Logical Agents

Knowledge-Based Agents

```
function KB-AGENT(percept) returns an action
    persistent: KB, a knowledge base
                t, a counter, initially 0, indicating time

    TELL(KB, MAKE-PERCEPT-SENTENCE(percept, t))
    action ← ASK(KB, MAKE-ACTION-QUERY(t))
    TELL(KB, MAKE-ACTION-SENTENCE(action, t))
    t ← t + 1
    return action
```

Wumpus World

- The wumpus world is a cave consisting of rooms connected by passageways.
- Lurking somewhere in the cave is the terrible wumpus, a beast that eats anyone who enters its room.
- The wumpus can be shot by an agent, but the agent has only one arrow.
- Some rooms contain bottomless pits that will trap anyone who wanders into these rooms (except for the wumpus, which is too big to fall in).
- The only redeeming feature of this bleak environment is the possibility of finding a heap of gold.

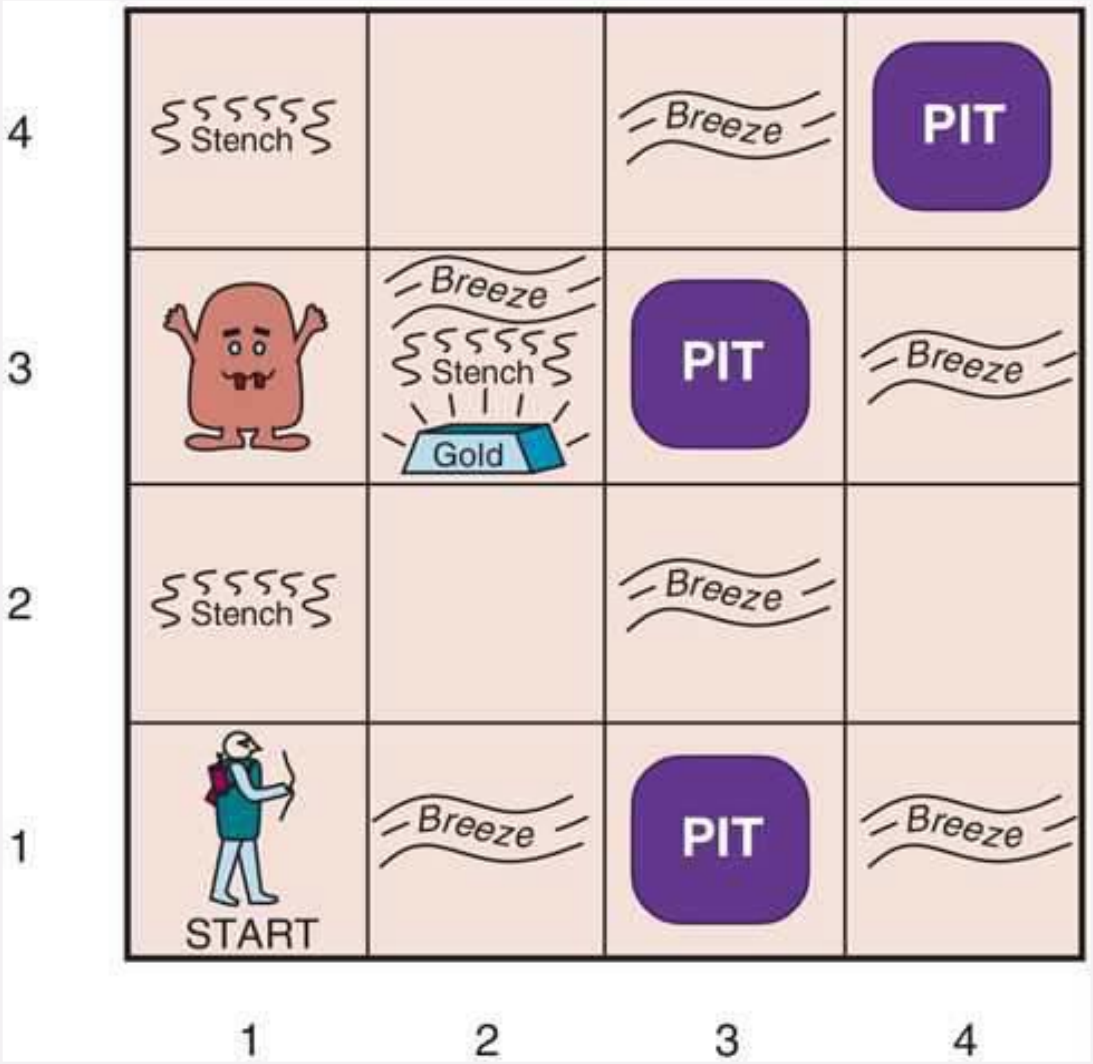
Wumpus World

Performance measure

Environment

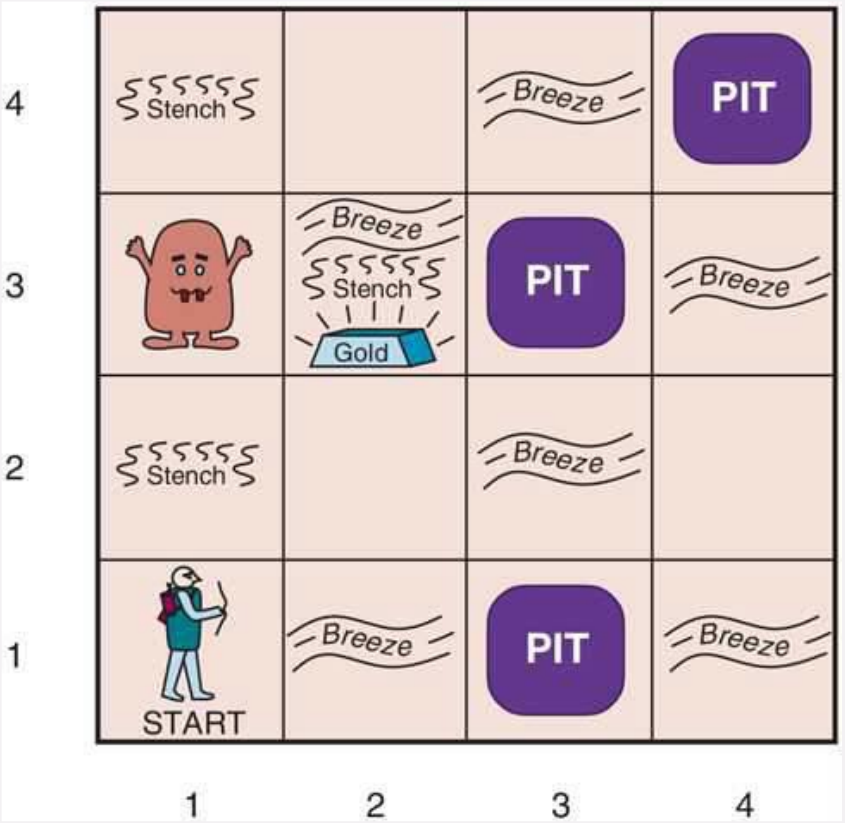
Actuators

Sensors



Performance measure

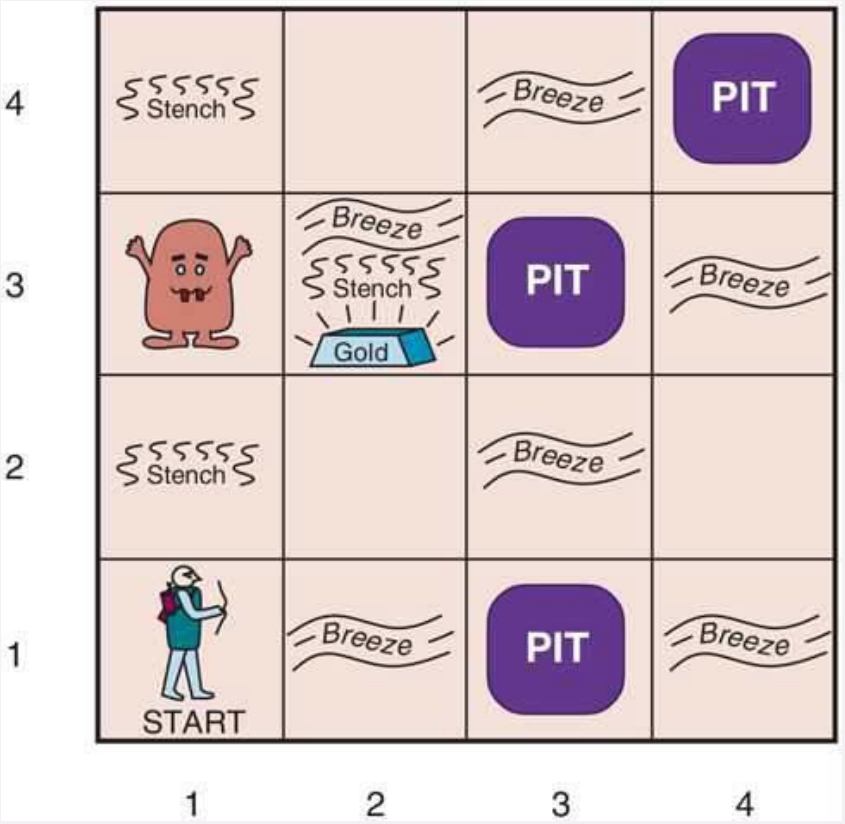
- +1000 for climbing out of the cave with the gold
- -1000 for falling into a pit or being eaten by the Wumpus
- -1 for each action taken, and
- -10 for using up the arrow.



