Praktische Informatik 2 — SoSe 98

Prof. Bernd Krieg-Brückner
Axel Lankenau
Shi Hui
Christoph Lüth

7. Übungsblatt

Ausgabe: 13.07.98

Abgabe: 10.08.98

$\boxed{1}$ *Bonustrack...* *Wichtung:* $2 \times 50\%$

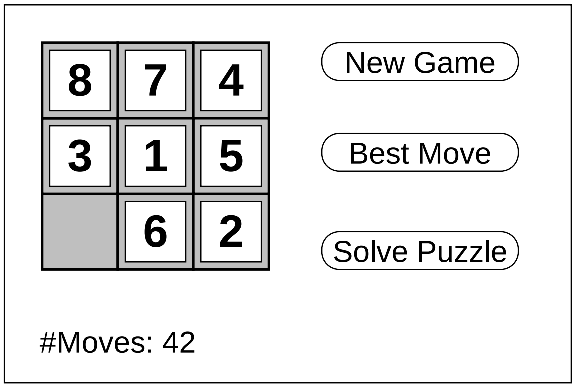
Implementieren Sie das in Anwendungsaufgabe 2 geforderte JAVA–Applet für das 9er–Puzzle, indem Sie jeder Position die Zugzahl, die zum Erreichen des Ziels noch nötig ist, zuordnen. Die Zielposition erhält also die 0 zugeordnet, die beiden Positionen, aus denen man die Zielposition erreichen kann, die 1, usw. Schließlich läßt sich für jede Position „durch Nachgucken" angeben, welches der beste Zug ist.

Besonders interessant ist hier der Algorithmus, der die Zuweisung der Entfernung zum Ziel übernimmt. Wie werden Zyklen vermieden? Wann weiß man, daß es keinen kürzeren Weg aus einer Position mehr geben kann? Wie werden unlösbare Positionen erkannt?

Die Oberfläche des Applets sollte durch eine definierte Schnittstelle von dem steuernden Algorithmus getrennt sein. Sie soll das Spielfeld visualisieren, Züge des Benutzers ausführen können und mindestens folgende Buttons bereitstellen:

`New Game`	Baut eine neue Startposition auf
`Best Move`	Schlägt den bestmöglichen Zug vor
`Solve Puzzle`	Löst die aktuelle Position, indem die nötigen Züge nach und nach ausgeführt werden

Die Anordnung der einzelnen Komponenten könnte in etwa wie folgt aussehen:



Die Zugeingabe kann z. B. durch Mausklick auf den zu bewegenden Stein realisiert werden.

Idealerweise finden sich für die Lösung dieser Aufgabe zwei Gruppen zusammen, die die Schnittstelle zwischen graphischer Darstellung und Algorithmus gemeinsam definieren. Danach kann dann eine Gruppe die Visualisierung realisieren, während sich die andere um die Implementierung des Algorithmus und der Datenstrukturen kümmert. Jede Teilaufgabe bringt bis zu 50% ein. ***Die in dieser Aufgabe erreichten Punkte können zum Ausgleich früherer Defizite verwendet werden!***