VL-18: Nachwort

(Berechenbarkeit und Komplexität, WS 2018)

Gerhard Woeginger

WS 2018, RWTH

Organisatorisches

- Vorlesung: Gerhard Woeginger (Zimmer 4024 im E1)
- Übungen: Jan Böker, Tim Hartmann
  Email: buk@lists.rwth-aachen.de

- Webseite:
  http://algo.rwth-aachen.de/Lehre/WS1819/BuK.php
  ($\longrightarrow$ Arbeitsheft zur Berechenbarkeit)
  ($\longrightarrow$ Arbeitsheft zur NP-Vollständigkeit)

Was haben wir in dieser Vorlesung diskutiert?

Stoffübersicht (1)

In den letzten Monaten haben wir folgendes gesehen:

Teil 1: Grundlagen

- Mathematische Modellierung von Berechnungen und Algorithmen
- Berechnungsmodell Turingmaschine (TM)
- Berechnungsmodell Registermaschine (RAM)
- Vergleich TM versus RAM
- Church-Turing-These

Stoffübersicht (2)

Teil 2: Berechenbarkeit

- Existenz unentscheidbarer Probleme
- Unentscheidbarkeit des Halteproblems
- Diagonalisierung / Unterprogrammtechnik / Reduktion
- Das Post'sche Correspondenzproblem
- Hilberts zehntes Problem
- Turing-Mächtigkeit

- WHILE- und LOOP-Programme
- Primitiv rekursive Funktionen
- μ-rekursive Funktionen

Stoffübersicht (3)

Teil 3: Komplexität

- Die Komplexitätsklassen P und NP
- Polynomielle Reduktionen
- NP-Vollständigkeit und der Satz von Cook & Levin
- Kochrezept für NP-Vollständigkeitsbeweise
- NP-Vollständigkeit zahlreicher Probleme
- Pseudo-polynomielle Algorithmen; stark NP-schwere Probleme
- Die Klassen coNP, PSPACE, EXPTIME

Weiterführendes

Vorlesungen

Weiterführende Vorlesungen:

- Komplexitätstheorie: Direkte Fortsetzung von BuK; untersucht Komplexitätsklassen und ihr Verhalten
- Rekursionstheorie: Theorie der berechenbaren Funktionen
- Effiziente Algorithmen: Fortsetzung von DSAL und BuK

Viele Spezialvorlesungen (Algorithmik):

- Algorithmische Geometrie
- Algorithmische Graphentheorie
- Algorithmische Spieltheorie (Britta Peis / WiWi)
- Approximationsalgorithmen (Marco Lübbecke / WiWi)
- Column Generation and Branch-and-Price (Marco Lübbecke / WiWi)
- Kryptographie
- Parametrisierte Komplexität
- Randomisierte Algorithmen
- Theorie Verteilter und Paralleler Systeme

Buchempfehlungen

Die folgenden Bücher zum Theme (und noch viele weitere) sind in der Informatikbibliothek zu finden:

- Uwe Schöning. Theoretische Informatik - kurzgefasst. Spektrum Akademischer Verlag, 2001.
- Michael Sipser. Introduction to the Theory of Computation. Cengage Learning, 2012.

Ein weiterführendes Buch ist

- Sanjeev Arora, Boaz Barak. Computational Complexity. Cambridge University Press, 2009.

Klausur

Anmerkungen zur Klausur (1)

Klausur:

Dienstag, 19. Februar 2019, 13:00h bis 16:00h

Mittwoch, 6. März 2019, 16:00h bis 19:00h

- Die Bearbeitungszeit beträgt 120 Minuten
- Bringen Sie Studierendenausweis und Lichtbildausweis mit

- Mobiltelefone müssen ausgeschaltet und weggepackt sein
- Bei der Klausur sind keine Bücher, keine Notizen, keine Mitschriften, keine Unterlagen, keine Taschenrechner erlaubt
- Werden Täuschungsversuche beobachtet, so wird die Klausur mit 0 Punkten bewertet

Anmerkungen zur Klausur (2)

- Schreiben Sie auf jedes Blatt Namen und Matrikelnummer
- Geben Sie am Ende der Klausur alle Blätter zusammen mit den Aufgabenblättern ab

- Schreiben Sie mit dokumentenechten Stiften (nicht mit roten oder grünen Stiften, und auf keinen Fall mit Bleistift)
- Schreiben Sie lesbar und sauber
- Formulieren Sie Ihre Antworten klar und eindeutig
- Geben Sie für jede Aufgabe maximal eine Lösung an (und streichen Sie alles andere durch).

