Ahmed AL Hilal

74869

CSC 4301 (01) – Intro. to Artificial Intelligence

Dr. Tajjeeddine RACHIDI

Spring 2022

Wumpus Game Logical Agent

In this project, we have been asked to create a logical agent for the Wumpus game. The goal of the agent is to kill the Wumpus in its corresponding room and find the hidden gold before moving to its room. The Agent was coded in prolog, where we created certain rules in the knowledge base, so that, the agent can reference the optimal path in the future and succeed in avoiding a failure.

First of all, we have started by creating the necessary predicates such as, Room, Breeze, and Pit … etc. These are the necessary blocks that will make the agent rational. I will explain the main predicates used in the program.

Room: Is a predicate where we give it a visited list, then we call a function called sense to make a perception using the knowledge base. We then, update the knowledge base and update the time and score of the game. Finally, we get the agent's location and result what the agents sees that surrounds him. Basically when the agent gets to a room it senses the surrounding rooms to avoid the Wumpus.

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

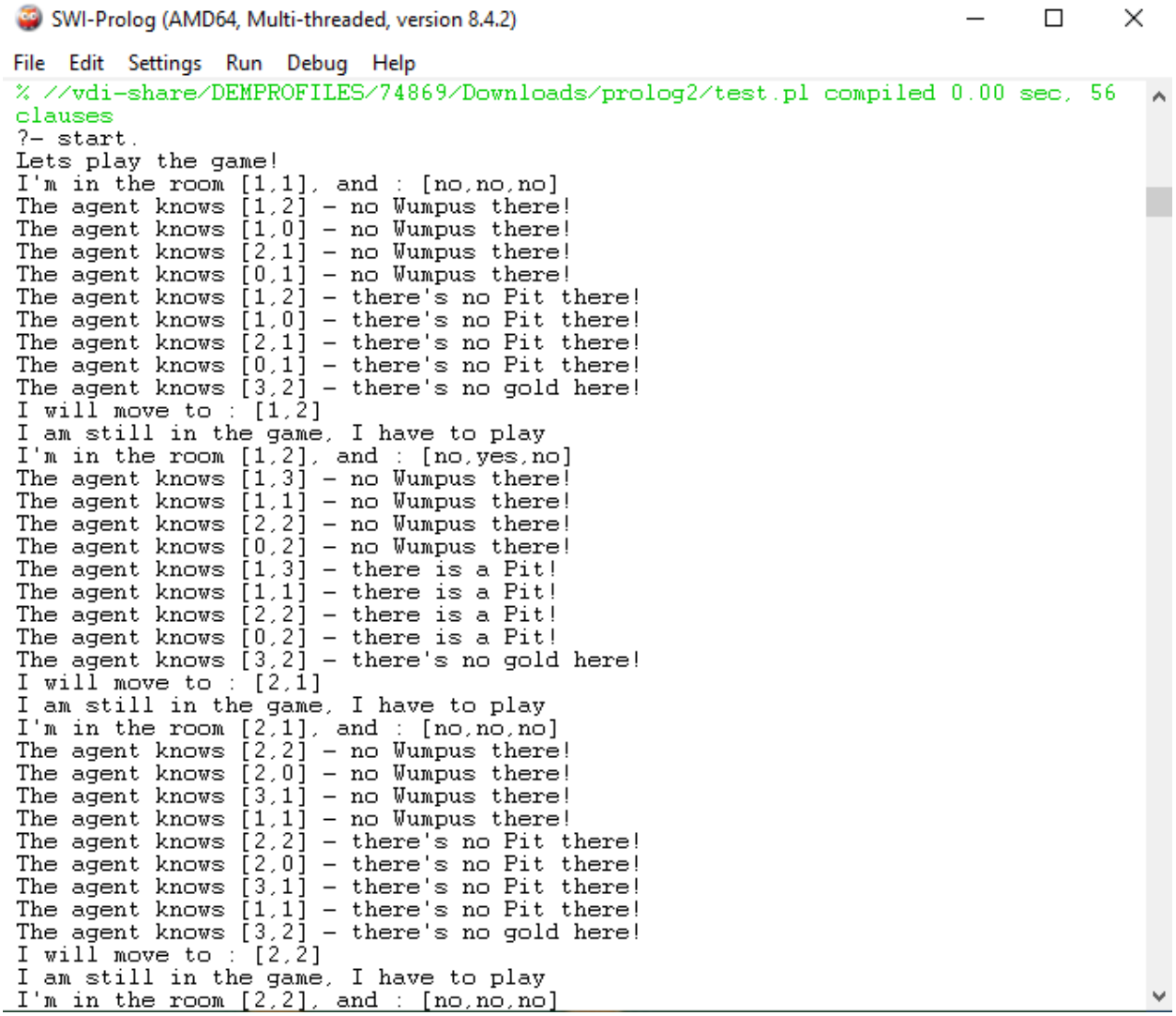
Update_score: when its called it updates the score of the game.