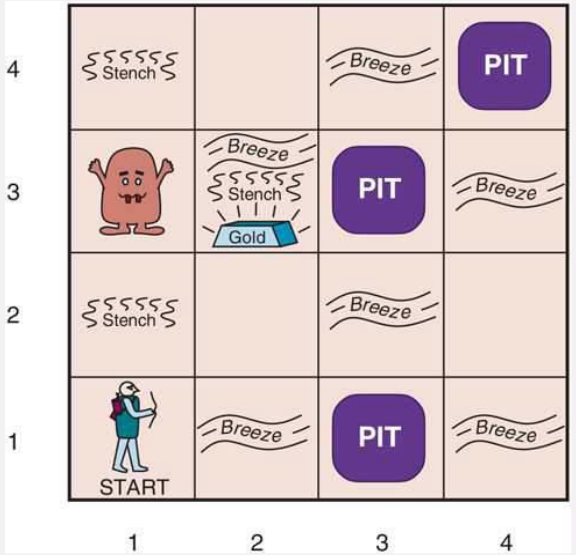
- The game ends either when the agent dies or when the agent climbs out of the cave.

Environment

- A 4*4 grid of rooms, with walls surrounding the grid.
- The agent always starts in the square labeled [1,1], facing to the east.
- The locations of the gold and the wumpus are chosen randomly, with a uniform distribution, from the squares other than the start square.
- In addition, each square other than the start can be a pit, with probability 0.2.



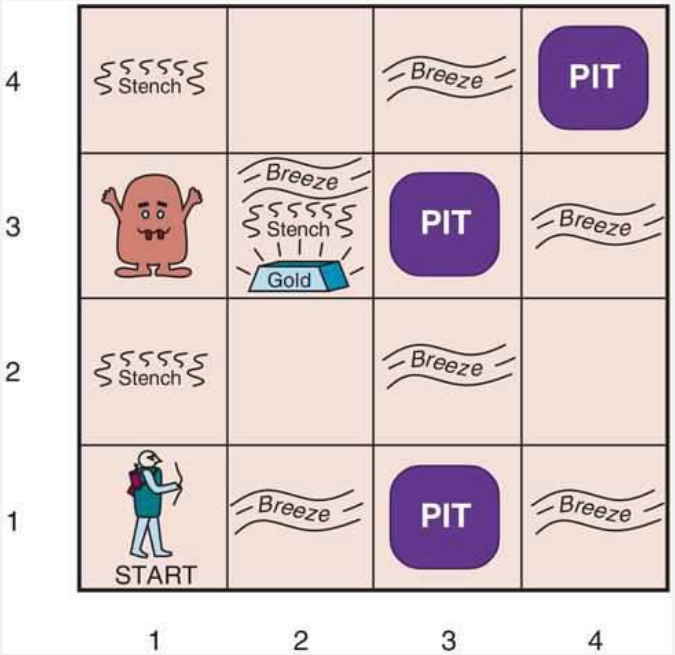
Actuators



- The agent can move Forward, TurnLeft by 90, or TurnRight by 90.
- The agent dies a miserable death if it enters a square containing a pit or a live wumpus.
- If an agent tries to move forward and bumps into a wall, then the agent does not move.
- The action Grab can be used to pick up the gold if it is in the same square as the agent.
- The action Shoot can be used to fire an arrow in a straight line in the direction the agent is facing. The arrow continues until it either hits (and hence kills) the wumpus or hits a wall.
- The agent has only one arrow, so only the first Shoot action has any effect.
- Finally, the action Climb can be used to climb out of the cave, but only from square [1,1].

Sensors

- In the squares directly (not diagonally) adjacent to the wumpus, the agent will perceive a *Stench*.
- In the squares directly adjacent to a pit, the agent will perceive a *Breeze*.
- In the square where the gold is, the agent will perceive a *Glitter*.
- When an agent walks into a wall, it will perceive a *Bump*.
- When the wumpus is killed, it emits a woeful *Scream* that can be perceived anywhere in the cave.