Anmerkungen zur Klausur (3)

- Die Ausarbeitung der Klausur muß unter Verwendung der in der Vorlesung eingeführten Notation erfolgen.
- Die Benotung der Klausur basiert ausschliesslich auf Ihrer schriftlichen Ausarbeitung.
- Für mündliche Ergänzungen und Erklärungen (im Rahmen der Einsichtnahme) werden keine Punkte vergeben.

Anmerkungen zur Klausureinsicht

Anmerkungen zur Klausureinsicht

- Bringen Sie Studierendenausweis und Lichtbildausweis mit.
- Nicht erlaubt sind: Dokumentenechte Stifte, Handys, andere elektronische Geräte.
- Falls es Ihnen nicht möglich ist, persönlich zur Klausureinsicht zu erscheinen, können Sie einen Bevollmächtigten benennen. Füllen Sie dazu ein Vollmacht-Formular aus (Vorlagen dazu gibt's im WWW), und geben Sie auch eine Kopie Ihres Studierendenausweises mit.
- Studenten des Bachelor-Studiengangs Informatik, die im dritten Versuch nicht bestanden haben, haben das Anrecht auf eine mündliche Ergänzungsprüfung.
  Falls Sie zu dieser Gruppe gehören und eine Ergänzungsprüfung wünschen, so müssen Sie sich während der Einsicht bei uns melden und Ihren Anspruch geltend machen.

Zum Inhalt der Klausur

Stoff der Klausur

Stoff der Klausur sind

alle Begriffe, Definitionen, Sätze, Beweise, Beweisskizzen, Hilfssätze, Lemmas, Korollare, Folgerungen, Beobachtungen, Anmerkungen, Beispiele, Zusammenfassungen, etc,

die in Vorlesung, Globalübung, Tutorium und/oder Hausübungen behandelt wurden.

Aufgabentypen bei der Klausur (1a)

Wissensabfragen: Definitionen, Sätze, Beweise

Beispiel 1

Definieren Sie die Komplexitätsklasse EXPTIME.

Beispiel 2

Formulieren Sie den Satz von Matijasevich.

Beispiel 3

Beweisen Sie, dass das Rucksack Problem NP-vollständig ist.

Aufgabentypen bei der Klausur (1b)

Wissensabfragen: Definitionen, Sätze, Beweise

Beispiel 4

Diese Aufgabe behandelt den Beweis des Satzes von Cook & Levin.

(a) Geben Sie die im Beweis verwendeten Variablentypen an und beschreiben Sie deren Bedeutung.

(b) Skizzieren Sie eine CNF-Formel, die den Umstand beschreibt, dass sich der Kopf zu jedem Zeitpunkt an genau einer Position befinden muß.

(c) Beschreiben Sie eine Formel (nicht notwendigerweise in CNF), die den Umstand beschreibt, dass für jeden Zustand an der aktuellen Kopfposition der korrekte Transitionsübergang realisiert wird.

Aufgabentypen bei der Klausur (2)

Anwenden von Sätzen und Methoden auf konkrete Probleme

Beispiel 1

Beweisen Sie mit Hilfe des Satzes von Rice, dass die folgende Sprache unentscheidbar ist: $L = \{ \langle M \rangle \mid L(M) \text{ ist leer} \}$

Beispiel 2

Zur Lösung dieser Aufgabe dürfen Sie voraussetzen, dass die Addition $x + y$ primitiv rekursiv ist.
Zeigen Sie, dass die Funktion $f : \mathbb{N}^2 \to \mathbb{N}$ mit $f(x, y) = (x + 1)^y$ primitiv rekursiv ist, indem Sie sie aus den Basisfunktionen und der Addition durch Komposition und primitive Rekursion zusammenbauen.

Aufgabentypen bei der Klausur (3a)

Beweisen und widerlegen von Sachverhalten

Beispiel 1

Beweisen oder widerlegen Sie, dass die folgende Sprache rekursiv aufzählbar ist: $L = \{ \langle M \rangle \mid M \text{ akzeptiert ein Palindrom} \}$

Beispiel 2

Beweisen oder widerlegen Sie:
Wenn L entscheidbar, dann ist auch L^2 entscheidbar.

Anmerkungen:

- Bei Aufgaben vom Typ "Beweisen oder widerlegen Sie folgende Aussage" werden Punkte nur für Argumente vergeben. Ein einfaches "Ja" oder "Nein" oder "Wahr" oder "Falsch" bringt keine Punkte.
- Allgemein: Punkte werden bei der Klausur hauptsächlich für Argumente vergeben. Begründen Sie Ihre Behauptungen.

