Prüfungsprotokoll Theorie - Indermark und Hromkovič
Compilerbau, Programmanalyse & Compileroptimierung, Algorithmische Grundlagen der Bioinformatik

Matthias Hensler

18. Oktober 2004

1 Indermark: Compilerbau - 20 Minuten

Grundlage: Vorlesung Indermark im Sommersemester 2004

Vorbereitung: Skript zur Vorlesung, Videoaufzeichnung zur Vorlesung

- Wann ist eine Sprache $LL(k)$? *(Definition)*
- Wie ist die Entscheidbarkeit für $LL(k)$ definiert? *(Definition des Lemma zur Disjunktheit zweier First-Mengen)*
- Warum sind die regulären Sprachen $LL(1)$? *(reguläre Sprache als Automat darstellen $\longrightarrow$ Zuständsübergänge als Regeln betrachten mit Ausgangszustand als linke Seite und Transistionssymbole konkateniert mit Zielzustand als rechte Seite)*
- Wie sind die Lookahead-Mengen für $LL(1)$ definiert?
- Wie läßt sich feststellen ob eine Sprache $LL(1)$ ist? *(la-Mengen Bilden und für Regelalternativen auf Disjunktheit/Eindeutigkeit überprüfen)*

2 Indermark: Programmanalyse & Compileroptimierung - 10 Minuten

Grundlage: Vorlesung Indermark im Sommersemester 2003

Vorbereitung: Skript zur Vorlesung, Videoaufzeichnung zur Vorlesung, einige über Google gefundene Dokumente zur Datenflußanalyse

- Wie funktioniert die Common-Subexpression Elimination bei SLC? *(AE-Analyse mit gen/kill-Funktion durchführen, für jede Zeile AE-Mengen bilden)*
- Wodurch unterscheiden sich die AE-Mengen bei IC-Programmen? *(Bitvektoren statt Expressions)*
- Bei IC: Was ist der Unterschied zwischen Flußdiagramm und Flußgraph? *(Diagramm mit Prädikaten, betrachten aller Berechnungspfade, beim Graphen weglassen der Prädikate und nichtdeterministische Programmpfade)*
- Wieso ist bei der MFP-Methode die Berechenbarkeit gegeben? *(wegen Monotonie und vollständigen Verband folgt mit dem 2. Fixpunktsatz die Existenz eines kleinsten Fixpunkts. Wegen ACC gilt nach dem 1. Fixpunktsatz die Berechenbarkeit des Fixpunkts)*
- Warum folgt aus ACC die Stetigkeit? *(wegen endlichen ω-Ketten folgt der Abbruch und daraus die Stetigkeit)*

3 Hromkovič: Algorithmische Grundlagen der Bioinformatik - 10 Minuten

Grundlage: Vorlesung Böckenhauer/Bongartz im Sommersemester 2003

Vorbereitung: eigene Mitschrift der Vorlesung, Buch der Vorlesung

- Kapitel 4: Welche Möglichkeiten gibt es um Matchings zwischen Text und Muster zu finden? *(betrachte naiven Ansatz mit $O(m \cdot (n-m))$. Verbesserung für gegebenes Muster mit String-Matching Automat (kurze Beschreibung und Konstruktion des Automaten). Verbesserung für gegebenen Text mit Suffixbäumen (kurze Beschreibung und Aufbau des einfachen Suffixbaums))*
- Kapitel 5: Welche Alternativen gibt es zum exakten Stringmatching? *(Alignmentverfahren mit verschiedenen Bewertungsfunktionen betrachten)*
- Kann man auch k Strings vergleichen? *(Ansatz mit dynamischer Programmierung immer noch möglich, aber exponentieller Aufwand...)*
- Exponentiell in was? *(...Anzahl der Strings, also k)*
- Ist eine Approximation möglich? *(r-Approximation in $O(1+\frac{\log r}{r})$, kurz das Verfahren angerissen)*
- Beweis für den Aufwand?
- Kapitel 10: Vergleich von Genomen möglich? Welche Aussagen sind möglich? *(als Taxa auffassen und in ultrametrischen Baum darstellen)*
- Was bedeutet Ultrametrik? *(Taxa nur in den Blättern, Kosten des Pfades von Wurzel zum beliebigen Blatt immer gleich)*
- Was für eine Möglichkeit gibt es noch? *(additive Bäume betrachten, mit Taxa im Baum und additiver Distanzfunktion)*

- Welche Bedeutung haben die Taxa innerhalb des Baumes? *(Lebewesen die in der Evolution auftauchen müssten aber bislang nicht nachgewiesen wurden)*

4 Fazit

Dies war meine erste Prüfung für die ich mit etwa 3 Monate (mit Verschnaufspausen) anhand der Skripte und Videoaufzeichnungen (sehr große Hilfe!) vorbereitet habe. Gelernt hab ich alleine, wobei eine Gruppe aus zwei bis drei Personen (eher nicht mehr) wohl effektiver ist.

In der Prüfung selber rast die Zeit und ist vorrüber ehe man wirklich mitbekommen hat das sie angefangen hat. Nervösität verfliegt schnell, es jedoch hilfreich (wenn man es kann) sich durch Verständnisproblemen bei den Fragen nicht verrückt zu machen (manchmal ist es nicht ganz so klar auf was die Frage tatsächlich abzielt). Herr Indermark fragt sehr gerne nach formalen Definitionen die man am besten parat hat.

Da ich diese oft nur in mehreren Anläufen hinschreiben konnte führte dies in Compilerbau zu einer Abwertung der Note. Insgesamt ist die Prüfung mit einer glatten zwei aber sehr fair bewertet. Die Professoren sind als Prüfer definitiv zu empfehlen, da die Prüfungsatmosphäre sehr angenehm ist.

Wer die Möglichkeit hat sich noch von Professor Indermark oder auch von Professor Hromkovič prüfen zu lassen sollte diese Chance auf jedenfall wahrnehmen!