Logical Agents

In which we design agents that can form representations of the world, use a process of inference to derive new representations about the world, and use these new representations to deduce what to do.

Knowledge-Based Logical Agents

- Two central AI concepts
  - Representation of knowledge
  - Reasoning processes acting on knowledge
- Play crucial role in “Partially Observable” environments
  - Combine general knowledge with current percepts to infer hidden aspects before acting
- Aids in agent flexibility
  - Learn new knowledge for new tasks
  - Adapt to changes in environment by updating relevant knowledge

Logic

- For <u>logical</u> agents, knowledge is <u>definite</u>
  - Each proposition is either “True” or “False”
- Logic has advantage of being simple representation for knowledge-based agents
  - But limited in its ability to handle uncertainty
- We will examine propositional logic and first-order logic

Knowledge Base

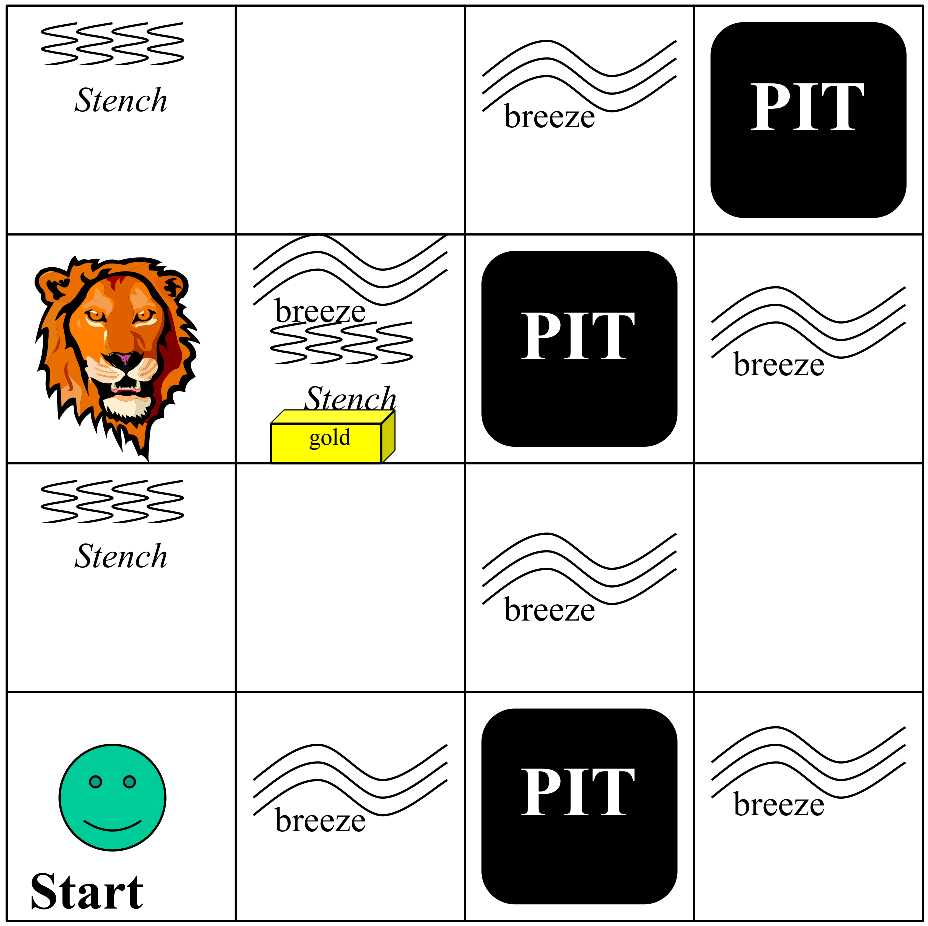
- Central component is its knowledge base (KB)
  - Contains set of "sentences" or factual statements
    - Some assertions about the world expressed with a knowledge representation language
  - KB initially contains some background knowledge
    - Innate knowledge
- How to add new information to KB?
  - TELL function
  - Inference: deriving new sentences from old ones
- How to query what is known?
  - ASK function
  - Answers should follow what has been told to the KB previously

A Simple Knowledge-Based Agent

- Agent needs to know
  - Current state of world
  - How to infer unseen properties of world from percepts
  - How world evolves over time
  - What it wants to achieve
  - What its own actions do in various circumstances

Wumpus World

Lion = wumpus →



“Wumpus World” Environment

- Simple environment to motivate logical reasoning
- Agent explores cave with rooms connected by passageways
- “Wumpus” beast lurking somewhere in cave
  - Eats anyone who enters its room
  - Agent has one arrow (can kill Wumpus)
- Some rooms contain bottomless pits
- Occasional heap of gold present
- Agent task
  - Enter cave, find the gold, return to entrance, and exit

