Name: boujrada mehdi project teammate: achraf ajrhourh

Id: 100696

CSC 4301(01) – into to AI

WUMPIS GAME LOGICAL AGENT

In this project we have been asked to use prolog to code a logical agent for the Wumpus game. The agent aim is to kill the Wumpus before moving into its room. And with a side objective of finding the gold. But the main objective of the game is killing the Wumps.

First and foremost, we established the essential predicates such as Room, Breeze, and Pit. … etc. These are the building components that will allow the agent to become rational. I'll go over the primary predicates that are used in the program. Room: Is a predicate in which we provide it a visited list and then call a sense function to make a perception utilizing the knowledge base. We then update the knowledge base as well as the game's time and score. Finally, we learn the agent's location as well as what he observes in his immediate surroundings. When the agent enters a room, it detects the surrounding rooms to avoid the Wumpus.

Update_time: when its called it updates the time to +1.

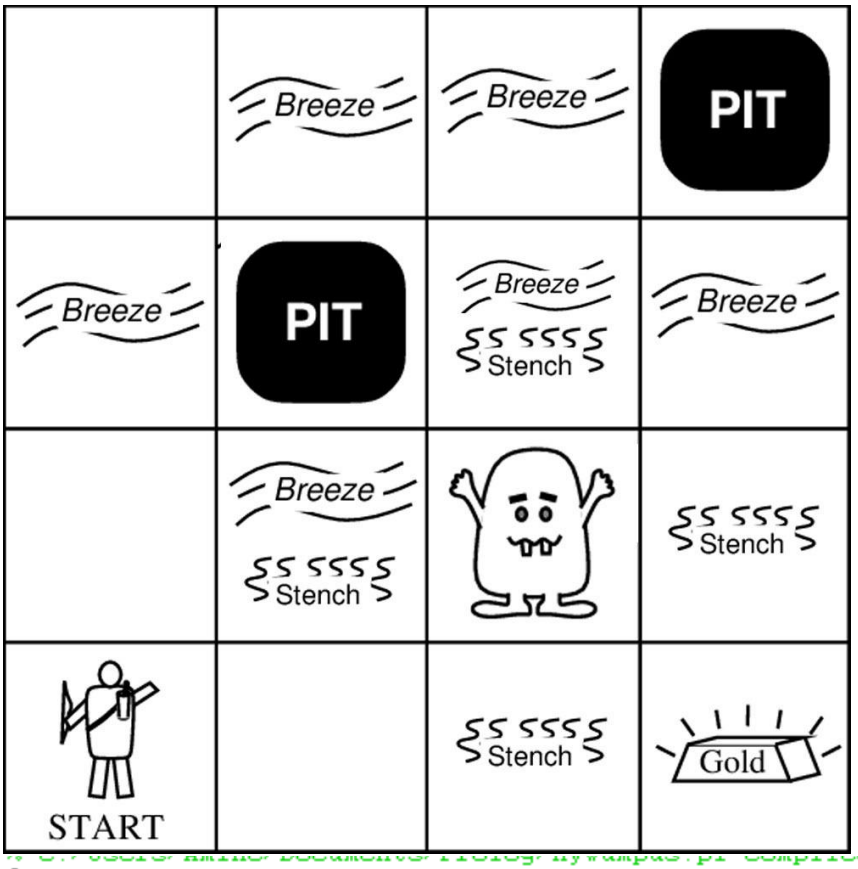
Update_score: when its called it updates the score of the game.

The wumpos location, gold location, and agent location are all called by this status. If the method is pit is true, the agent fails the game because he fell into a pit; otherwise, it checks to see if a wumpus ate the agent or if the agent found the goal. If none of these occur, the agent continues to play the game.