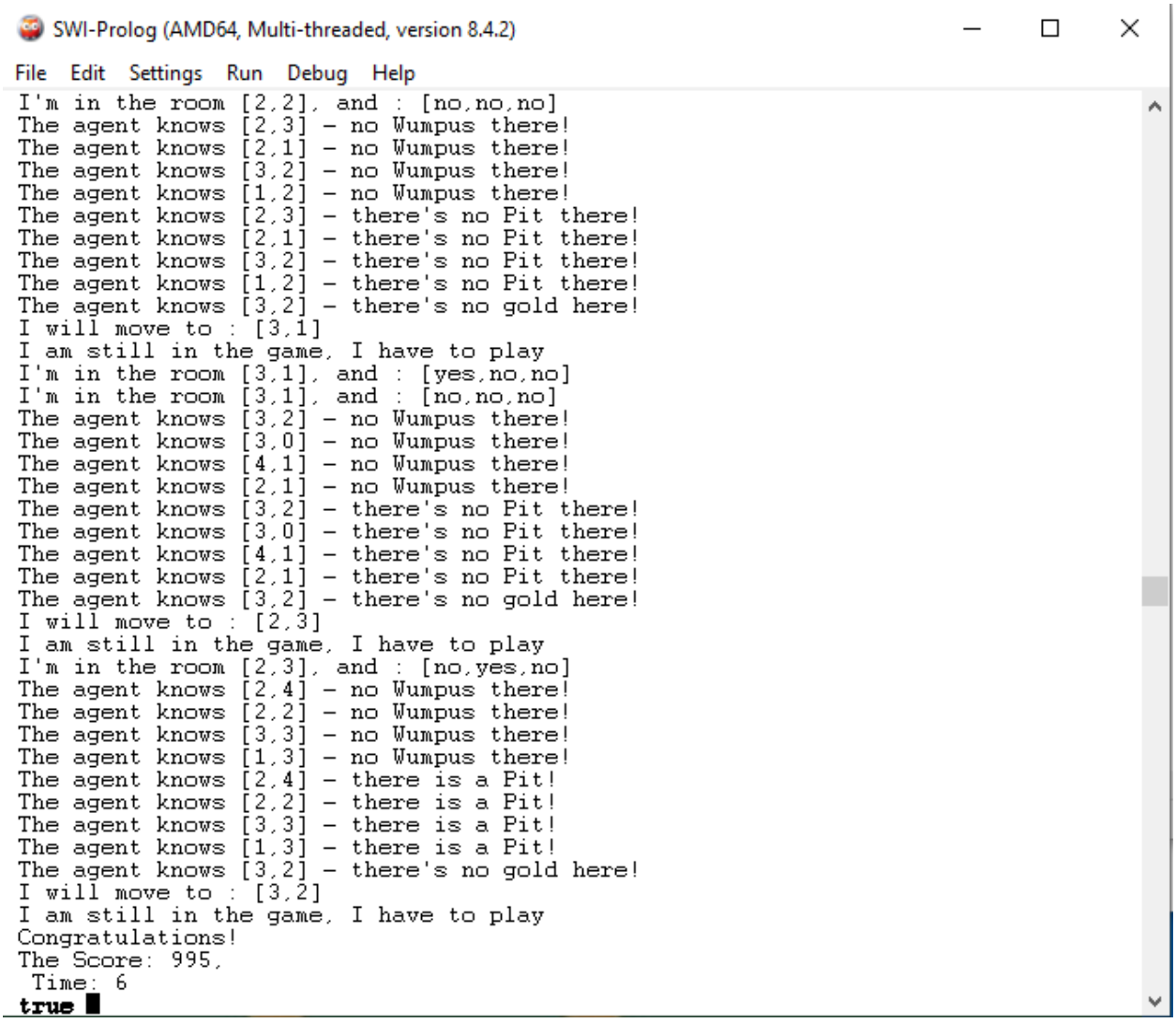
Status: it calls the wumpos location, gold location, and agent location . if the procedure is_pit is true, then the agent fails the game as he has fallen in a pit, otherwise it checks if a wumpus ate the agent or found the goal. If none of these happened then the agent keeps playing the game.

