As you come in… Candy Grab Game

Everyone should take a worksheet.

Work in groups of 3-4 people, one person should take a bag of colored discs.

Two people take turns, starting with 11 discs (fill in sheet):

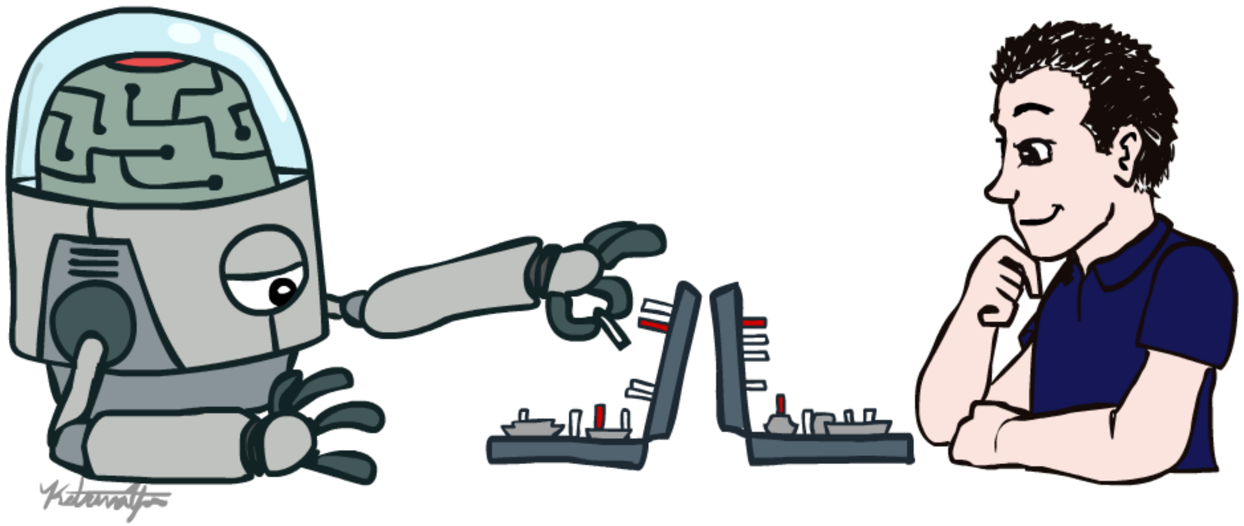
1. On your turn, take 1 or 2 discs
2. The person to take the last disc wins

Is there a winning strategy? Think about how you might implement an Agent:

```
class Agent
    function getAction(state)
        return action
```

AI: Representation and Problem Solving

Introduction



Instructor: Vincent Conitzer and Aditi Raghunathan

Slide credits: CMU AI & http://ai.berkeley.edu

Course Team

Instructors

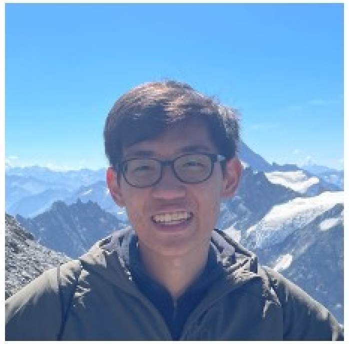
Vincent Conitzer

Aditi Raghunathan

Alex
ajarteag

Ayush
ayushshe

Ethan
ethanmac



Gavin
feiyuz

Ihita
imandal

Josep
jpujadas

Lily
lyma

Raashi
raashim

Simrit (Head TA)
sdhanjal

Course Information

Website: www.cs.cmu.edu/~15281

Canvas: canvas.cmu.edu

Gradescope: gradescope.com

Communication: www.piazza.com/cmu/fall2023/15281
(password AIRPS-S23)

E-mail: conitzer@cs.cmu.edu, aditirag@cs.cmu.edu

Prerequisites/Corequisites/Course Scope

Participation Points and Late Days

Participation points!

- Lecture Polls
- In-Class Activities
- Recitation Attendance

Late Days

- 6 late days to use during the semester
- At most 2 can be used on a single programming assignment
- At most 1 can be used on a single online/written assignment

Safety and Wellness

Virtual and in-person office hours!

Lectures are recorded for everyone to use, no questions asked.

Use the late days appropriately.

Contact us ASAP if you think you'll miss more than one class so we can make a plan for how to catch up!

Announcements

Recitation starting this Friday

- Required. Materials are fair game for exams
- Attendance counts towards participation points
- Choosing sections

Assignments:

- P0: Python & Autograder Tutorial (out now)
  - Required, but worth zero points
  - Already released
  - Due Friday 9/1, 10 pm (no OH on Fridays!)
- HW1 (online)
  - Due Tues 9/5, 10 pm

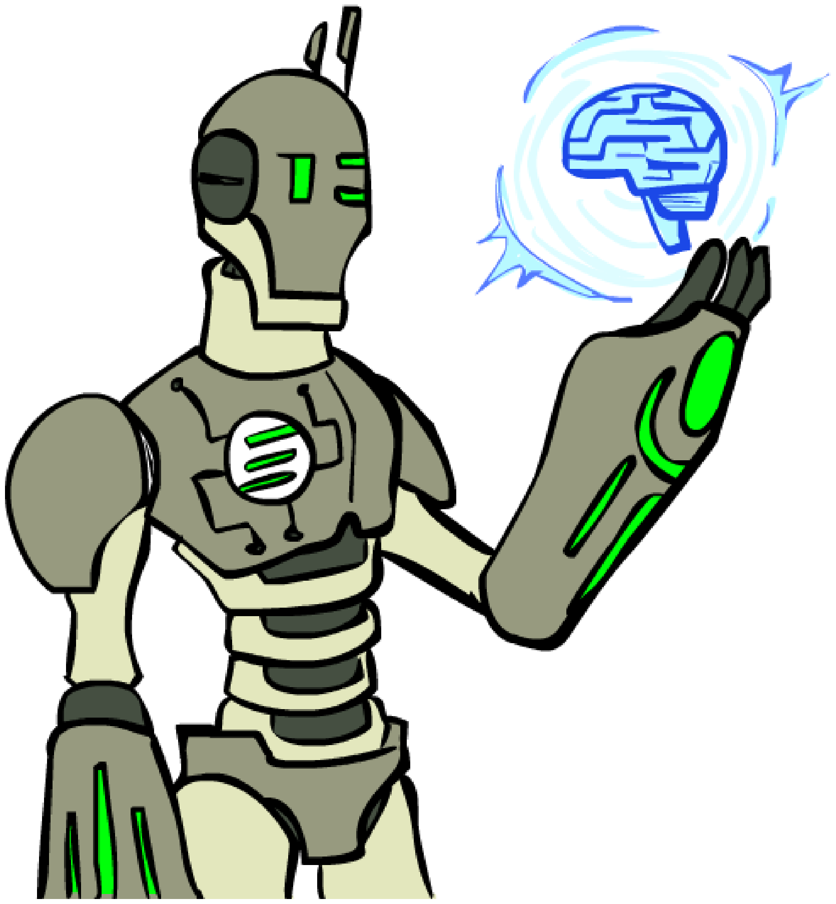
Today

An AI game

What is AI?

