CZ3005 - Artificial Intelligence - Assignment 2 Report

Submitted by Team Hello Kitty:
Agarwal Gopal (U1922859F)
Sneha Ravisankar (U2023734D)
Chai Youxiang (U2022575A)

Contributions
All members contributed to the agent, driver and the report equally.

Introduction
Wumpus World is a finite grid world (3x3 Square in this Assignment), which is populated by an Agent, a Wumpus, Confundus Portals and Coins. In this world the Agent is mobile and all other items remain fixed through the game. There can only be one item per cell. Our goal while working on this project was to create an Agent using prolog which will be able to navigate the Wumpus World successfully. In order to do this the agent must have to take deterministic actions which prevents it from being caught by the Wumpus, while collecting all the coins on the map. Additionally, we also created a driver that tests our agent.

In this report, we will elaborate on our approach to each task. We will then display our output under different circumstances in order to demonstrate correctness of our implementation. Please reference

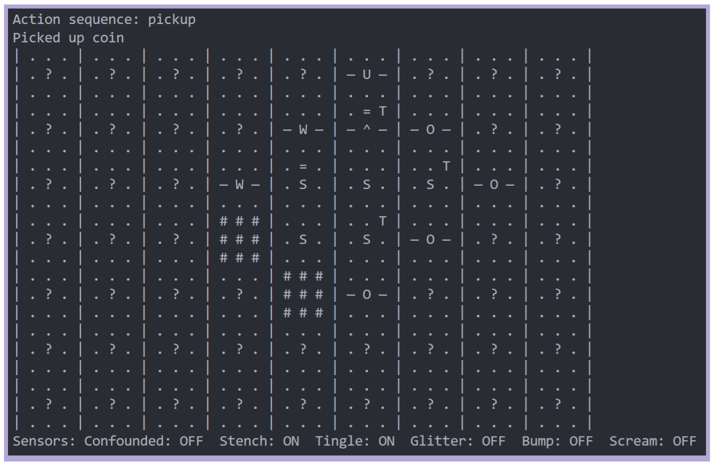
Agent

• reborn/0
reborn implements Agent's reset due to arriving into a cell inhabited by the Wumpus. For this task, we retracted all indicators, walls, wumpus, confundus, current and safe cells. We also reset the current position to (0,0) facing north, and set it as safe and visited. We give the agent the arrow. Additionally, we retract all the other flags we have created to support our implementation too. Essentially, this resets the game if the agent runs into the Wumpus.

• reposition(L)
reposition(L) is called when the agent gets Confounded. Inorder to implement this predicate we reset the game in a manner similar to the reborn task. We set the current cell to be (0,0) and mark it as safe and visited. Update the board with the relevant sensory information.

• move(A,L)
Move(A,L) implements Agent's reasoning response to executing action A and receiving sensory input L. For this task we implemented the agent's reasoning to each action in the following manner:
The move function accepts the following **A**ctions as variable **A:**

1) **shoot**
   Asserts that the Wumpus has been killed if the agent perceives a **Scream** (Sc). Shooting wumpus also calls a loop function to have the arrow “travel” down a straight line.
2) **turnleft**
   Rotates the agent’s face/heading 90deg leftwards on the spot by retracting previously defined predicates and asserting new ones that reflect the agent’s new heading.
3) **turnright**
   Rotates the agent’s face/heading 90deg rightwards on the spot by retracting previously defined predicates and asserting new ones that reflect the agent’s new heading.
   For both left and right turns, we first retrieve the current position of the agent using **current(X,Y,D)** where X and Y are cell coordinates and D is the agent’s heading.
   Eg, if D == rnorth, we retract the current D and assert reast (For agent turning right)
4) **pickup**
   Let’s the agent pick up the coin and retracts the **glitter** percept pertaining to a specific X,Y cell coordinate.