Init: it initializes the game, the land which is the size of the game and the pit location and gold location, init agent which is the location of the agent and, init wumpus for initializing the wumpus location.

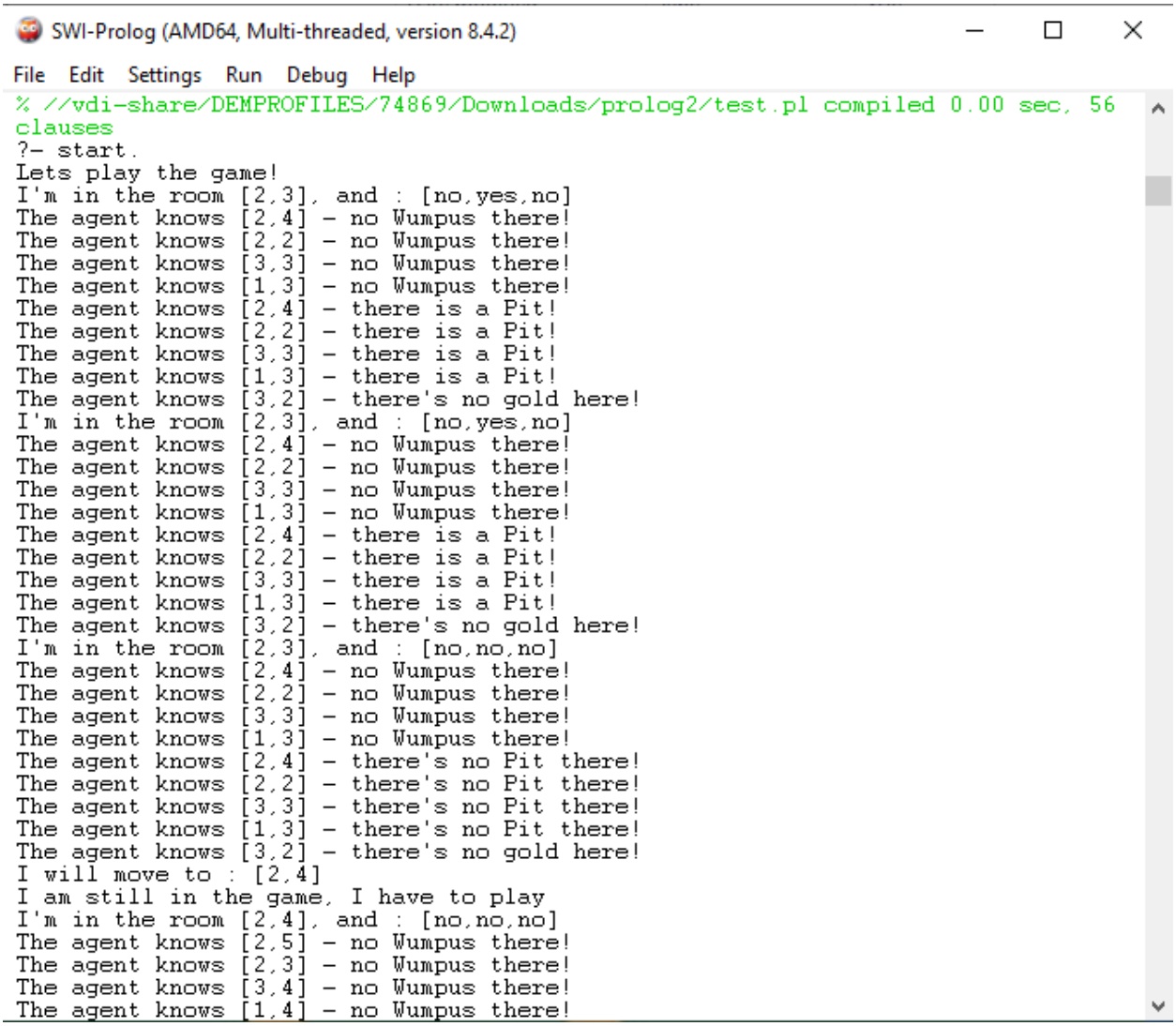
Adjacency: we initialize some rooms as safe rooms then a predicate called adjacent to. This will let us know when is smelly or is breezy call it to know if a wumpus or pit are adjacent to the agent.