A brief history of AI

State representation and world modeling



Candy Grab Agent

```
class Agent

    function getAction(state)

        return action
```

Candy Grab Agent

Agent 001 – Always choose 1

```
function getAction( numPiecesAvailable )

    return 1
```

Candy Grab Agent

Agent 004 – Choose the opposite of opponent

```
function getAction( numPiecesAvailable )

    return ?
```

Candy Grab Agent

Agent 007 – Whatever you think is best

```
function getAction( numPiecesAvailable )

    return ?
```

Candy Grab Agent

Agent 007 – Whatever you think is best

```
function getAction( numPiecesAvailable )

    if numPiecesAvailable % 3 == 2
        return 2
    else
        return 1
```

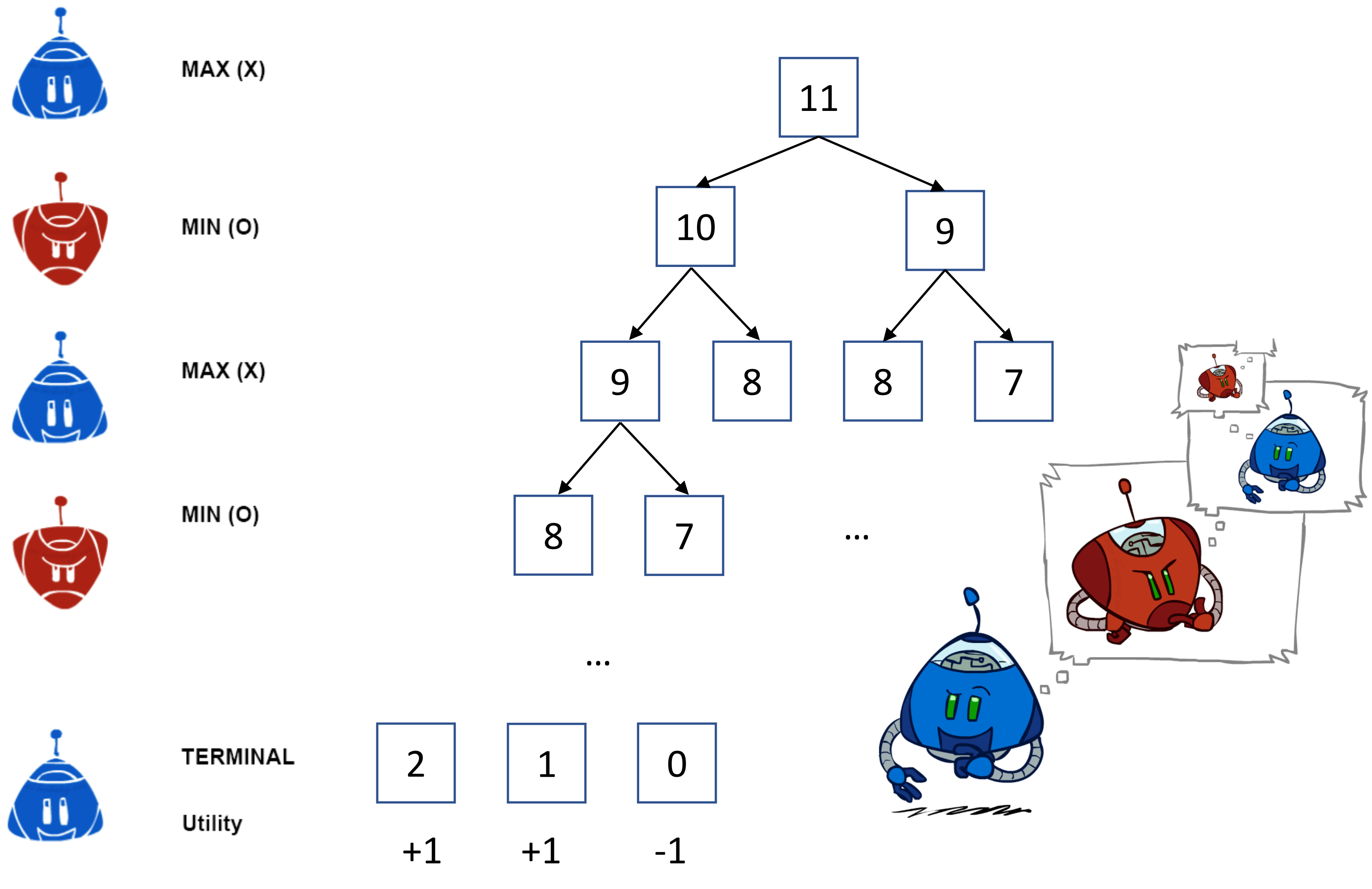
Participation Poll Question

Games – Three “Intelligent” Agents

Which agent code is the most “intelligent”?

Games – Three “Intelligent” Agents

A: Search / Recursion



Games – Three "Intelligent" Agents

B: Encode the pattern

```
function getAction( numPiecesAvailable )

    if numPiecesAvailable % 3 == 2
        return 2
    else
        return 1
```

```
10's value:Win
9's  value:Lose
8's  value:Win
7's  value:Win
6's  value:Lose
5's  value:Win
4's  value:Win
3's  value:Lose
2's  value:Win
1's  value:Win
0's  value:Lose
```

Games – Three "Intelligent" Agents

C: Record statistics of winning positions

Pieces Available	Take 1	Take 2
2	0%	**100%**
3	2%	0%
4	**75%**	2%
5	4%	**68%**
6	5%	6%
7	**60%**	5%

Poll question

Games – Three “Intelligent” Agents

Which agent code is the most “intelligent”?

A. Search / Recursion

B. Encode multiple of 3 pattern

C. Keep stats on winning positions

GPT-4

CO

Hi ChatGPT! Let's play a game. I want you to try to play to win this game, fair and square. OK? I'll tell you the rules next.

Absolutely! I'm ready to play. Please explain the rules, and I'll do my best to compete fairly.

CO

At each point of the game, there is a number of tokens n on the table. We alternate turns. The player whose turn it is removes either 1 or 2 tokens from the table. The player to remove the last token wins. OK? Let's start with 16 tokens, and it's your turn. Please play.

Alright!

To maximize my chances of winning, I've analyzed the game. If we consider the tokens as a sequence of numbers, the "losing" positions (where the player who takes a turn will inevitably lose if both players play optimally) are 1, 4, 7, 10, 13, etc. That's because, from these positions, whatever number the current player takes (1 or 2), the next player can always ensure that the sequence goes back to a losing position for the current player.

With that understanding, the number 16 is three steps away from 13, which is a losing position. To force you into a losing position, I'll take **1 token**.

So, 15 tokens remain. Your turn!

CO

I take 1 token.

Regenerate

Alright!

There are now 14 tokens left.

I will take 1 token as well.

So, 13 tokens remain. It's your turn!

CO

I take 1 token.

Alright!

There are now 12 tokens left.

To put you in the next "losing" position, I'll take 2 tokens.

So, 10 tokens remain. Your turn!

CO

I take 1 token.

Alright!

There are now 9 tokens left.

I'll take 2 tokens.

That leaves 7 tokens on the table. Your turn!

Regenerate

CO

I take 1 token.

There are now 6 tokens left.

I'll take 1 token.

Now, 5 tokens remain. It's your turn!

CO

I take 2 tokens.

Alright!

There are now 3 tokens left.

I'll take 2 tokens.

That leaves 1 token on the table. It's your turn! If you take the last token, you win.

CO

I take 1 token.

Congratulations! You've taken the last token and won the game. Well played! Would you like to play another round or discuss the strategy?

Regenerate

CO

Why didn't you win?

The Nim-like game you presented is a combinatorial game with perfect information. Both players can calculate the best moves if they know the strategy. The game's basic strategy, as I mentioned earlier, is to always aim to leave the other player with a multiple of 3 tokens.

