CIS 421/521:
ARTIFICIAL INTELLIGENCE

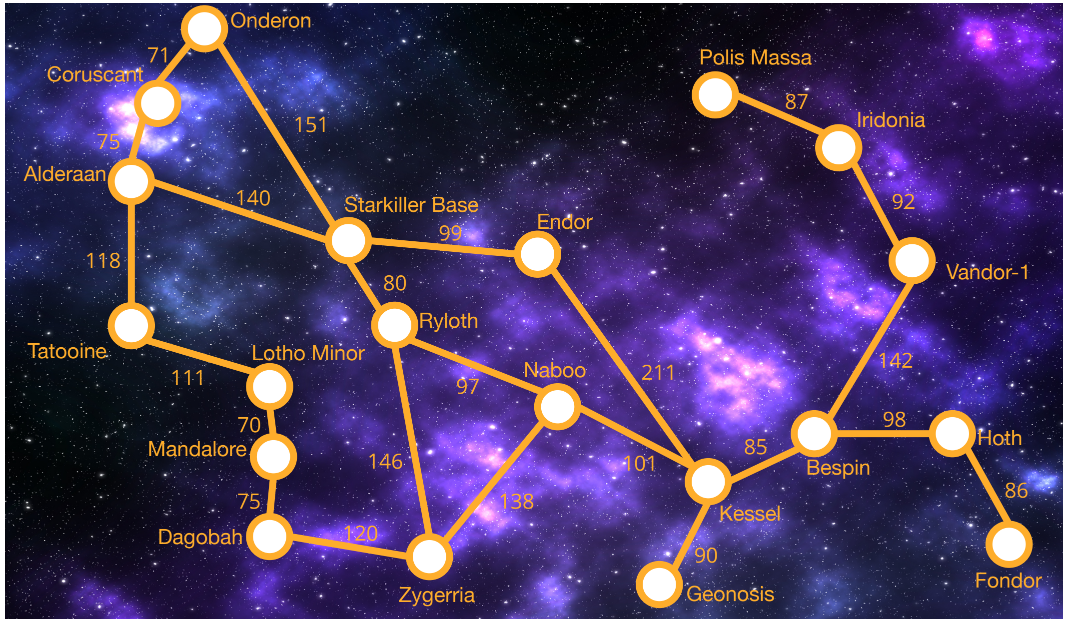
Logical Agents

Penn Engineering
UNIVERSITY of PENNSYLVANIA

Knowledge-based Agents

Knowledge-based agents use a process of **reasoning** over an internal **representation** of knowledge to decide what action to take.

So far, our **problem-solving** agents have performed a **search** over **states** in order to find a plan. The representation of states has been **atomic**.



Limited to commands like "Navigate to Kessel"

"Take me to the nearest habitable planet where I can store my perishable cargo"

Knowledge-based Agents

A central component of a knowledge-based agent is a **knowledge base** or KB.

A KB contains a set of **sentence** that are written in a **knowledge representation language.** The sentence contains some assertion about the world.

Natural language sentences	Knowledge representation language sentence
Hoth is a planet	`planet(hoth)`
Hoth is habitable	`habitable(hoth)`
Hoth is far from its sun	`far_from(hoth, sol)`
If a planet is far from its sun then it is cold	`planet(x) and sun(y) and far_from(x,y) →cold(x)`

Knowledge-based Agents

There are two kinds of sentences:

- **Axioms** – a sentence that is given
- **Derived sentences** – a new sentence that is derived from others sentences

The process of deriving new sentences from old sentences is called **inference**.

A KB can initially contain some **background information** about the world, and a knowledge-based agent can add to the information in the KB through its observations of the world.

In addition to asserting new knowledge into its KB, a knowledge-based agent can also query the KB and ask it to derive new knowledge in order to select what action it should take.

Hunt the Wampas

Wampa World

Our **knowledge-based agent**, R2D2, explores **a cave** consisting of **rooms** connected by passageways.

Lurking somewhere in the cave is the **Wampa**, a beast that eats any agent that enters its room.

Some rooms contain bottomless **pits** that trap any agent that wanders into the room.

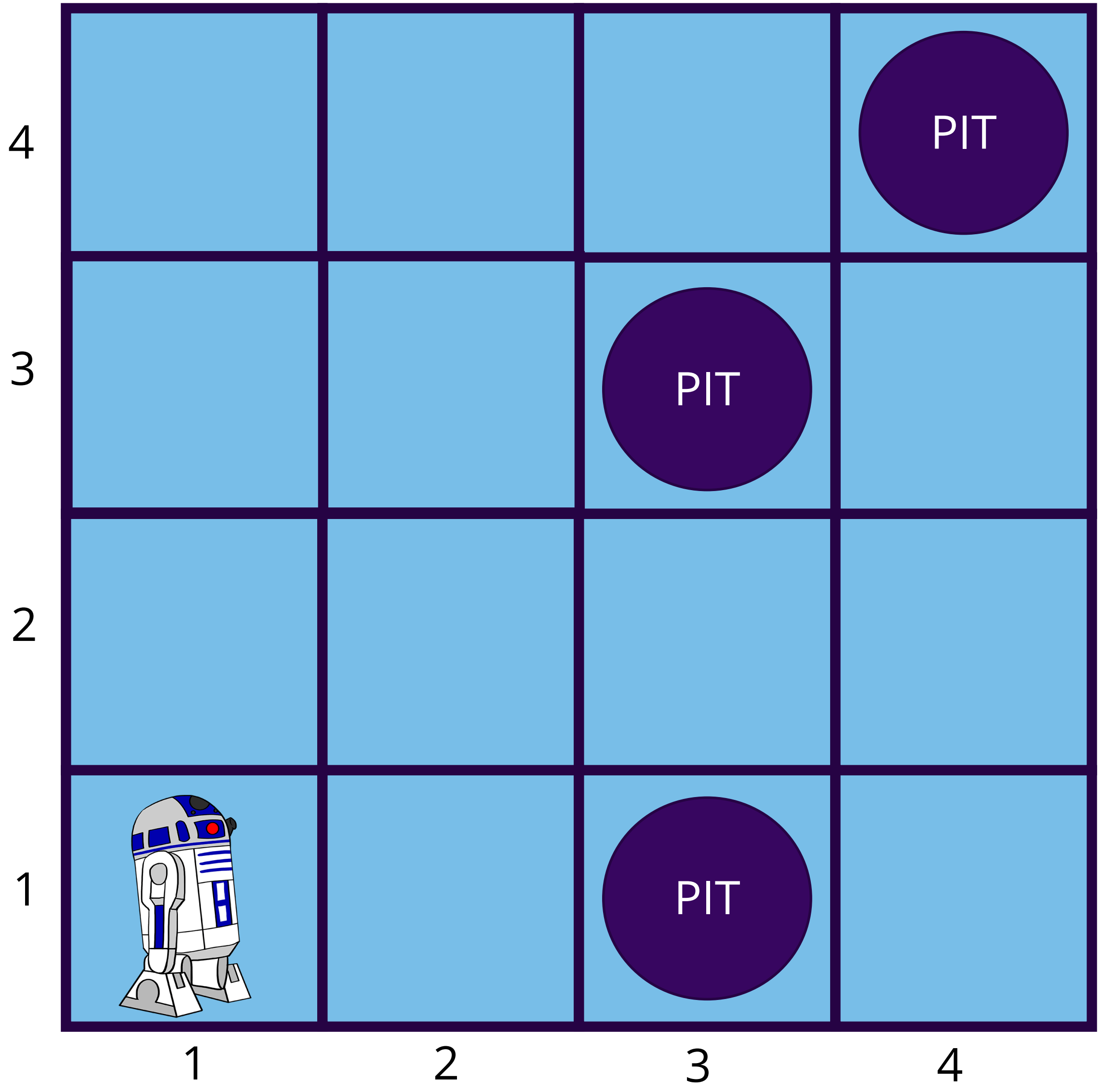
In one room is master **Luke**.