Playing the game – the basic approach

- The agent knows its goal

  - getting the gold without falling in a pit or getting killed by Wumpus

- Agent knows that the layout is a rectangular grid of rooms
- Agent can sense breeze and stench

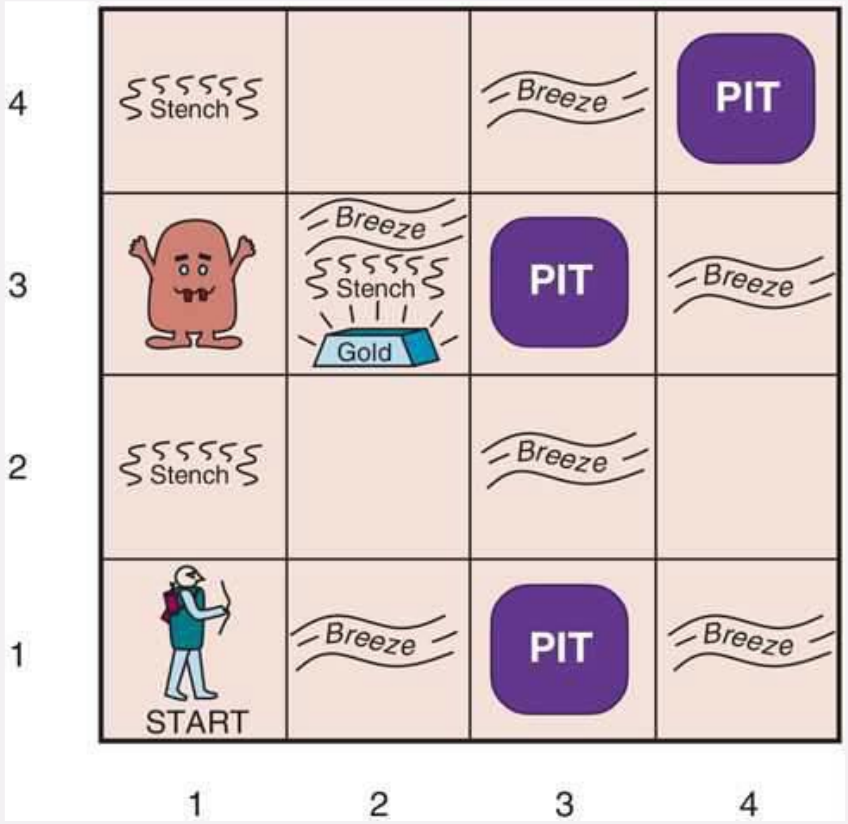
  - infers that neighboring room has pit or Wumpus

- The agent needs to build a map of its world

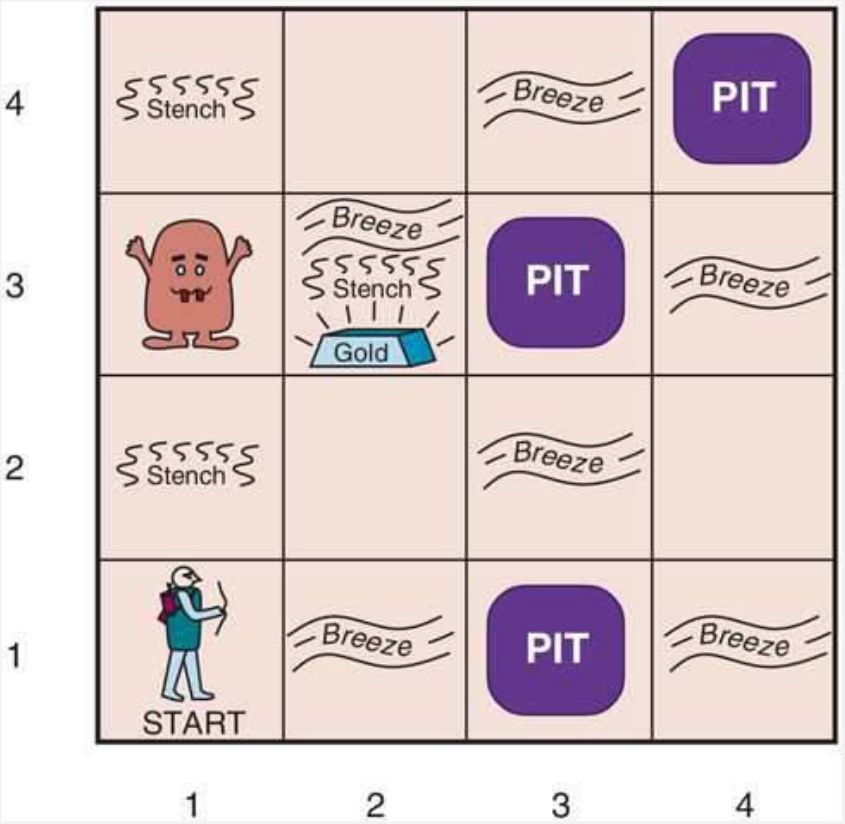
  - based on the inputs it receives from its sensors it has to infer

