GWV – Grundlagen der Wissensverarbeitung

Tutorial 3 : Searching

Class Exercise 3.1 : (Scotland yard again)

In Scotland Yard, you can take taxis, busses, and subways to move around. Each time you use one of these, you have to pay one ticket that fits. Suppose you are Mister X. You have tickets for taxi, bus, and subway as well as black tickets which can be used for all three modes of transportation. The tickets differ in value
(black > subway > bus > taxi).

1. Again, Mister X takes some steps and then the detectives do (and then the game is over). Find a way to escape the detectives and keep the best tickets. What kind of search would you use?
2. Suppose you had a fixed set of tickets and were tasked to go as far as possible from a given location (measured in centimeters on the board) using these tickets. How would you compute the best moves?

Formal Requirements for Programming Assignments

You can use whatever programming language you like. Strengthen your programming skills in a language you're familiar with or try something new? It's up to you! But: Bear in mind that everybody from a group must be able to explain what the program does when looking at the program code. We will have a look at and talk about the program code next week.

Please hand in your solutions as two attachments:

- your program code in a compressed archive file (`.tar.gz`, `.zip`, . . .)
- one PDF file

Please keep using the format for the subjects of your e-mails as specified before and `name1_name2_...nameN_TutorialX.pdf` as name for the PDF file and use the same format (replace `.pdf` with `.tar.gz`, `.zip`, . . .) for the archive file.

Put the following in the PDF file for each subtask (if applicable):

- the location where the code implementing that subtask can be found (e.g. file and method name)
- if your code does not work: what you tried to do and what does not work
- an output (screenshot, text, . . .) of your program and an explanation what the output shows
- the answer/solution to the subtask

If you don't mention a subtask in your PDF, it is assumed that you did not work on that subtask.

Exercise 3.2 : (Blind Search)

of
12

```
xxxxxxxxxxxxxxxxxxxx
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
xxxxxxxxxxxxxxxxxxxx
```

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.
2. Your program should be able to read the ASCII labyrinths provided in the nats wiki.
3. The program should be able to display / print search states in some way.

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

Version: October 23, 2019
Achievable score on this sheet: 12