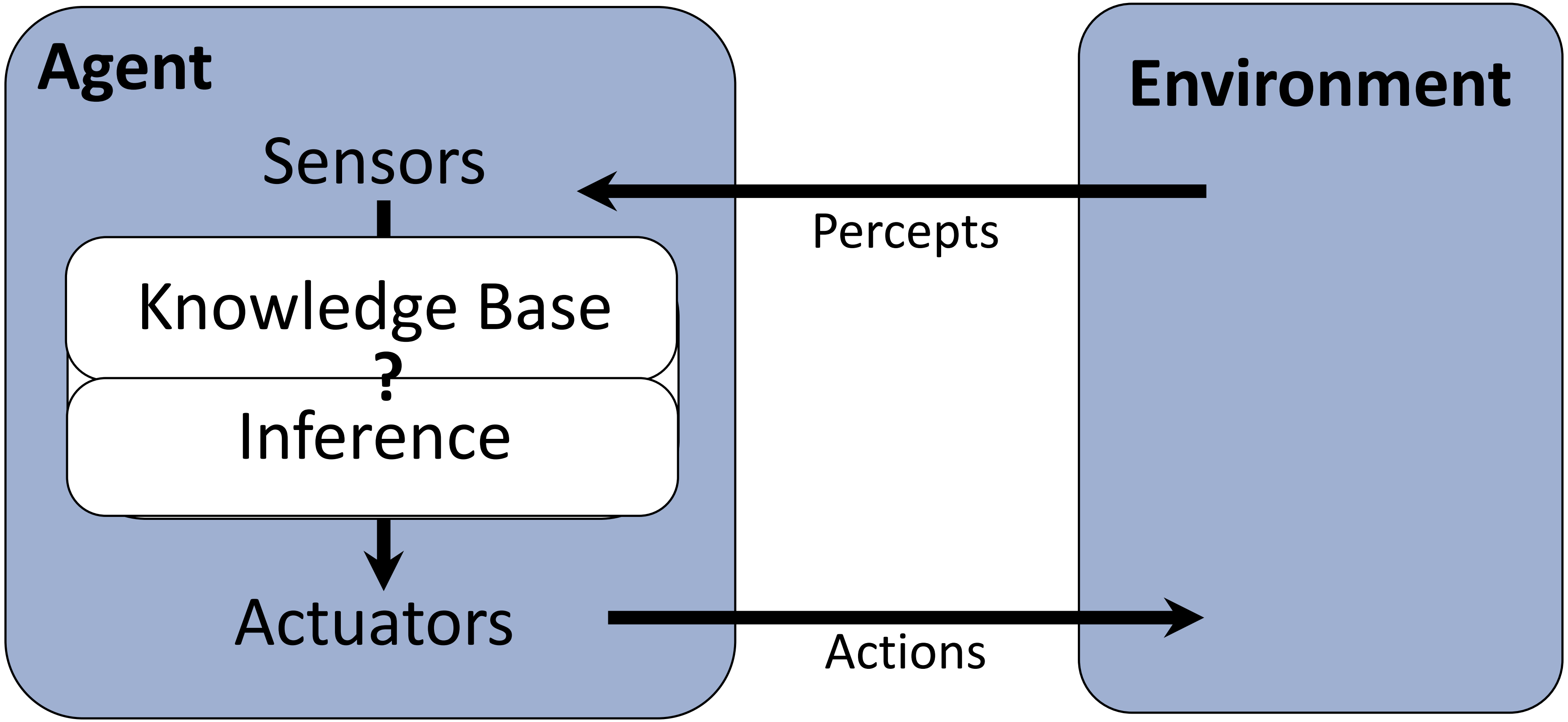
Knowledge Based Agents

Outline

- Knowledge-based agents
- Wumpus world
- Logic in general - models and entailment
- Propositional (Boolean) logic
- Equivalence, validity, satisfiability

Logical Agents

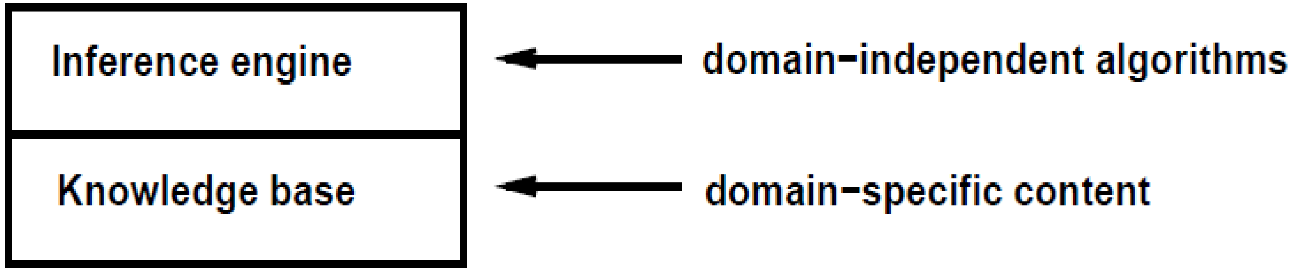
Logical agents and environments



Knowledge Based Agents

- **Knowledge-based agents** are agents that know about their world and reason about their courses of action.
- A **knowledge-based agent** can combine general knowledge with current percepts to infer hidden aspects of the current state prior to selecting actions.
- **Knowledge base**, or KB is a set of **sentences** expressed in a language called a **knowledge representation language** and represents some assertion about the world.