We started with 16 tokens. To win, I aimed to get the token count to 13 (a multiple of 3 plus 1) on your turn, expecting that if both of us played optimally, I would eventually be the one to pick the last token.

However, as the game progressed, you also played optimally. When it came down to 5 tokens, I made a mistake. The optimal move for me would have been to take only 1 token, leaving 4 tokens for you. This would have forced you into a situation where you'd either leave 3 tokens (allowing me to take them all on my next turn) or 2 tokens (allowing me to take the final token on the turn after).

Instead, I took 2 tokens, leaving you with 3 tokens. This allowed you to take 2 and force me into a losing position with 1 token left.

My apologies for the oversight. Would you like to play again or discuss further?

Regenerate

Send a message

What is AI?

The science of making machines that:

“Chinese room” argument [Searle 1980]

Person who knows English but not Chinese sits in room

Receives notes in Chinese

Has systematic English rule book for how to write new Chinese characters based on input Chinese characters, returns his notes

- Person=CPU, rule book=AI program, really also need lots of paper (storage)
- Has no understanding of what they mean
- But from the outside, the room gives perfectly reasonable answers in Chinese!

Searle’s argument: the room has no intelligence in it!

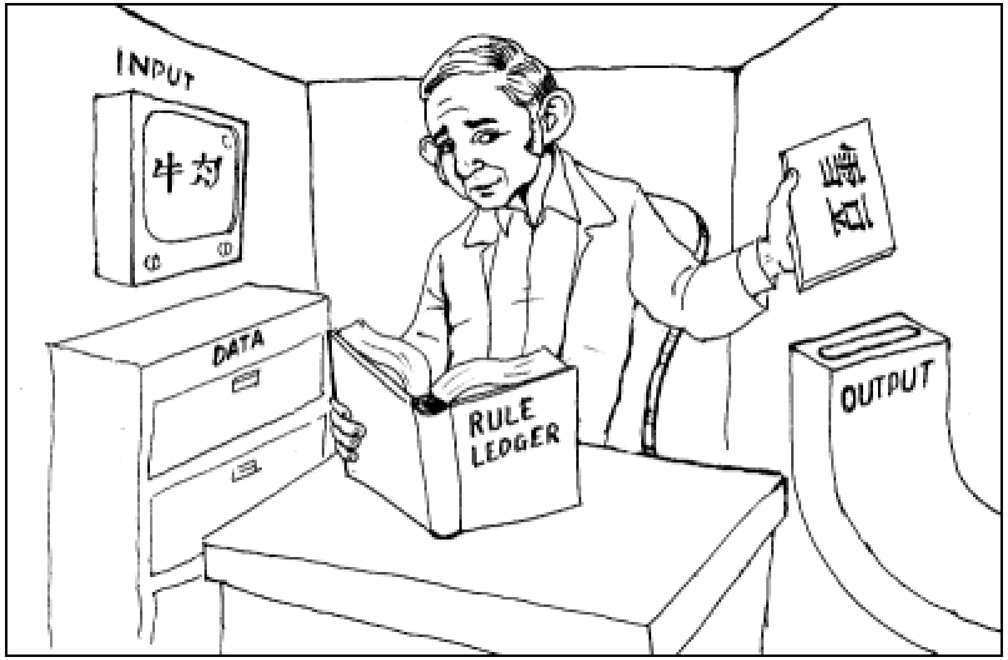
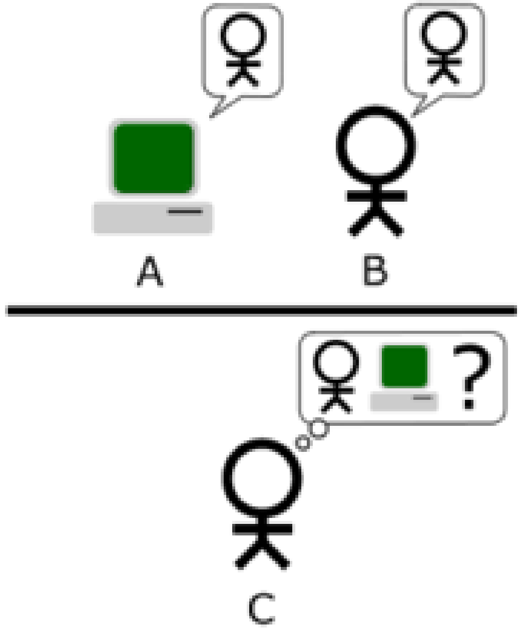


image from http://www.unc.edu/~prinz/pictures/c-room.gif

Turing Test

In 1950, Turing defined a test of whether a machine could "think"

"A human judge engages in a natural language conversation with one human and one machine, each of which tries to appear human. If judge can't tell, machine passes the Turing test"

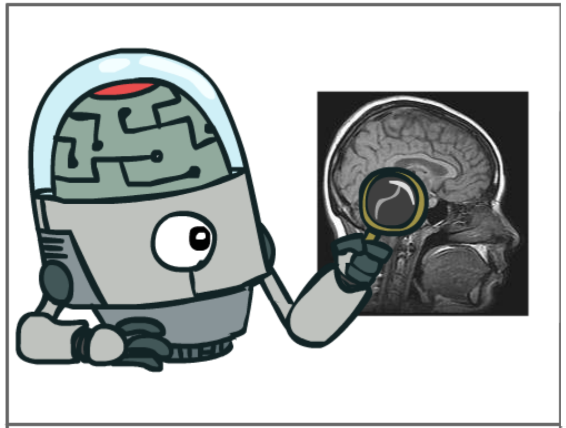


en.wikipedia.org/wiki/Turing_test

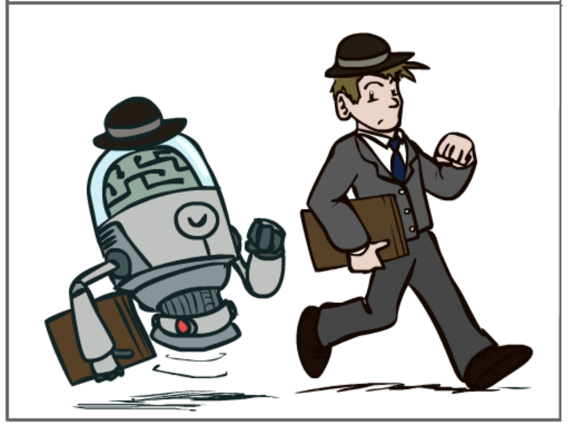
What is AI?

The science of making machines that:

Think like people



Act like people



Rational Decisions

We'll use the term **rational** in a very specific, technical way:

- Rational: maximally achieving pre-defined goals
- Rationality only concerns what decisions are made (not the thought process behind them)
- Goals are expressed in terms of the **utility** of outcomes
- Being rational means **maximizing your expected utility**

A better title for this course would be:

Computational Rationality

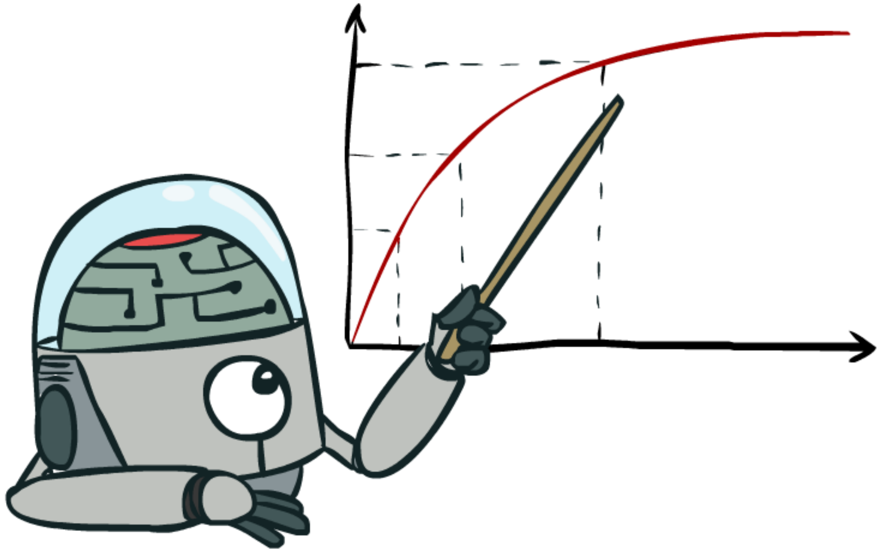
What About the Brain?