Wumpus World PEAS Description

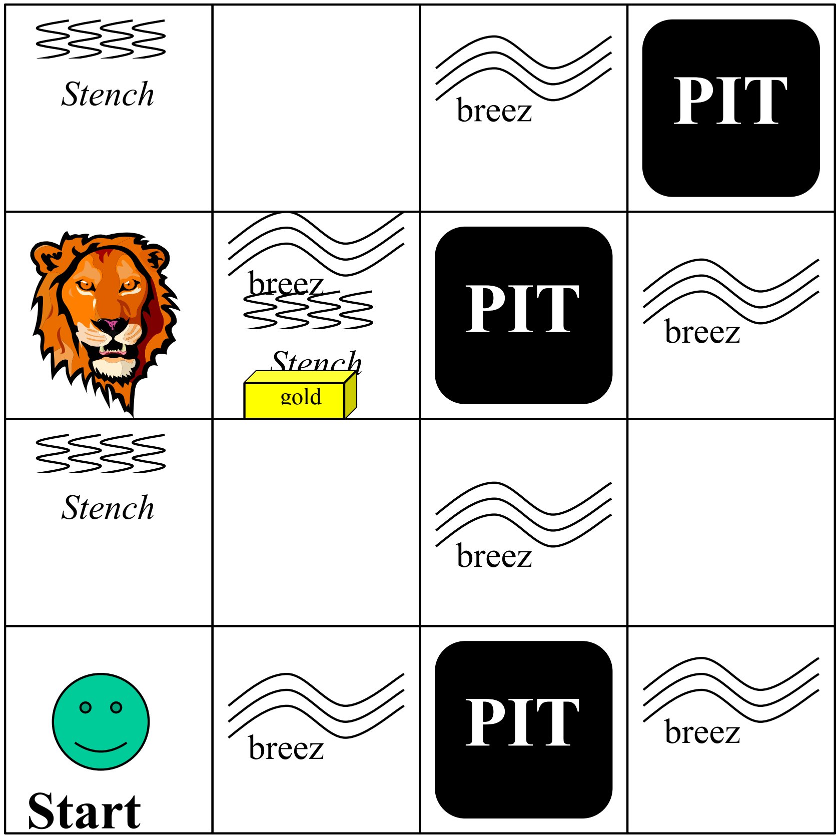
- (P)erformance measure
  - Receive +1000 for picking up gold
  - Cost of –1000 for falling into pit or being eaten by Wumpus (GAME OVER!)
  - Cost of –1 for each action taken
  - Cost of –10 for using up the only arrow
- (E)nvironment
  - 4x4 grid of rooms
  - Agent starts in square [1,1]
  - Wumpus and gold locations chosen randomly
  - Probability of square being a pit is .2
    - [0=*no*, …, 0.5=*maybe*, …, 1=*yes*]

Wumpus World PEAS Description

- (A)ctuators
  - Move forward, turn left, turn right
    - Note: die if enter pit or live wumpus square
  - Grab (gold)
  - Shoot (arrow)
    - Kills wumpus if facing its square
- (S)ensors
  - Nose: squares adjacent to wumpus are “smelly”
  - Skin/hair: Squares adjacent to pit are “breezy”
  - Eye: “Glittery” if and only if gold is in the same square
  - Percepts: [Stench, Breeze, Glitter]

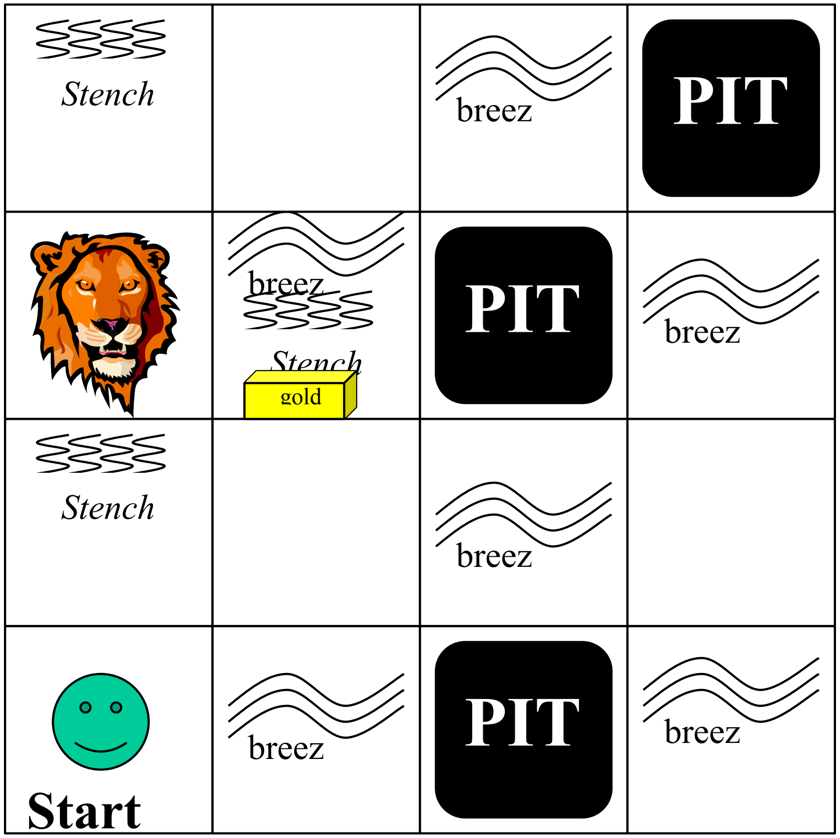
Wumpus World Characterization

- Is the world deterministic?
- Is the world fully observable?
- Is the world static?
- Is the world discrete?



