Gedächtnisprotokoll zur Diplomprüfung

Theoretische Informatik

Stefan Hoferer

24. Juni 2005

Prüfer:	Prof. Dr. Klaus Indermark
Beisitzer:	René Thiemann
Prüfungsgebiete:	Compilerbau (Indermark, SS 04)
	Logikprogrammierung (Indermark, WS 03/04)
	Termersetzung (Giesl, SS 04)
Datum, Uhrzeit:	13.06.05, 11:00 Uhr
Note	1.0

Bemerkungen

Wie in fast allen Protokollen zu lesen ist, kann auch ich bestätigen, dass die Atmosphäre sehr angenehm und locker ist. Die Nervosität verflog bereits mit der ersten Frage, und ich konnte mich gut auf die weiteren Fragen konzentrieren. Links neben mir saß der Beisitzer, rechts Herr Indermark. Es gab leider nur eine Armbanduhr, die zu weit lag um die Uhrzeit erkennen zu können. Anhand der Zeichen die Herr Thiemann neben den Fragen notierte konnte ich ungefähr abschätzen, wie gut meine Antworten waren.
Herr Indermark hilft, wenn man nicht weiter weiß, oder kurz davor ist Unsinn von sich zu geben: "Vorsicht, sie bewegen sich da gerade auf sehr dünnem Eis...".
Kleinere Fehler fallen ebenfalls kaum ins Gewicht.
Da ich während meiner Vorbereitung eine Zusammenfassung der Prüfungsfragen erstellt habe, verzichte ich hier auf Angabe der Lösung bzw. des genauen Wortlautes. Die Zusammenfassungen können auf www.shoferer.de runtergeladen werden.

Termersetzungssysteme (ca. 15 min)

- Satz von Birkhoff, dabei Ersetzungsrelation definieren, Wortproblem erläutern
- Folgerungsbegriff erklären

- Ersetzungsrelation graphisch darstellen.
- Sprung zu kritischen Paaren. Welche Möglichkeiten gibt es (jeweils einzeichnen), welche davon kann man immer zusammenführen, warum kann man diese zusammenführen.
- Entscheidbarkeit des Wortproblems.
- äquivalenz Gleichungssystem und TES
- Kongruenzabschluss, was muss man machen um ein Entscheidungsverfahren zu erhalten.
- Konvergenz, Church-Rosser-Eigenschaft, Beweis der Äquivalenz

Compilerbau (ca. 15 min)

- reguläre Ausdrücke um Komplement erweitern, invariante bei Thompson-Konstruktion, Bedeutung für die Komplexität
- Minimalisierung des Produktautomaten. Geht das? Darf man unproduktive Zustände weglassen? Wie muss man den Äquivalenzbegriff für Zustände bei diesem Automat definieren?
- Welche Zustände hat der Backtrackautomat?
- Wie entscheidet man bei der TD-Analyse, welche Regel man anwendet? Wie sind la-Mengen definiert? Wann ist ε in einer la-Menge? Induktiven Algorithmus angeben, der testet, ob ε aus Nichtterminal ableitbar ist. Wie sieht ein Algorithmus aus, der testet, ob ε in der Follow-Menge liegt.
- Attributgrammatiken, welche Regel bestimmt ein Attribut. (Unterschied zwischen inherit und synthetisch war gewünscht, hab ich aber nur mit viel Rumstammeln hinbekommen)
- L-Attributgrammatiken, Definition, expand/reduce-Schritte, Wie geht hier der index-Shift?
- Welche Deklarationsumgebungen sind in einer Prozedur sichtbar?

Logikprogrammierung (ca. 15 min)

- Was ist ein Logikprogramm? Fakten, Regeln, Kann man ein Logikprogramm als eine Formel darstellen?
- Skolemnormalform, Pränexnormalform
- Fixpunktsemantik, $trans_{\Pi}$, Stetigkeit, Monotonie, kleinste obere Schranke

- Programm für Addition angeben, Fp-Semantik darauf anwenden
- SLD-Baum, unterschied zu Prolog, wie erkennt man unendliche Pfade, wie groß ist ein Resolvent?

Die notierten Fragen stellen nur einen groben Überblick der angesprochenen Themen dar. In der Prüfung waren die Fragen sehr kleinschrittig und erlaubten nur selten weit auszuholen und so etwas Zeit zu schinden. Wer es etwas genauer haben möchte, kann sich gerne bei mir melden

Viel Erfolg bei euren Prüfungen!