The goal is:

- collect Luke
- exit the world
- without being eaten

Wampa World

Environment: A 4x4 grid of rooms. The agent starts in the square [1,1]. Wampa and Luke are randomly placed in other squares. Each square can be pit with 20% probability.

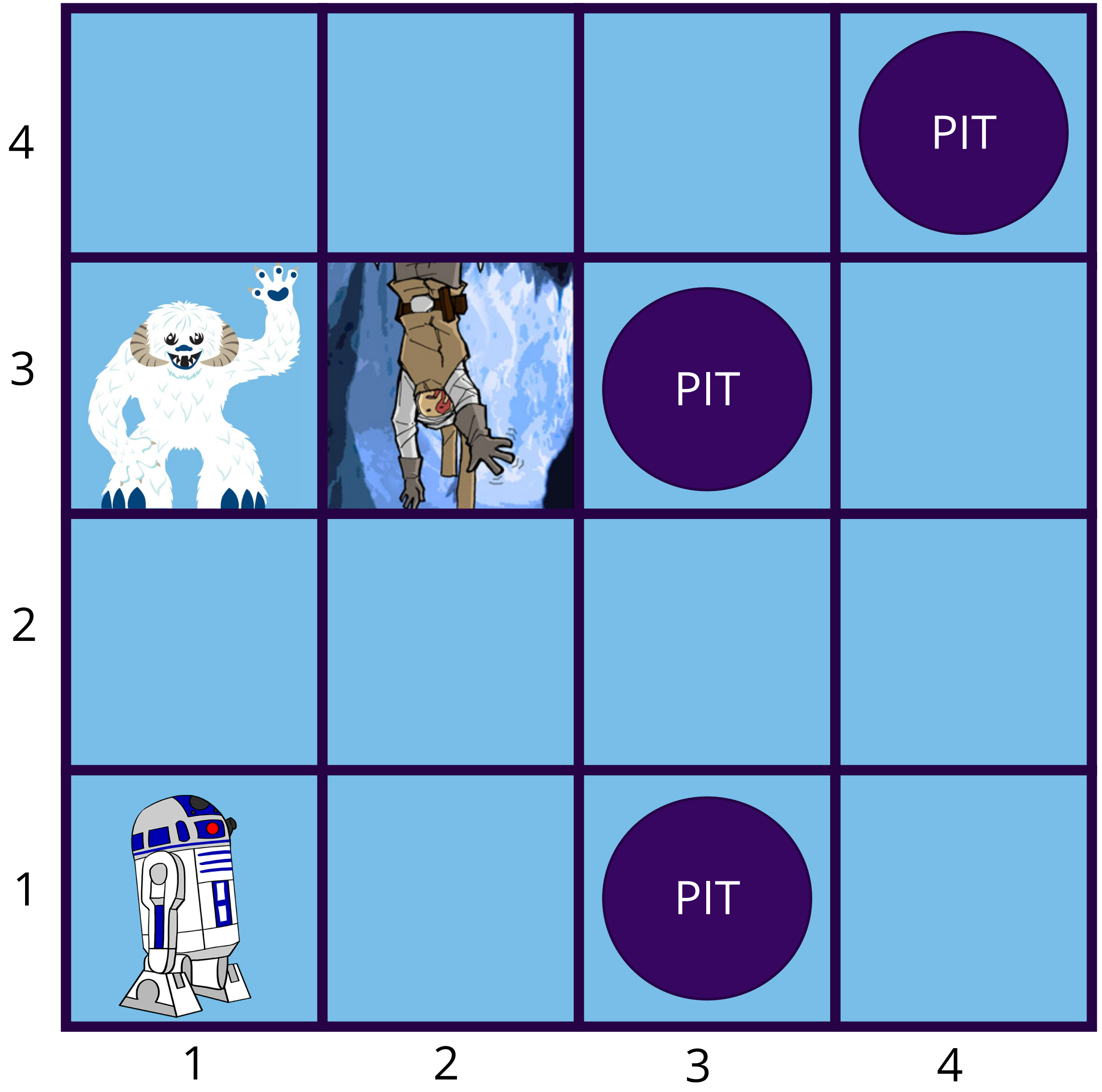


Wampa World

Performance measure:

+1000 points for rescuing Luke and leaving the cave

-1000 for falling into a pit or being eaten by the Wampa
-1 for each action taken
-10 for using up your blaster fire



Wampa World

Actuators:

R2 can move *Forward, TurnLeft, Turn right.*

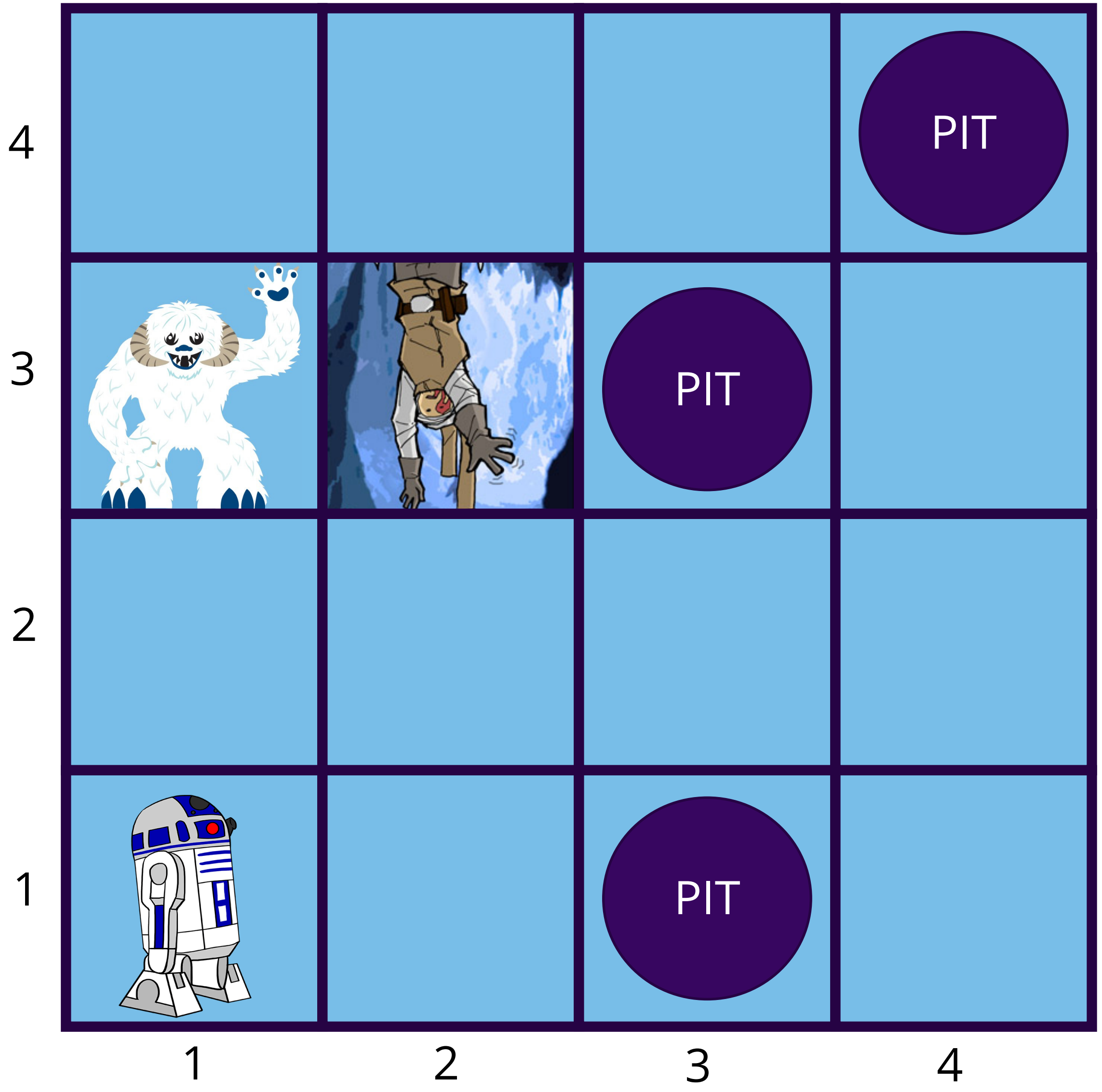
Agent dies if it moves into a pit or a Wumpus square.

Grab can pick up Luke.

Shoot fires blaster bolt in a straight line in the direction that R2D2 is facing.

If the blaster hits the Wampa, it dies. R2 only has enough power for one shot.

Climb gets R2 out of the cave but only works in [1, 1]



Wampa World

Sensors:

In each square adjacent to the Wampa, R2D2's olfactory sensor perceives a *Stench*

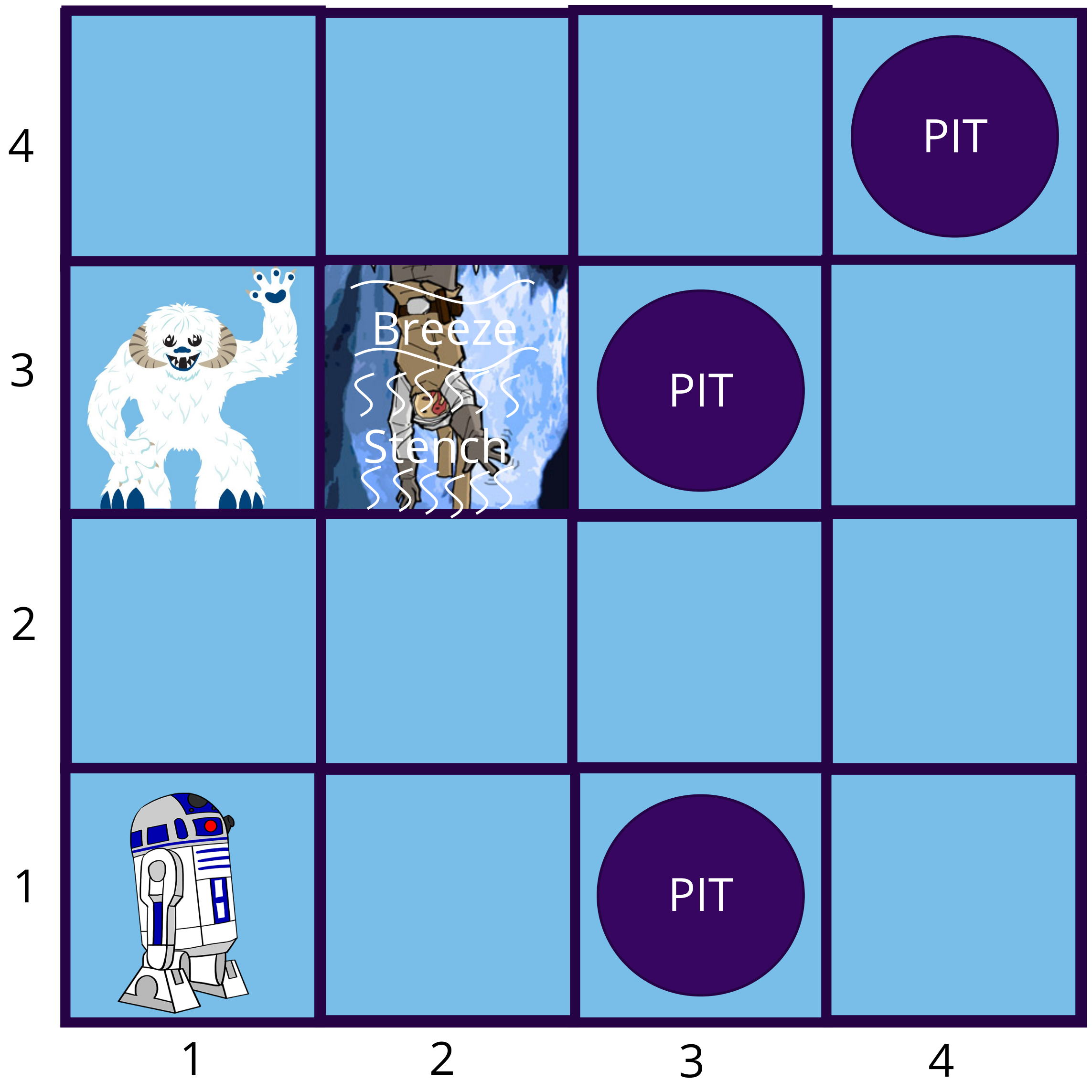
In each square adjacent to a pit, R2D2's wind sensor perceives a *Breeze*

In the square with Luke, R2D2's audio sensor perceives a *Gasp*

When R2D2 walks into a wall it perceives a *Bump*

When the Wampa is killed , R2D2's audio sensor perceives a *Scream*

Percept=[*Stench, Breeze, None, None None*]



