

Aus der Modulbeschreibung

C144 Theoretische Informatik: Berechenbarkeit und Komplexität

Arbeitsaufwand: Präsenzzeit 56 h (= 2 h V + 2 h Ü je Woche)
Vor- und Nachbereitungszeit 94 h ($\approx$ 6 h je Woche)

Voraussetzungen: anwendungsbereite Kenntnisse auf den Gebieten
Modellierung, Logik, Formale Sprachen,
Maschinenmodelle, Algorithmen und Datenstrukturen,
Aufwandsabschätzungen

Lernziele: Nach erfolgreichem Abschluss des Moduls sind die Studierenden in der Lage, fundiert die prinzipiellen Möglichkeiten und Grenzen der verschiedenen Berechenbarkeitsmodelle einzuschätzen.

Sie besitzen ein Grundverständnis grundlegender Komplexitätsklassen. Sie können die Komplexität ausgewählter Problembeispiele beurteilen und algorithmisch unlösbare oder schwer handhabbare Probleme als solche erkennen.

Inhalt der Lehrveranstaltung

- ▶ Berechnungsmodelle:
TM, Goto, While, Loop
konkrete und abstrakte Syntax, Semantik (Interpreter),
Äquivalenzen (Compiler)
- ▶ berechenbare Funktionen
- ▶ partiell / primitiv rekursive Funktionen
- ▶ Entscheidbarkeit und Aufzählbarkeit
von Mengen / Sprachen
REC, Diagonalisierung, RE, Reduktion $\leq_m$
Unentscheidbare Probleme: Halteprobleme, PCP, FOL
- ▶ Komplexität
Nichtdeterminismus, Zeit, Platz, Reduktion $\leq_p$
Komplexitätsklassen P, NP, NP_c, PSPACE, NPSPACE
- ▶ Logiken: SAT $\in$ NP_c, QBF $\in$ PSPACE_c, FOL $\in$ RE \ REC

Berechnungsmodelle

- ▶ TM M : von M berechnete Funktion, akzeptierte / entschiedene Menge
- ▶ Goto-Programme: Syntax, Semantik (small-step), berechnete Funktion
- ▶ Loop / While-Programme: Syntax, Semantik (big-step), berechnete Funktion
- ▶ Übersetzungen, Unterschiede

zu big-step / small-step-Semantik z.B.

<http://concrete-semantics.org/slides-semantics.pdf>

Partiell / primitiv rekursive Funktionen

- ▶ Basisfunktionen:
konstant 0, Nachfolger, Projektion
- ▶ Operatoren (Verknüpfungen):
Substitution, Rekursion, μ

(Werte berechnen, definierte Funktion angeben,
Beispiele angeben)

Beziehungen zu Maschinenmodellen

Codierungen

von

- ▶ Wörtern
- ▶ Paaren
- ▶ Tupeln
- ▶ Listen
- ▶ Bäumen / Termen
- ▶ Mengen, TM, Programmen

(Werte berechnen, Beispiele angeben)

mit Ressourcen ... berechenbar?

Aufzählbare / entscheidbare Mengen

- ▶ Aufgaben / Probleme als Mengen
Instanzen, Lösungen
- ▶ Aufzählbarkeit / Entscheidbarkeit
- ▶ Abschlusseigenschaften
- ▶ Unentscheidbare Probleme
 - ▶ Halteprobleme
 - ▶ PCP
 - ▶ FOL

Komplexität

- ▶ Ressourcen: Zeit, Platz
- ▶ Nichtdeterminismus, Orakel
- ▶ Reduktion $\leq_p$
- ▶ Komplexitätsklassen P, NP, PSPACE, NPSPACE
- ▶ Vollständigkeit
- ▶ Einordnung von
SAT (allgemein, CNFSAT, DNFSAT, 2SAT, 3SAT),
QBF

<https://theory.cs.princeton.edu/complexity/book.pdf>

Inklusion: <https://www.math.ucdavis.edu/~greg/zoology/diagram.xml>

Prüfung Theoretische Informatik WS25/26

Klausur (90 min)

am Donnerstag, 12.2.2026 um 15:00 Uhr in NI001

Aufgabentypen ähnlich Übungsaufgaben

einziges zulässiges Hilfsmittel:

beidseitig handbeschriebenes A4-Blatt

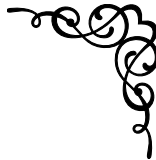
Zur Prüfung ist zugelassen, wer alle folgenden Bedingungen erfüllt:

Autotool: $\geq \lfloor 19/2 \rfloor = 9$ Punkte für Pflicht-Aufgaben **und**

Übungen: ≥ 3 Punkte für erfolgreiches Vorrechnen

(individuelle Ergebnisse in Opal)

WS25/26: 31 neu erworbene Zulassungen



Gewinner
des Autotool-Highscore
Theoretische Informatik WS25/26



Julian Binger
mit 209 Punkten