Erinnerungsprotokoll Diplomprüfung
Teilprüfung Theoretische Informatik
Kurs 1820 Formal Languages

Prüfer: Prof. Dr. Verbeek
Beisitzer: Dr. Hertling
Datum: 26.4.2001

Erklärung der Chomsky Hierarchie?

Es gibt vier Chomsky Klassen.
Sie sind mit Grammatiken definiert.
Die höhreren sind echte Teilmengen der unteren.
Klasse 0: Keine Einschränkung der Regelmengen String mit mindestens einem NT geht in beliebigen String
über.
Klasse 1: NT dürfen nur im Kontext ersetzt werden und (wichtig!): Keine Verkürzung der Regeln.
Klasse 2: Ersetzung von einem einzelnen NT links und A ✍? zulässig
Klasse 3: (nicht mehr verlangt)

Warum sind Chomksy 2 eine echte Teilmenge der Chomsky 1?

(Hier hatte mußte ich nach einigen halbherzigen Versuchen Grammatik überführen in NF, ... passen). Idee mit Iterationslemma.

Iterationslemma erklären mit Konstante K, Positionenmenge , Aufteilung in 5 Teilwörter, die bezüglich der Positionen bestimmten Bedingungen genügen müssen.

Konstante hängt ab von der Anzahl NT und den Längen der rechten Seite.
Mit dem Iterationslemma läßt sich nur nachweisen, daß eine Sprache nicht kfS ist. Für den umgekehrten Fall kann man es nicht anwenden.

Unterschied zum Pumpinglemma (Positionen, Iterationslemma ist stärker?)

Beweisidee des Iterationslemma: (Baum, der für mindestens ein NT auf einem Weg zweimal das gleiche NT gelabelt hat). Dieses NT kann iteriert (aufgeblasen) werden.

Wie entscheidet man kfS?

Mit CKY Algorithmus, der in O(n*n*n) entscheidet und O(n*n) parst.

Gesamteindruck: Relativ gelassenen Prüfungsatmosphäre. Schwerpunkt liegt bei Kapiteln 1-4. Kapitel 6-7 werden höchstens gestreift. Bei Definitionen (Sprache, Grammatik, Ableitung) unbedingt auf Einzelheiten achten. Bei einigen Antworten war ich nicht spontan genug, was zum Nachhaken führte.

Die Benotung war großzügig. Trotz erkennbarer Mängel: 2

Kleine Erklärungsprobleme, weil die Prüfung auf deutsch stattfand, der Kurstext nur englisch erhältlich ist.

Lernempfehlung: Buch von John Hopcroft und Jeffrey Ullman: Automatentheorie, Formale Sprachen und Komplexitätstheorie (Addison Wesley).

Diplomprüfung Theoretische Informatik
bei
Prof. Dr. Verbeek

Formal Languages

Roman Kawe

Matr. Nr. 3462277

26.Februar 1997

1 KontextSensitive Sprachen

Von welchen Grammatiken werden die ksS erzeugt? Insbesondere: Welche Form haben die Regeln?

2 Chomsky-Hierarchie

Wie sehen die jeweiligen Regeln der Grammatiken aus? Warum sind die kontextfreien Sprachen eine echte Teilmenge der kontextsensitiven Sprachen, das kann man an der Form der Regeln doch garnicht ablesen?

Wodurch werden die Sprachen der Chomsky-Hierarchie erkannt? Bei welchen Sprachen gibt es Unterschiede zwischen deterministischen und nicht-deterministischen Sprachen? Insbesondere: Warum gilt dkfS $\subset$, aber nicht $\subseteq$ kfS?

3 LL

Wie verhalten sich die von LL-Grammatiken erzeugten Sprachen zu den dkfS? Er hat mir dann noch zusätzlich erklärt, dass auch $\bigcup_{i=0}^{\infty} \mathrm{LL}(i)$ eine echte Teilmenge der dkfS ist.

4 CYK

Was bedeuten die A_{ij}? Warum funktioniert der Algorithmus?

5 Eindrücke...

Ein netter Prüfer: Er lässt einen nicht hängen, zeigt einem aber, dass man es doch hätte wissen können: Ich konnte z. B. nicht begründen, warum ksS $\subset$ kfS ist, obwohl die Regeln das ja nicht nahe legen, da machte er den Umweg über die Entfernung von ε-Regeln, worauf es mir wie Schuppen aus den Haaren fiel und er nur sagte, "Sehen Sie, Sie wissen es ja doch...". Das Ergebnis war jedenfalls eine 1,7, schließlich hatt ich mit den A_{ij} auch so meine Schwierigkeiten :-)