SOLUTION Informed Search - Prac Sheet

Last modified on Sunday, 21 March 2021 by f.maire@qut.edu.au

This prac is a preparation for the first assignment. In the first assignment, the main task is to define and implement a class derived from the *search.Problem* class in order to solve a planning problem. In this prac, you also define and implement a class derived from the *search.Problem* as a way to find optimal sequences of actions, but on a much simpler problem.

Exercise 1 – A* implementation

Edit the file *W04_search.py* to implement the function *astar_graph_search.*

Hint: look at the implementation of *astar_tree_search.*

See file SOLUTION_W04_search.py

Exercise 2 – Creating your own *search.Problem* class

Create a class derived from the *search.Problem* class to solve the "pancake puzzle" introduced in Week 03 lecture. Implement a search heuristic function for this problem. Once your *PancakePuzzle* class is completed, compare BFS and A* search.

Recall that in Week 03 prac, you used code that defined a class *Sliding_puzzle.* This class gives you a good example of the definition of a derived class of the *search.Problem* class.

See file SOLUTION_pancake_puzzle.py

Exercise 3 – Admissible heuritics (non-programming exercise)

Assuming that h1 and h2 are admissible, which of the following expressions are also guaranteed to be admissible?

(i) *h1 + h2* ***no, consider h1=h2=h* 2 h* is not smaller or equal to h****

(ii) *h1 * h2* ***no, same counter example as for (i)***

(iii) max(*h1, 0.3*h2*) ***yes, both expressions smaller than h****

(iv) min(*h1 , 3*h2*) ***yes, same reason as for (iii)***

(v) *0.94*h1 + 0.08*h2* ***no, same counter example as for (i)***