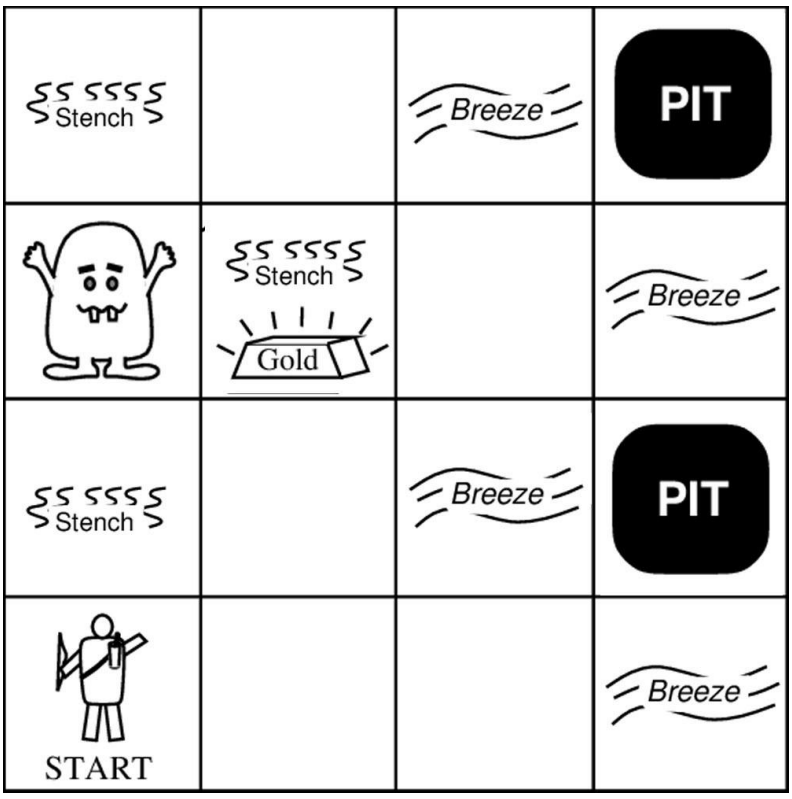
It initializes the game, the land, which is the game's size, as well as the pit and gold locations, init agent, which is the agent's location, and init wumpus, which is the wumpus' location.

Adjacency: After setting up some rooms as safe rooms, we use the adjacent to predicate. When it's smelly or gusty, we'll call it to see if a wumpus is nearby.