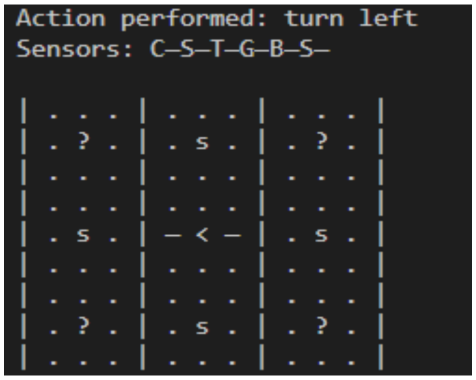


```
Action sequence: pickup
Picked up coin
| . . . | . . . | . . . | . . . | . . . | . . . | . . . | . . . | . . . |
| . ? . | . ? . | . ? . | . ? . | . ? . | — U — | . ? . | . ? . | . ? . |
| . . . | . . . | . . . | . . . | . . . | . . . | . . . | . . . | . . . |
| . . . | . . . | . . . | . . . | . . . | . = T | . . . | . . . | . . . |
| . ? . | . ? . | . ? . | . ? . | — W — | — ^ — | — O — | . ? . | . ? . |
| . . . | . . . | . . . | . . . | . . . | . . . | . . . | . . . | . . . |
| . . . | . . . | . . . | . . . | . = . | . . . | . . T | . . . | . . . |
| . ? . | . ? . | . ? . | — W — | . S . | . S . | . S . | — O — | . ? . |
| . . . | . . . | . . . | . . . | . . . | . . . | . . . | . . . | . . . |
| . . . | . . . | . . . | # # # | . . . | . . T | . . . | . . . | . . . |
| . ? . | . ? . | . ? . | # # # | . S . | . S . | — O — | . ? . | . ? . |
| . . . | . . . | . . . | # # # | . . . | . . . | . . . | . . . | . . . |
| . . . | . . . | . . . | . . . | # # # | . . . | . . . | . . . | . . . |
| . ? . | . ? . | . ? . | . ? . | # # # | — O — | . ? . | . ? . | . ? . |
| . . . | . . . | . . . | . . . | # # # | . . . | . . . | . . . | . . . |
| . . . | . . . | . . . | . . . | . . . | . . . | . . . | . . . | . . . |
| . ? . | . ? . | . ? . | . ? . | . ? . | . ? . | . ? . | . ? . | . ? . |
| . . . | . . . | . . . | . . . | . . . | . . . | . . . | . . . | . . . |
| . . . | . . . | . . . | . . . | . . . | . . . | . . . | . . . | . . . |
| . ? . | . ? . | . ? . | . ? . | . ? . | . ? . | . ? . | . ? . | . ? . |
| . . . | . . . | . . . | . . . | . . . | . . . | . . . | . . . | . . . |
Sensors: Confounded: OFF  Stench: ON  Tingle: ON  Glitter: OFF  Bump: OFF  Scream: OFF
```

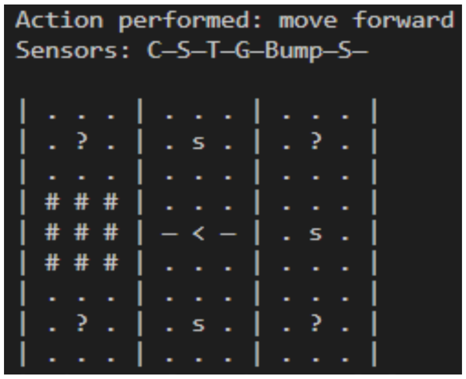
5) **movefoward**
   Checks are done to see if moving forward will cause the agent to **Bump** into a wall and if the agent steps into a portal which causes loss of pre-existing perception information.

   If the checks are cleared, the **goforward** function is called to assert the new agent’s coordinates, incrementing in either the X or Y direction, then asserting the new cell has been **visited.** There are a few exception cases regarding the “Move Forward” action, here is how we handled them:

a) The intended new location is inhabited by a Wall:
   Check to see if the bump percept is turned on. Assert wall(X,Y).