Knowledge Base

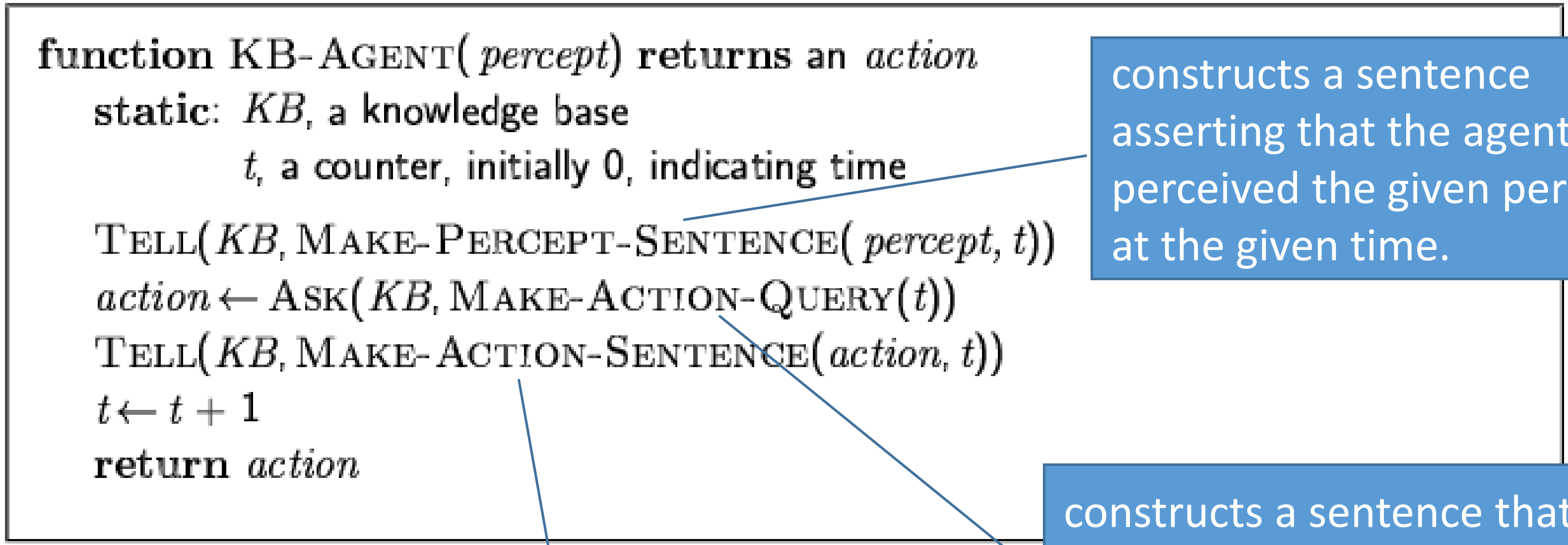


- **Knowledge base**: Set of sentences in a formal language
  - A database of facts (knowledge)
- **Inference Engine**: Set of rules for deducing conclusions
  - A database of rules
- Declarative approach for building an agent
  - **TELL** it what it needs to know
  - Then it can **ASK** itself what to do - answers should follow from the KB
  - As an example, think of a child

Knowledge Based Agents

- It takes a percept as input and returns an action.
- The agent maintains a **knowledge base**, KB, which may initially contain some **background knowledge.**
- Each time the agent program is called, it does three things.
  - First, it **TELLS** the knowledge base what it perceives.
  - Second, it **ASKS** the knowledge base what action it should perform.
    - Reasoning may be done about the current state of the world, about the outcomes of possible action sequences, and so on.
  - Third, the agent records its choice with **TELL** and executes the action. TELL is necessary to let the knowledge base know that the *action* has actually been executed.

A simple knowledge-based agent



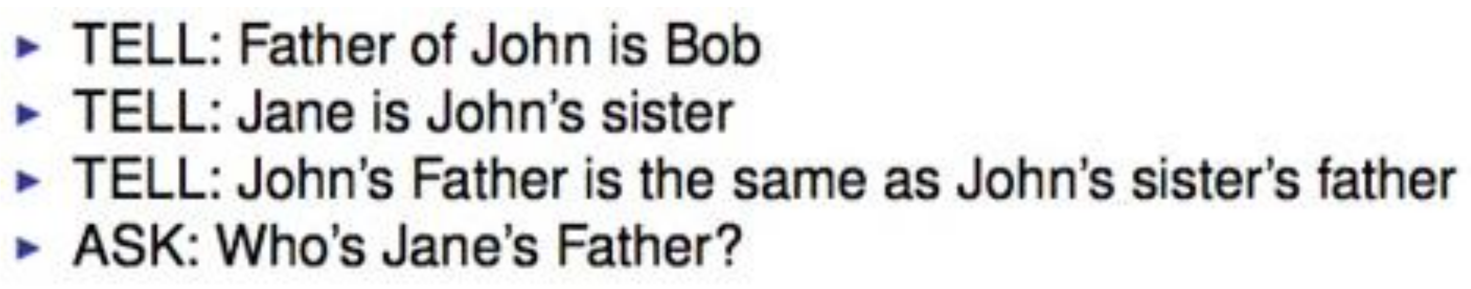
```
function KB-AGENT(percept) returns an action
    static: KB, a knowledge base
            t, a counter, initially 0, indicating time

    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
```

constructs a sentence asserting that the agent perceived the given percept at the given time.

constructs a sentence that asks what action should be done at the current time.

constructs a sentence asserting that the chosen action was executed.



- TELL: Father of John is Bob
- TELL: Jane is John's sister
- TELL: John's Father is the same as John's sister's father
- ASK: Who's Jane's Father?