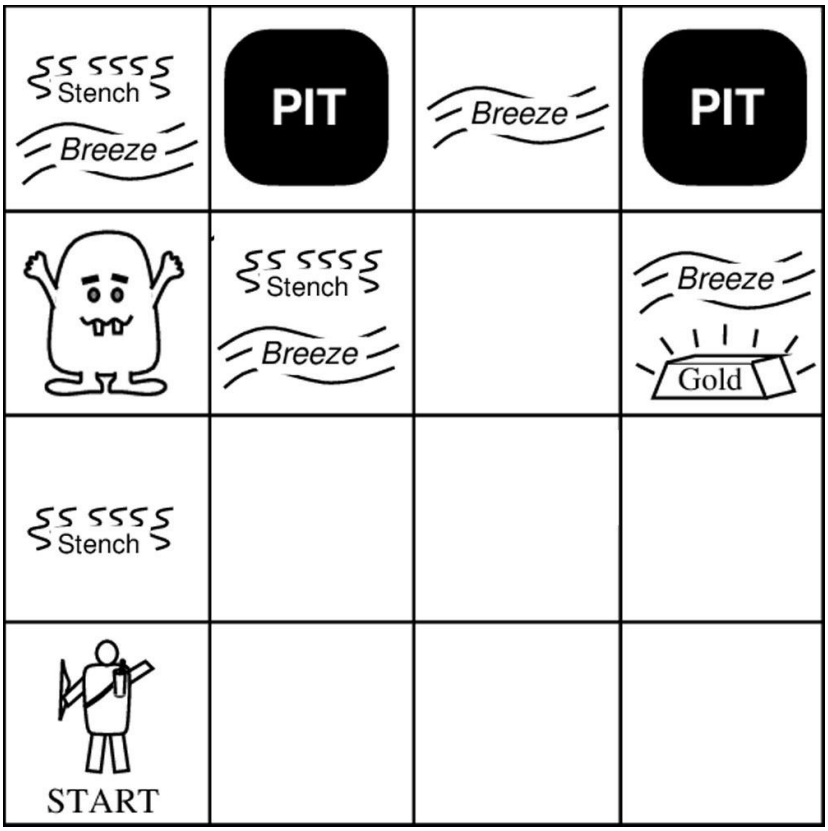
Some situations when the agent won the game:




```
?-
|    start.
I am in [1,1]
I am in [2,1]
I am in [1,1]
I am in [1,2]
I am in [2,1]
I am in [3,1]
there is a stench in [3,1]
I am in [1,2]
I am in [2,2]
there is a breeze in [2,2]
there is a stench in [2,2]
I am in [1,2]
I am in [1,3]
there is a breeze in [1,3]
I have found GOLD, Score is now 520
I am in [3,1]
there is a stench in [3,1]
I am in [4,1]
I am in [4,2]
there is a stench in [4,2]
The wumpus is located in [3,2]! I am shooting my bullet!
Score: 517
timer: 13
WON!
true.
```




```
?- start.
I am in [1,1]
I am in [2,1]
I am in [1,1]
I am in [1,2]
there is a stench in [1,2]
I am in [2,1]
I am in [3,1]
I am in [1,2]
there is a stench in [1,2]
I am in [2,2]
I am in [3,1]
I am in [4,1]
there is a breeze in [4,1]
I am in [2,2]
I am in [3,2]
there is a breeze in [3,2]
I am in [2,2]
I am in [2,3]
there is a stench in [2,3]
I have found GOLD, Score is now 516
I am in [3,3]
I am in [4,3]
there is a breeze in [4,3]
there is a breeze in [4,3]
I am in [3,3]
I am in [3,4]
there is a breeze in [3,4]
I am in [2,4]
I am in [1,4]
there is a stench in [1,4]
The wumpus is located in [1,3]! I am shooting my bullet!
Score: 510
timer: 20
WON!
true.
```

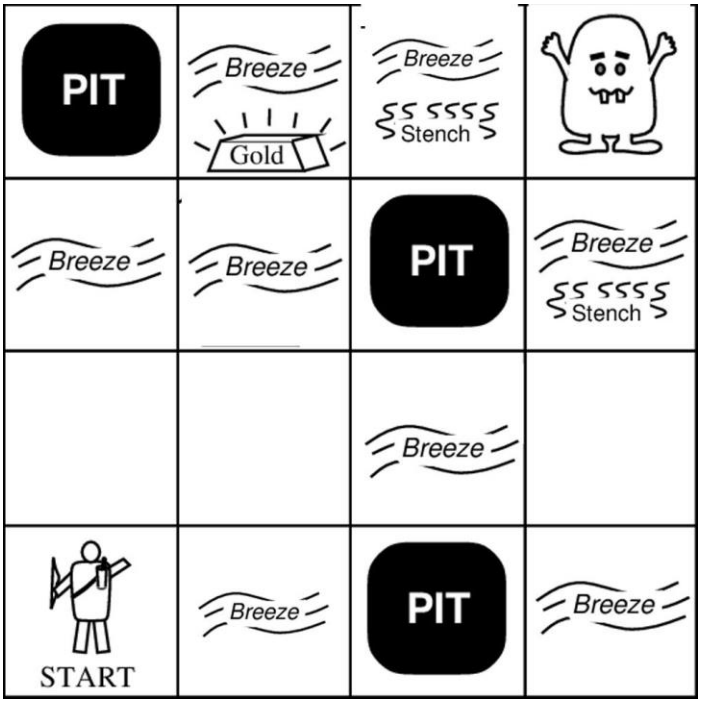



```
?- start.
I am in [1,1]
I am in [2,1]
I am in [1,1]
I am in [1,2]
there is a stench in [1,2]
I am in [2,1]
I am in [3,1]
I am in [1,2]
there is a stench in [1,2]
I am in [2,2]
I am in [3,1]
I am in [4,1]
I am in [2,2]
I am in [3,2]
I am in [2,2]
I am in [2,3]
there is a breeze in [2,3]
there is a stench in [2,3]
I am in [3,2]
I am in [4,2]
I am in [2,3]
there is a breeze in [2,3]
there is a stench in [2,3]
I am in [3,3]
I am in [4,3]
there is a breeze in [4,3]
I have found GOLD, Score is now 511
I am in [3,3]
I am in [3,4]
there is a breeze in [3,4]
there is a breeze in [3,4]
The wumpus is located in [1,3]! I am shooting my bullet!
Score: 509
timer: 21
WON!
true.
```