Aufgabentypen bei der Klausur (3b)

Beweisen und widerlegen von Sachverhalten

Beispiel 3

Beweisen oder widerlegen Sie für $L \subseteq \{0,1\}^*$:

(a) Wenn L abzählbar ist, dann ist L auch aufzählbar.

(b) Wenn L aufzählbar ist, dann ist L auch abzählbar.

Beispiel 4

Gilt $\mathrm{SAT} \leq_p H$? (Anmerkung: H bezeichnet das Halteproblem.)

Anmerkungen:

- Auch wenn eine Aufgabe als Entscheidungsfrage formuliert wird, gibt es für ein einfaches Ja oder Nein keine Punkte.
- Allgemein: Punkte werden bei der Klausur hauptsächlich für Argumente vergeben. Begründen Sie Ihre Behauptungen.

Aufgabentypen bei der Klausur (4a)

Beweisen von Sachverhalten

Beispiel 1

L_1 und L_2 seien semi-entscheidbare Sprachen über $\Sigma = \{0, 1\}$ mit $L_1 \cup L_2 = \Sigma^*$ und $101 \in L_1 \cap L_2$. Beweisen Sie, dass $L_1 \leq L_1 \cap L_2$ gilt.

Beispiel 2

Es seien a, b, c drei positive ganze Zahlen. Beweisen Sie, dass die Funktion $f : \mathbb{N} \to \mathbb{N}$ mit $f(n) = an^2 + bn + c$ LOOP-berechenbar ist.

Aufgabentypen bei der Klausur (4b)

Beweisen von Sachverhalten

Beispiel 3

Beweisen Sie, dass das folgende Problem NP-schwer ist.

EINGABE: Ein Graph $G = (V, E)$; für jede Kante $e \in E$ eine ganze Zahl $L(e)$
FRAGE: Gibt es eine Einbettung $f : V \to \mathbb{Z}$ der Knoten in die ganzen Zahlen, sodass für jede Kante $e = \{u, v\} \in E$ der Abstand zwischen $f(u)$ und $f(v)$ genau $L(e)$ beträgt?

Beispiel 4

Liegt das folgende Problem in coNP? Liegt es in NP?

EINGABE: Eine logische Formel φ in CNF
FRAGE: Besitzt φ höchstens eine erfüllende Variablenbelegung?

Eine Klausuraufgabe von 2018

Aufgabe (1)

- Wir betrachten n Prozesse $P_1, \ldots, P_n$ mit Prozesszeiten $t_1, \ldots, t_n$ und Verfügbarkeitsintervallen $[\ell_i, r_i]$.
- Diese Prozesse sollen in geeigneter Reihenfolge sequentiell auf einem einzelnen Prozessor abgearbeitet werden.
- Jeder Prozess P_i $(i = 1, \ldots, n)$ ist dabei ab dem Zeitpunkt ℓ_i verfügbar, muß für t_i unmittelbar aufeinanderfolgende Zeiteinheiten auf dem Prozessor bearbeitet werden, und muss spätestens zum Zeitpunkt r_i fertig sein.
- Der Prozessor kann zu jedem Zeitpunkt höchstens einen Prozess bearbeiten.

Problem: PROZESS-PLANUNG

Eingabe: Ganze Zahlen $t_1, \ldots, t_n \geq 1$ und $\ell_1, \ldots, \ell_n$ und $r_1, \ldots, r_n$
Frage: Gibt es einen Plan, der alle Prozesse rechtzeitig fertig stellt?

Aufgabe (2)

(a) Betrachten Sie die folgende Instanz des PROZESS-PLANUNG Problems, die aus sieben Prozessen besteht.

Prozess	P_1	P_2	P_3	P_4	P_5	P_6	P_7
t_i	1	2	2	3	3	4	4
ℓ_i	9	0	0	0	0	0	0
r_i	10	19	19	19	19	19	19

Ist diese Instanz eine JA-Instanz?

Aufgabe (3)

(b) Formulieren Sie die Zertifikat-Charakterisierung von NP.

(c) Zeigen Sie, dass das Problem PROZESS-PLANUNG die Zertifikat-Charakterisierung von NP erfüllt:
Beschreiben Sie Ihr Zertifikat und **analysieren** Sie seine Länge.
Beschreiben Sie das Verhalten Ihres Verifizierers und **analysieren** Sie seine Laufzeit.

Aufgabe (4)

(d) Beweisen Sie durch eine polynomielle Reduktion:
PROZESS-PLANUNG ist **NP-schwer**.