# GWV – Grundlagen der Wissensverarbeitung

## Tutorial 3: Searching

**Exercise 1.1: (Search Space Construction 1)**

| |
|---|
| of |
| 0 |

CC-BY enwiki user Annielogue

In peg solitaire (depicted above), you need to clear the board (with only one peg left) by repeatedly jumping with one peg over another one, removing the latter from the board. Construct an appropriate search space for the game.

**Exercise 1.2: (Search Space Construction 2)**

| |
|---|
| of |
| 0 |

In the game Scotland Yard, Mister X has to evade several detectives using different means of transportation (and spending tickets). Suppose (against the rules of the game) you as Mister X has a fixed amount of steps before the detectives may move a fixed amount of steps. How would you find a place to go where the detectives can't reach you?

**Exercise 1.3: (Blind Search)**

```
xxxxxxxxxxxxxxxxxxx
x                 x
x       xxx       x
x       x xxxxx   x
x   s     x       x
x       x x  xxxxxxx
x  xx xxxxx       x
x      x      g   x
x      x          x
xxxxxxxxxxxxxxxxxxx
```

The above figure shows an environment for a robot in an ASCII-Art representation. The robot starts in the field s (start) and wants to get to the field g (goal). The robot can move one field at a time in any of the four directions (up, down, left, right). The fields with an x denote a blocked field that the robot can not enter.

*This assignment teaches the basics of blind search strategies and should also familiarize you with the practical aspect of artificial intelligence, that is you get to write a small program.*

1. Build up an internal representation of the environment that is suitable for searching. *Hint: It's easier if your program can read the environment from the ASCII files provided in the nats wiki (`http://nats-www.informatik.uni-hamburg.de/GWV1415/`).*
2. Implement the blind search strategy "breadth first" to find a path for the robot. (4 Pt.)
3. Implement the blind search strategy "depth first" to find a path for the robot. (4 Pt.)
4. Describe problems of the search strategies you encounter. (1 Pt.)
5. What other problems could your search strategies encounter in other environments? Provide example environments. (2 Pt.)
6. Can you think of possible ways to cope with the problems you identified? (1 Pt.)

Hand in the documented program code and a suitable human-readable output of the search process.

*Version: October 23, 2015*
*Achievable score on this sheet: 12*