

## Algorithmen und Datenstrukturen

Sommersemester 2013

### 3. Übungsblatt

#### Aufgabe 1: Java - Finden eines Weges in einem Labyrinth mittels Rekursion

In dieser Aufgabe soll eine **rekursive** Methode realisiert werden, welche einen Weg von einem vorgegebenen Startpunkt zu einem vorgegebenen Zielpunkt in einem Labyrinth findet. Das Labyrinth wird durch ein zweidimensionales Feld mit `int` - Werten modelliert. Der Wert 0 repräsentiert ein leeres Feld. Felder mit dem Wert 9 sind blockiert (Wand), d.h. über dieses Feld kann kein Weg führen. In einem Schritt kann ein Weg zum rechten Nachbarfeld, zum unteren Nachbarfeld, zum linken Nachbarfeld oder zum oberen Nachbarfeld führen.

Beispiel:

	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	9	0	9	0	9
2	0	0	9	0	0	0
3	9	0	9	0	9	9
4	0	0	9	0	0	0
5	0	9	0	9	0	0

Start sei das Feldelement mit den Koordinaten (5,0), Ziel sei das Feldelement (5,5).

Ein Weg soll in dem Feld wie folgt markiert werden:

- 1: Der Weg geht nach rechts
- 2: Der Weg geht nach unten
- 3: Der Weg geht nach links
- 4: Der Weg geht nach oben
- 5: Der Weg endet, da das Ziel erreicht ist

Ein Weg von (5,0) nach (5,5) kann damit wie folgt angegeben werden:

	0	1	2	3	4	5
0	1	1	1	1	2	0
1	4	9	0	9	2	9
2	4	3	9	2	3	0
3	9	4	9	2	9	9
4	1	4	9	1	1	2
5	4	9	0	9	0	5

Implementieren Sie eine **rekursive** Methode

```
public static boolean probieren(int zeile, int spalte),
```

die einen Weg von einem gegebenen Startfeld zu einem gegebenen Zielfeld wie oben angegeben markiert und *true* zurückliefert, wenn es einen Weg gibt. Wenn es keinen Weg gibt, soll die Methode dies erkennen und *false* zurückliefern.

Auf der Website zur Übung befindet sich ein Grundgerüst, so dass es genügt die Methode an der mit `TODO Übung 3 Aufgabe 1` markierten Stelle zu implementieren. Kommentieren Sie Ihren Quellcode sinnvoll.

**Wichtig:** Es ist **nicht** gefordert, einen kürzesten Weg zu finden. Es genügt, irgendeinen Weg zu finden. Testen Sie Ihre Methode ausführlich. (15 Punkte)

## Aufgabe 2: Stackverwaltung in rekursiven Methoden

Eine rekursive Lösung des Problems der Türme von Hanoi sei analog zur Vorlesung wie folgt gegeben:

```

1 public static void rekhanoi(int start, int ziel, int hilf, int hoehe){
2     /* Start von rekhanoi */
3     if (hoehe>0) {
4         rekhanoi(start, hilf, ziel, hoehe-1) /* nach 1. Rekursion */;
5         zug(start, ziel);
6         rekhanoi(hilf, ziel, start, hoehe-1) /* nach 2. Rekursion */;
7     }
8 }

10 public static void main(String[] args){
11     rekhanoi(1,2,3,2) /* nach Aufruf in main */;
12 }
```

Zeichnen Sie den gesamten Programmstack (vgl. Kap. 0.6 Folie 119) jeweils an der mit `Start von rekhanoi` bezeichneten Programmstelle. Verwenden Sie zur Bezeichnung der Rückkehrpunkte die im obigen Programm angegebenen Kommentare. (10 Punkte)

**Beachten** Sie die Hinweise auf dem ersten Übungsblatt.

---

**Abgabetermin:** Donnerstag, den 25. April bis 10 Uhr im Institut für Informationssysteme, 2. OG im Informatik-Neubau (Raum 2.071, Küche neben Sekretariat)