

Gedächtnisprotokoll Formale Methoden

Auch bei mir war es sehr ähnlich zu den Prüfungen der Anderen. Scheinbar gibt es pro Tag eine KIV-Aufgabe (die aber auch immer ähnlich ist).

Gerhard hat bei mir die Fragen gestellt.

AL

Begonnen hat es mit dem Aussagenlogischen Kalkül. Dazu wurde mir eine aussagenlogische Formel vorgelegt (etwas wie $(A \wedge B) \vee C \rightarrow (A \wedge B) \vee C$). Ich sollte erst Mal nennen wie unser Kalkül heißt (Aussagenlogischen Sequenzenkalkül^^) und dann anhand der Regeln die Sequenz "per Hand" beweisen. Dann kam wie bei allen die Frage, was der Simplifier hier getan hätte. Der hätte Regeln mit maximaler einer Prämisse angewandt und dann die restliche Prämisse mit kontextabhängiger Simplifikation geschlossen (hier musste ich auch angeben, was genau er bei kontextabhängiger Simpl. machen würde).

PL

Beim Übergang zu PL würde ich zunächst gefragt, wie denn hier die Formeln aufgebaut würden (dabei auch Aufbau der Terme, Erklärung Signatur, etc.).

Was kann PL mehr? \rightarrow Quantoren und Gleichheiten.

Bei der Frage "Welche Regeln sind für Gleichheit hinzugekommen?" sorgt ihr zuverlässig für einen Lacher bei Alex als auch einem Schmunzeln von Gerhard (wie ich bereits einen Protokoll von vorherigen Jahrgängen entnehmen konnte), wenn ihr die Abkürzung "Refl" verwendet.

Dann hat mich Gerhard gebeten, die Regel $(L \forall)$ aufzuschreiben. Bei der verunsichernden Frage, ob man denn jetzt beliebige Terme für "t" einsetzen darf oder ob es Einschränkungen gäbe, müsst ihr sehr bestimmt mit "Ja, man kann ganz beliebige Terme einsetzen!", um ein anerkennendes Nicken von beiden zu erhalten.

"Welche Eigenschaften gelten denn jetzt in PL nicht, die in AL gelten?" - "Entscheidbarkeit." -

"Welche Eigenschaften interessieren uns noch?" - "Vollständigkeit und Korrektheit".

"Was können wir mit PL nicht?" - "Natürliche Zahlen monomorph spezifizieren." - "Warum konnten wir das nicht? Was war das Problem?" - "Weil man 'künstlich' eine Konstante zur Trägermenge jedes Modells von Nat hinzufügen kann und erhält ein weiteres Modell, das nicht isomorph zu Nat ist."

"Was macht man daher?" - "Generiertheit einführen. (Erklären, wie Gen.-Klauseln definiert sind und wie man dadurch Trägermengen einschränkt)".

"Welche weitere Art der Generiertheit gibt es noch (Freie Generiertheit!) und was kann die mehr (Injektiv auf die Trägermenge \rightarrow für jedes Element in der Trägermenge gibt es genau (!) einen Konstruktorterm)."

"Wie kann man das im KIV spezifizieren?" - "Data Specifications: Da werden die nötigen Axiome dann automatisch generiert." - "Schreib mir doch bitte die Regeln für Listen auf!"

"Welche freien Datentypen haben wir denn schon kennengelernt?" - "Listen, Nat, Aufzählungstypen, Tupel, Binärbäume (das letzte wusste ich leider nicht; der Tip "Die hatten wir in den Übungsaufgaben" hat mir leider auch nicht geholfen, da ich die leider nicht gemacht hatte (peinliche Situation))

“Wie setze ich solche freien Datentypen in JAVA (oder anderen Programmiersprachen) um?”
- “Abstrakte Klasse für den Typ und Unterklassen für jede Art von Konstruktoren.”

Simplifier

Was kann der Simplifier alles? Meine Antwort “Rewriteregel”, “Äquivalenzregel”, “Forwardregel” haben sie zwar akzeptiert, wollten aber dennoch “Assoziativität und Kommutativität” hören und wie diese vom Simplifier angewandt werden (alle anderen Regeln bis auf Assoz. / Kommut. anwenden). Alex hat meine anfängliche Verwirrtheit gefeiert, als ich zunächst statt Assoziativität “Transitivität” gesagt hab^^ Hab schließlich aber noch auf die richtige Terminologie umgeschwitcht.

Dann sollte ich die Rewriteregel aufschreiben und erklären wie die angewandt wird. (Wir haben bisschen rumgehackt darauf, was t bei alles sein kann (alles nur keine Variable) und dass man beliebige Substitutionen der freien Variablen verwenden kann).

An die restlichen Fragen kann ich mich hier nicht mehr erinnern. Es ging auf jeden Fall noch um Anreicherungen (und, dass man bei data specification keine Axiome angeben kann -- war mir neu), was genau unter Hierarchiepersistenz und Eindeutigkeit zu verstehen ist, und ob Anreicherungen mit struk.-rek. / nicht-rek. definierten Formeln diese Eigenschaften haben.

KIV

Auch ich musste die intersection-Funktion zunächst definieren. Dazu wurde schon alles Relevante von Anderen geschrieben. Witzig war noch, dass zwischendurch plötzlich das Telefon geklingelt, alle 10 Sekunden Schockstarre verharrt waren und schließlich Alex mit den Worten “Ja, geh halt hin!” an Gerhard gerichtet, diese Spannung intelligent aufgelöst hatte und Gerhard schließlich mit dem Ausruf “Ja, Kilian. Was gibt’s denn?” den Hörer aufgenommen hatte und nach weiteren 3 Sekunden - etwas verwirrt - “Ja.. Äh... Ja.. Da müssen Sie sich wohl verwählt haben!” - wieder auflegte.

Bewertung

Die Bewertung war sehr nett; kurze Denkpausen und Verwirrungen wurden einfach ignoriert. Jedoch kam hier nochmals die Frage auf “Ja, warum geben Sie denn keine Hausaufgaben ab? Etwas besorgt waren wir ja schon.”, die ich dann etwas verlegen mit einer spontanen Ausrede beantwortet hatte. Das wurde mir aber bei der Bewertung nicht negativ angerechnet, daher war es also letztlich kein Problem.