

CSE 512 – Winter 2019 – Lab 34

Instructor: Kerstin Voigt
Tuesdays 1:30-3:20pm in JB 359

In this lab you will complete a Python implementation of a 8-puzzle solving program. Your program will solve the 8-puzzle first by “best-first search” and then by the superior A-star (A*) search algorithm. Both programs are variants of the general “graph search algorithm” in Nilsson, Chapter 8.

Exercise 1: Obtain copies of files

- bestfirst_astar_search_lab4.py and
- puzz8_lab4.py

The code in bestfirst_astar_search_lab4.py will have been discussed in class; because of its relative complexity, some additional lab time should be spent to understand it.

The code in puzz8_lab4.py is INCOMPLETE. Fill in the missing parts (there will be some running discussion on how to this in the lab). Once done, load and run bestfirst_astar_search_lab.py. This file is set up to run as “main”. See what you get ...

Exercise 2: Obtain a copy of file Tiles1to8.py. The instructor will assist you in using this program to graphically play out the solution path for the 8-puzzles that were run in Exercise 1.

Exercise 3: With minimally changes (no more than replacing the parameters to the search functions), the bestfirst- and A-star –implementations can be run for virtually any other problem domain that has a

1. A defined *representation of “state”* (the problem being the finding of a sequence of operations that will convert an initial artifact into a goal artifact).
2. A **goal function** (true with a state matches up with a goal state).
3. A **successor function** (given a state it will produce all possible next states the can be produced in one move)
4. An **evaluation function** to judge the proximity of a state to the goal state; the evaluation should ideally be “admissible.”

Go over puzz8_lab4.py again and confirm that each of these four ingredients has been provided. This can be done for any other puzzle. Demonstrate this fact by providing the equivalent to puzz8_lab4.py for the “Water Jug Problem”.

The Water Jug Problem: There are two water jugs; one holds 3 liters, the other holds 4 liters. The jugs may be handled in the following ways: a jug can be filled to capacity, a jug can be emptied completely, one may pour water from one jug into the other jug until either the first jug is empty, or the second jug is at capacity. There is no manner of “measuring” amounts of water. An unlimited supply of water is provided. Initially, both jugs are empty. ***Which sequence of actions (fill, dump, pour) will result in a goal state where the 4-liter jug holds 2 liters and the 3-liter jug is empty?***

The **completion of Exercise 3** is also your **Homework Assignment 2**: The assignment is due on *Thursday, Feb 7, at the beginning of the lecture*. Submit (1) a hardcopy of `bestfirst_astart_search_lab4.py` with appropriately changed parameters to the search function calls, (2) a copy of your file `jugs34.py` (analogous to `puzz8_lab4.py`), (3) a hard copy of a typescript/screenshot (Idle has a File -> ‘Print Window’) which show how best-first and A-star solve the Water Jug Problem.

Credit for this lab: (1) Work diligently on the exercises above. Keep in mind that Exercise 3 is also a next homework assignment, so make good use of your time. (2) Sign your name on the sign-up-sheet which will be circulated toward the end of this lab session.