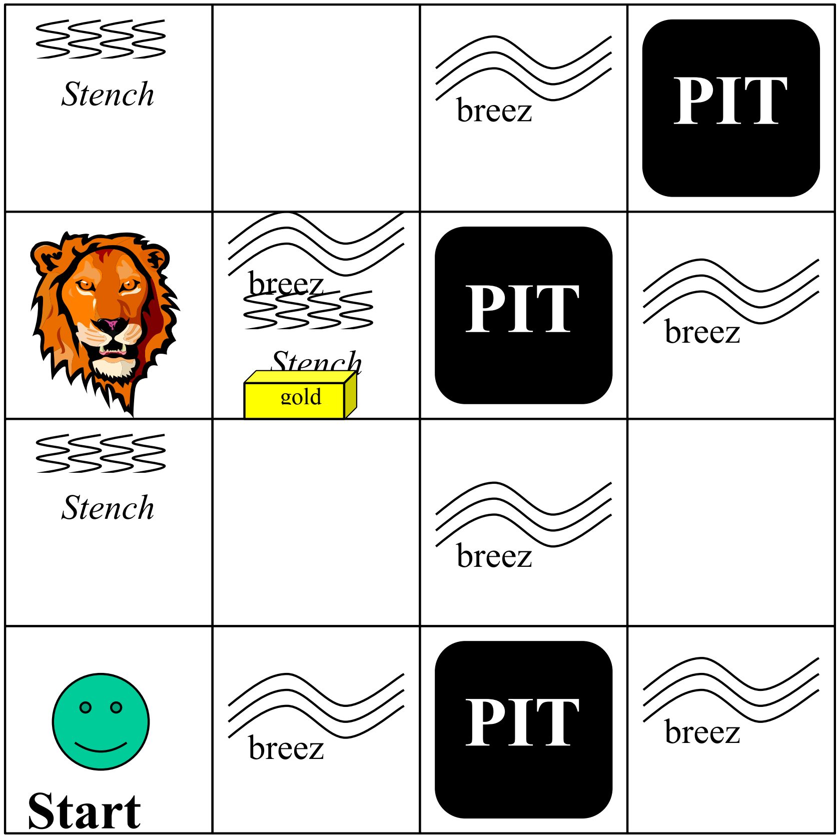
Wumpus World Characterization

- Is the world deterministic?
  - Yes, outcomes exactly specified
- Is the world fully observable?
- Is the world static?
- Is the world discrete?



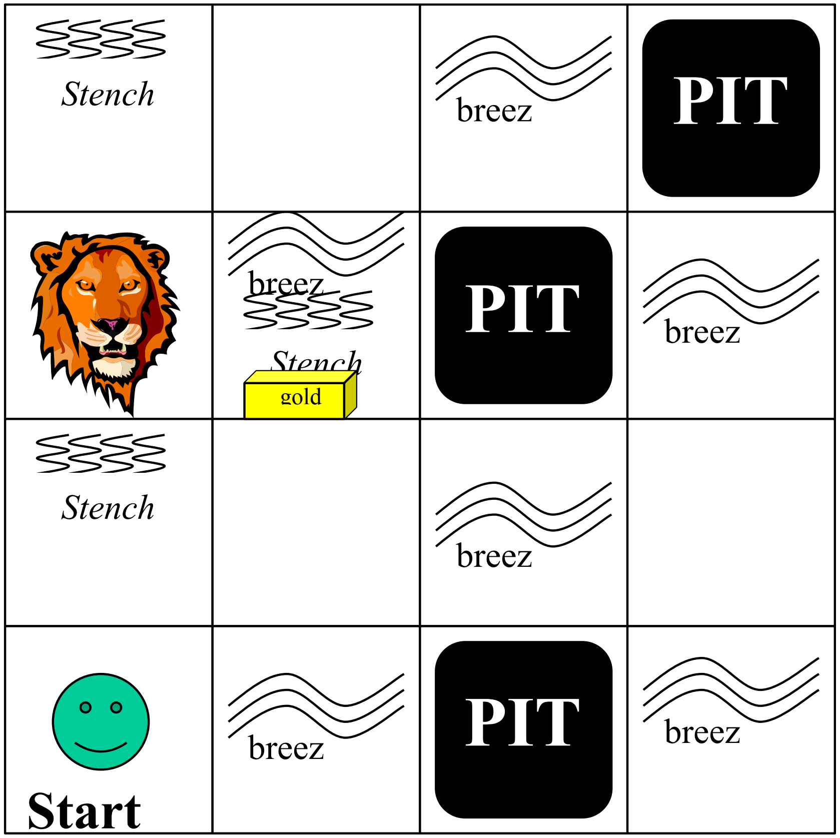
Wumpus World Characterization

- Is the world deterministic?
  - Yes, outcomes exactly specified
- Is the world fully observable?
  - No, only <u>local</u> percepts
- Is the world static?
- Is the world discrete?