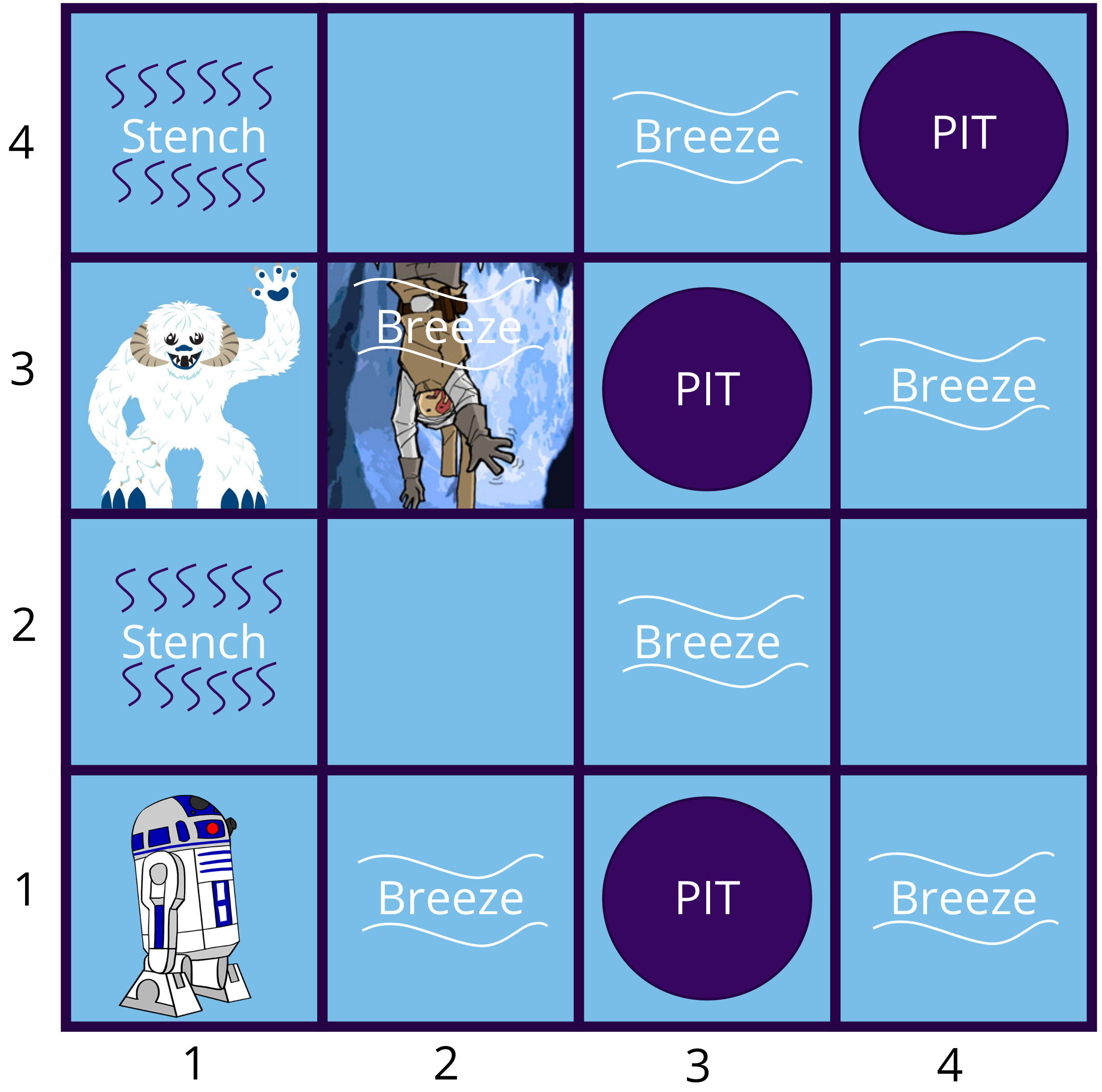
Wampa World

Deterministic, discrete, static, single-agent (Wampa doesn't move)

Sequential because reward doesn't come for many steps

Partially observable because some parts of the state are not directly perceivable:

- Location of Luke, Wampa, and pits aren't directly observable.