Logical Agents

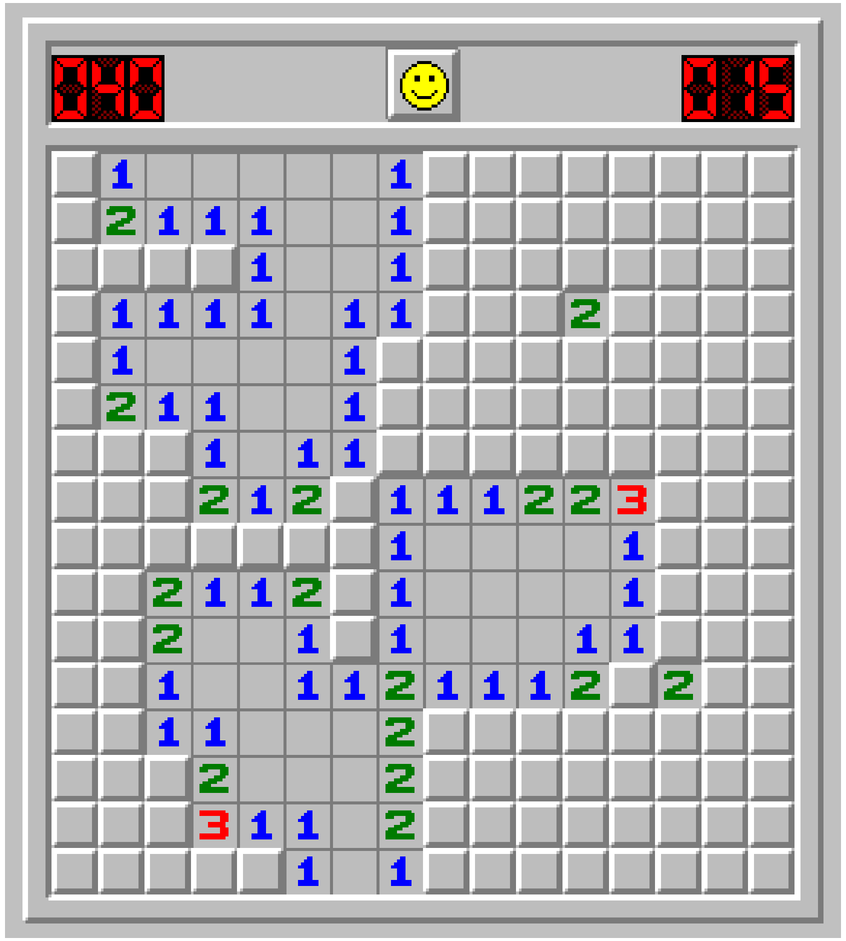
- So what do we TELL our knowledge base (KB)?
    - Facts (sentences)
        - The grass is green
        - The sky is blue
    - Rules (sentences)
        - Eating too much candy makes you sick
        - When you’re sick you don’t go to school
    - Percepts and Actions (sentences)
        - Pat ate too much candy today

- What happens when we ASK the agent?
    - Inference – new sentences created from old
        - Pat is not going to school today

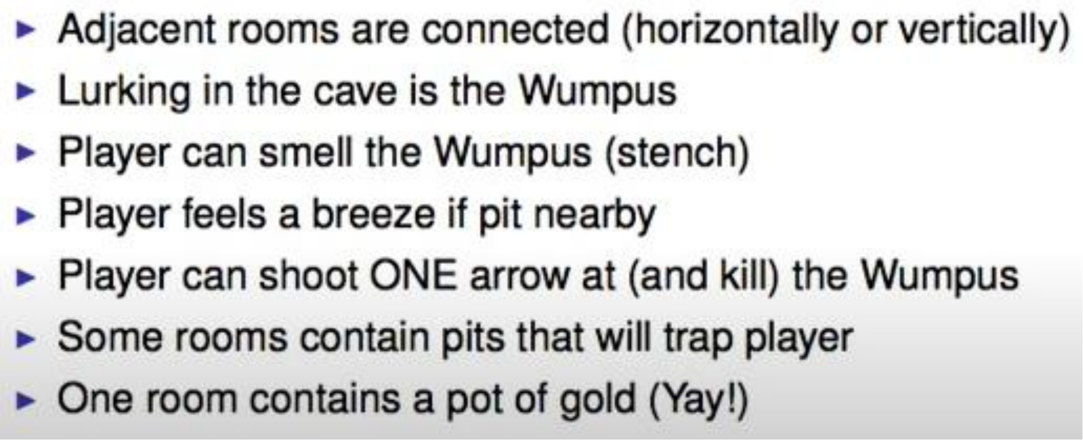
Knowledge Based Agents

- Abilities of Knowledge Based Agents
- The agent must be able to:
  - Represent states, actions, etc.
  - Incorporate new percepts
  - Update internal representations of the world
  - Deduce hidden properties of the world
  - Deduce appropriate actions