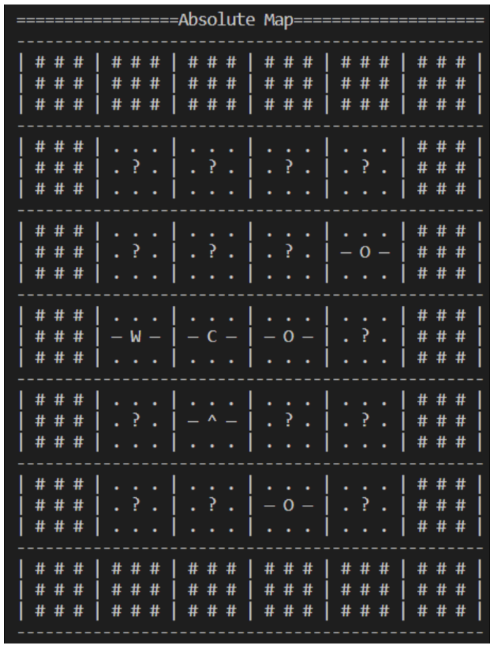


```
Action performed: turn left
Sensors: C—S—T—G—B—S—

| . . . | . . . | . . . |
| . ? . | . s . | . ? . |
| . . . | . . . | . . . |
| . . . | . . . | . . . |
| . s . | — < — | . s . |
| . . . | . . . | . . . |
| . . . | . . . | . . . |
| . ? . | . s . | . ? . |
| . . . | . . . | . . . |
```

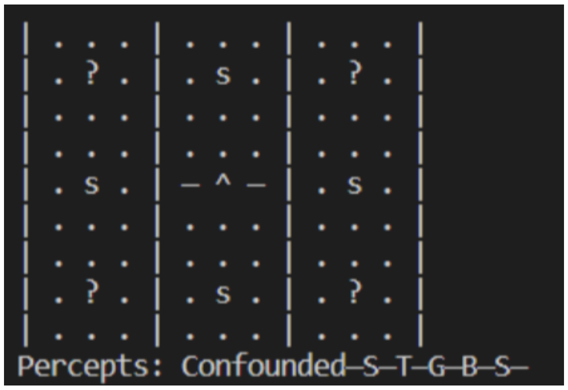


```
Action performed: move forward
Sensors: C—S—T—G—Bump—S—

| . . . | . . . | . . . |
| . ? . | . s . | . ? . |
| . . . | . . . | . . . |
| # # # | . . . | . . . |
| # # # | — < — | . s . |
| # # # | . . . | . . . |
| . . . | . . . | . . . |
| . ? . | . s . | . ? . |
| . . . | . . . | . . . |
```

b) The intended new location is inhabited by Confundus Portal:



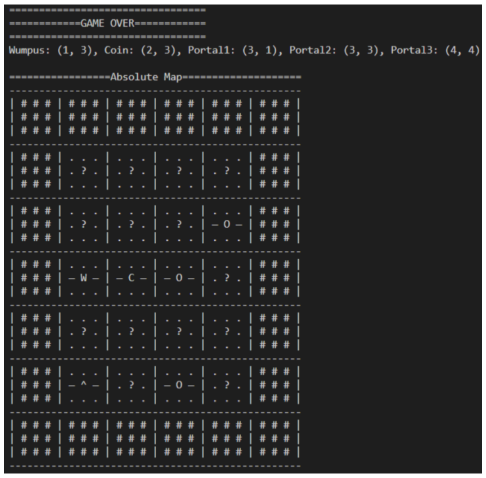
```
================Absolute Map===================
-----------------------------------------------
| # # # | # # # | # # # | # # # | # # # | # # # |
| # # # | # # # | # # # | # # # | # # # | # # # |
| # # # | # # # | # # # | # # # | # # # | # # # |
-----------------------------------------------
| # # # | . . . | . . . | . . . | . . . | # # # |
| # # # | . ? . | . ? . | . ? . | . ? . | # # # |
| # # # | . . . | . . . | . . . | . . . | # # # |
-----------------------------------------------
| # # # | . . . | . . . | . . . | . . . | # # # |
| # # # | . ? . | . ? . | . ? . | — O — | # # # |
| # # # | . . . | . . . | . . . | . . . | # # # |
-----------------------------------------------
| # # # | . . . | . . . | . . . | . . . | # # # |
| # # # | — W — | — C — | — O — | . ? . | # # # |
| # # # | . . . | . . . | . . . | . . . | # # # |
-----------------------------------------------
| # # # | . . . | . . . | . . . | . . . | # # # |
| # # # | . ? . | — ^ — | . ? . | . ? . | # # # |
| # # # | . . . | . . . | . . . | . . . | # # # |
-----------------------------------------------
| # # # | . . . | . . . | . . . | . . . | # # # |
| # # # | . ? . | . ? . | — O — | . ? . | # # # |
| # # # | . . . | . . . | . . . | . . . | # # # |
-----------------------------------------------
| # # # | # # # | # # # | # # # | # # # | # # # |
| # # # | # # # | # # # | # # # | # # # | # # # |
| # # # | # # # | # # # | # # # | # # # | # # # |
-----------------------------------------------
```



