

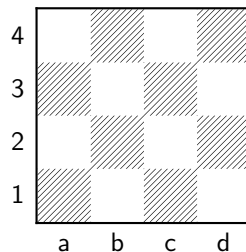


Artificial Intelligence, Wintersemester 2012/2013 Übungsblatt 9

Abgabe: 10.01.2013, Besprechung: 17.01.2013

Aufgabe 1 Vier-Damen-Problem [4 Punkte]

Vier Damen ♔ sollen auf dem unten abgebildeten Schachbrett so platziert werden, dass keine eine andere bedroht.



- (a) Definieren Sie zunächst die Variablen und die Definitionsbereiche.
- (b) Lösen Sie dieses Problem mit einfachem **Backtracking** und stellen Sie die Suche graphisch in Form eines Baumes dar.
- (c) Verwenden Sie zur Lösung nun **Backtracking** mit **AC-3** und kennzeichnen Sie die einzelnen Schritte, die sich durch AC-3 ergeben.

Aufgabe 2 CSP: Kartographische Färbung (map coloring) [8 Punkte]



Ziel ist es, jeder Region mittels **Backtracking** eine der Farben *rot* (r), *grün* (g) oder *blau* (b) zuzuweisen, so dass benachbarte Regionen jeweils unterschiedliche Farben haben.

- (a) Lösen Sie das Problem mittels **Backtracking mit Forward Checking**. Belegen Sie die Variablen in der Reihenfolge WA, NSW, NT, V, Q, SA, T und weisen Sie die Farben in der Reihenfolge r, g, b zu.

Formatieren Sie Ihre Abgabe bitte wie in folgendem Beispiel und geben Sie eine gewählte Belegung durch den entsprechenden Großbuchstaben (z. B. R) an:

	D_{WA} rgb	D_{NSW} rgb	D_{NT} rgb	D_V rgb	D_Q rgb	D_{SA} rgb	D_T rgb
1) $WA \leftarrow r$	R	rgb	gb	rgb	rgb	gb	rgb
2)
3)

- (b) Wieso ist der einfache Forward-Checking-Algorithmus nicht optimal?
 (c) Lösen Sie das Problem nun mit **Forward Checking** und **AC-3**.

Aufgabe 3 CSP: Sudoku [8 Punkte]

In der Datei `uebung09.zip` finden Sie im Paket `csp` ein Framework für das Lösen von CSPs. Im Unterpaket `csp.sudoku` ist Sudoku als CSP in diesem Framework implementiert. Im Unterpaket `csp.solver` finden Sie einen teilweise implementierten Solver für CSPs, den Sie vervollständigen sollen.

- (a) Implementieren Sie den **Backtracking**-Algorithmus, indem Sie dazu alle mit `TODO` markierten Stellen in der Klasse `BackTrackingSearch` vervollständigen.
 (b) Implementieren Sie den **AC-3**- und den **MAC**-Algorithmus, indem Sie dazu alle mit `TODO` markierten Stellen in der Klasse `ArcConsistency` vervollständigen.

Ihre Implementierungen können Sie an den drei mitgelieferten Sudokus in den Dateien `sudoku1.txt`, `sudoku2.txt` und `sudoku3.txt` (dieses braucht länger) testen.

Senden Sie Ihre kommentierte Implementierung sowie Ihre Ergebnisse an florian.mittag@uni-tuebingen.de und andreas.draeger@uni-tuebingen.de mit dem Betreff „Abgabe KI-Uebung 09“. Bitte geben Sie in jeder Datei Ihren vollständigen Namen an, und senden Sie nur die Quellcode-Dateien ein, die Sie laut Aufgabenstellung modifiziert haben.