

# Wumpus World Report

---

1. Wumpus implementation
2. Wumpus game rules
3. World notions and rules
4. Possible worlds
  - Possible without shooting
  - Possible with shooting only
  - Impossible worlds
5. World outputs
  - World 1 output
  - World 2 output

## Wumpus implementation

This report was created as a part of the author's implementation of the Wumpus World on Prolog. You can find Prolog rules within `wumpus/wumpus.pl` file of this assignment package. Worlds can be found in `wumpus/world*.pl`, where `*` is the world number.

## Wumpus game rules

The rules are considered in classical understanding with following changes.

First of all, let us define, that game field is in a rectangular shape with positive coordinates, starting from [1, 1]. The agent will always start from this [1, 1], this starting position will be always clear from any other items. The agent can move up, down, left, right within 1 unit. Winning the game is acquired by taking Gold. There is only one Wumpus and one Gold, with number of pits from 0 to (world\_width \* world\_height - 2), these items do not overlap. Hence the original task of finding **all possible ways**, counting score does not lead to acquiring significant amount of information. By the same concern, we will have possible paths, where Wumpus will be shot, thus, game may be considered without Wumpus at all, but to conform with rules, let us say, that due to no difference in shooting turn number, we will consider shooting to nearby cells with Wumpus only. And again, with the same concern, there is no need to percept stinch, breezes, glitter and bumps. Returning back from the cave is not necessary, since the path back will be the same, as forward, but reversed.

## World notions and rules

In author's agent implementation, the world rules are following:

```
agent([_, _]).      % coordinates of agent
wumpus([_, _]).     % coordinates of wumpus
pit([_, _]).        % coordinates of pit
agent([_, _]).      % coordinates of gold
```

```
world_width(_).    % world width (X-coordinate)
world_height(_).   % world height (Y-coordinate)
```

## Possible worlds

Possible worlds are those, where the agent can acquire Gold with or without shooting Wumpus by at least 1 way. Possible worlds outputs are situated below.

Possible without shooting

*The agent can acquire Gold even if it cannot shoot at all.*

From `world1.pl`

```
-----
|   |   | P |   | P |
-----
|   |   |   |   |   |
-----
|   | P |   | W |   |
-----
|   |   | P |   | G |
-----
| A |   |   | P |   |
-----
```

This world is traversable by A\* algorithm and thus is possible. No shot Wumpus needed. One of the possible ways is: `Path = [[1, 1], [1, 2], [1, 3], [1, 4], [2, 4], [3, 4], [4, 4], [5, 4], [5, 3], [5, 2]]`

Possible with shooting only

*The only path to Gold lies through Wumpus.*

From `world2.pl`

```
-----
|   |   | P |   | P |
-----
| W |   |   |   |   |
-----
|   | P |   |   |   |
-----
|   |   | P |   | G |
-----
| A |   |   | P |   |
-----
```

This world is traversable by A\* algorithm only if Wumpus is getting shot and thus is also possible. One of the possible ways is: `Path = [[1, 1], [1, 2], [1, 3], [1, 4], [2, 4], [3, 4], [4, 4], [5, 4], [5, 3], [5, 2]]`

## Impossible worlds

*Those where the agent cannot acquire Gold in any way.*

From `world3.pl`

```
-----  
|   |   | P |   | P |  
-----  
| P |   |   |   |   |  
-----  
|   | P |   | W |   |  
-----  
|   |   | P |   | G |  
-----  
| A |   |   | P |   |  
-----
```

Pits disposition does not allow the agent to come across and thus it makes the world impossible. Prolog will output `false` on this.

## World outputs

### World 1 output

```
Path = [[1, 1], [1, 2], [1, 3], [1, 4], [1, 5], [2, 5], [2, 4], [3, 4],  
[3, 3], [4, 3], [4, 4], [5, 4], [5, 3], [5, 2]]  
Path = [[1, 1], [1, 2], [1, 3], [1, 4], [1, 5], [2, 5], [2, 4], [3, 4],  
[3, 3], [4, 3], [4, 2], [5, 2]]  
Path = [[1, 1], [1, 2], [1, 3], [1, 4], [1, 5], [2, 5], [2, 4], [3, 4],  
[3, 3], [4, 3], [5, 3], [5, 2]]  
Path = [[1, 1], [1, 2], [1, 3], [1, 4], [1, 5], [2, 5], [2, 4], [3, 4],  
[4, 4], [5, 4], [5, 3], [5, 2]]  
Path = [[1, 1], [1, 2], [1, 3], [1, 4], [1, 5], [2, 5], [2, 4], [3, 4],  
[4, 4], [5, 4], [5, 3], [4, 3], [4, 2], [5, 2]]  
Path = [[1, 1], [1, 2], [1, 3], [1, 4], [1, 5], [2, 5], [2, 4], [3, 4],  
[4, 4], [4, 3], [4, 2], [5, 2]]  
Path = [[1, 1], [1, 2], [1, 3], [1, 4], [1, 5], [2, 5], [2, 4], [3, 4],  
[4, 4], [4, 3], [5, 3], [5, 2]]  
Path = [[1, 1], [1, 2], [1, 3], [1, 4], [2, 4], [3, 4], [3, 3], [4, 3],  
[4, 4], [5, 4], [5, 3], [5, 2]]  
Path = [[1, 1], [1, 2], [1, 3], [1, 4], [2, 4], [3, 4], [3, 3], [4, 3],  
[4, 2], [5, 2]]  
Path = [[1, 1], [1, 2], [1, 3], [1, 4], [2, 4], [3, 4], [3, 3], [4, 3],  
[5, 3], [5, 2]]  
Path = [[1, 1], [1, 2], [1, 3], [1, 4], [2, 4], [3, 4], [4, 4], [5, 4],
```



[illegible]