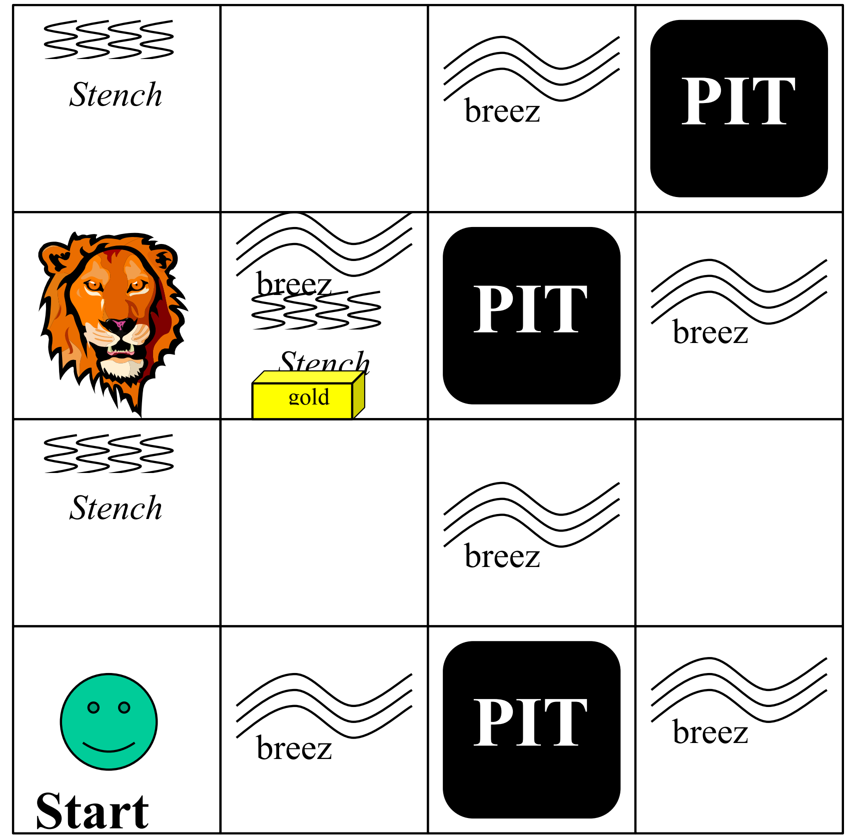
Wumpus World Characterization

- Is the world deterministic?
  - Yes, outcomes exactly specified
- Is the world fully observable?
  - No, only <u>local</u> percepts
- Is the world static?
  - Yes, Wumpus and pits do not move (though would be interesting!)
- Is the world discrete?



Wumpus World Characterization

- Is the world deterministic?
  - Yes, outcomes exactly specified
- Is the world fully observable?
  - No, only <u>local</u> percepts
- Is the world static?
  - Yes, Wumpus and pits do not move (though would be interesting!)
- Is the world discrete?
  - Yes, blocks/cells



Exploring a Wumpus World

A = agent

B = breeze

G = glitter, gold

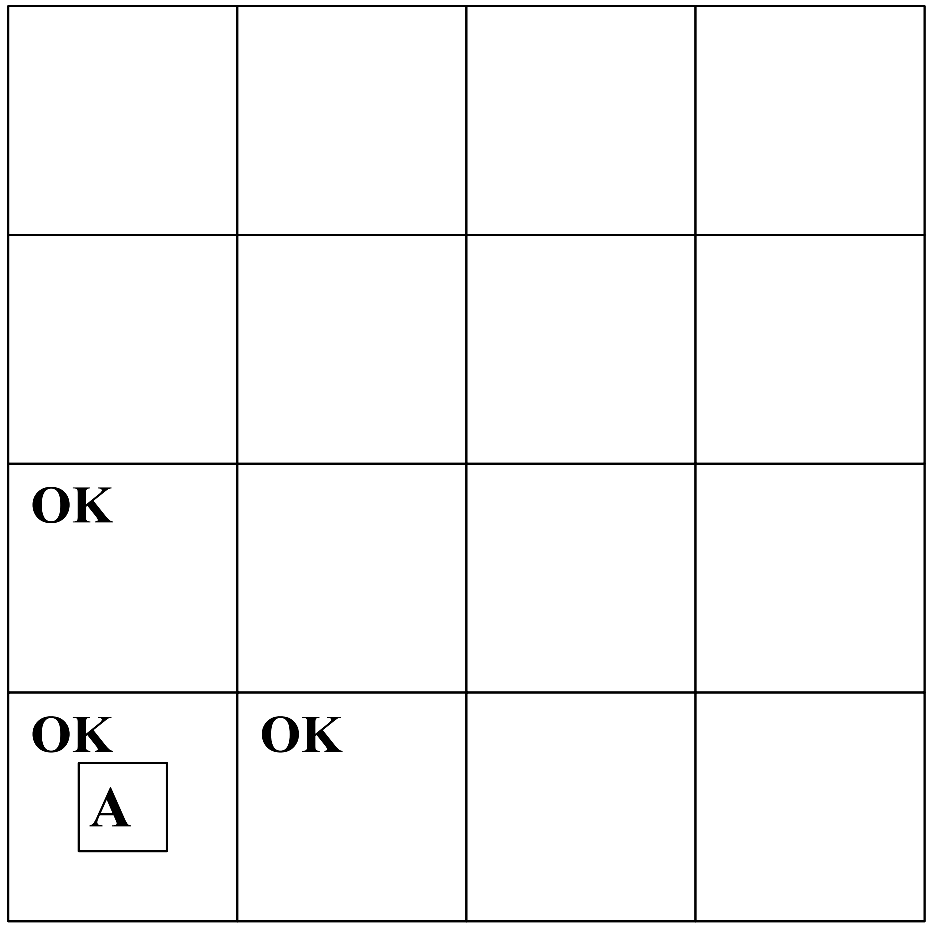
OK = safe square

P = pit

S = stench

V = visited

W = Wumpus



From local percepts, determines that {(1,1), (1,2), (2,1)} are free from danger.