As can be seen, the score and time have been modified in favor of the winning team.

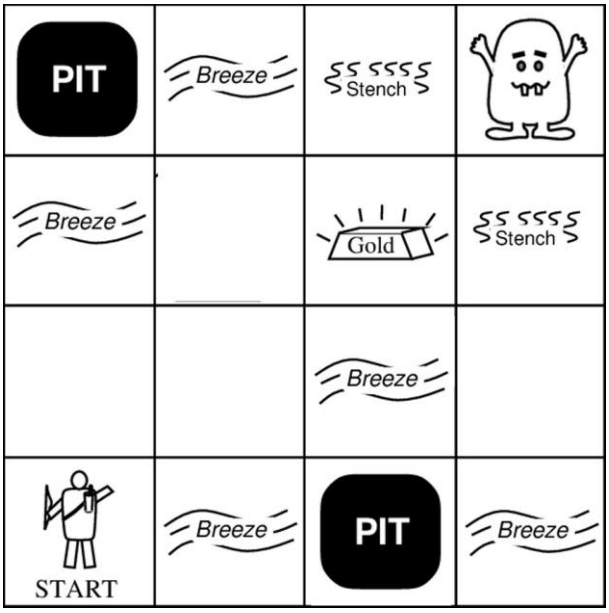
Because the agent does not always win the game, it all relies on where we position the pit, wumpus, and gold.

And this is a situation where the agent fails the game:




```
?- start.
I am in [1,1]
I am in [2,1]
there is a breeze in [2,1]
I am in [1,1]
I am in [1,2]
I am in [2,2]
I am in [1,2]
I am in [1,3]
there is a breeze in [1,3]
I am in [2,2]
I am in [3,2]
there is a breeze in [3,2]
there is a breeze in [3,2]
I am in [1,3]
there is a breeze in [1,3]
I am in [2,3]
there is a breeze in [2,3]
I am in [3,2]
there is a breeze in [3,2]
there is a breeze in [3,2]
I am in [4,2]
I am in [4,3]
there is a breeze in [4,3]
there is a stench in [4,3]
I am in [4,2]
I am in [4,1]
there is a breeze in [4,1]
I am in [4,3]
there is a breeze in [4,3]
there is a stench in [4,3]
I am in [3,3]
I am in [3,4]
there is a breeze in [3,4]
there is a stench in [3,4]
I am in [2,4]
there is a breeze in [2,4]
I have found GOLD, Score is now 510
The agent failed to find the Wumpus
FAILED !
true.
```

And one error we had in the implementation, is that the agent got out of the WUMPUS world:




```
?- start.
I am in [1,1]
I am in [2,1]
I am in [1,1]
I am in [1,2]
there is a breeze in [1,2]
I am in [2,1]
I am in [3,1]
I am in [1,2]
there is a breeze in [1,2]
I am in [2,2]
I am in [3,1]
I am in [4,1]
I am in [2,2]
I am in [3,2]
I am in [2,2]
I am in [2,3]
there is a breeze in [2,3]
I am in [3,2]
I am in [4,2]
I am in [2,3]
there is a breeze in [2,3]
I am in [3,3]
I have found GOLD, Score is now 512
I am in [4,3]
there is a stench in [4,3]
I am in [3,3]
I have found GOLD, Score is now 1010
I am in [3,4]
there is a stench in [3,4]
I am in [2,4]
there is a breeze in [2,4]
The wumpus is located in [3,5]! I am shooting my bullet!
Score: 1008
timer: 22
WON!
true.
```

As we can see, the agent's attempts to manipulate the positions of the pit, gold, and Wumpus have failed, as the Wumpus have murdered him.

