

Aufgabenblatt 2

7	2	4
5		6
8	3	1

	1	2
3	4	5
6	7	8

Abb. 1: Startzustand S (links) und Zielzustand Z (rechts) des 8-Puzzle.

Aufgabe 1 Parität (10 %)

Ein Zustand des 8-Puzzle (z.B. Zustand S in Abb. 1) kann verändert werden, indem ein dem leeren Feld benachbarter Stein (linker, rechter, oberer oder unterer Stein) in das leere Feld verschoben wird.

Ganz allgemein kann ein Zustand z_1 in ein Zustand z_2 durch eine Folge von Zügen überführt werden, wenn die beiden Zustände z_1 und z_2 dieselbe *Parität* haben (<https://de.wikipedia.org/wiki/15-Puzzle>).

Die *Parität* (im 8-Puzzle) gibt an, ob die Anzahl der *falschen Zahlenpaare* gerade oder ungerade ist. Ein *Zahlenpaar* (x,y) ist *falsch*, falls $x < y$ und der Stein x auf dem Stein y folgt, gleich wie viele Steine dazwischen liegen. Das leere Feld wird dabei ignoriert. Bei einem Zustand mit der Folge 1, 4, 2, 6, 7, 8, 3, 5 gibt es folgende falsche Zahlenpaare: (2,4), (3,8), (3,7), (3,6), (3,4), (5,8), (5,7), (5,6). Das sind 8 Paare und damit ist die Parität gerade. Daher kann dieser Zustand überführt werden in den Zielzustand 1, 2, 3, 4, 5, 6, 7, 8, da dieser mit 0 falschen Zahlenpaaren eine ebenfalls gerade Parität hat.

- Welche Parität hat der Zustand S in Abb. 1?
- Implementieren Sie eine Funktion zur Berechnung der Parität eines Zustands.

Aufgabe 2 Heuristiken (30 %)

Schauen Sie sich das in der Vorlesung besprochene A*-Verfahren und die beiden Heuristiken h_1 und h_2 an.

- Für einen Zustand z ist $h_1(z)$ die Anzahl der falsch platzierten Steine in Bezug auf den Zielzustand. In Abb. 1 ist $h_1(S) = 8$. Wieso ist h_1 eine monotone Heuristik?
- Für einen Zustand z ist $h_2(z)$ die Summe der Manhattan-Distanzen der Steine von ihren Zielpositionen. In Abb. 1 ist $h_2(S) = 3 + 1 + 2 + 2 + 2 + 3 + 3 + 2 = 18$. Wieso ist h_2 eine monotone Heuristik?
- Wieso ist $h_1(n) \leq h_2(n)$? Welche Heuristik ist also besser?
- Implementieren Sie beide Heuristiken.

Aufgabe 3 **Suchverfahren IDS und A*** **(60 %)**

- a) Implementieren Sie Suchverfahren für das 8-Puzzle in zwei Varianten:
- Iterativ vertiefende Suche (iterative deepening depth-first search IDS).
 - A* mit einer der Heuristiken aus Aufgabe 2. Gerne dürfen Sie auch eine andere Heuristik einsetzen. Ausgangspunkt ist die in der Vorlesung besprochene uniforme Kostensuche auf S. 2-42, die um eine Heuristik erweitert wird.
- b) Testen Sie Ihre Suchverfahren für zufällig generierte Startzustände. Beachten Sie dabei die Paritätsüberlegung in Aufgabe 1.
- c) Bestimmen Sie die Anzahl der vom Suchverfahren generierten Zustände und die Länge der Lösungsfolge für verschiedene Startzustände.
- d) Sind Ihre Zugfolgen optimal? Wenn ja, warum?
- e) Welches Problem könnte entstehen (nicht ausprobieren!), falls A* für die Lösung des 15-Puzzle eingesetzt werden würde?

Implementierungshinweise

- Auf der Moodle-Seite finden Sie ein Programmgerüst in Java. Viele Teile sind schon vorgegeben, so dass Sie sich auf die wesentlichen Aspekte der Suchverfahren konzentrieren können.
- Sie sind natürlich frei eine komplett eigene Implementierung in einer Programmiersprache Ihrer Wahl zu erstellen. Besprechen Sie das aber bitte vorher mit mir.