    - which rooms are safe
    - which rooms have pit
    - which room has Wumpus
    - which room has gold
    - return path from room with gold to the Start node

- Building of map requires inferencing



Playing the game – some details



- The agent starts with some knowledge

  - about gold, Wumpus, pits, breeze, stench etc.

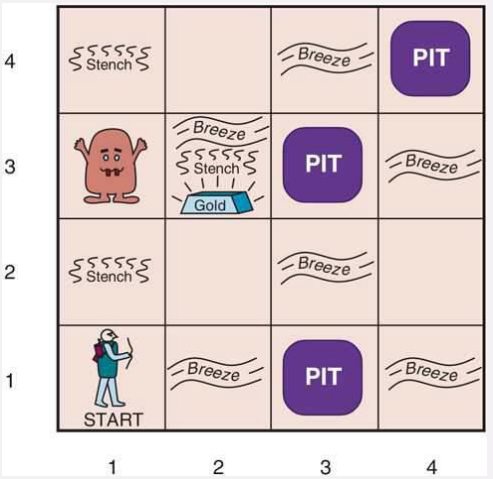
- The agent starts from Start node and knows that node is safe
- Knows nothing about the state of other rooms
- Can sense breeze, stench or nothing in a given room
- This gives some information about adjacent rooms

  - e.g. if agent is in room (i, j) and feels a breeze then at least one of the 4-neighbors has a pit

  - *how can the agent know which neighbor has a pit?*

- Based on this type of information, it has to build a map of its world

The ingredients



- **Knowledge base**
  - need a representation
  - a matrix of room numbers
  - info about the room i.e. world view

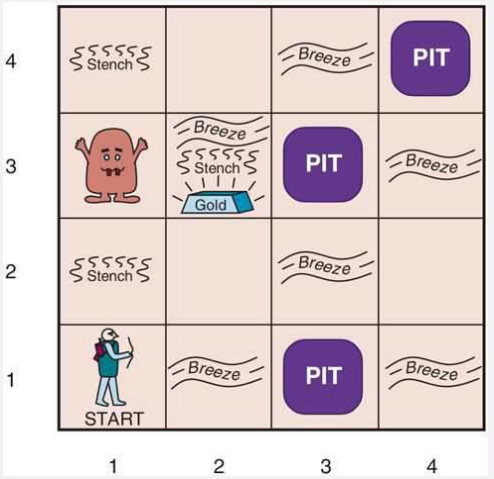
- **The inferencing system**
  - uses the prior knowledge
  - uses the percept sequence
  - infers new knowledge
  - adds it to the prior knowledge

- **After each step**
  - Inputs from sensors collected
  - Inferencing system uses these inputs and the KB to infer something about neighboring rooms
  - Adds this information to the KB
  - Agent takes next step by querying KB