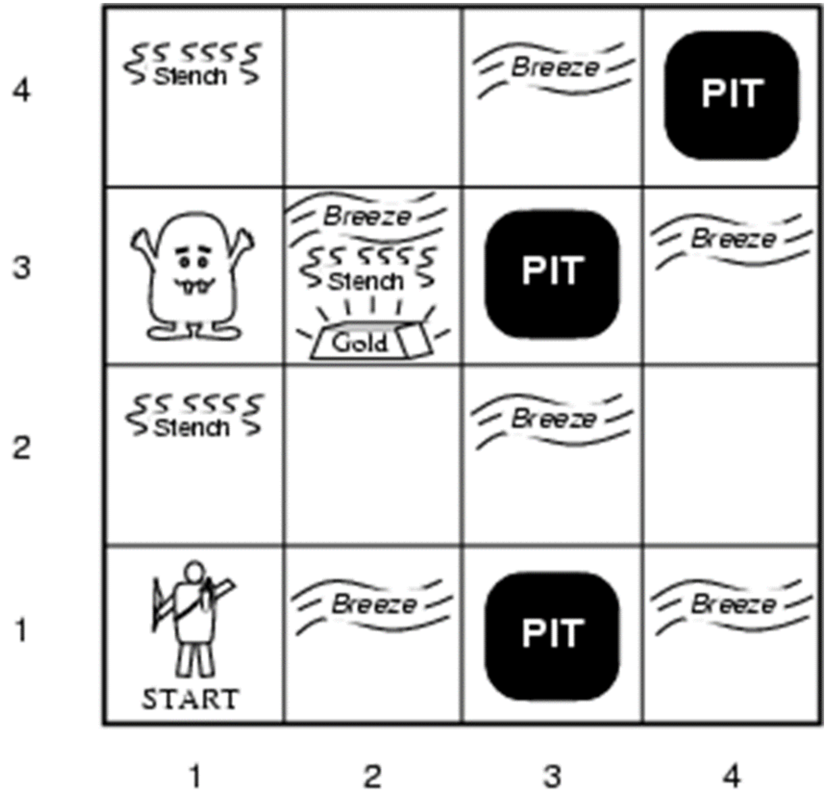
Minesweeper



Wumpus World

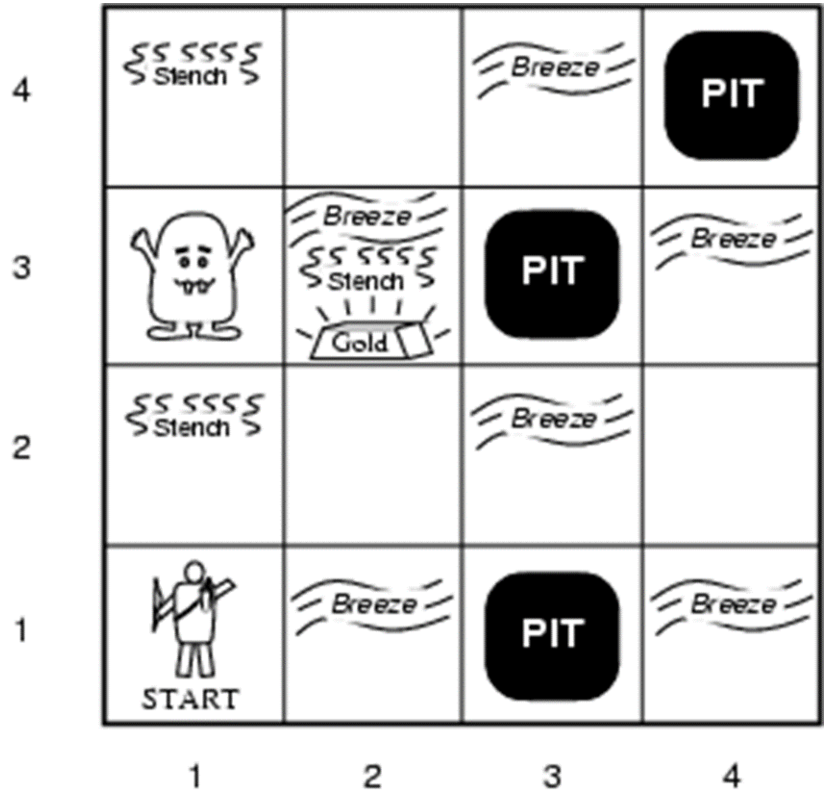


- Adjacent rooms are connected (horizontally or vertically)
- Lurking in the cave is the Wumpus
- Player can smell the Wumpus (stench)
- Player feels a breeze if pit nearby
- Player can shoot ONE arrow at (and kill) the Wumpus
- Some rooms contain pits that will trap player
- One room contains a pot of gold (Yay!)



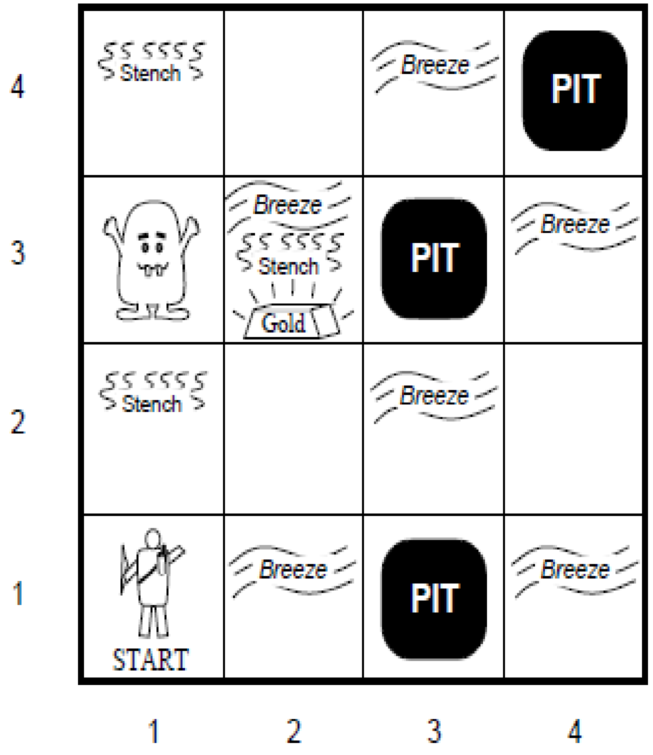
Wumpus World

- The Wumpus computer game
- The agent explores a cave consisting of rooms connected by passageways
- Lurking somewhere in the cave is the Wumpus, a beast that eats any agent that enters its room
- Some rooms contain bottomless pits that trap any agent that wanders into the room
- Occasionally, there is a heap of gold in a room.
- The goal is to collect the gold and exit the world without being eaten