Exploring a Wumpus World

A = agent

B = breeze

G = glitter, gold

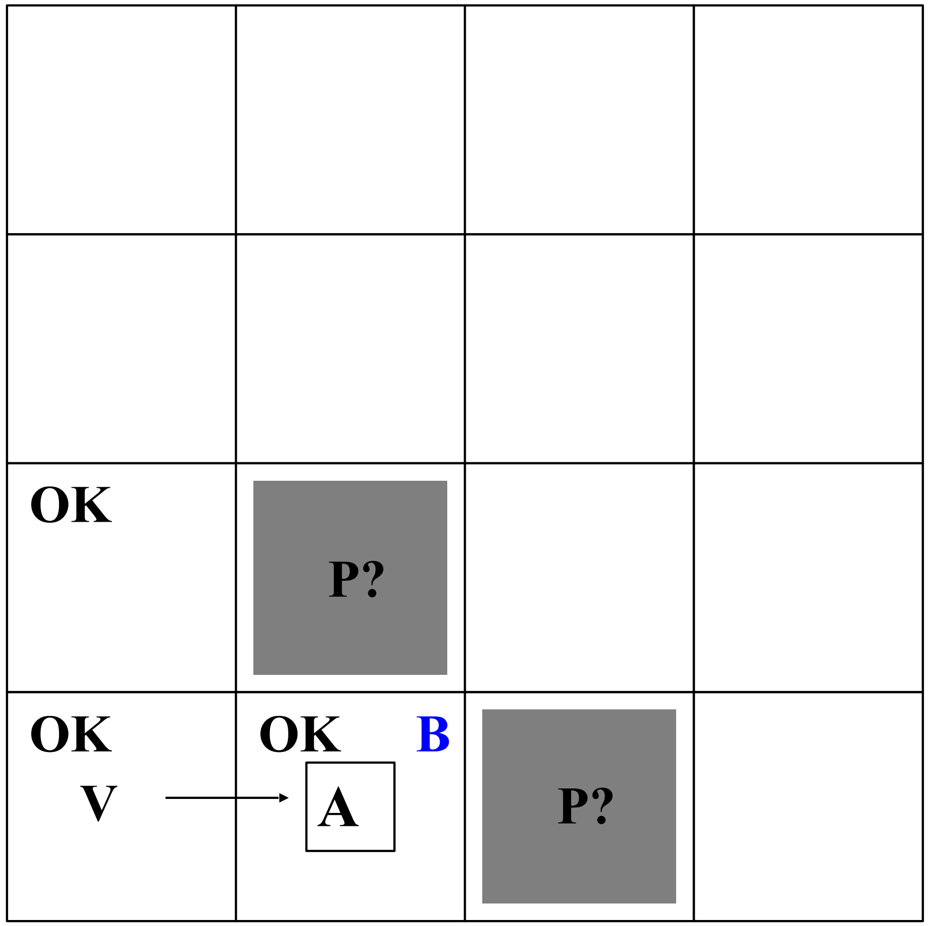
OK = safe square

P = pit

S = stench

V = visited

W = Wumpus



From breeze percept, determines that (2,2) or (3,1) is a pit. Go back to (1,1) and move up to (1,2).

Exploring a Wumpus World

A = agent

B = breeze

G = glitter, gold

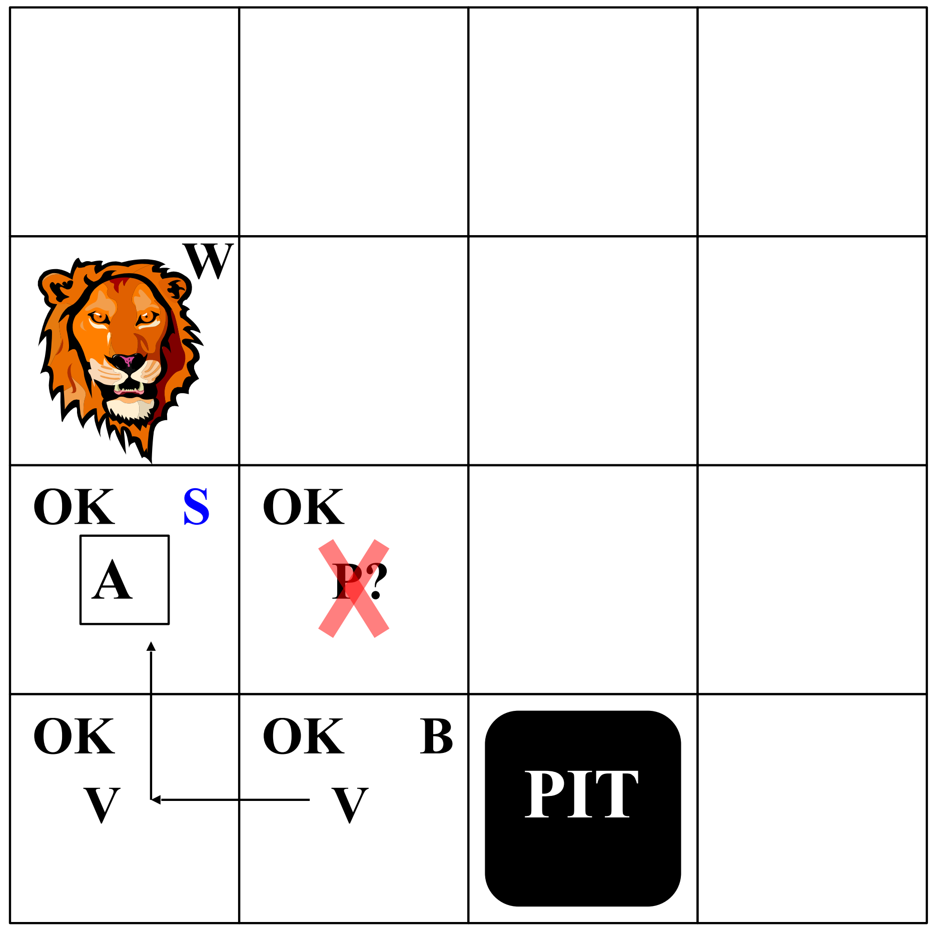
OK = safe square

P = pit

S = stench

V = visited

W = Wumpus



From stench and no-breeze percept in (1,2), determines that Wumpus in (1,3), pit in (3,1), and (2,2) clear.