In terms of our coding errors, we've run into a few issues, such as the game not ending in some circumstances when the agent falls into a pit; I suspect the issue stemmed from the placement of the pit and agent, but we're not sure. We also couldn't come up with a means for the agent to kill the Wumpus and win the game this way, so in this version, the agent must just avoid the Wumpus and find the gold to win.

The init of all the games that I used:

game 1(won game)

```
init:-
   retractall(timer(_)),
   assert(timer(0)),
   retractall(score(_)),
   assert(score(30)),
   retractall(gold_location(_)),
   assert(gold_location([1,3])),
   retractall(wumpus_location(_)),
   assert(wumpus_location([3,2])),
   retractall(pit_location(_)),
   assert(pit_location([2,3])),
   assert(pit_location([4,4])),
   retractall(agent_location(_)),
   assert(agent_location([1,1])),
   retractall(wumpus_final_location(_)),
   assert(wumpus_final_location([-1,-1])).
```

game 3: (won game)

```
init:-
   retractall(timer(_)),
```

```
    assert(timer(0)),
    retractall(score(_)),
    assert(score(30)),
    retractall(gold_location(_)),
    assert(gold_location([2,3])),
    retractall(wumpus_location(_)),
    assert(wumpus_location([1,3])),
    retractall(pit_location(_)),
    assert(pit_location([4,2])),
    assert(pit_location([4,4])),
    retractall(agent_location(_)),
    assert(agent_location([1,1])),
    retractall(wumpus_final_location(_)),
    assert(wumpus_final_location([-1,-1])).
```

game 4: (won game)

init:-

```
    retractall(timer(_)),
    assert(timer(0)),
    retractall(score(_)),
    assert(score(30)),
    retractall(gold_location(_)),
    assert(gold_location([4,3])),
    retractall(wumpus_location(_)),
```

```
    assert(wumpus_location([1,3])),
    retractall(pit_location(_)),
    assert(pit_location([2,4])),
    assert(pit_location([4,4])),
    retractall(agent_location(_)),
    assert(agent_location([1,1])),
    retractall(wumpus_final_location(_)),
    assert(wumpus_final_location([-1,-1])).

game 9: (failed game)
init:-
    retractall(timer(_)),
    assert(timer(0)),
    retractall(score(_)),
    assert(score(30)),
    retractall(gold_location(_)),
    assert(gold_location([2,4])),
    retractall(wumpus_location(_)),
    assert(wumpus_location([4,4])),
    retractall(pit_location(_)),
    assert(pit_location([1,4])),
    assert(pit_location([3,1])),
    assert(pit_location([3,3])),
    retractall(agent_location(_)),
```

```
    assert(agent_location([1,1])),
    retractall(wumpus_final_location(_)),
    assert(wumpus_final_location([-1,-1])).
```

game 2 (out of the WUMPS world game)

```
init:-
    retractall(timer(_)),
    assert(timer(0)),
    retractall(score(_)),
    assert(score(30)),
    retractall(gold_location(_)),
    assert(gold_location([3,3])),
    retractall(wumpus_location(_)),
    assert(wumpus_location([4,4])),
    retractall(pit_location(_)),
    assert(pit_location([1,4])),
    assert(pit_location([3,1])),
    retractall(agent_location(_)),
    assert(agent_location([1,1])),
    retractall(wumpus_final_location(_)),
    assert(wumpus_final_location([-1,-1]))
```