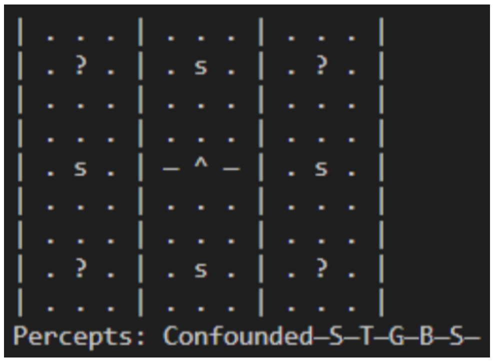
```
| . . . | . . . | . . . |
| . ? . | . s . | . ? . |
| . . . | . . . | . . . |
| . . . | . . . | . . . |
| . s . | — ^ — | . s . |
| . . . | . . . | . . . |
| . . . | . . . | . . . |
| . ? . | . s . | . ? . |
| . . . | . . . | . . . |
Percepts: Confounded—S—T—G—B—S—
```

The agent gets confunded. Reposition(L) is called. The print out on the left shows where the agent is currently located on the absolute map. The printout above shows the agent's new relative map.

c) The intended new location is inhabited by the Wumpus:



```
================================
===========GAME OVER============
================================
Wumpus: (1, 3), Coin: (2, 3), Portal1: (3, 1), Portal2: (3, 3), Portal3: (4, 4)

=================Absolute Map===================
------------------------------------------------
| # # # | # # # | # # # | # # # | # # # | # # # |
| # # # | # # # | # # # | # # # | # # # | # # # |
| # # # | # # # | # # # | # # # | # # # | # # # |
------------------------------------------------
| # # # | . . . | . . . | . . . | . . . | # # # |
| # # # | . ? . | . ? . | . ? . | . ? . | # # # |
| # # # | . . . | . . . | . . . | . . . | # # # |
------------------------------------------------
| # # # | . . . | . . . | . . . | . . . | # # # |
| # # # | . ? . | . ? . | . ? . | — O — | # # # |
| # # # | . . . | . . . | . . . | . . . | # # # |
------------------------------------------------
| # # # | . . . | . . . | . . . | . . . | # # # |
| # # # | — W — | — C — | — O — | . ? . | # # # |
| # # # | . . . | . . . | . . . | . . . | # # # |
------------------------------------------------
| # # # | . . . | . . . | . . . | . . . | # # # |
| # # # | . ? . | . ? . | . ? . | . ? . | # # # |
| # # # | . . . | . . . | . . . | . . . | # # # |
------------------------------------------------
| # # # | . . . | . . . | . . . | . . . | # # # |
| # # # | — ^ — | . ? . | — O — | . ? . | # # # |
| # # # | . . . | . . . | . . . | . . . | # # # |
------------------------------------------------
| # # # | # # # | # # # | # # # | # # # | # # # |
| # # # | # # # | # # # | # # # | # # # | # # # |
| # # # | # # # | # # # | # # # | # # # | # # # |
------------------------------------------------
```



```
| . . . | . . . | . . . |
| . ? . | . s . | . ? . |
| . . . | . . . | . . . |
| . . . | . . . | . . . |
| . s . | — ^ — | . s . |
| . . . | . . . | . . . |
| . . . | . . . | . . . |
| . ? . | . s . | . ? . |
| . . . | . . . | . . . |
Percepts: Confounded—S—T—G—B—S—
```

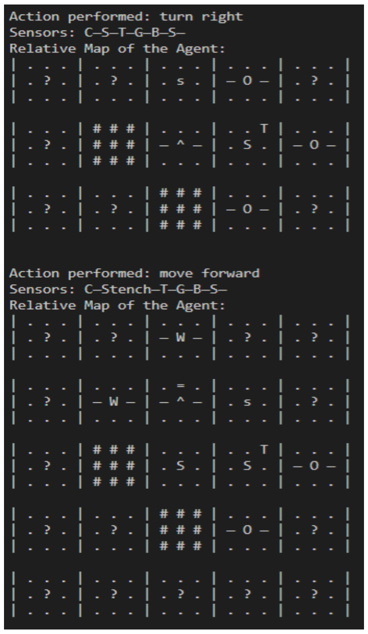
Agent is killed by the Wumpus, reborn/0 is called. The game restarts.

