

Washington State University  
School of Electrical Engineering and Computer Science  
Fall 2018

CptS 440/540 Artificial Intelligence

**Homework 5**

Due: October 4, 2018 (11:59pm)

**General Instructions:** Your solution to this homework should consist of Agent.h and Agent.cc files (or a PyAgent.py file), perhaps with a readme.txt file. Put your files into one zip file, with the name <yourLastName>-<yourFirstName>.zip and submit as an attachment under Content → Homework 5 for the course CptS 440 Pullman (all sections of CptS 440 and 540 are merged under the CptS 440 Pullman section) on the Blackboard Learn system by the above deadline. Note that you may submit multiple times, but we will only grade the most recent entry submitted before the above deadline.

1. For this problem, you will implement a problem-solving agent, where search is used to move your agent to desired locations. The file “wumpus-search.zip” that accompanies this homework description<sup>1</sup> provides an A\* search using the city-block heuristic. The search algorithm is implemented in the files Search.h, Search.cc and Search.py. The zip file also includes modified Agent.h, Agent.cc and PyAgent.py files that demonstrate the use of the search capability. Copy all these files, including the updated Makefile, to a copy of the simulator code and recompile to use the search capability. You will mainly need to use the following three methods of the SearchEngine class:
  - `AddSafeLocation(x,y)` – Adds a location to the search engine’s list of safe locations. The search engine considers only safe locations in its solution.
  - `RemoveSafeLocation(x,y)` – Removes a safe location from the search engine’s list of safe locations (e.g., in case a previously-thought safe location turns out to be unsafe).
  - `FindPath(startLocation,startOrientation,goalLocation,goalOrientation)` – Returns a list of actions to get from the start state to the goal state. If no safe path is possible, then returns an empty list.

Specifically, your agent should do the following.

- a. In addition to the agent’s state information, you should also keep track of the wumpus location, the gold location, and any pit locations, as you learn this information. You will be playing each world for multiple tries, so keeping track of what you learned from previous tries is useful (and required).
- b. If you perceive a Glitter, then Grab.
- c. If you have the gold and are in the (1,1) location, then Climb.
- d. If you know the gold’s location, and you don’t have the gold, then use the search engine to find a sequence of actions to get you there.

---

<sup>1</sup> Also available at <http://www.eecs.wsu.edu/~holder/courses/AI/hw5/wumpus-search.zip>.

- e. If you have the gold, but aren't in the (1,1) location, then use the search engine to find a sequence of actions to get you to (1,1).
- f. Otherwise, update the safe location information in the search engine, determine a safe unvisited location, and use the search engine to find a sequence of actions to get there.

Submit your Agent.h and Agent.cc files, or PyAgent.py file, along with an optional readme.txt file containing any information you think we may need about your agent. Your agent should not require any user input. Your agent will be tested by copying only your Agent.h and Agent.cc files, or PyAgent.py file, into a fresh copy of the simulator code (that includes the new Makefile, Search.h, Search.cc and Search.py), compiling and running it on several test worlds. *The test worlds will all be 4x4, and there will always be a way to safely get the gold without having to shoot the wumpus.* Your agent will be given 10 tries on each test world. Your grade will be based on satisfying the above requirements, good programming style<sup>2</sup>, and your agent's scores on the test worlds.

---

<sup>2</sup> Good programming style guides for C++ and Python are available on the course website.