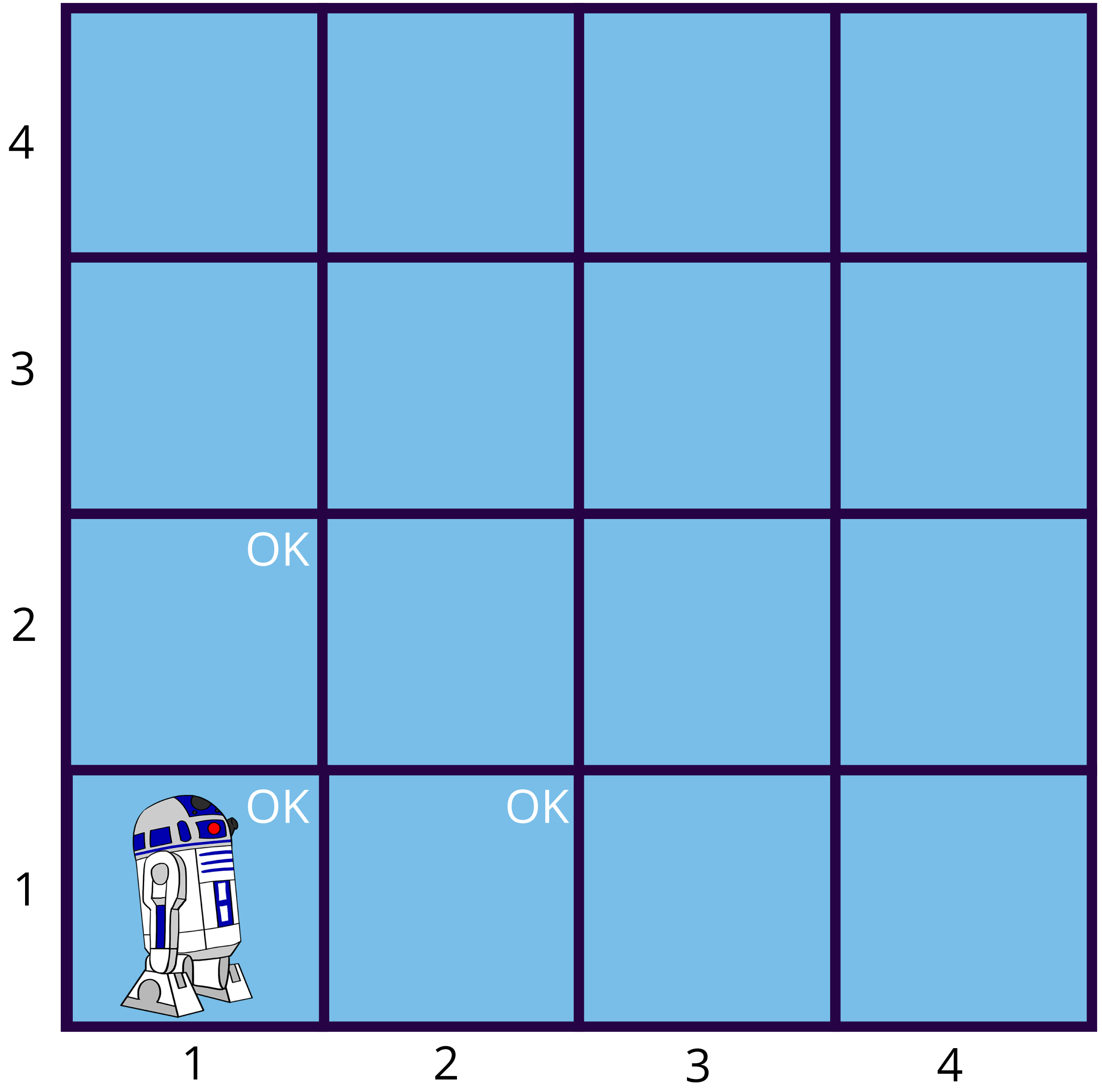


Wampa World Walkthrough

R2D2 starts in [1,1]

Percept=[*None, None, None, None None*]

What can we conclude about [1,2] and [2,1]?

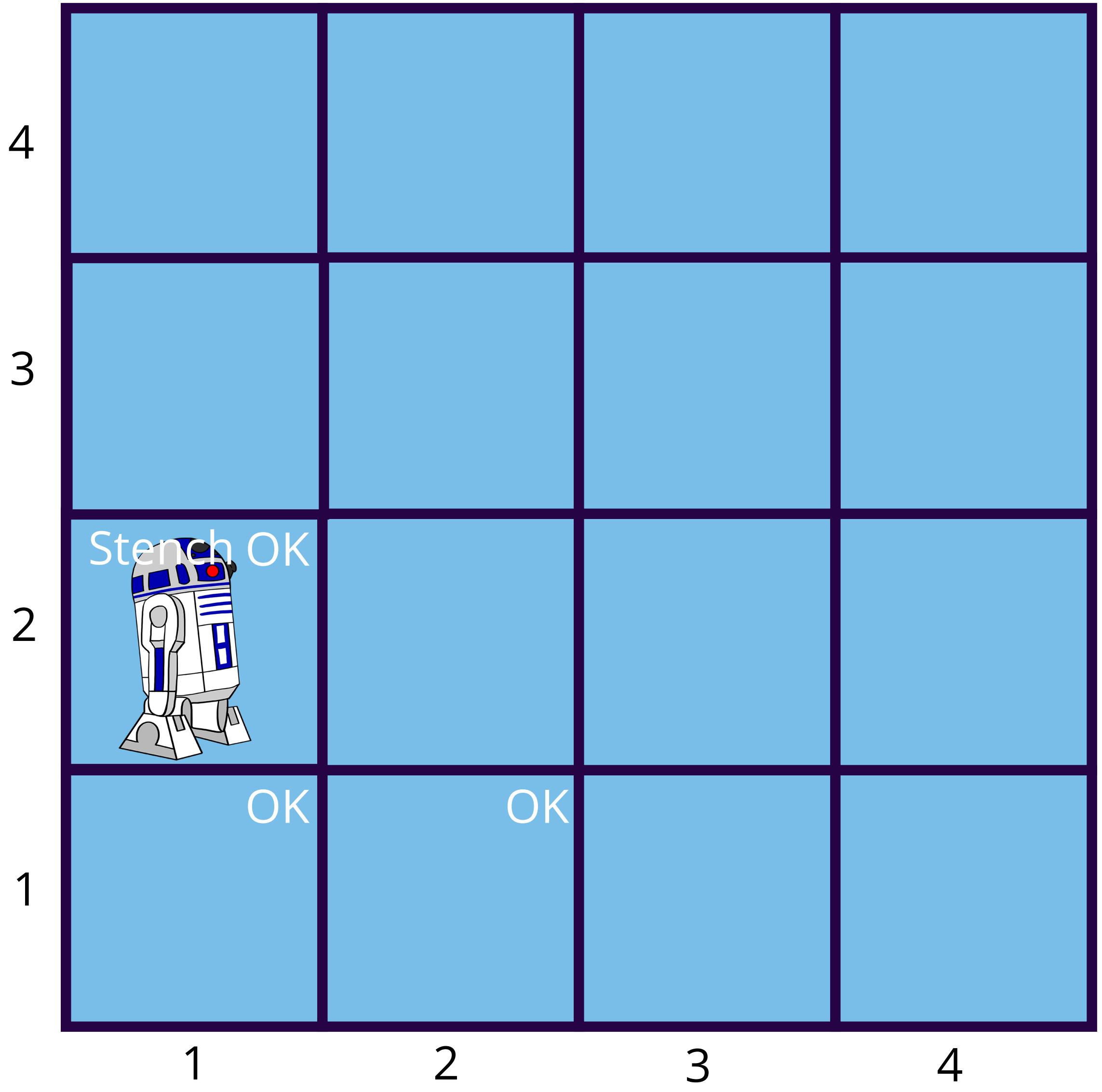


Wampa World Walkthrough

R2D2 moves to [1,2]

Percept=[*Stench, None, None, None None*]

What can we conclude about [3,1]?



