

ECS 170: Introduction to Artificial Intelligence

UC Davis

Spring 2024

Homework 1

1. Instructions

In this assignment, you will implement the A* algorithm in Python, and apply it to the problem below. The helper code you need is provided on Canvas, you just need to fill in the missing parts.

You need to implement:

- function `Astar` in `search.py`
- function `heuristic` in `search.py`
- the following methods of the `TilesNode` class:
 - `TilesNode.is_goal`
 - `TilesNode.generate_children`

Resources: You can use built-in functions & data structures, and the standard library, but you cannot use/import anyone else's code or any package that is not in the Python standard library.

Testing: You can use `test.py` to test your code:

```
python -m unittest test.py
```

However, your code will be tested on some secret instances of the problems; therefore, you should be careful about the boundary cases. Note that you should not necessarily expect your algorithms to solve every instance. Difficult instances may require too much time or memory; that does not mean your implementation is incorrect. Of course, your code is expected to output the right answer when it outputs something.

Submission: You should submit `problems.py` and `search.py` to the Gradescope assignment "Homework 1", and you should upload them directly without zipping.

2. Problem Description

The problem that you will solve using A* is the classic fifteens puzzle (the four-by-four version of the eights puzzle studied in class) where you can move any horizontally or vertically adjacent tile into the empty slot.

For example, here is the goal state:

1	2	3	4
---	---	---	---

5	6	7	8
9	10	11	12
13	14	15	

The following states are one move away from the goal state:

1	2	3	4
5	6	7	8
9	10	11	12
13	14		15

1	2	3	4
5	6	7	8
9	10	11	
13	14	15	12

Every move has a cost of 1, and of course, since you are using A*, your program should find the optimal (lowest-cost) solution. In principle, there may be more than one optimal solution; your program is expected to give one of these – it does not matter which one.

You are free to design your own heuristic – keep in mind what we discussed in class about admissibility and consistency.