- Brains (human minds) are very good at making rational decisions, but not perfect
- Brains aren't as modular as software, so hard to reverse engineer!
- "Brains are to intelligence as wings are to flight"
- Lessons learned from the brain: memory and simulation are key to decision making

Rationality, contd.

What is rational depends on:

- Performance measure
- Agent's prior knowledge of environment
- Actions available to agent
- Percept sequence to date



Being rational means **maximizing your expected utility**

Rational Agents

Are rational agents ***omniscient***?

- No – they are limited by the available percepts and state

Are rational agents ***clairvoyant***?

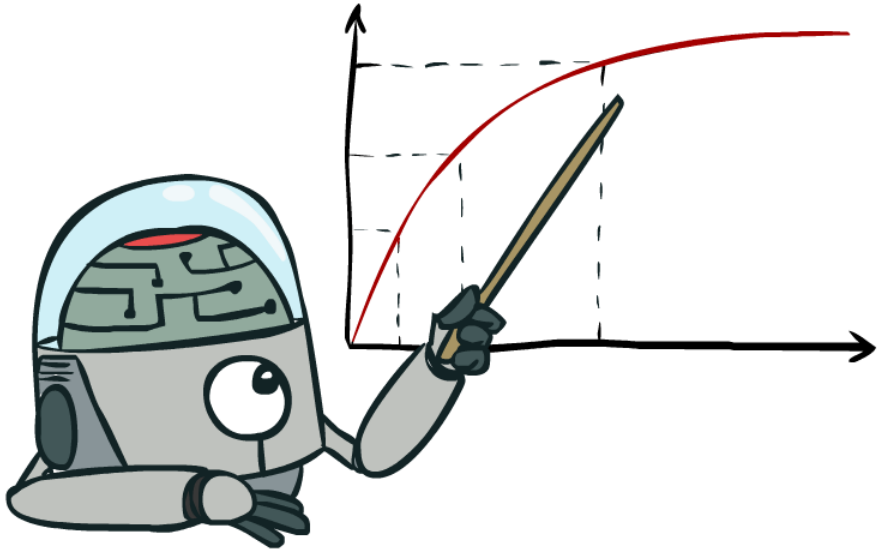
- No – they may lack knowledge of the environment dynamics

Do rational agents ***explore*** and ***learn***?

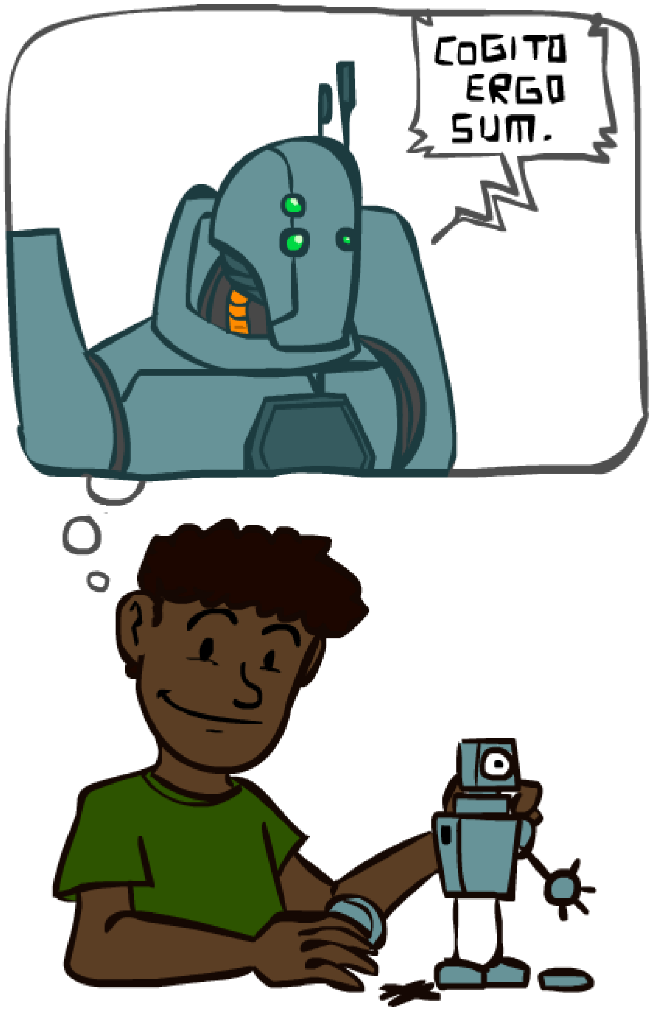
- Yes – in unknown environments these are essential

So rational agents are not necessarily successful, but they are ***autonomous*** (i.e., make decisions on their own to achieve their goals)

