Prof. Dr. Hans-Jörg Kreowski, Dr. Sabine Kuske
Studiengang Informatik
Linzer Str. 9a
OAS 3001, 3005
Tel.: 2956, 2335, 3697 (Sekr.), Fax: 4322
E-Mail: {kreo,kuske}@informatik.uni-bremen.de
www.informatik.uni-bremen.de/theorie

Juli 2006
Fragenkatalog

# Theoretische Informatik 2

## Ein paar Fragen

Wie verabredet, soll am Ende der Lehrveranstaltung ein Fachgespräch stattfinden, in dem von jeder Teilnehmerin und jedem Teilnehmer gemäß Prüfungsordnung einige Fragen beantwortet werden sollen. Zu einem Fachgespräch könnt ihr einzeln oder in Gruppen bis zu vier Personen erscheinen, wobei bei einem Gruppenfachgespräch natürlich jeder einzeln befragt wird. Die folgenden Fragenkataloge sind dafür die Grundlage. Dabei geht es nicht nur darum, eine einzelne Frage beantworten zu können, sondern über die thematischen Komplexe (wie CE-S, Sortieren, Matrizenmultiplikation, monotone Grammatiken, kontextfreie Grammatiken, Turing-Maschinen usw.) etwas zu wissen und sagen zu können. Außerdem kommt es nicht nur auf die Konzepte an (Katalog 1), sondern auch auf deren Eigenschaften (Katalog 2).

## Katalog 1: Was ist das? Wie ist es konstruiert?

### CE-S

1. CE-S-Spezifikation
2. Term
3. Werteterm
4. Bedingte Gleichung
5. Wertzuweisung
6. Substitution
7. Auswertungsschritt
8. Gleichungsanwendung
9. Berechnung einer CE-S-Spezifikation
10. Gleichwertigkeit von Termen
11. Vorwärtsinterpreter für CE-S
12. Gleichungsbeweiser für CE-S
13. CE-S-berechenbare Funktion
14. Aufwand einer CE-S-Spezifikation

### Aufwandsklassen

1. Das große $O$
2. Aufwandsklasse
3. die Klasse $P$
4. die Klasse $NP$
5. $NP$-Vollständigkeit
6. $P$=$NP$-Problem
7. die Klasse $PSPACE$
8. die Klasse $NPSPACE$

### Chomsky-Grammatiken

1. Chomsky-Grammatik
2. Produktion
3. Anwendung einer Produktion
4. direkte Ableitung
5. Ableitung
6. von einer Chomsky-Grammatik erzeugte Sprache
7. Typ-3-Grammatik
8. Typ-2-Grammatik
9. Typ-1-Grammatik
10. Typ-0-Grammatik
11. monotone Grammatik
12. kontextfreie Grammatik

13. rechtslineare Grammatik
14. reguläre Grammatik

### Entscheidbarkeit

1. Wortproblem
2. Entscheidbarkeit eines Problems
3. Postsches Korrespondenzproblem
4. Cocke-Kasami-Younger-Verfahren

### Turing-Maschinen

1. Turing-Maschine
2. Konfiguration
3. deterministische Turing-Maschine
4. Turing-berechenbare Funktion
5. Churchsche These

### While-S

1. *while*-$S$-Programm
2. graphische Darstellung eines *while*-$S$-Programms
3. Berechnung eines *while*-$S$-Programms

## Katalog 2: Was gilt? Was nicht? Warum?

### Aufwand

1. Das Sortierproblem liegt in $O(n)$.
2. Das Sortierproblem liegt in $O(n \cdot \mathrm{ld}\ n)$.
3. Das Sortierproblem liegt in $O(n^2)$.
4. $O(ldn) \subseteq O(n) \subseteq O(n^2) \subseteq O(n^3) \subseteq \cdots \subseteq O(2^n)$.
5. $P \subseteq NP$.
6. $NP \subseteq P$.
7. $P \subseteq PSPACE$.
8. $PSPACE \subseteq P$.
9. $PSPACE \subseteq NPSPACE$.
10. $NPSPACE \subseteq PSPACE$.
11. $NP \subseteq NPSPACE$.
12. $NPSPACE \subseteq NP$
13. Das Wortproblem für monotone Grammatiken ist in *NPSPACE*.

### Entscheidbarkeit

1. Das Wortproblem für kontextfreie Grammatiken ist in kubischer Zeit lösbar.
2. Das Wortproblem für Chomsky-Grammatiken ist entscheidbar.
3. Das Wortproblem ist für monotone Sprachen entscheidbar.
4. Das Wortproblem ist für kontextfreie Sprachen nicht entscheidbar.
5. Das Leerheitsproblem ist für kontexfreie Sprachen entscheidbar.
6. Die Leerheit des Durchschnitts zweier kontextfreier Sprachen ist entscheidbar.
7. Die Leerheit des Durchschnitts zweier regulärer Sprachen ist entscheidbar.
8. Das Postsche Korrespondenzproblem ist entscheidbar.

### Berechenbarkeit

1. Jede Turing-berechenbare Funktion ist CE-S-berechenbar und umgekehrt.
2. Jede *while*-$S$-berechenbare Funktion ist CE-S-berechenbar und umgekehrt.
3. Jede Turing-berechenbare Funktion ist *while*-$S$-berechenbar und umgekehrt.