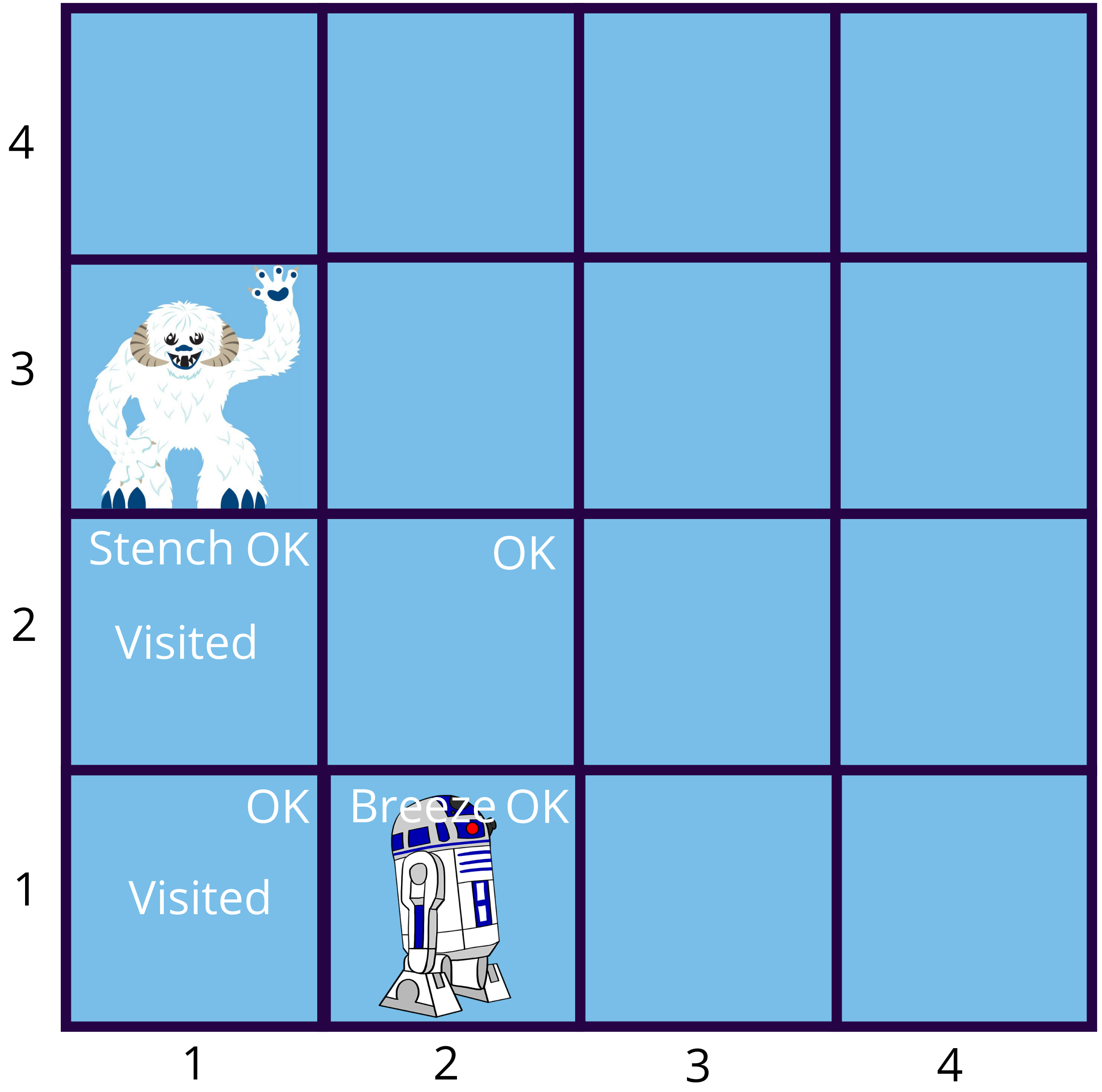
Wampa World Walkthrough

R2D2 moves to [2,1]

Percept=[*None, Breeze, None, None None*]

What can we conclude about [3,1]?

What can we conclude about [2,2]?

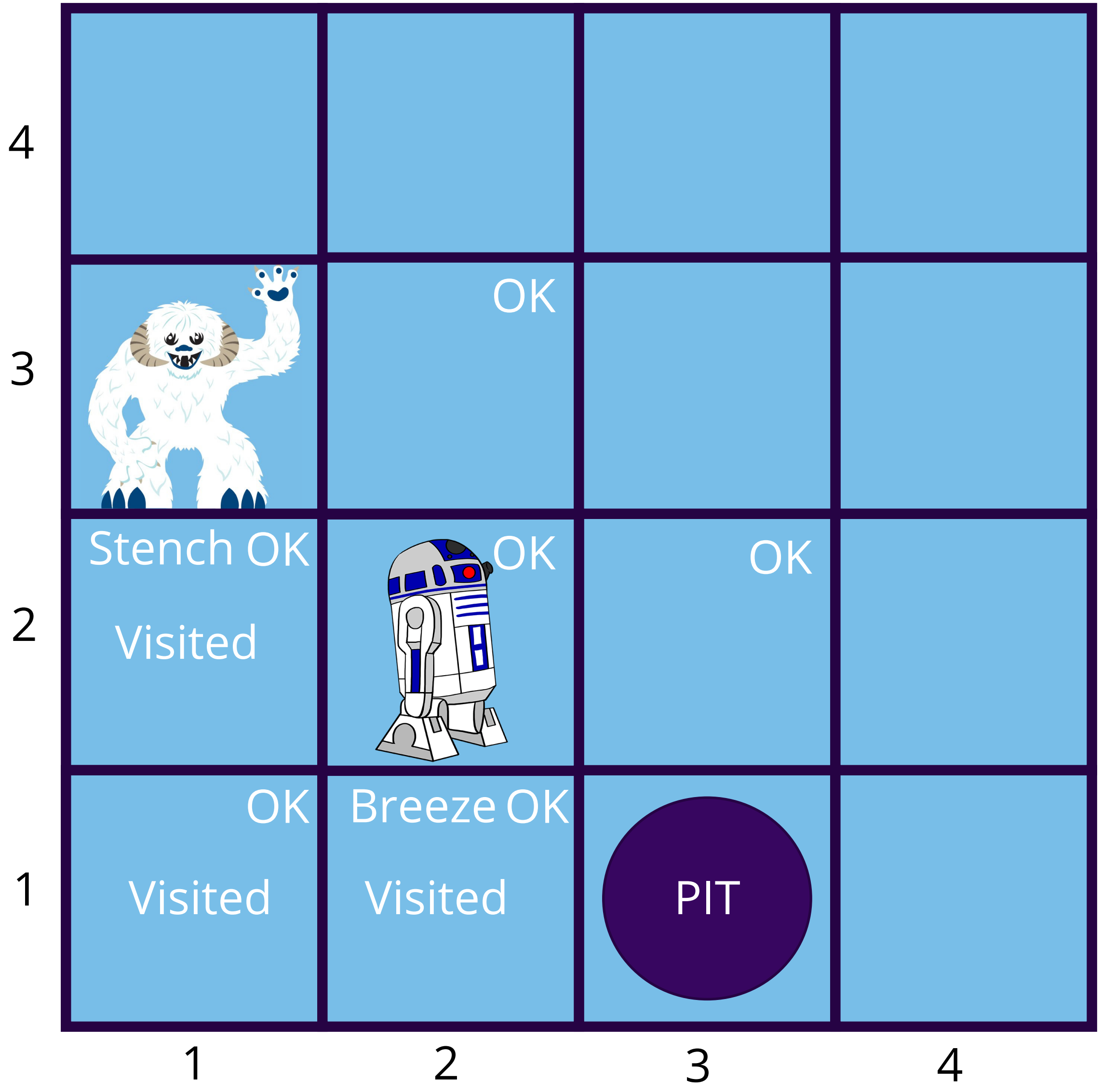


Wampa World Walkthrough

R2D2 moves to [2,2]

Percept=[*None, None, None, None None*]

What can we conclude about [3,2] and [2,3]?