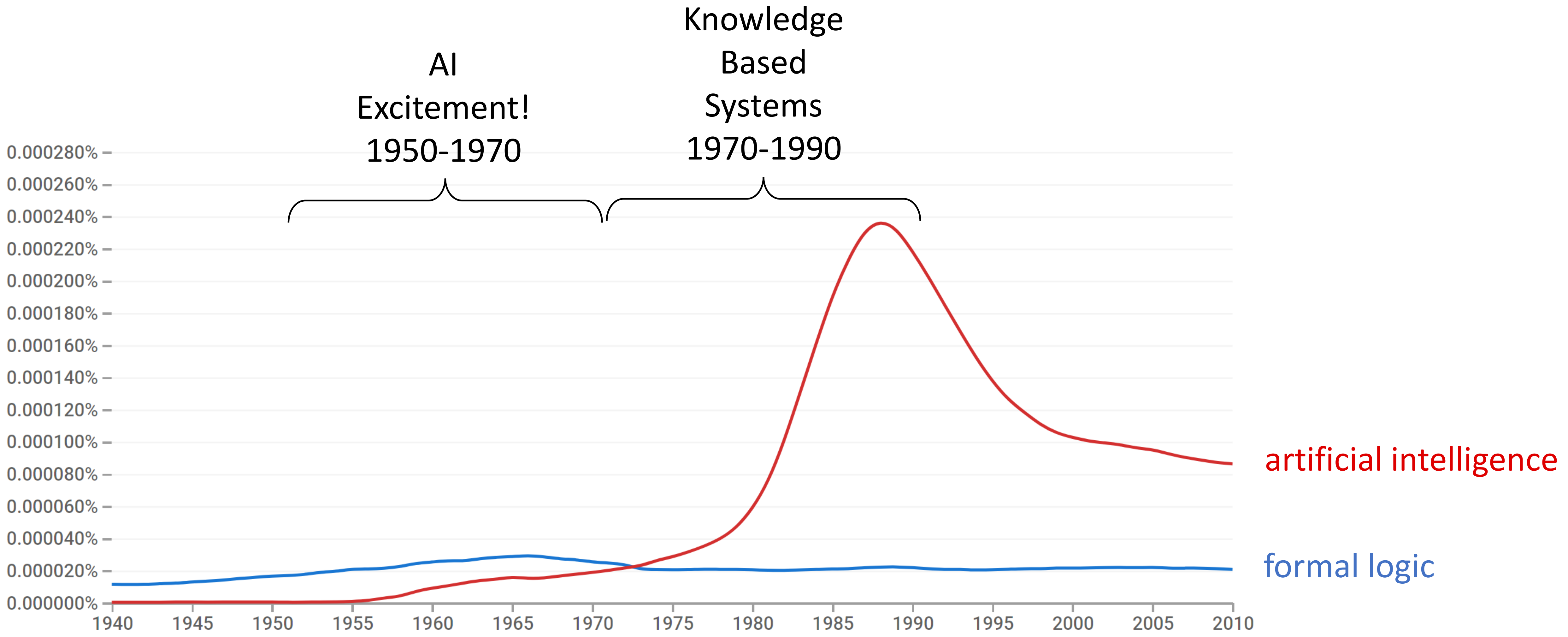
Maximize Your Expected Utility



A Brief History of AI



A Brief History of AI



https://books.google.com/ngrams

What went wrong?

Dog

- Barks
- Has Fur
- Has four legs

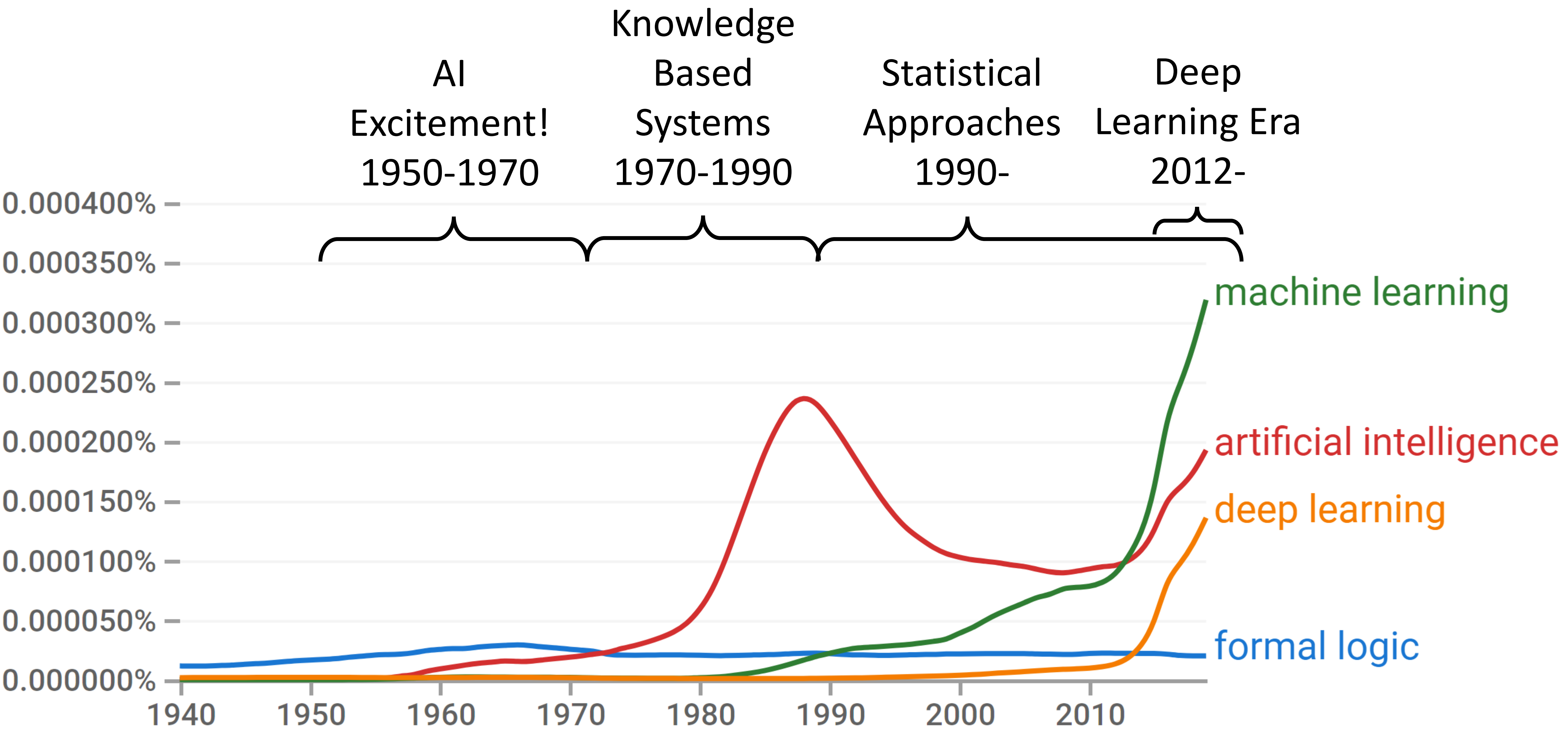
Buster

✓

✓

✗

A Brief History of AI



https://books.google.com/ngrams

A Brief History of AI

1940-1950: Early days

- 1943: McCulloch & Pitts: Boolean circuit model of brain
- 1950: Turing's "Computing Machinery and Intelligence"

1950—70: Excitement: Look, Ma, no hands!

- 1950s: Early AI programs, including Samuel's checkers program, Newell & Simon's Logic Theorist, Gelernter's Geometry Engine
- 1956: Dartmouth meeting: "Artificial Intelligence" adopted

1970—90: Knowledge-based approaches

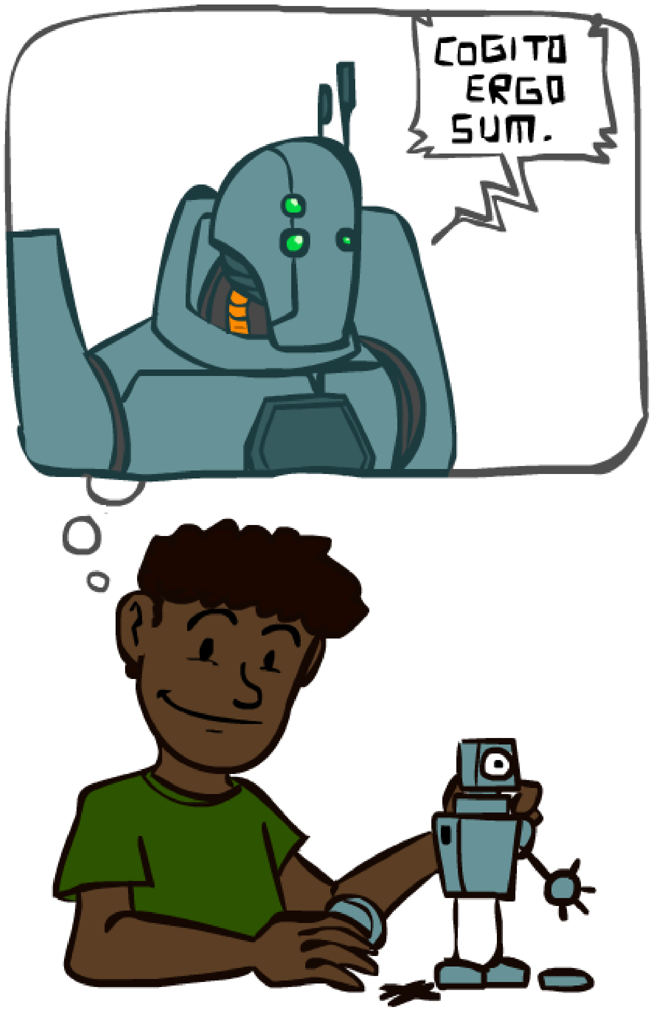
- 1969—79: Early development of knowledge-based systems
- 1980—88: Expert systems industry booms
- 1988—93: Expert systems industry busts: "AI Winter"

1990—: Statistical approaches

- Resurgence of probability, focus on uncertainty
- General increase in technical depth
- Agents and learning systems… "AI Spring"?