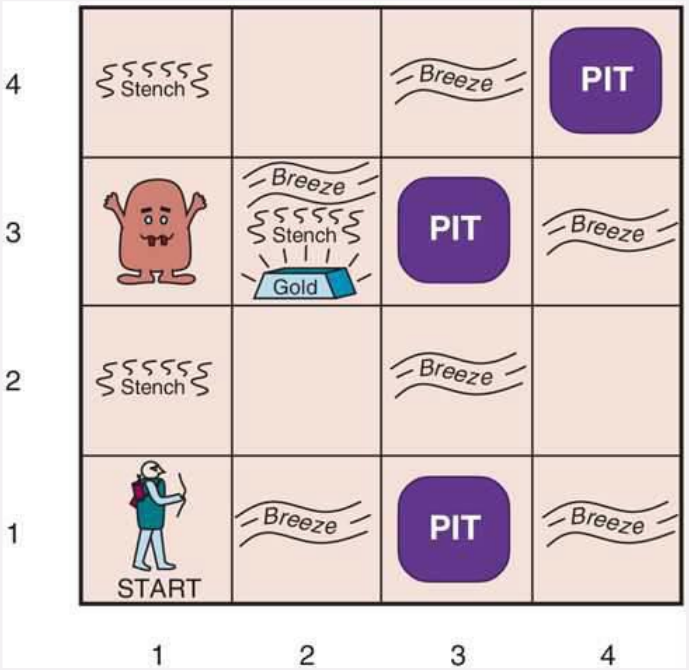
The knowledge base for Wumpus World

- The Wumpus world is defined by
  - the rooms – labelled as (x, y)
  - the state of the rooms
- What are the possible states of any room?
  - Not Known (NK)
  - Safe (S)
  - Wumpus in neighborhood(WN)
  - Pit in neighborhood (PN)
  - Goal (G)

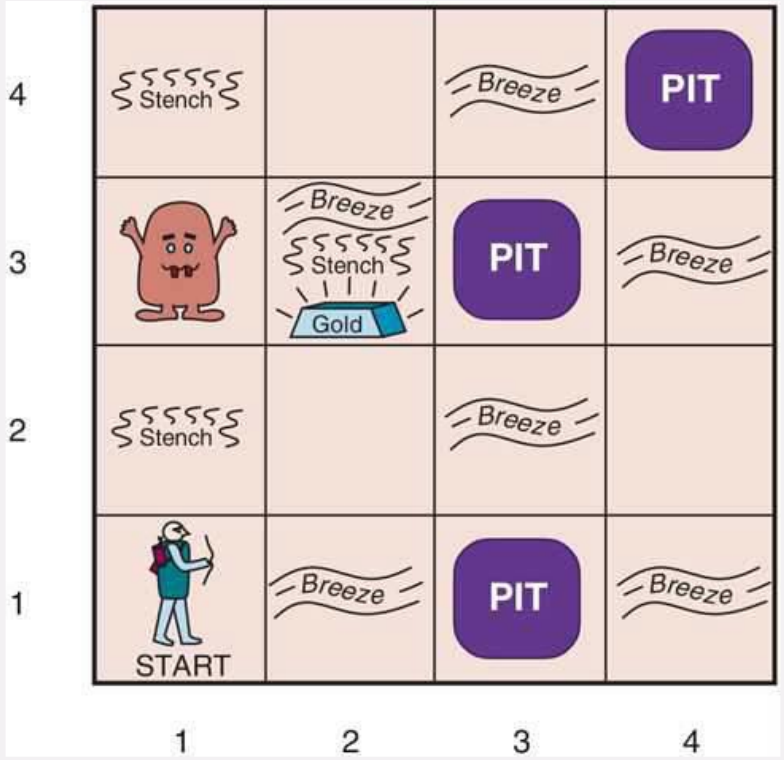
- Initially, all rooms except Start, are NK
- As the agent progresses, the states of rooms will change based on
  - inputs from sensors
  - current knowledge of the world



Understanding the Knowledge Base



- Knowledge base has to
  - store the knowledge relevant for the specific problem
  - allow the inferencing system to interact with it
  - allow new knowledge to be added by inferencing system
- *How do we represent knowledge?*
- For the Wumpus world it was sufficient to store current information about each room



1,4	2,4	3,4	4,4
1,3	2,3	3,3	4,3
1,2 OK	2,2	3,2	4,2
1,1 A OK	2,1 OK	3,1	4,1

A = Agent
B = Breeze
G = Glitter, Gold
OK = Safe square
P = Pit
S = Stench
V = Visited
W = Wumpus

1,4	2,4	3,4	4,4
1,3	2,3	3,3	4,3
1,2 OK	2,2 P?	3,2	4,2
1,1 V OK	2,1 A B OK	3,1 P?	4,1

None, None, None, None, None,

None, Breeze, None, None, None,

1,4	2,4	3,4	4,4
1,3 W!	2,3	3,3	4,3
1,2 A S OK	2,2 OK	3,2	4,2
1,1 V OK	2,1 B V OK	3,1 P!	4,1

A = Agent
B = Breeze
G = Glitter, Gold
OK = Safe square
P = Pit
S = Stench
V = Visited
W = Wumpus

1,4	2,4 P?	3,4	4,4
1,3 W!	2,3 A S G B	3,3 P?	4,3
1,2 S V OK	2,2 V OK	3,2	4,2
1,1 V OK	2,1 B V OK	3,1 P!	4,1

Stench, None, None, None, None,

Stench, Breeze, Glitter, None, None,