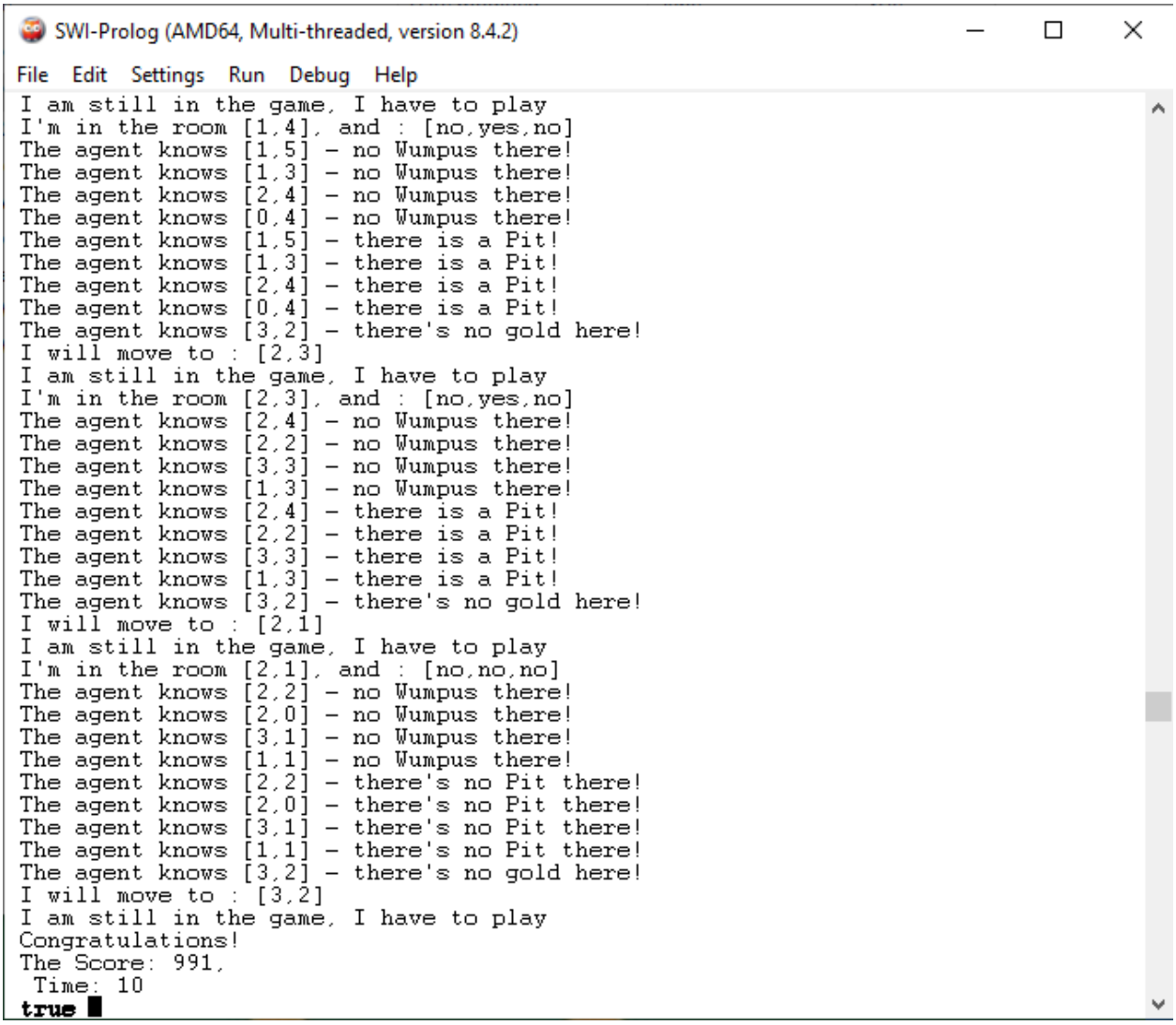
I will showcase now a picture when the agent wins the game:




SWI-Prolog (AMD64, Multi-threaded, version 8.4.2)
File Edit Settings Run Debug Help
I'm in the room [2,2], and : [no,no,no]
The agent knows [2,3] - no Wumpus there!
The agent knows [2,1] - no Wumpus there!
The agent knows [3,2] - no Wumpus there!
The agent knows [1,2] - no Wumpus there!
The agent knows [2,3] - there's no Pit there!
The agent knows [2,1] - there's no Pit there!
The agent knows [3,2] - there's no Pit there!
The agent knows [1,2] - there's no Pit there!
The agent knows [3,2] - there's no gold here!
I will move to : [3,1]
I am still in the game, I have to play
I'm in the room [3,1], and : [yes,no,no]
I'm in the room [3,1], and : [no,no,no]
The agent knows [3,2] - no Wumpus there!
The agent knows [3,0] - no Wumpus there!
The agent knows [4,1] - no Wumpus there!
The agent knows [2,1] - no Wumpus there!
The agent knows [3,2] - there's no Pit there!
The agent knows [3,0] - there's no Pit there!
The agent knows [4,1] - there's no Pit there!
The agent knows [2,1] - there's no Pit there!
The agent knows [3,2] - there's no gold here!
I will move to : [2,3]
I am still in the game, I have to play
I'm in the room [2,3], and : [no,yes,no]
The agent knows [2,4] - no Wumpus there!
The agent knows [2,2] - no Wumpus there!
The agent knows [3,3] - no Wumpus there!
The agent knows [1,3] - no Wumpus there!
The agent knows [2,4] - there is a Pit!
The agent knows [2,2] - there is a Pit!
The agent knows [3,3] - there is a Pit!
The agent knows [1,3] - there is a Pit!
The agent knows [3,2] - there's no gold here!
I will move to : [3,2]
I am still in the game, I have to play
Congratulations!
The Score: 995,
Time: 6
true

This is another version where we changed the position of the agent:

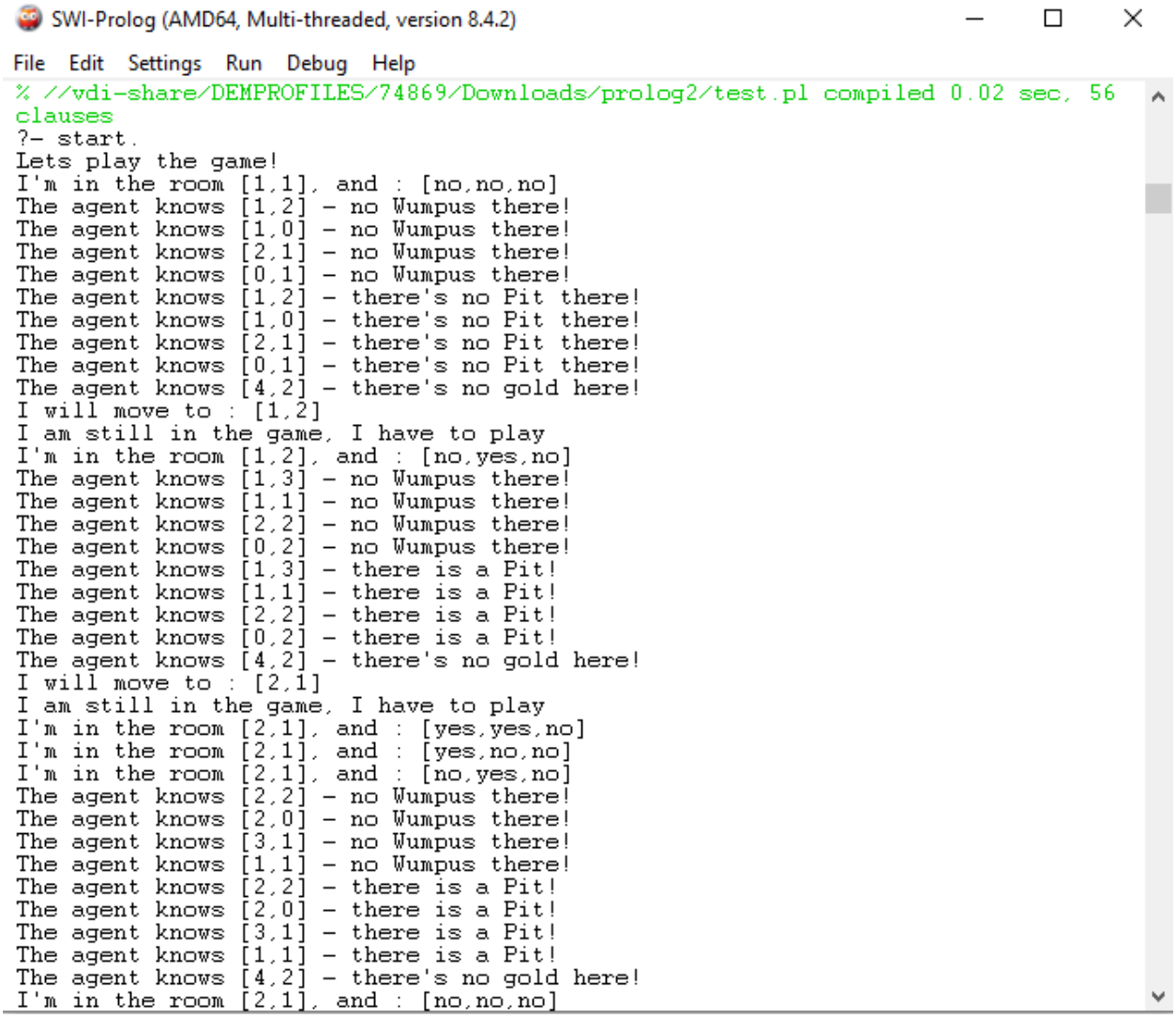


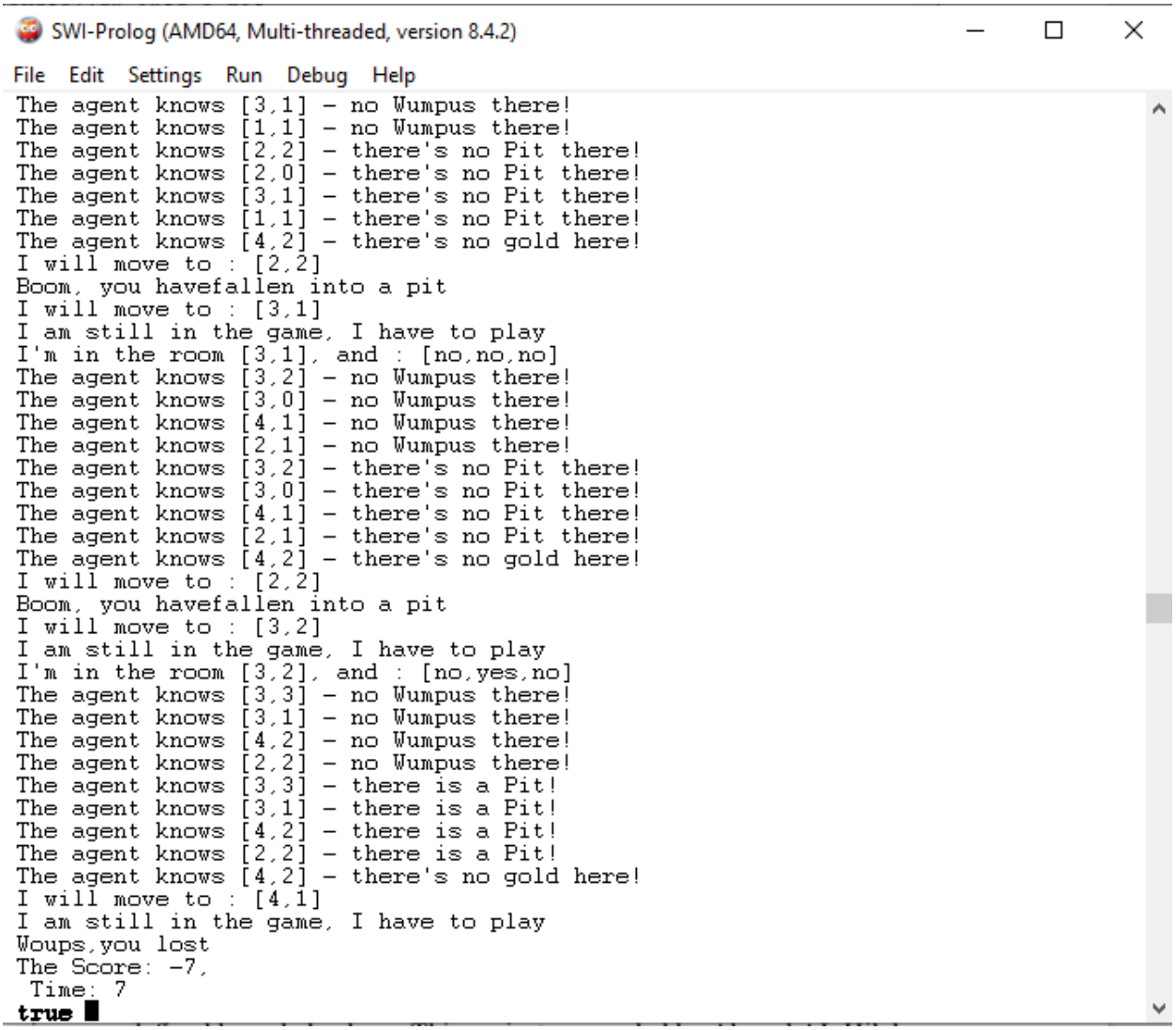


As we can see, the score and time has been changed with a winning outcome.

It all depends on where we put the pit, wumpus, and gold as in some cases the agent fails to win the game.

This is a special case where the agent has failed to win:





As we can see, by manipulating the positions of the pit, gold, and Wumpus the agent has unfortunately failed since the Wumpus have killed him.

As for our fails in the code, we have encountered a few bugs, such as, the game doesn't end when the agent falls into a pit in some cases, I believe that the problem came from the placement of the pit and agent but we are not sure. We also couldn't implement a mechanism to kill the Wumpus so that he can win the game that way, therefore, in this version the agent should only avoid the Wumpus and finds the gold to win.

We have learned a lot from this project on how to use a smart agent that learns and perceive its surroundings using a predefined knowledge base. This project was coded by Ahmed AL Hilal and my teammate Yasmine Najd.