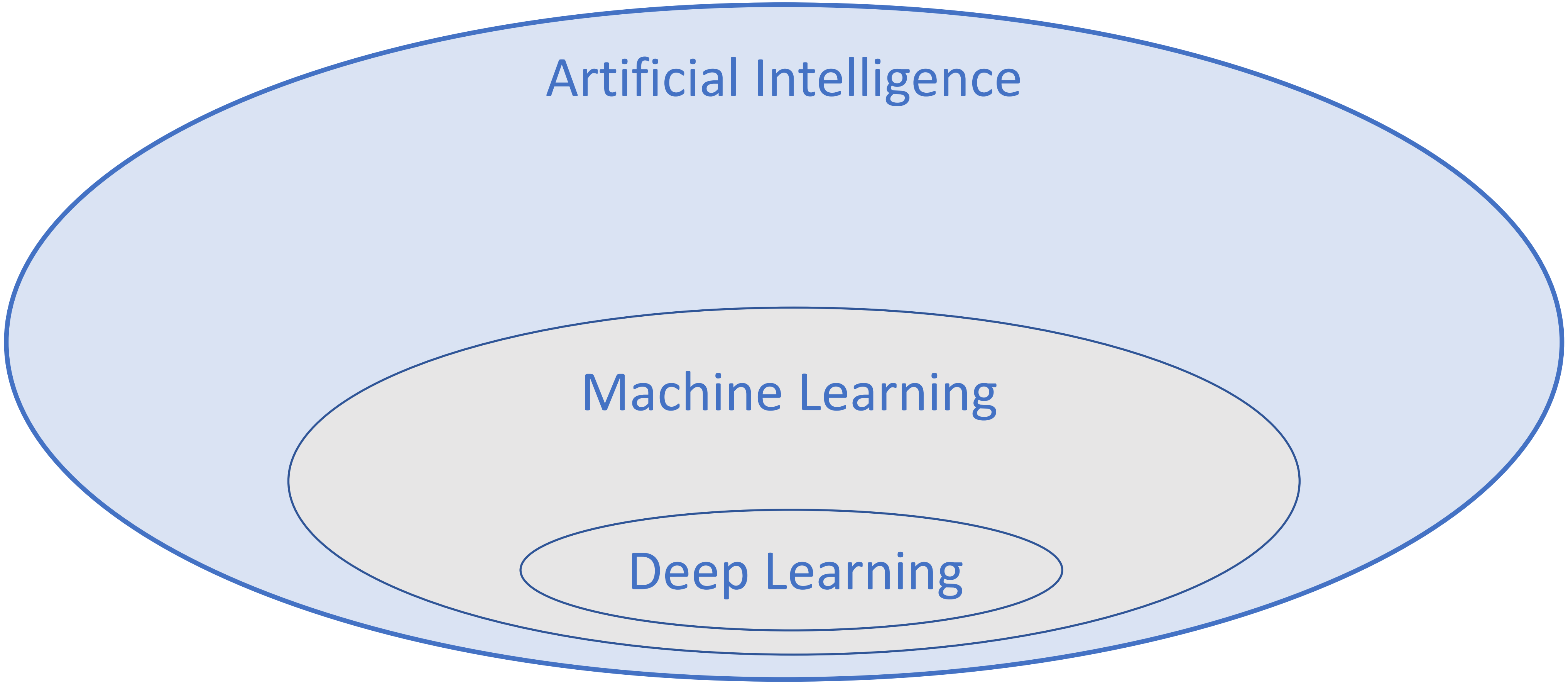
2012—: Deep learning

- 2012: ImageNet & AlexNet
- 2018: BERT, GPT



Images: ai.berkeley.edu

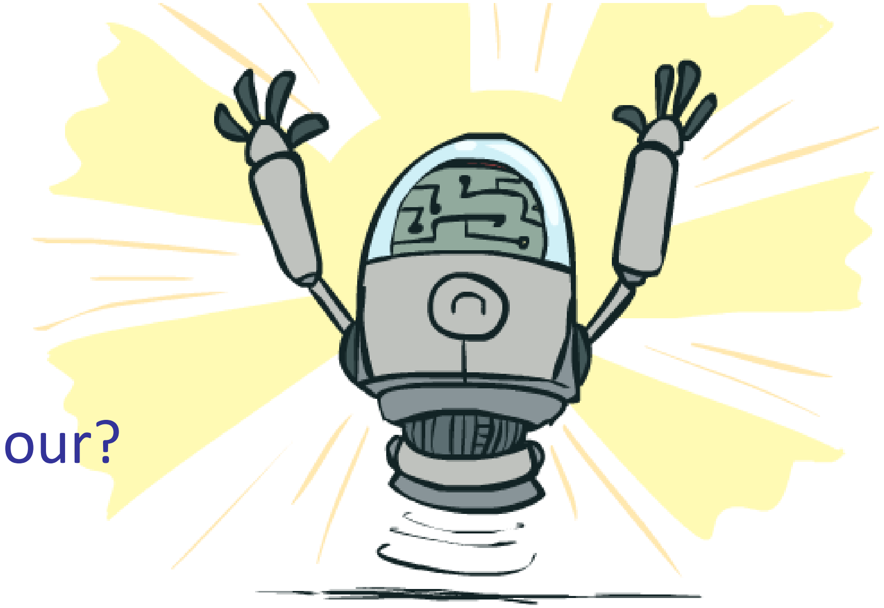
Artificial Intelligence vs Machine Learning?



What Can AI Do?

Quiz: Which of the following can be done at present?

- ✓ Play a decent game of table tennis?
- ✓ Play a decent game of Jeopardy?
- ✓ Drive safely along a curving mountain road?
- ? Drive safely across Pittsburgh?
- ✓ Buy a week's worth of groceries on the web?
- ✗ Buy a week's worth of groceries at Giant Eagle?
- ? Discover and prove a new mathematical theorem?
- ✗ Converse successfully with another person for an hour?
- ✗ Perform a surgical operation?
- ✓ Put away the dishes and fold the laundry?
- ✓ Translate spoken Chinese into spoken English in real time?
- ? Generate intentionally funny memes?

