

### 1. 8-Puzzle mit Uniform Cost verbessern?!

Erweitern Sie Ihre Lösung des 8-Puzzle aus dem letzten Seminar dahingehend, dass Sie eine

- Uniform Cost Expansion ( $f(n) = g(n)$ )

implementieren.

Welchen Einfluss hat dies auf die Anzahl der expandierten Knoten für die Suche (erweitern Sie die Tabelle aus Seminar 7)? Wie unterscheidet sich diese Suche von einer Breitensuche? Ist das Ergebnis der Suche optimal?

### 2. Gib Kruskal eine Chance

Implementieren Sie den Algorithmus von Kruskal zur Bestimmung des minimalen Spannbaums. Achten Sie auf Effizienz bei der Entwicklung der *Disjoint Sets*.

Zur Kantensortierung eignet sich der Einsatz einer *PriorityQueue* – oder zur Effizienzsteigerung ein *Fibonacci Heap* (der allerdings nicht in den Java Standardklassen zu finden ist).