Exploring a Wumpus World

A = agent

B = breeze

G = glitter, gold

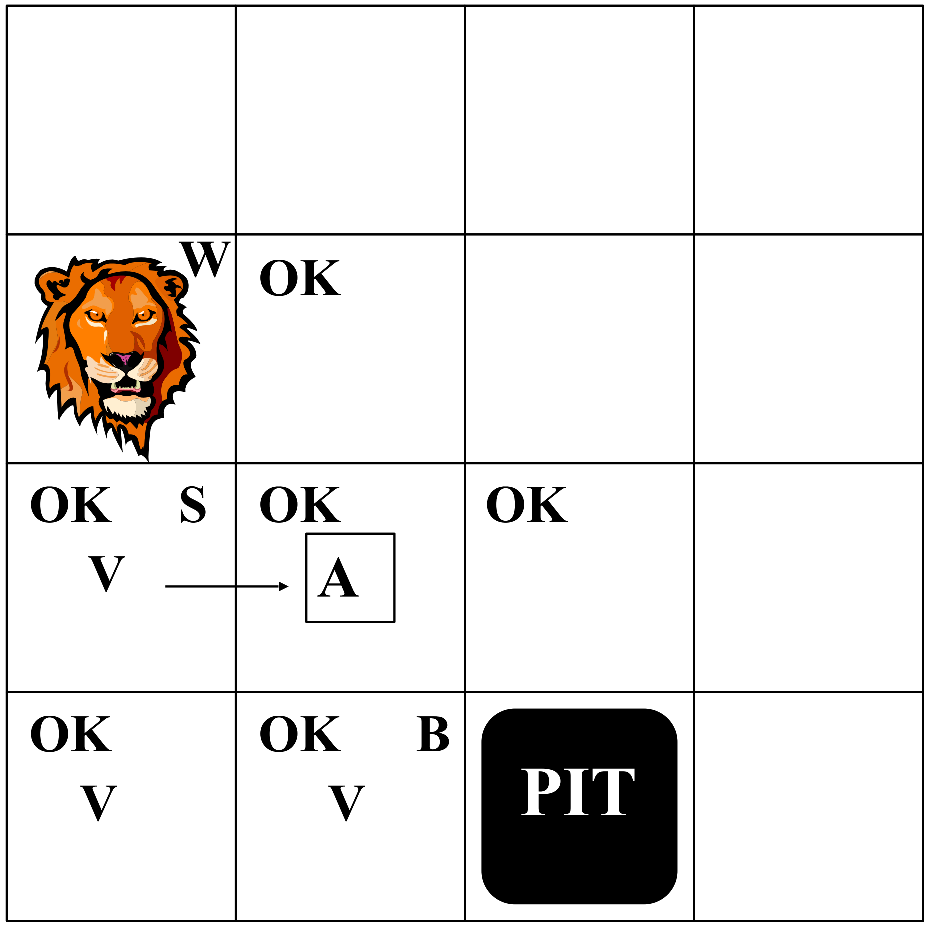
OK = safe square

P = pit

S = stench

V = visited

W = Wumpus



From local percepts, it is OK to move up or right.

Exploring a Wumpus World

A = agent

B = breeze

G = glitter, gold

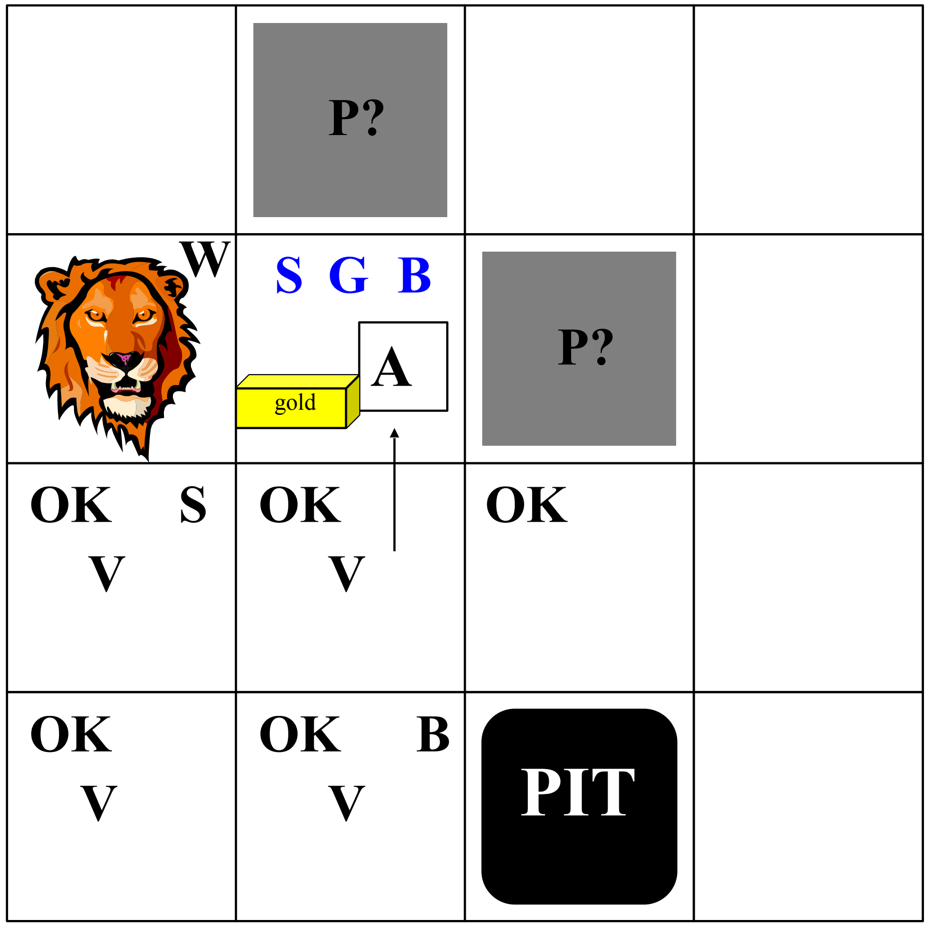
OK = safe square

P = pit

S = stench

V = visited

W = Wumpus



Found gold! No need to explore further. Time to head back.