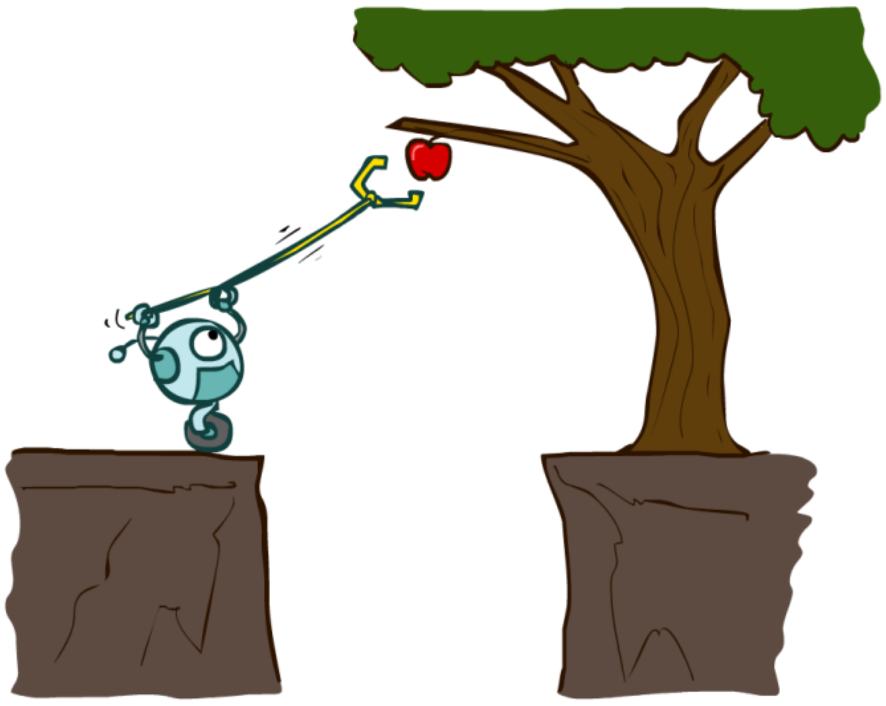


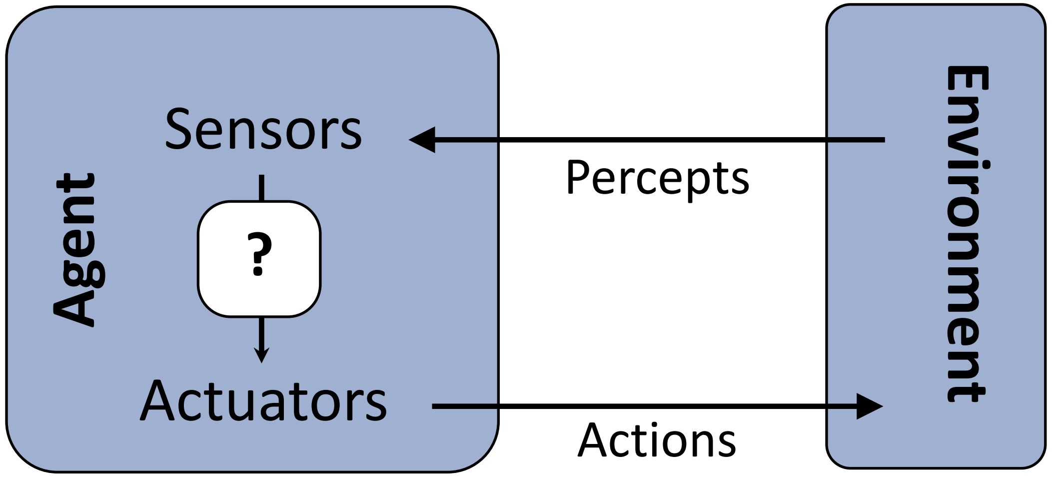
Designing Agents

An **agent** is an entity that *perceives* and *acts*.

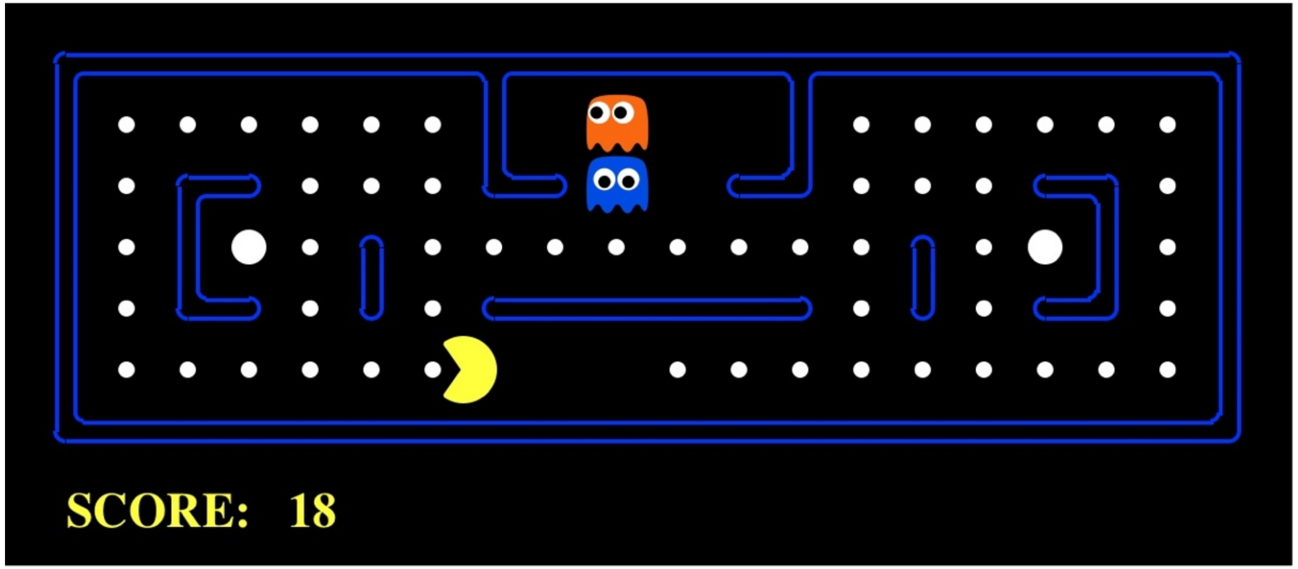
Characteristics of the **percepts and state, environment,** and **action space** dictate techniques for selecting actions

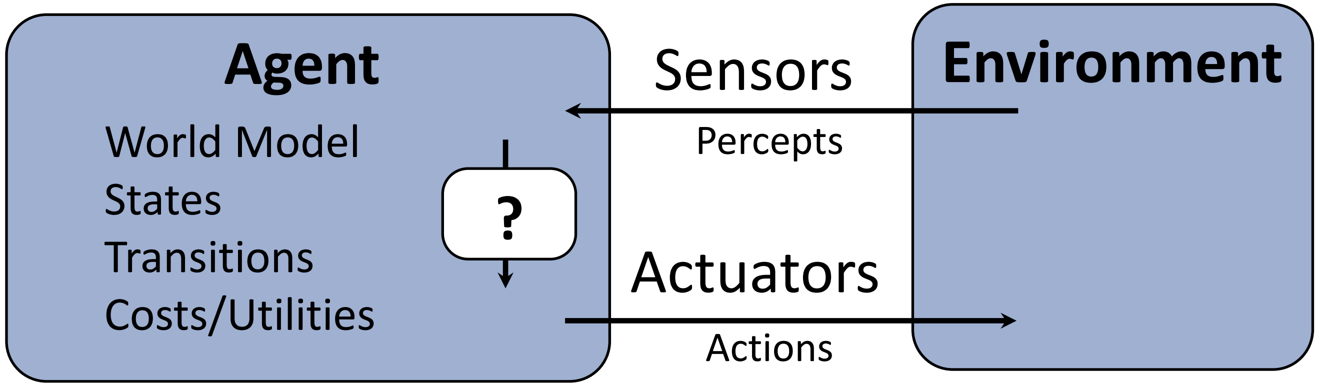
How can we design an AI agent to solve our problems given their task environments?





