	S	Prak.	Proj.	Σ	Kreditpunkte: 6	Turnus angeboten in jedem SoSe
	2	2	0	0	0	0	4		
Formale Voraussetzungen: -									
Inhaltliche Voraussetzungen: -									
Vorgesehenes Semester: ab 1. Semester									
Sprache: Deutsch									
Ziele: <ul style="list-style-type: none"> • Fundamentale Konzepte und Ergebnisse aus den Gebieten Berechenbarkeit, Komplexität und Prädikatenlogik kennen und verinnerlicht haben. • Verschiedene Berechnungsmodelle kennen und die Grenzen der Berechenbarkeit einschätzen können. • Die Komplexität von typischen Informatik-Problemen einschätzen können und sensibilisiert sein für die Existenz schwieriger Probleme. • Induktionsbeweise über die Struktur von Zahlen, Wörtern, Berechnungssequenzen und/oder ähnliche Strukturen nachvollziehen und selbständig durchführen können. • Selbständig Algorithmen entwerfen und formal spezifizieren können. • In Gruppen Probleme analysieren und gemeinsam Lösungsstrategien entwickeln und präsentieren können. 									

Inhalte: 1) Berechenbarkeit

- Turingmaschinen
- Linear beschränkte Automaten
- Grammatiken der Typen 0 und 1, Abschlusseigenschaften
- LOOP-Programme und WHILE-Programme
- Primitiv rekursive Funktionen und -rekursive Funktionen
- Unentscheidbarkeit
- Unentscheidbare Probleme für Turingmaschinen
- Satz von Rice
- Postsches Korrespondenzproblem
- Äquivalenzproblem kontextfreier Grammatiken
- Semi-Entscheidbarkeit und Rekursive Aufzählbarkeit
- Universelle Turingmaschinen
- Reduktionen

2) Komplexität

- Zeit- und Platzbeschränkte Turingsmaschinen
- Komplexitätsklassen P, NP, PSpace, ExpTime
- P vs NP-Problem
- NP-Vollständigkeit
- NP-vollständige Probleme aus verschiedenen Gebieten
- Komplemente und coNP
- Approximation NP-harter Probleme
- Satz von Savitch

Unterlagen (Skripte, Literatur, Programme usw.):

- J.E. Hopcroft, R. Motwani, J.D. Ullman: Einführung in die Automatentheorie, Formale Sprachen und Komplexitätstheorie, Pearson Studium 2011
- J.E. Hopcroft, R. Motwani, J.D. Ullman: Introduction to Automata Theory, Languages, and Computation (3rd edition). Pearson Education, 2014
- C. Lutz: Theoretische Informatik 2, Skript

Form der Prüfung:

i. d. R. Bearbeitung von Übungsaufgaben und Fachgespräch

Arbeitsaufwand	Präsenz	56 h
	Übungsbetrieb/Prüfungsvorbereitung	124 h
	Summe	180 h

Lehrende:
Prof. Dr. C. Lutz u.a.

Verantwortlich:
Prof. Dr. C. Lutz