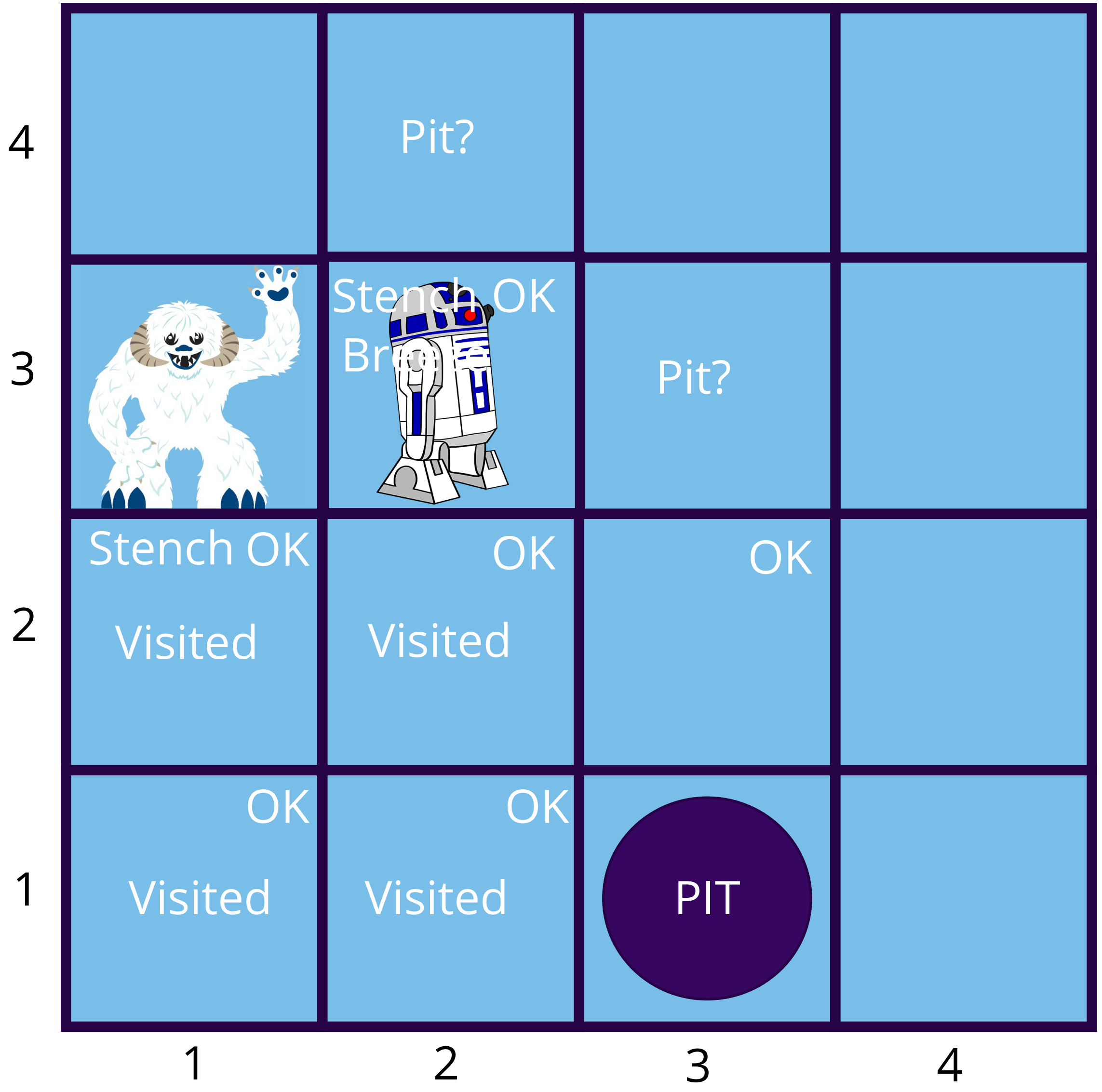
Wampa World Walkthrough

R2D2 moves to [2,2]

Percept=[*Stench, Breeze, Gasp, None None*]

Who is here?

What is in [2,4] and [3,3]?



Logic

Logic can serve as a general class of **representations** for knowledge-based agents. Here we are going to examine **Propositional Logic.**

- KB consists of **sentences** in the **representation language**
- Representation language has a **syntax** that specifies sentences that are well-formed
- A logic defines the **semantics** of the sentences, which is their **meaning**
- The semantics defines the **truth** of each sentence with respect to a **possible world,** which we will often call a **model**.

Possible worlds and models

- Models are mathematical abstractions that have a fixed set of **truth values** which are **{true, false}** for each sentence.
- If sentence α is true in model **m** then we say
  - **m** satisfies α, or
  - **m** is a model of α
- We use the notation **M(α)** to mean the **set of all models** of α.

For instance, α could be a sentence that means “there is no pit in [2,2]”. In that case, **M(α)** would be all instances of Wampa World where [2,2] doesn’t have a pit.

Logical Entailment

Once we have a notion of truth, we can start to define **logical reasoning**. Logical reasoning involves the **entailment** relation between sentence.

In plain English, entailment is the idea that a sentence **follows logically** from another sentence.

To write sentence α entails sentence **β in** mathematical notation we use the ⊨ **symbol:**

$$\alpha \models \beta$$

The definition is

$$\alpha \models \beta \text{ if and only if } M(\alpha) \subseteq M(\beta)$$

This means that α is more specific, or stronger than, **β.** For instances, **β** could mean that "The agent is a robot" and α could mean "The agent is an astromech".

Knowledge Base

The KB can be thought of set as a set of sentences.

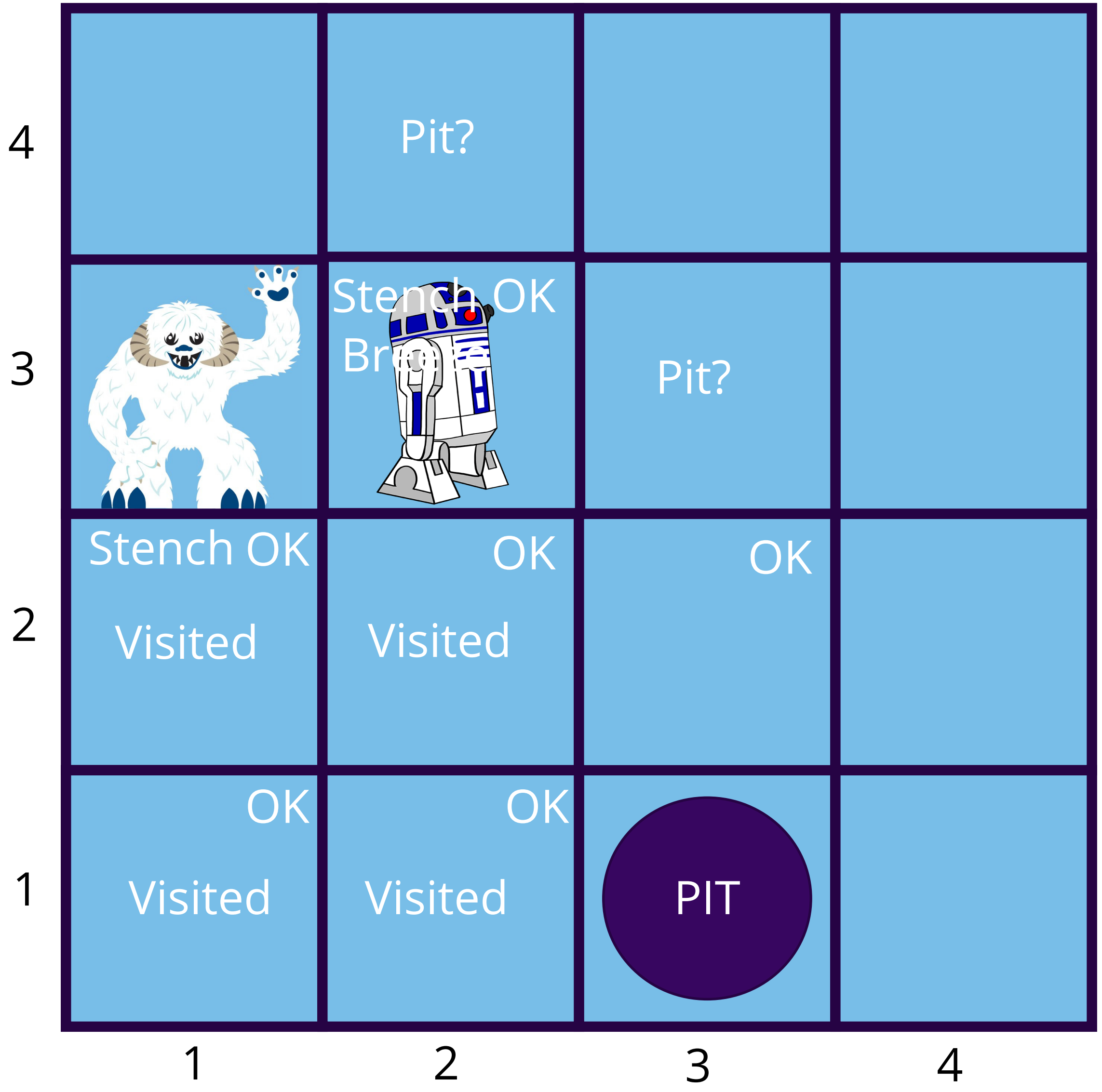
α_1 **=** “There is no pit in [1,2]”