Exploring a Wumpus World

A = agent

B = breeze

G = glitter, gold

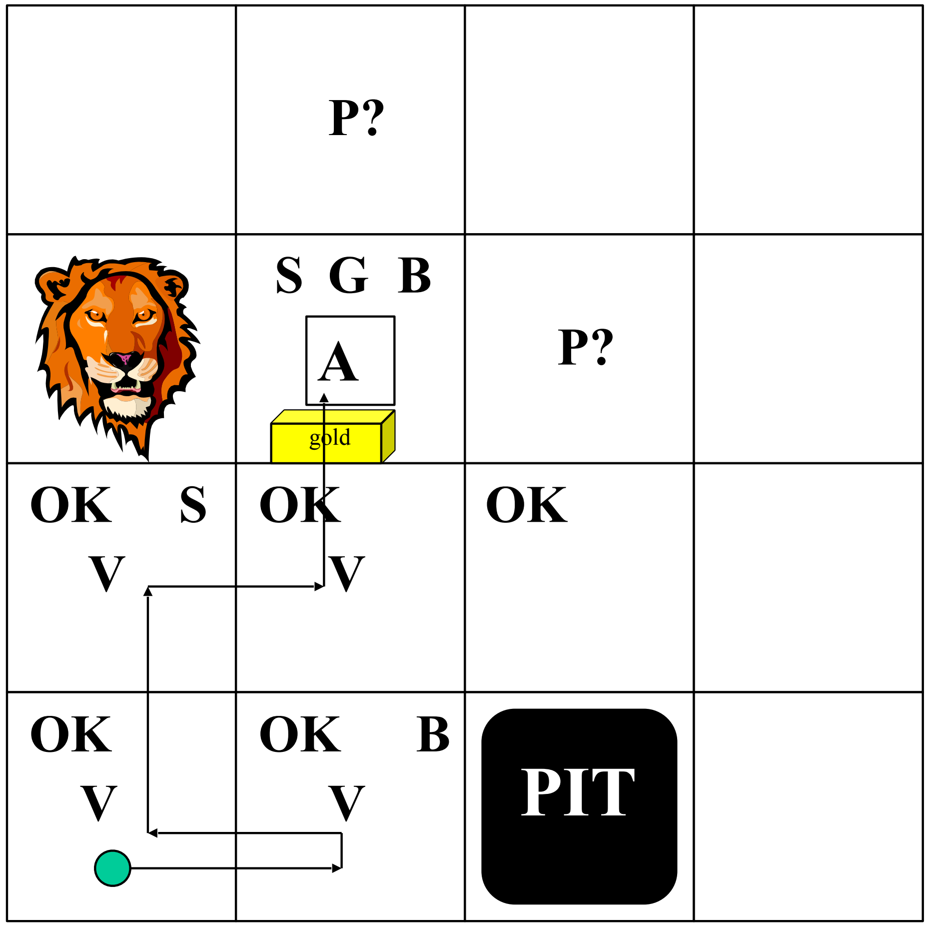
OK = safe square

P = pit

S = stench

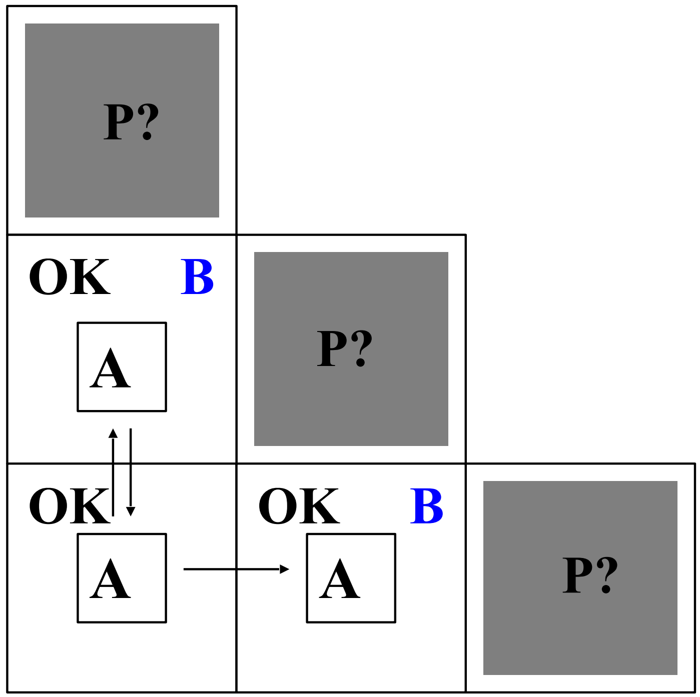
V = visited

W = Wumpus



Then go home using **OK** squares (retrace route).

