Übung zur Vorlesung

Automatisierte Logik und Programmierung

Prof. Chr. Kreitz

Universität Potsdam, Theoretische Informatik — Sommersemester 2009

Blatt 5 — Abgabetermin: 23.7.2009

Aufgabe 5.1 (Optimierung)

In Übung 4 haben wir einen Globalsuch-Algorithmus für das n-Damen Problem synthetisiert.

```
FUNCTION queens(n:ℤ):Set(Seq(ℤ)) WHERE n≥1
   RETURNS {nq | perm(nq,{1..n}) ∧ safe(nq)}
≡ if nodups([]) ∧ safe([]) then queens_gs(n,[]) else ∅
FUNCTION queens_gs(n:ℤ,V:Seq(ℤ)):Set(Seq(ℤ))
  WHERE n≥1 ∧ range(V)⊆{1..n} ∧ nodups(V) ∧ safe(V)
  RETURNS {nq | perm(nq,{1..n}) ∧ V⊑nq ∧ safe(nq)}
≡     {nq | nq∈{V} ∧ perm(nq,{1..n}) ∧ safe(nq)}
  ∪ ⋃{ queens_gs(n,W) | W∈{V·i | i∈{1..n}} ∧ nodups(W) ∧ safe(W)}
```

Optimieren Sie diesen Algorithmus mit den in der Vorlesung vorgestellten Techniken.

Aufgabe 5.2 (Rückblick, 1. Teil)

Die folgenden Aufgaben sind vorgesehen als Kontrollfragen zur Überprüfung des eigenen Kenntnisstandes. Sie entsprechen in ihrer Thematik dem Spektrum einer mündlichen Prüfung. Die Antworten sind größtenteils im Skript bzw. auf den Folien enthalten, allerdings nur selten an auffälliger Stelle. Versuchen sie, diese zunächst ohne Ihre Unterlagen zu beantworten.

5.2–a Beschreiben Sie die wichtigsten Bestandteile eines Beweisassistenzsystems.

5.2–b Welche prinzipiellen Möglichkeiten einer Computerunterstützung für formaler Beweise gibt es?

5.2–c Welche Doppelrolle spielt die Sprache ML für interaktiver Beweissysteme?

5.2–d Schreiben Sie ein ML-Programm `prim`, welches einen Primzahltest durchführt.

5.2–e Durch welche Datenstruktur werden Sequenzenbeweise in ML repräsentiert?
Nennen sie auch die wichtigsten Zugriffs- und Manipulationsoperatoren.

5.2–f Welcher Mechanismus ist erforderlich, um konservative Erweiterungen der Typentheorie durch Abstraktionen fehlerfrei zu ermöglichen?

5.2–g Erklären Sie die besondere Rolle von Taktiken bei der Beweisführung.

5.2–h Warum sind Beweise, die mit Taktiken geführt werden, immer korrekt?

5.2–i Auf welche Arten kann ein Benutzer Taktiken konstruieren/programmieren? Nennen Sie Beispiele.

5.2–j Was ist der Unterschied zwischen Rewriting und taktischer Beweisführung? Wie kann man die beiden Techniken kombinieren?

5.2–k Unter welchen Voraussetzungen kann man Entscheidungsprozeduren in Beweissystemen einsetzen?

5.2–l Warum gibt es keine Entscheidungsprozedur für die gesamte Arithmetik?

5.2–m Welche Art von Problemen entscheidet die Prozedur `arith`?

5.2–n Erklären Sie die grundsätzliche Arbeitsweise der Prozedur `arith`.

5.2–o Erklären Sie die grundsätzliche Arbeitsweise der Prozedur `supinf`.

5.2–p Erklären Sie die grundsätzliche Arbeitsweise der Prozedur `equality`.

5.2–q Welche Probleme besitzen Beweissysteme, die sich ausschließlich auf Entscheidungsprozeduren und Benutzerinteraktion stützen?

5.2–r Welche Arten von Beweisführung gibt es für die Logik erster Stufe und wie funktionieren sie?

5.2–s Beschreiben Sie die Funktionsweise von JProver.

5.2–t Aus welchen Kompoenenten besteht eine formale Theorie? Geben Sie ein beispiel.

5.2–u Beschreiben Sie die wichtigsten Aufgaben formaler Wissensbanken?
Auf welche Arten kann man formale Wissensbanken gestalten?

5.2–v Auf welche Arten kann man das Benutzerinterface von Beweisassistenten gestalten. Geben Sie Vor- und Nachteile an.
Welche Besonderheit besitzt der Termeditor von NuPRL und welche Vorteile ergeben sich daraus?

Aufgabe 5.3 (Rückblick, 2.Teil)

5.3–a Warum kann es ein vollständiges Programmsynthesesystem niemals geben?

5.3–b Beschreiben Sie die Phasen einer formalen Entwicklung von Software.

5.3–c Welche Komponenten von Programmen und Spezifikationen sind für Programmsynthese von Bedeutung?

5.3–d Geben Sie eine präzise Definition der Begriffe “Programmkorrektheit” und “Erfüllbarkeit von Spezifikationen” auf der Basis der oben genannten Komponenten.

5.3–e Welche grundsätzlichen Paradigmen für Programmsynthese kennen Sie? Beschreiben Sie die grundsätzlichen Unterschiede bei der Darstellung des Problems und der Vorgehensweise zur Lösung.

5.3–f Welches Problem führt dazu, daß das Prinzip “Beweise-als-Programme” in der Praxis nur von geringer Bedeutung ist?

5.3–g Was ist die charakterisierende Eigenschaft einer Algorithmentheorie?

5.3–h Beschreiben Sie das generelle Verfahren zur Synthese von Algorithmen mit Hilfe von Algorithmenschemata. Wodurch wird dieses Vorgehen gerechtfertigt?

5.3–i Welche Vorteile hat die Synthese mit Algorithmenschemata gegenüber anderen Verfahren?

5.3–j Beschreiben Sie die allgemeine Struktur von Divide & Conquer Algorithmen und die Voraussetzungen für ihre Korrektheit.
Beschreiben Sie eine Strategie zur Erzeugung von Divide & Conquer Algorithmen (an einem Beispiel)

5.3–k Beschreiben sie die allgemeine Struktur von Globalsuchalgorithmen und die Voraussetzungen für ihre Korrektheit. Beschreiben Sie die allgemeine Strategie zur Erzeugung von Globalsuchalgorithmen.

5.3–l Auf welche Art könnte man Synthesestrategien in das formale Konzept typentheoretischer Schlußfolgerungen integrieren?

5.3–m Beschreiben Sie die allgemeine Struktur von Problemreduktionsgeneratoren und die Voraussetzungen für ihre Korrektheit. Beschreiben Sie die Strategie zur Erzeugung von Problemreduktionsgeneratoren.

5.3–n Beschreiben Sie die allgemeine Struktur von Lokalsuchalgorithmen und die Voraussetzungen für ihre Korrektheit. Beschreiben Sie die allgemeine Strategie zur Erzeugung von Lokalsuchalgorithmen.

5.3–o Beschreiben Sie diewichtigsten Techniken zur Optimierung von synthetisierten Algorithmen.