α_2 **=** “There is a pit in [3,1]”

α_3 **=** “There is a wampa in [1,2]”

The KB is false in models that contradict what the agent knows. For example, the KB is false in any model **m** where [1,2] contains a pit.

Model checking is the process of enumerating all possible models that are compatible with the KB. $\mathbf{M}(\mathrm{KB}) \subseteq \mathbf{M}(\alpha_1)$



Logical inference

Entailment can be applied to derive conclusions, which is the process of **logical inference.**

We can think about the consequences of a KB as a large set of additional sentences that are entailed given the sentences that have been added to the KB.

We would like to design **inference algorithms** to enumerate these sentences.

When an inference algorithm **i** allows us to conclude that α is true, then we write

$$\mathbf{KB} \vdash_i \alpha$$

"α is derived from **KB** by i"

Propositional Logic

Atomic sentences are represented with a single **propositional symbols**.
Propositional symbols **stand for a statement** that can be true or false.

For example, $\mathbf{W_{1,3}}$ is a propositional symbol that we choose to stand for

"There is a Wampa at location [1,3]"

That statement can be true or false.

The symbol **FacingEast** could stand for "The agent is currently facing East".

Propositional Logic

Complex sentences are constructed from simpler ones using parentheses and **logical connectives**.

Logical Connective	Meaning
¬ (not)	$\neg \mathbf{W_{1,3}}$ is the negation of $\mathbf{W_{1,3}}$
∧ (and)	$\mathbf{W_{1,3}} \land \mathbf{P_{3,1}}$ is called a conjunction
∨ (or)	$\mathbf{W_{1,3}} \lor \mathbf{P_{3,1}}$ is called a disjunction
⟹ (implies)	$\mathbf{W_{1,3}} \Rightarrow \mathbf{B_{1,2}}$ is called an implication. $\mathbf{W_{1,3}}$ is its **premise or antecede** and $\mathbf{B_{1,2}}$ is its **conclusion or consequence**
⇔ (if and only if)	$\mathbf{W_{1,3}} \Leftrightarrow \mathbf{W_{3,4}}$ is called an biconditional

Truth Tables

Negation

P	**¬P**
T	F
F	T

"**It is not the case that** the moon is made of cheese" is **true** because "the moon is made of cheese" is **false**.

"**It is not the case that** grass is green" is **false** because "grass is green" is **true**.

Truth Tables

Conjunction

P	**Q**	**P $\wedge$ Q**
T	T	T
T	F	F
F	T	F
F	F	F

P and Q are true:
“The sky is blue **and** grass is green.” (this sentence is true)

P is true, but Q is false:
“Grass is green **and** dogs are birds.” (this sentence is false)

P is false, but Q is true:
“Snakes are insects **and** the sky is blue.” (this sentence is false)

P and Q are both false:
“Snakes are insects **and** dogs are birds.” (this sentence is false)

Truth Tables

Disjunction

P	**Q**	**P $\vee$ Q**
T	T	T
T	F	T
F	T	T
F	F	F

P and Q are true:
"The sky is blue **or** grass is green." (this sentence is true)

P is true, but Q is false:
"Grass is green **or** dogs are birds." (this sentence is true because grass is green)

P is false, but Q is true:
"Snakes are insects **or** the sky is blue." (this sentence is true because the sky is blue)

P and Q are both false:
"Snakes are insects **or** dogs are birds." (this sentence is false)

Truth Tables

Conditional

P	Q	$P \to Q$
T	T	T
T	F	F
F	T	T
F	F	T

Truth Tables

Conditional

P	Q	P → Q
T	T	T
T	F	F
F	T	T
F	F	T

Which of these scenarios did I lie in, or break my promise to you?

To understand why "→" is defined this way, it may help to consic
yourself: **In which of these four scenarios did I tell a lie**?

I say to you, "If you come over and help me move my couch on Saturc ...en I will buy you some pizza." Translation: $P \rightarrow Q$

<u>Scenario 1:</u> You DO help me, and I DO buy you pizza (P and Q are both true).
<u>Scenario 2:</u> You DO help me, but I do NOT buy you pizza (P is true, Q is false).
<u>Scenario 3:</u> You do NOT help me, but I DO buy you pizza anyway (P is false, Q is true).
<u>Scenario 4:</u> You do NOT help me, and I do NOT buy you pizza (P and Q are both false).

Truth Tables

Shorthand for $P \Longrightarrow Q \wedge Q \Longrightarrow P$

Bi-Conditional

P	Q	P ↔ Q
T	T	T
T	F	F
F	T	F
F	F	T

Which of these scenarios did I lie in?

Scenario 1: Peggy goes to the party, AND Quinn goes too. (P and Q are both true).
Scenario 2: Peggy goes to the party, but Quinn does NOT go. (P is true, Q is false).
Scenario 3: Peggy does NOT go to the party, but Quinn does go. (P is false, Q is true).
Scenario 4: Peggy does NOT go, and neither does Quinn. (P and Q are both false).

A Simple Knowledge Base

Now that we've defined the **semantics** of propositional logic, we can construct a knowledge base for Wampa World.