Wumpus World description

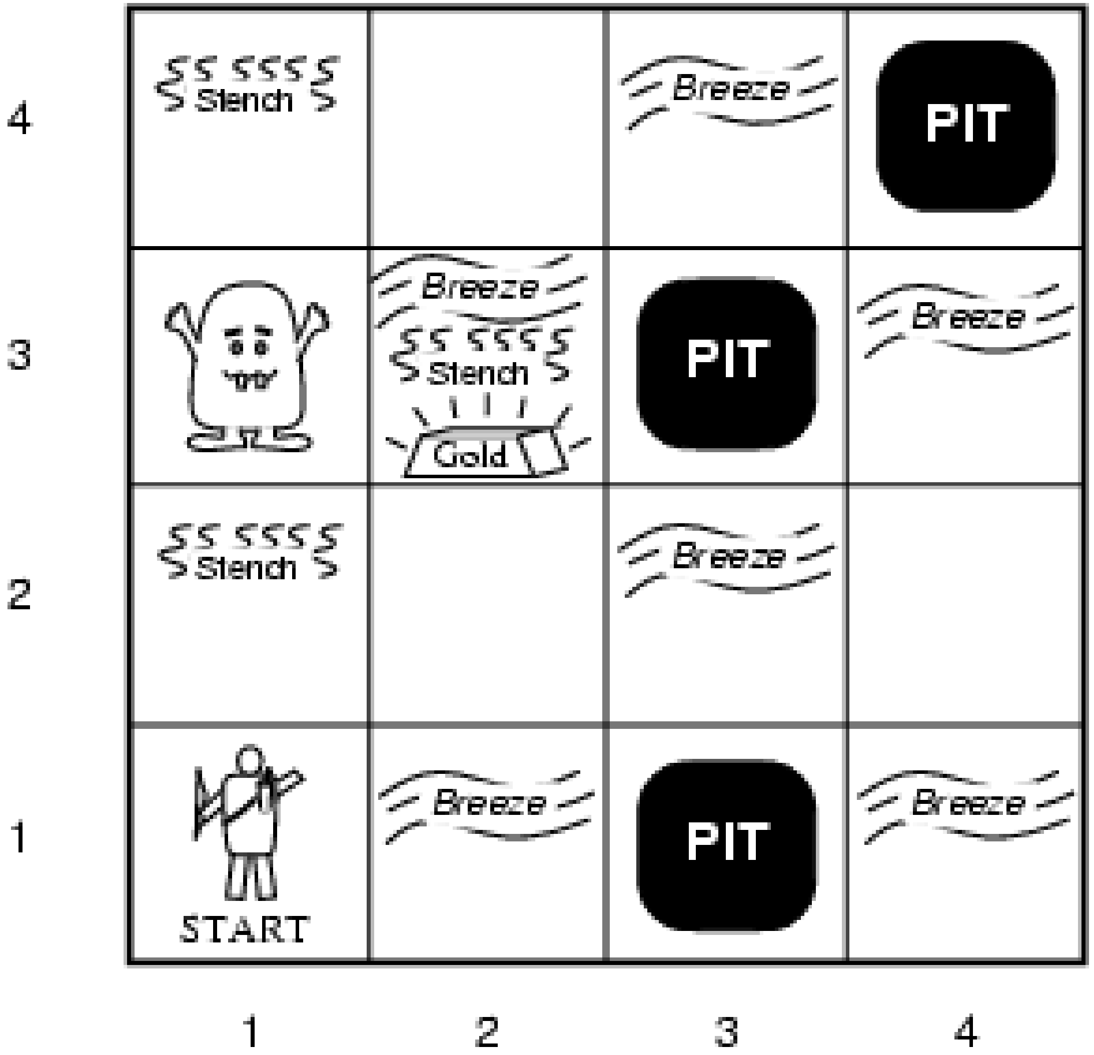
- **Performance measure**
  - gold +1000, death -1000
  - -1 per step, -10 for using the arrow
- **Environment**
  - Squares adjacent to wumpus are smelly (stench)
  - Squares adjacent to pit are breezy
  - Glitter iff gold is in the same square
  - Shooting kills wumpus if you are facing it
  - Shooting uses up the only arrow
  - Grabbing picks up gold if in same square
  - Releasing drops the gold in same square
- **Sensors**: Stench, Breeze, Glitter, Bump, Scream
- **Actuators**: Left turn, Right turn, Forward, Grab, Release, Shoot



Wumpus World

- The agent always starts in the field [1,1]

- Agent’s task is to find the gold, return to the field [1,1] and climb out of the cave



Agent in a Wumpus world: Percepts

- The agent perceives
  - a **stench** in the square containing the Wumpus and in the adjacent squares (not diagonally)
  - a **breeze** in the squares adjacent to a pit
  - a **glitter** in the square where the gold is
  - a **bump**, if it walks into a wall
  - a woeful **scream** everywhere in the cave, if the Wumpus is killed
- The percepts are given as a five-symbol list. If there is a stench and a breeze, but no glitter, no bump, and no scream, the percept is

  [Stench, Breeze, None, None, None]
- The agent cannot perceive its own location

Agent in a Wumpus world: Actions

- **go forward**
- **turn right** 90 degrees
- **turn left** 90 degrees
- **grab**: Pick up an object that’s in the same square as the agent
- **shoot**: Fire an arrow in a straight line in the direction the agent is facing. It continues until it hits and kills the Wumpus or hits the outer wall. The agent has only one arrow, so only the first shoot action has any effect
- **climb** is used to leave the cave and is only effective in the start square
- **die**: This action automatically and irretrievably happens if the agent enters a square with a pit or a live Wumpus

Agent in a Wumpus world : Goal

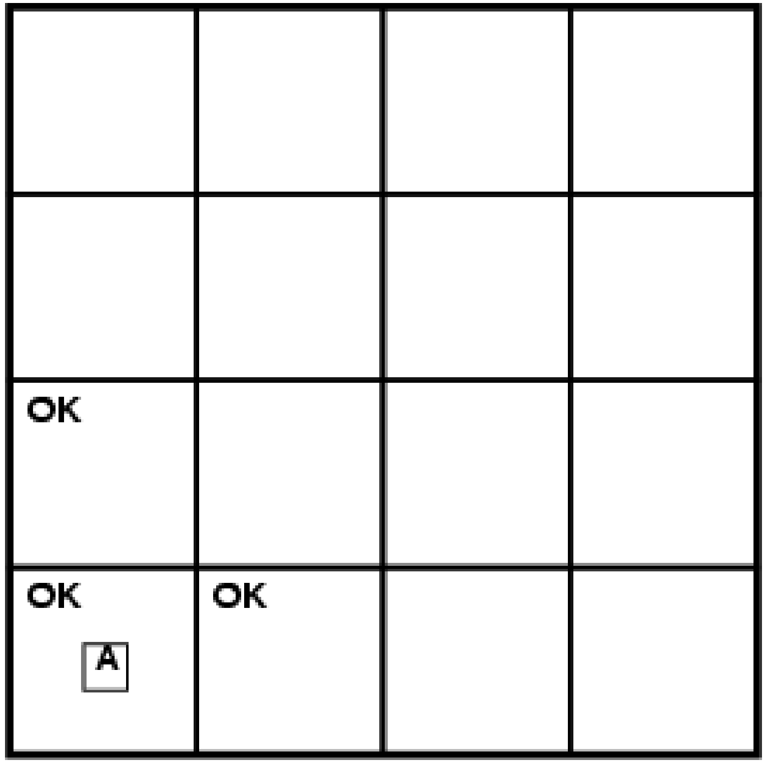
The agent’s goal is to find the gold and bring it back to the start square as quickly as possible, without getting killed

- 1,000 points reward for climbing out of the cave with the gold
- 1 point deducted for every action taken
- 10,000 points penalty for getting killed

Wumpus world characterization

- **Fully Observable:**
  - No – only local perception
- **Deterministic:**
  - Yes – outcomes exactly specified
- **Episodic:**
  - No – sequential at the level of actions
- **Static:**
  - Yes – Wumpus and Pits do not move
- **Discrete:**
  - Yes
- **Single-agent:**
  - Yes – Wumpus is essentially a natural feature