Localisation and mapping

visited(X,Y), safe(X,Y) and wall(X,Y), glitter(X,Y), wumpus(X,Y) and **confundus(X,Y):** Returns True according to what is present in cell (X,Y)

manage_tingle(X,Y): Checks surrounding cells to see if previous assertions exist. If tingle is "on", asserts confundus(X,Y) on all surrounding cells with a potential portal.

handle_stench(X,Y): Asserts wumpus(X,Y) on potential cells with the wumpus if previously unvisited and if stench percept is on



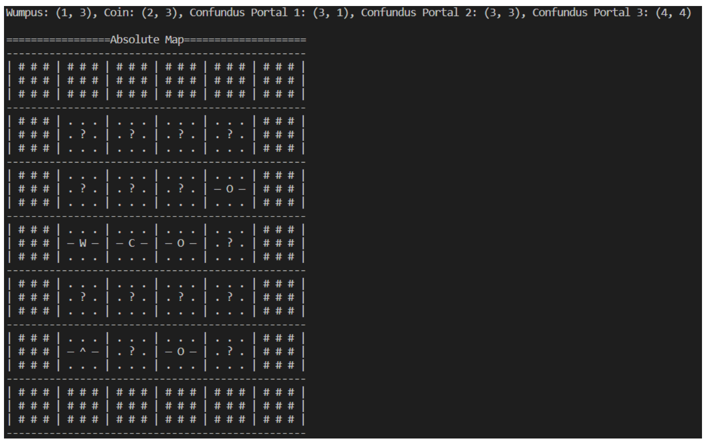
```
Action performed: turn right
Sensors: C—S—T—G—B—S—
Relative Map of the Agent:
| . . . | . . . | . . . | . . . | . . . |
| . ? . | . ? . | . s . | — O — | . ? . |
| . . . | . . . | . . . | . . . | . . . |

| . . . | # # # | . . . | . . T | . . . |
| . ? . | # # # | — ^ — | . S . | — O — |
| . . . | # # # | . . . | . . . | . . . |

| . . . | . . . | # # # | . . . | . . . |
| . ? . | . ? . | # # # | — O — | . ? . |
| . . . | . . . | # # # | . . . | . . . |


Action performed: move forward
Sensors: C—Stench—T—G—B—S—
Relative Map of the Agent:
| . . . | . . . | . . . | . . . | . . . |
| . ? . | . ? . | — W — | . ? . | . ? . |
| . . . | . . . | . . . | . . . | . . . |

| . . . | . . . | . = . | . . . | . . . |
| . ? . | — W — | — ^ — | . s . | . ? . |
| . . . | . . . | . . . | . . . | . . . |

| . . . | # # # | . . . | . . T | . . . |
| . ? . | # # # | . S . | . S . | — O — |
| . . . | # # # | . . . | . . . | . . . |

| . . . | . . . | # # # | . . . | . . . |
| . ? . | . ? . | # # # | — O — | . ? . |
| . . . | . . . | # # # | . . . | . . . |

| . . . | . . . | . . . | . . . | . . . |
| . ? . | . ? . | . ? . | . ? . | . ? . |
| . . . | . . . | . . . | . . . | . . . |
```

• **explore(L)** is true if the list L contains a sequence of actions that leads the Agent to inhabit a safe and non-visited location.
explore(L) calls **plan(L)** to check surrounding cells in the NSEW directions using predicates such as ~wumpus ^ ~wall. It also takes into account if the particular cell has been visited before, pointing to a route previously taken. It also calls **danger(L)** to shoot the arrow if the **wumpuslocated** flag is set to true, meaning wumpus has been located.