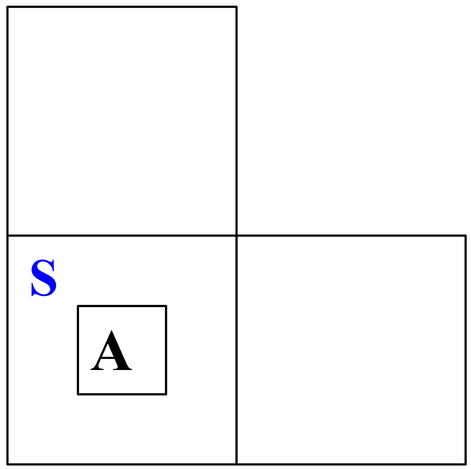
Tight Spots



<u>Breeze</u> in (1,2) and (2,1) → no safe actions!

Pit may actually only be in (2,2), but can't tell.

More Tight Spots



<u>Smell</u> in (1,1) → Cannot move!

Possible action: shoot arrow straight ahead

Logical Agent

- Need agent to represent beliefs
  - “There is a pit in (2, 2) or (3, 1)”
  - “There is no Wumpus in (2, 2)”
- Need to make inferences
  - If available information is correct, draw a conclusion that is guaranteed to be correct
- Need <u>representation</u> and <u>reasoning</u>
  - Support the operation of knowledge-based agent

Knowledge Representation

- For expressing knowledge in computer-tractable form
- Knowledge representation language defined by
  - **Syntax**
    - Defines the possible well-formed configurations of sentences in the language
  - **Semantics**
    - Defines the "meaning" of sentences (need interpreter)
    - Defines the truth of a sentence in a world (or model)

The Language of Arithmetic

Syntax: "$x + 2 \geq y$" is a sentence

"$x2 + y >$" is not a sentence

Semantics: $x + 2 \geq y$ is true iff the number $x + 2$ is no less than the number y

$x + 2 \geq y$ is True in a world where x=7, y=1

$x + 2 \geq y$ is False in a world where x=0, y=6

Inference

- Sentence is <u>valid</u> iff it is true under all possible interpretations in all possible worlds
  - Also called <u>tautologies</u>
  - "There is a stench at (1,1) or there is not a stench at (1,1)"
  - "There is an open area in front of me" is <u>not valid</u> in all worlds
- Sentence is <u>satisfiable</u> iff there is some interpretation in some world for which it is true
  - "There is a wumpus at (1,2)" could be true in some situation
  - "There is a wall in front of me and there is no wall in front of me" is <u>unsatisfiable</u>

Propositional Logic: Syntax

- *True*, *False*, S_1, S_2, … are sentences
- If S is a sentence, $\neg S$ is a sentence
  - Not (negation)
- $S_1 \land S_2$ is a sentence, also $(S_1 \land S_2)$
  - And (conjunction)
- $S_1 \lor S_2$ is a sentence
  - Or (disjunction)
- $S_1 \Rightarrow S_2$ is a sentence
  - Implies (conditional)
- $S_1 \Leftrightarrow S_2$ is a sentence
  - Equivalence (biconditional)

Propositional Logic: Semantics

- Semantics defines the rules for determining the truth of a sentence
  - (wrt a particular model)
- $\neg S$, is true iff S is false
- $S_1 \wedge S_2$, is true iff S_1 is true <u>and</u> S_2 is true
- $S_1 \vee S_2$, is true iff S_1 is true <u>or</u> S_2 is true
- $S_1 \Rightarrow S_2$, is true iff S_1 is false <u>or</u> S_2 is true
- $S_1 \Leftrightarrow S_2$, is true iff $S_1 \Rightarrow S_2$ is true <u>and</u> $S_2 \Rightarrow S_1$ is true
  - (S_1 same as S_2)

Semantics in Truth Table Form

P	Q	$\neg P$	$P \wedge Q$	$P \vee Q$	$P \Rightarrow Q$	$P \Leftrightarrow Q$
False	False	True	False	False	True	True
False	True	True	False	True	True	False
True	False	False	False	True	False	False
True	True	False	True	True	True	True

Propositional Inference: Enumeration Method

- Truth tables can test for <u>valid</u> sentences
  - True under all possible interpretations in all possible worlds
- For a given sentence, make a truth table
  - Columns as the combinations of propositions in the sentence
  - Rows with all possible truth values for proposition symbols
- If sentence true in every row, then valid

Propositional Inference: Enumeration Method

- Test $((P \vee H) \wedge \neg H) \Rightarrow P$

P	H	$P \vee H$	$\neg H$	$(P \vee H) \wedge \neg H$	$((P \vee H) \wedge \neg H) \Rightarrow P$
False	False	False	True	False	True
False	True	True	False	False	True
True	False	True	True	True	True
True	True	True	False	False	True

Propositional Inference: Enumeration Method

- Test $((P \vee H) \wedge \neg H) \Rightarrow P$

P	H	$P \vee H$	$\neg H$	$(P \vee H) \wedge \neg H$	$((P \vee H) \wedge \neg H) \Rightarrow P$
False	False	False	True	False	True
False	True	True	False	False	True
True	False	True	True	True	True
True	True	True	False	False	True

Practice

- Test $(P \wedge H) \Rightarrow (P \vee \neg H)$

Practice

- Test $(P \wedge H) \Rightarrow (P \vee \neg H)$

P	H	$P \wedge H$	$\neg H$	$(P \vee \neg H)$	$(P \wedge H) \Rightarrow (P \vee \neg H)$
False	False	False	True	Ture	True
False	True	False	False	False	True
True	False	False	True	True	True
True	True	True	False	True	True