

# Grundlagen der Künstlichen Intelligenz (CS 205)

Prof. Dr. M. Helmert  
Dr. M. Wehrle  
Frühjahrssemester 2014

Universität Basel  
Fachbereich Informatik

## Übungsblatt 2

Abgabe: 21. März 2014

### Aufgabe 2.1 (3+1 Punkte)

- (a) Formalisieren Sie den Zustandsraum des 15-Puzzles aus der Vorlesung. Spezifizieren Sie hierzu die Menge der Zustände, der Aktionen mit entsprechenden Aktionskosten und der Transitionen. Geben Sie weiterhin den Anfangszustand und die Menge der Zielzustände an.
- (b) Wie viele Zustände hat das 15-Puzzle?

### Aufgabe 2.2 (0+6+2 Punkte)

Bei dieser Aufgabe handelt es sich um eine Programmieraufgabe. Wir erwarten, dass Sie Ihre Implementierung selbständig, das heisst ohne Anwendung von fremdem Code z.B. aus dem Internet erstellen. Uns ist bewusst, dass Programmieraufgaben aufwendiger sind als die üblichen theoretischen Aufgaben und helfen bei technischen Schwierigkeiten und Verständnisproblemen gerne weiter. Bitte wenden Sie sich dazu *mit genügend zeitlichem Abstand zum Abgabetermin* an Lukas Beck oder Martin Wehrle.

- (a) Auf der Vorlesungsseite befinden sich Beispielimplementierungen in Java für einen trivialen Zustandsraum sowie für den Zustandsraum des Blocks-World-Problems aus der Vorlesung. Die Zustände des trivialen Zustandsraums bestehen aus den Integern 1,2 und 3, die man mit entsprechenden Aktionen inkrementieren und dekrementieren kann. Beide Beispielimplementierungen implementieren die bereitgestellten Interfaces `State` und `Action` sowie das Black-Box-Interface `StateSpace`, das Sie in der Vorlesung kennengelernt haben. Machen Sie sich mit diesen Implementierungen vertraut. Testen Sie die Implementierungen mit der Klasse `StateSpaceTest`, die vom initialen Zustand aus auf zufällige Art Nachfolgezustände berechnet.
- (b) Implementieren Sie nach dem Vorbild von (a) den Zustandsraum für *entweder* das 15-Puzzle *oder* das Pancake-Problem, das Sie in den Anwesenheitsübungen kennengelernt haben (je nach dem, welches Problem Ihnen besser gefällt). Implementieren Sie hierzu die bereitgestellten Interfaces `State`, `Action` und `StateSpace`.
- (c) Implementieren Sie nach dem Vorbild der Blocks-World-Implementierung von (a) eine Methode `buildFromCmdline`, die ein Input-Puzzle oder Input-Pancake parst (als Datei oder direkt von der Konsole) und einen entsprechenden Zustandsraum erzeugt. Ein Input-Puzzle kann in einer Zeile kodiert werden, die aus einer Permutation von 16 Zahlen zwischen 0 und 15 besteht, die den Anfangszustand des Puzzles beschreibt. Analog kann ein Input-Pancake in einer Zeile als Permutation von Zahlen kodiert werden. Testen Sie ihre Implementierung, indem Sie die bereitgestellte Klasse `StateSpaceTest` entsprechend erweitern.

Reichen Sie bitte den Code inklusive Hinweisen zu Kompilierung und Aufruf bei `courses` ein (sofern nicht selbsterklärend).

*Die Übungsblätter dürfen in Gruppen von zwei Studierenden bearbeitet werden. Bitte schreiben Sie beide Namen auf Ihre Lösung.*