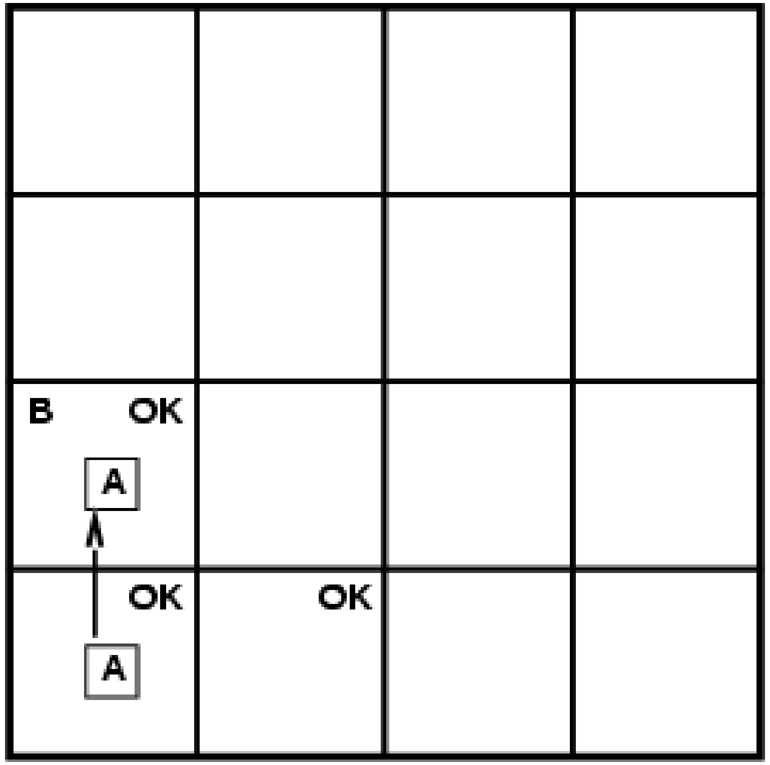
Exploring a wumpus world



A	agent
B	breeze
G	glitter
OK	safe cell
P	pit
S	stench
W	wumpus

- Agent’s initial knowledge: Agent in [1,1] ; and [1,1] is a safe square
- The initial situation, after percept is [None, None, None, None, None] in (1,1)
- So agent concludes that [1,2] and [2,1], are free of dangers

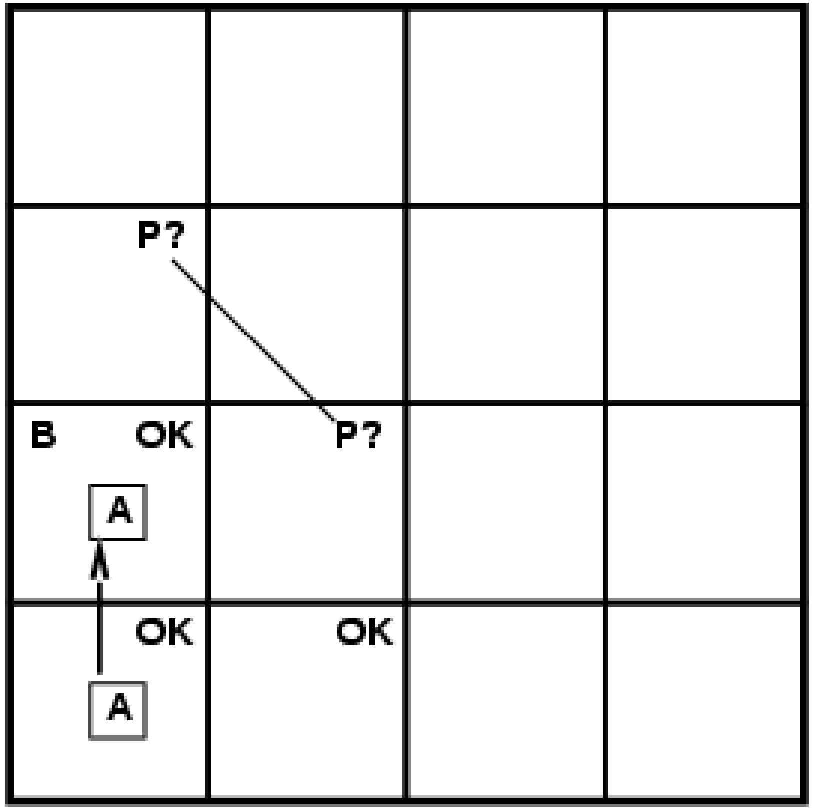
Exploring a wumpus world



A	agent
B	breeze
G	glitter
OK	safe cell
P	pit
S	stench
W	wumpus

- After one move, with percept [None, Breeze, None, None, None]

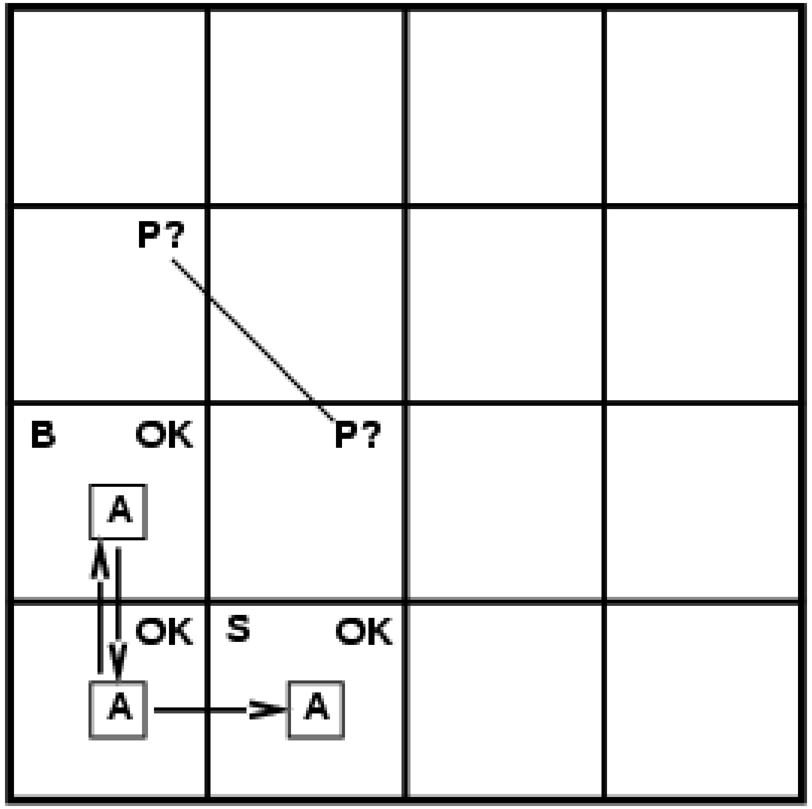
Exploring a wumpus world



A	agent
B	breeze
G	glitter
OK	safe cell
P	pit
S	stench
W	wumpus

- so there must be a pit in a neighboring square.
- Either [2, 2] or [3,1]; or both

