

Theoretische Informatik 2 <i>Theoretical Computer Science 2</i>								Modulnummer:	
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"> • Formale Grundlagen und elementare Fragestellungen der Informatik kennen und die fundamentale Rolle der Theorie in der Informatik verstehen. • Konzepte zur formalen Beschreibung und Analyse von Informatiksystemen kennen. • Beherrschung der grundlegenden Methoden aus den Bereichen der Berechenbarkeit und Komplexität. • Beherrschung elementarer Beweistechniken und Beweise selbst durchführen können. • Probleme analysieren, von spezifischen Gegebenheiten abstrahieren und formale Modelle in mathematischen Definitionen darstellen können. • Algorithmen für diese Probleme kennen und auf neue Problemvarianten anwenden können. • Korrektheit von Algorithmen beweisen und Eigenschaften von Algorithmen analysieren können. • Eigenständig und in Gruppen Lösungsstrategien für formale Problemstellungen entwickeln können und Lösungen verständlich präsentieren. 									
Inhalte: . 1 Berechenbarkeit: <ul style="list-style-type: none"> • 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 • Postisches Korrespondenzproblem • Äquivalenzproblem kontextfreier Grammatiken • Semi-Entscheidbarkeit und Rekursive Aufzählbarkeit • Universelle Turingmaschinen • Reduktionen 									

Inhalte 2: .

2 Komplexität:

- Zeit- und Platzbeschränkte Turingmaschinen
- 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

Lehrveranstaltung(en):

- 03-IBGT-TH12 Theoretische Informatik 2: Berechenbarkeit und Komplexität

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, Skript 2. Teil

Form der Prüfung:

MP, Fachgespräch, Klausur, ggf. mit Bonusprüfung

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

Lehrende:

Prof. Dr. C. Lutz, Prof. Dr. S. Siebertz

Verantwortlich:

Prof. Dr. C. Lutz