

Modulbezeichnung	Theoretische Informatik 1								
Modulverantwortliche(r)	Prof. Dr. C. Lutz								
Modulart	Pflicht/Wahl <input checked="" type="checkbox"/> Wahlpflicht <input type="checkbox"/>								
Spezialisierungsbereich									
Dauer des Moduls	1 Semester								
Kreditpunkte	9 CP								
Arbeitsaufwand	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 60%;">Berechnung des Workloads</td> <td></td> </tr> <tr> <td>Präsenz</td> <td style="text-align: right;">84 h</td> </tr> <tr> <td>Übungsbetrieb/Prüfungsvorbereitung</td> <td style="text-align: right;">186 h</td> </tr> <tr> <td>Summe</td> <td style="text-align: right; border-top: 1px solid black;">270 h</td> </tr> </table>	Berechnung des Workloads		Präsenz	84 h	Übungsbetrieb/Prüfungsvorbereitung	186 h	Summe	270 h
Berechnung des Workloads									
Präsenz	84 h								
Übungsbetrieb/Prüfungsvorbereitung	186 h								
Summe	270 h								
Turnus des Moduls	angeboten in jedem WiSe								
Voraussetzung für die Teilnahme	Keine <input checked="" type="checkbox"/> Folgende								
Lehr- und Lernformen	Seminar <input type="checkbox"/> Vorlesung <input checked="" type="checkbox"/> Tutorium <input checked="" type="checkbox"/> Praktikum <input type="checkbox"/> Projekt <input type="checkbox"/>								
Lernziele	<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 Automatentheorie, formalen Sprachen und Algorithmen. • 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. 								
Lerninhalte	<p>A) Automaten und formale Sprachen</p> <p>1 Endliche Automaten und Reguläre Sprachen:</p> <ul style="list-style-type: none"> • Definition und Grundbegriffe • Nichtdeterminismus • Nichterkennbarkeit von Sprachen und Pumping-Lemma • Abschlusseigenschaften • Potenz- und Produktautomat • Leerheits-, Wort- und Äquivalenzproblem • reguläre Ausdrücke • Minimale Automaten und Nerode-Rechtskongruenz • Rechtslineare Grammatiken 								

Lerninhalte 2	<p>.</p> <p>2 Kontextfreie Sprachen:</p> <ul style="list-style-type: none"> ● Grammatiken und Chomsky-Hierarchie ● kontextfreie Grammatiken ● Chomsky Normalform ● Leerheits-, Wort- und Äquivalenzproblem ● CYK-Algorithmus ● Abschlusseigenschaften ● Pumping-Lemma ● Kellerautomaten <p>B) Algorithmentheorie</p> <ul style="list-style-type: none"> ● Algorithmenbegriff ● Korrektheit und Komplexität von Algorithmen ● Suchen und Rekursionen ● Sortieren ● Graphen und elementare Graphenalgorithm: Graphdurchläufe, MST und SP ● Algorithmen Paradigmen: Divide and Conquer, Greedy Algorithmen, Dynamische Programmierung <p>Lehrveranstaltung(en): Dieses Modul besteht aus zwei Veranstaltungen, jeweils im Format 2VL+1UE, die beide verpflichtend sind.</p> <ul style="list-style-type: none"> ● 03-IBGT-AFS Automaten und formale Sprachen ● 03-IBGT-AT Algorithmentheorie
Prüfungsformen	TP, PL1: 50%, PL2: 50%, Klausur, Fachgespräch, ggf. Bonusprüfung
Literatur	<ul style="list-style-type: none"> ● 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 1, Skript ● D. Kozen: Automata and Computability, Springer, 2007 ● Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein: Introduction to Algorithms, MIT Press, 2003 ● Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein: Algorithmen - Eine Einführung, Walter de Gruyter GmbH & Co KG, 2017 ● Martin Dietzfelbinger, Kurt Mehlhorn, Peter Sanders: Algorithmen und Datenstrukturen: Die Grundwerkzeuge, Springer-Verlag, 2014 ● T. Ottmann and P. Widmayer. Algorithmen und Datenstrukturen. Spektrum Verlag, 2002.