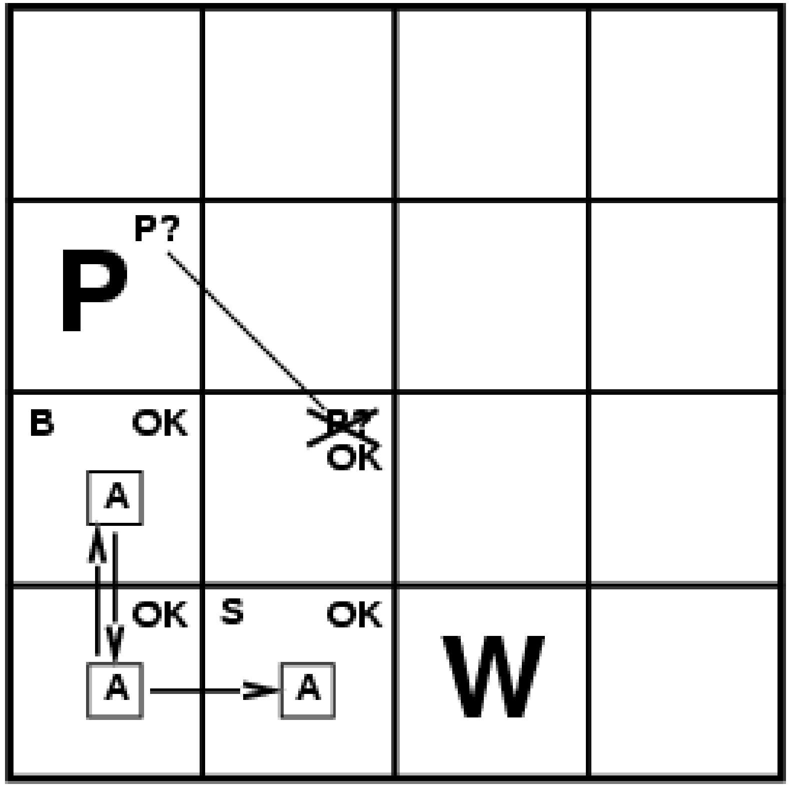
Exploring a wumpus world



A	agent
B	breeze
G	glitter
OK	safe cell
P	pit
S	stench
W	wumpus

- After the third move, with percept [Stench, None, None, None, None].
- The stench in [1,2] means that there must be a wumpus nearby.