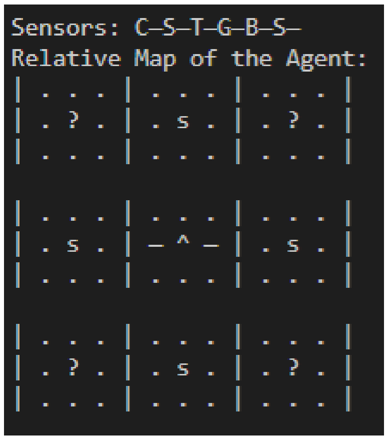
• current(X,Y,D) is true if (X,Y) is the current relative position and D is the current relative orientation of the Agent. Possible relative orientations are described by constants {rnorth,rwest,reast,rsouth}, and the relative position consists of two integers.

• hasarrow/0 returns true if the Agent has the arrow, and begins to return false after the shoot action has been executed by the Agent.

Driver
Initial absolute map of the world depicted by an array of map cells and the initial relative map:



```
Wumpus: (1, 3), Coin: (2, 3), Confundus Portal 1: (3, 1), Confundus Portal 2: (3, 3), Confundus Portal 3: (4, 4)

================Absolute Map===================
-----------------------------------------------
| # # # | # # # | # # # | # # # | # # # | # # # |
| # # # | # # # | # # # | # # # | # # # | # # # |
| # # # | # # # | # # # | # # # | # # # | # # # |
-----------------------------------------------
| # # # | . . . | . . . | . . . | . . . | # # # |
| # # # | . ? . | . ? . | . ? . | . ? . | # # # |
| # # # | . . . | . . . | . . . | . . . | # # # |
-----------------------------------------------
| # # # | . . . | . . . | . . . | . . . | # # # |
| # # # | . ? . | . ? . | . ? . | — O — | # # # |
| # # # | . . . | . . . | . . . | . . . | # # # |
-----------------------------------------------
| # # # | . . . | . . . | . . . | . . . | # # # |
| # # # | — W — | — C — | — O — | . ? . | # # # |
| # # # | . . . | . . . | . . . | . . . | # # # |
-----------------------------------------------
| # # # | . . . | . . . | . . . | . . . | # # # |
| # # # | . ? . | . ? . | . ? . | . ? . | # # # |
| # # # | . . . | . . . | . . . | . . . | # # # |
-----------------------------------------------
| # # # | . . . | . . . | . . . | . . . | # # # |
| # # # | — ^ — | . ? . | — O — | . ? . | # # # |
| # # # | . . . | . . . | . . . | . . . | # # # |
-----------------------------------------------
| # # # | # # # | # # # | # # # | # # # | # # # |
| # # # | # # # | # # # | # # # | # # # | # # # |
| # # # | # # # | # # # | # # # | # # # | # # # |
-----------------------------------------------
```



```
Sensors: C-S-T-G-B-S-
Relative Map of the Agent:
| . . . | . . . | . . . |
| . ? . | . s . | . ? . |
| . . . | . . . | . . . |

| . . . | . . . | . . . |
| . s . | — ^ — | . s . |
| . . . | . . . | . . . |

| . . . | . . . | . . . |
| . ? . | . s . | . ? . |
| . . . | . . . | . . . |
```

From this printout, we can see that the relative origin cell of the agent is always depicted in the center of the map. The sequence begins with the map of initial knowledge before the first actions are applied. Each map is accompanied by the percepts provided to the agent by the driver. Therefore, all requirements are satisfied.

Conclusion
We've learnt the basic application of decision making using predicate through the use of Prolog for a simple game-world such as in this lab assignment. The importance of perception as a form of constraint and guidance to achieve the agent's intended goal and the difficulty of translating "rules" from the Knowledge Base into actions that the agent can carry out.

Although this assignment takes place in a simple world with few considerations, it is immediately apparent how these "rules" translate to real-world situations such as self-driving cars where agents need to constantly perceive their surroundings and make decisions without intervention from an actual human, distilling the true essence of Artificial Intelligence.

References

1. Derek Banas. (2015, August 13). *Prolog Tutorial* [Video]. YouTube. https://www.youtube.com/watch?v=SykxWpFwMGs
2. Being Passionate Learner. (2020, May 13). *The Wumpus World* [Video]. YouTube. https://www.youtube.com/watch?v=Bz_cFuf4aQg
3. *wumpus.pl*. Home page of Markus Triska. (n.d.). Retrieved April 29, 2022, from https://www.metalevel.at/wumpusworld/wumpus.pl