

Protokoll 15.05.2017

Team: 7

Beginn: 16:00

Ende: 18:30

Protokoll: Manuel Grandt

Inhalt

- Aufgabenstellung
- Lösung
- Abgabe
- Ergebnis

Top 1: Aufgabenstellung

Erstellen Sie ein Programm, das ein 9x9 Sudoku löst.

Gegeben ist die folgende Startbelegung:

Programmieren Sie ein Prolog-Programm, das dieses Rätsel komplett und konsistent mittels Constraints als CSP löst.

- Nutzen Sie die SWI-Prolog-Library `clpfd`.
- Gestalten Sie Ihr Programm derart, dass auch andere Startbelegungen angegeben werden können (reicht im Programm selbst).
- Erzeugen Sie eine übersichtliche Ausgabe, aus der Zeilen, Spalten und die Belegung der einzelnen Felder ersichtlich werden.

Top 2: Lösung

Jedes Feld wird angesprochen und erklärt, dass sie eine Zahl zwischen 1 und 9 enthält.

Anschließend wird über jede Zeile, Spalte und jeden Block iteriert und immer erklärt, dass die Felder jeweils unterschiedlich sein müssen. Es wurde mit `all_distinct` und `all_different` experimentiert, wobei wir festgestellt haben, dass schwerere Aufgaben mit `all_distinct` besser gelöst werden als mit `all_different`, für einfache Aufgaben `all_different` allerdings deutlich besser abschneidet.

Anschließend werden die Felder gelabelt.

Top 3: Abgabe

Die erste Vorstellung unserer Lösung wurde nicht abgenommen, da wir zu viel ausgeschrieben hatten.

Nach einer Überarbeitung haben wir erneut versucht unsere Lösung abzugeben, es wurde bemängelt, dass wir nicht mit „`maplist`“ arbeiten, sondern mit selbst geschriebenen Hilfsfunktionen.

Es wurden Fragen zur Arbeitsweise von Prolog beantwortet und was für Schwachstellen Constraints haben. Wir wurden aufgefordert ein Sudoku mit unserer Lösung laufen zu lassen, welches in der ersten Zeile keine vordefinierten Werte hat.

Die Lösung wurde nicht abgenommen.

Nachdem wir unsere Hilfsfunktionen durch „maplist“ ersetzt haben und ein Sudoku gefunden und mit unserem Programm gelöst haben, haben wir erneut um Abnahme der Lösung gebeten.

Diesmal wurde die Lösung abgenommen.

Top 4: Ergebnis

Die besten Ergebnisse:

- Das Sudoku aus der Aufgabenstellung lösen wir mit 69.831 Inferenzen mit `all_different` und `labeling([ff],sudoku)`.
- Das Sudoku aus dem Praktikum lösen wir mit 108.485 Inferenzen mit `all_different` und `labeling([],sudoku)`.
- Das schwere Sudoku, welches in der ersten Zeile keine Werte hat lösen wir mit 657.141 Inferenzen mit `all_distinct` und `labeling([],sudoku)`.

In der ZIP liegen bei:

`aufgabe2.pl`

`start_belegung.pl`

`aufgabe_2_test.pl`

Der Constraintsolver

Die Sudoku mit denen wir gearbeitet haben

Unsere Tests