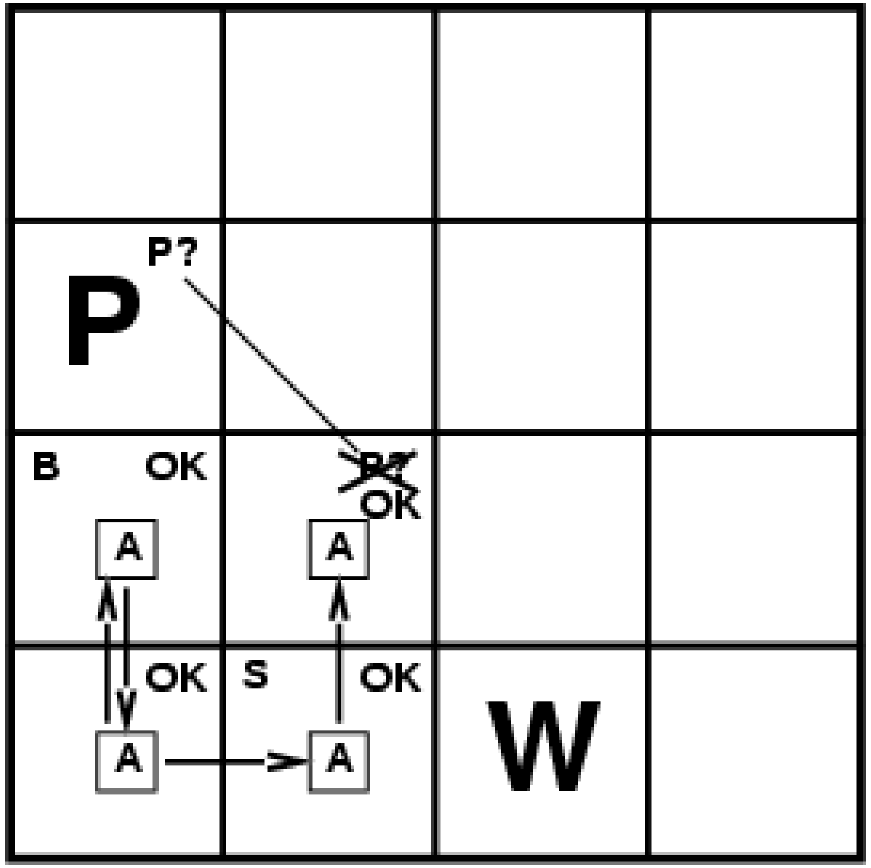
Exploring a wumpus world



A	agent
B	breeze
G	glitter
OK	safe cell
P	pit
S	stench
W	wumpus

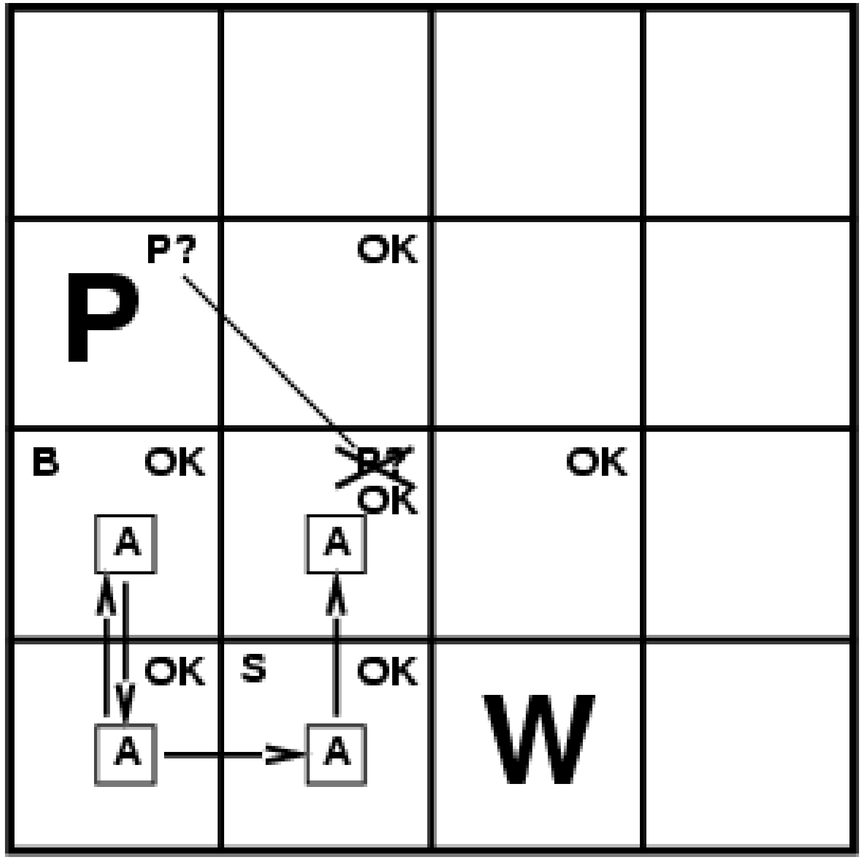
- But Wumpy cannot be in [2,2] (the agent would have detected a stench when it was in [2,1]).
- Similarly, the lack of a breeze in [1,2] implies that there is no pit in [2,2].
- So pit must be in [3,1]

Exploring a wumpus world



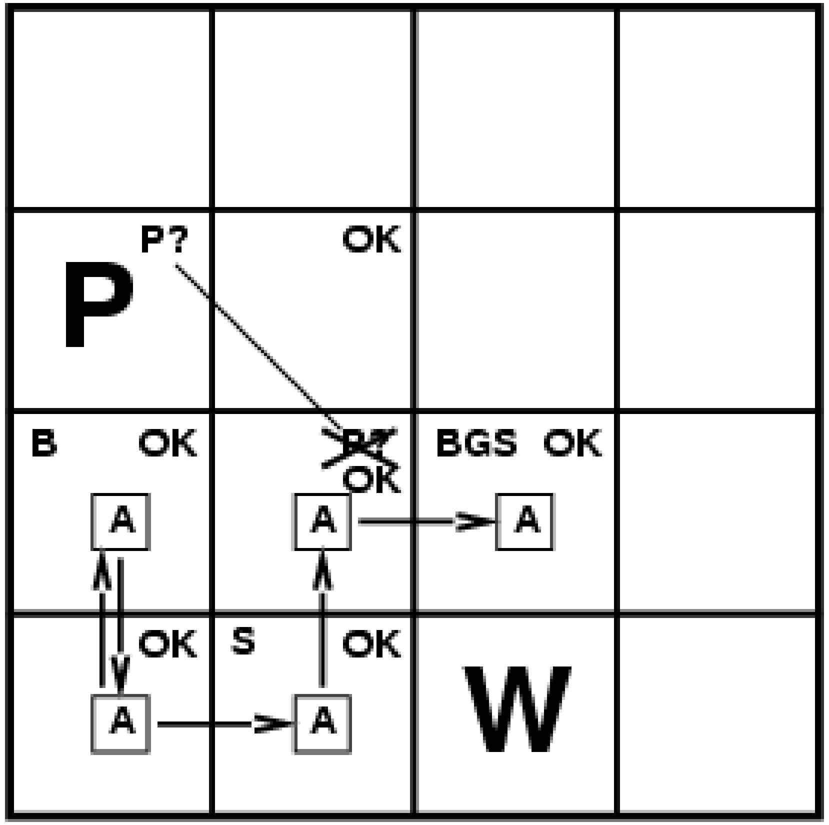
A	agent
B	breeze
G	glitter
OK	safe cell
P	pit
S	stench
W	wumpus

Exploring a wumpus world



A	agent
B	breeze
G	glitter
OK	safe cell
P	pit
S	stench
W	wumpus

Exploring a wumpus world



A	agent
B	breeze
G	glitter
OK	safe cell
P	pit
S	stench
W	wumpus

Representing Knowledge

- The agent that solves the Wumpus world can most effectively be implemented by a knowledge-based approach
- Need to represent states and actions, update internal representations, deduce hidden properties and appropriate actions
- Need a formal representation for the KB
- And a way to reason about that representation

Logic in general

- **Logics** are formal languages for representing information such that conclusions can be drawn
- **Syntax** defines the sentences in the language
- **Semantics** define the "meaning" of sentences
  - i.e., define **truth** of a sentence in a world
- E.g., the language of arithmetic
  - x+2 ≥ y is a sentence; x2+y > {} is not a sentence
  - x+2 ≥ y is true iff the number x+2 is no less than the number y
  - x+2 ≥ y is true in a world where x = 7, y = 1
  - x+2 ≥ y is false in a world where x = 0, y = 6

Representation, reasoning, and logic

- The object of knowledge representation is to express knowledge in a computer-tractable form, so that agents can perform well.
- A knowledge representation language is defined by:
  - its **syntax,** which defines all possible sequences of symbols that constitute sentences of the language.
    - Examples: Sentences in a book, bit patterns in computer memory.
  - its **semantics**, which determines the facts in the world to which the sentences refer.
    - Each sentence makes a claim about the world.
    - An agent is said to believe a sentence about the world.