CSC14003 – Artificial Intelligence

PROJECT 02: LOGIC

Project description

The purpose of this project is to design and implement a logical search agent for a partially-observable environment. This will be accomplished by implementing an agent that navigates through the Wumpus World.

In summary, the Wumpus World presents key features:

- A network of interconnected 2D caves.
- Rooms that may harbor deadly pits, signaled by a perceivable breeze.
- Presence of a Wumpus in one of the rooms, detectable through a discernible stench.
- We have one arrow that we can shoot in the direction we are facing.
- A quest for a hidden pot of gold.
- Movement options: forward, backward, left, or right by 90 degrees.

The primary objectives encompass locating the gold and potentially eliminating the Wumpus to ensure success in this environment.

Wumpus World

We will modify the Wumpus world as such:

- The world will be limited in (10 x 10) instead of (4 x 4).
  - Room (1, 1) will still be the bottom-left one
  - Room (10, 10) the top-right one.
  - First number is room position in horizontal coordinate.
  - Second number is room position in vertical coordinate.
- Agent can appear in any Room (x_a, y_a) and always facing to the right. This room is the only room have the cave door.
- There may be any number of pits and gold in the world.
- There is at least one Wumpus.

- The agent carries an infinite number of arrows.
- The game will end when one of the following three conditions occurs:
  - The agent dies
  - The agent kills all of the Wumpus AND grabs all the gold
  - The agent climbs out of the cave

The scores are as such:

- Add 100 points for picking up each gold.
- Reduce 100 points for shooting an arrow.
- Reduce 10000 points for dying (by being eaten by the Wumpus, falling in a pit).
- Add 10 point for climbing out of the cave.
- Reduce 10 points for moving from one room to the next.

Tasks

Your group must implement code to explore the Wumpus World and get the highest score possible, using either Propositional Logic or First-Order Logic (or both).

Your implementation should output information about the search, including the precepts at every room the agent enters, the content of or change in the knowledge base after each new precept, and the action decided upon by the agent. The program should also output the score of the agent.

Having your implementation generate worlds randomly can help you test your agent.

Specifications

Input: the given map is represented by matrix, which is stored in the input file, for example, map1.txt. The input file format is described as follows:

- The first line contains an integer N, which is the size of map.
- N next lines with each line represents a string. If room empty, it is marked by hyphen character (-). If room has some things or signal such as Wumpus, Pit, Breeze, Stench, or Agent, it is marked by first capitalized character in name of each type and written next to each other. Between two adjacent rooms is separated by a dot (.)

- For example: -.BS.W.BS.P.B.-.-.-.-

	BS	W	BS	P	B				

Output: a result with path for agent, game point will be displayed on screen and/or written in output text file such as result1.txt.

Requirements

No.	Specifications	Scores
1	Finish problem successfully.	50%
2	Graphical demonstration of each step of the running process. You can demo in console screen or use any other graphical library.	10%
3	Generate at least 5 maps with difference structures such as position and number of Pit, Gold and Wumpus.	20%
4	Report your algorithm, experiment with some reflection or comments.	20%
Total		100%

Notice

This assignment will be completed in **groups**, with a maximum of 3-5 members per group. To prepare, you will need to create a folder that contains various subfolders, including source, input, output, and document. The report must give the following information:

- Your detailed information (Student Id, Full Name)
- Assignment Plan
- Environment to compile and run your program.
- Estimating the degree of completion level for each requirement.
- References (if any)

Your team can use any programming language to be, but Python is encouraged

Any plagiarism, tricks, or any lie will have 0 points for the course grade.

Contact this email if you have any questions about project (quochuyy2000@gmail.com).