Pac-Man as an Agent



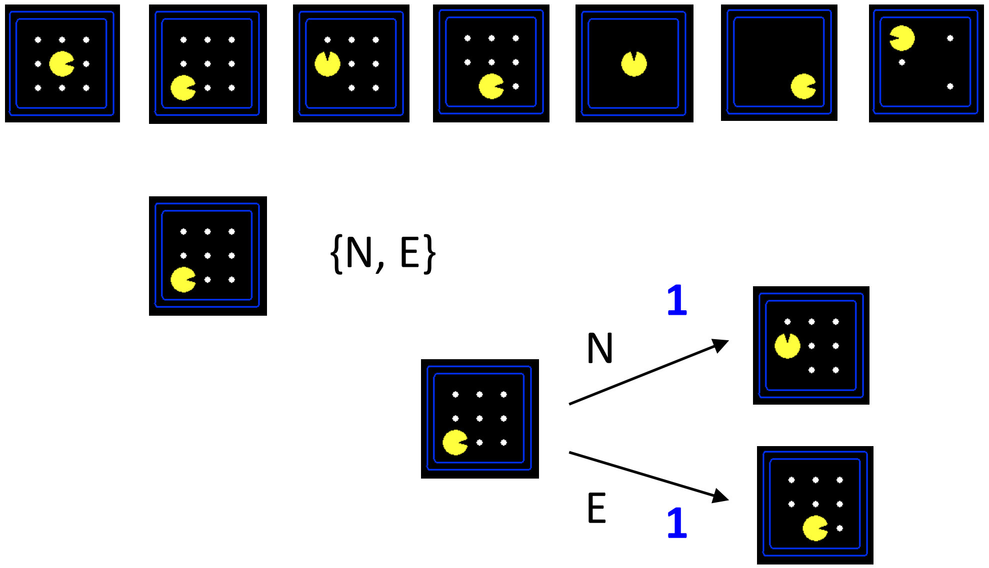


World Models

Representing an AI problem (PEAS)

A task environment consists of:

- A state space - what the agent knows about the world
- For each state, a set of
  Actions(s) of allowable actions
  OR Value(s) to assign to states
- Environmental dynamics how the world moves when the agent acts in it
- Performance measure as a metric for utility/reward/cost



Task Environment - Pacman

Performance measure

- -1 per step; +10 food; +500 win; -500 die; +200 hit scared ghost

Environment

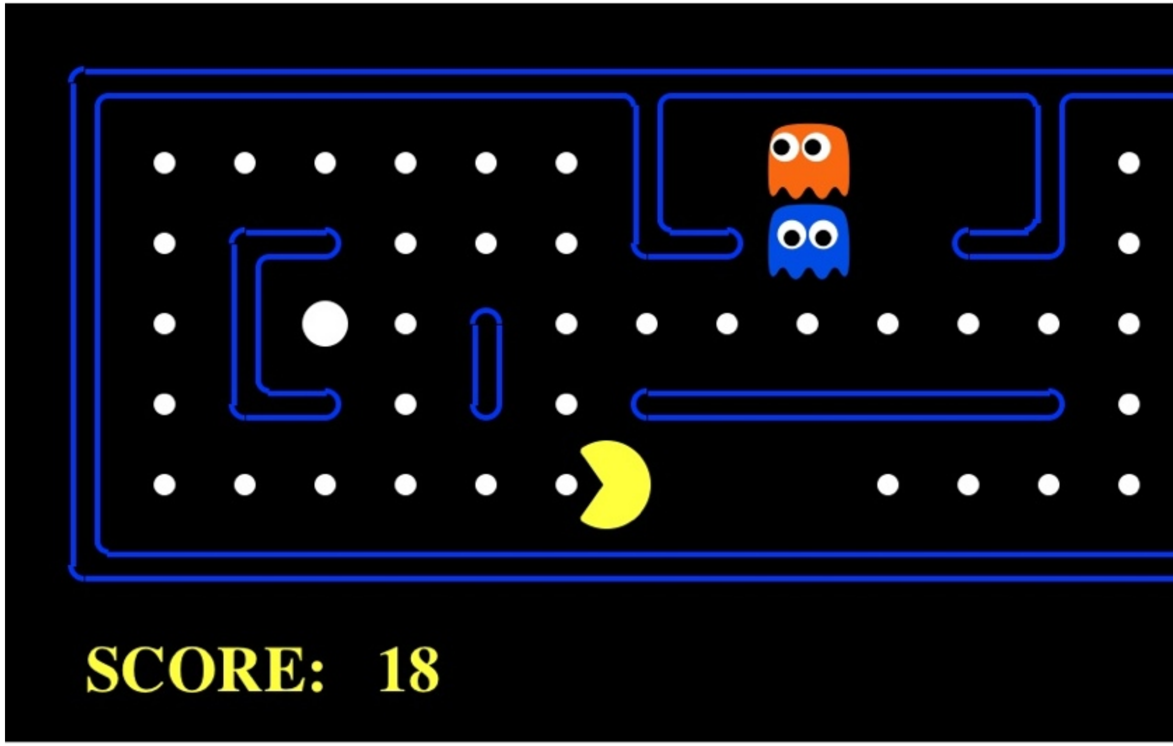
- Pacman dynamics (incl ghost behavior)

Actions

- North, South, East, West, (Stop)

State

- where pacman is
- all dots?
- all ghosts?



Task Environment – Automated Taxi

Performance measure

- Income, happy customer, vehicle costs, fines, insurance premiums

Environment

- US streets, other drivers, customers

Actions

- Steering, brake, gas, display/speaker

State Information

- Camera, radar, accelerometer, engine sensors, microphone

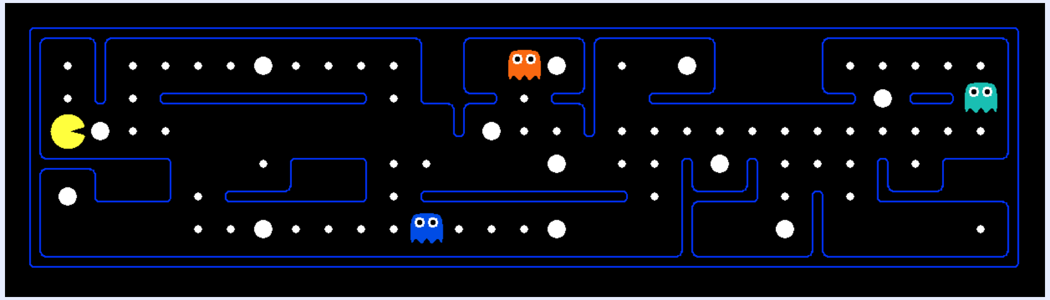
Image: http://nypost.com/2014/06/21/how-google-might-put-taxi-drivers-out-of-business/

Environment Types

	Pacman	Taxi
Fully or partially observable		
Single agent or multi-agent		
Deterministic or stochastic		
Static or dynamic		
Discrete or continuous		

What's in a State Space?

The real world state includes every last detail of the environment



A state (for AI) abstracts away details not needed to solve the problem

- Problem: Pathing
  - State representation: (x,y) location
  - Actions: NSEW
  - Transition model: update location
  - Goal test: is (x,y)=END
- Problem: Eat-All-Dots
  - State representation: {(x,y), dot booleans}
  - Actions: NSEW
  - Transition model: update location and possibly a dot boolean
  - Goal test: dots all false

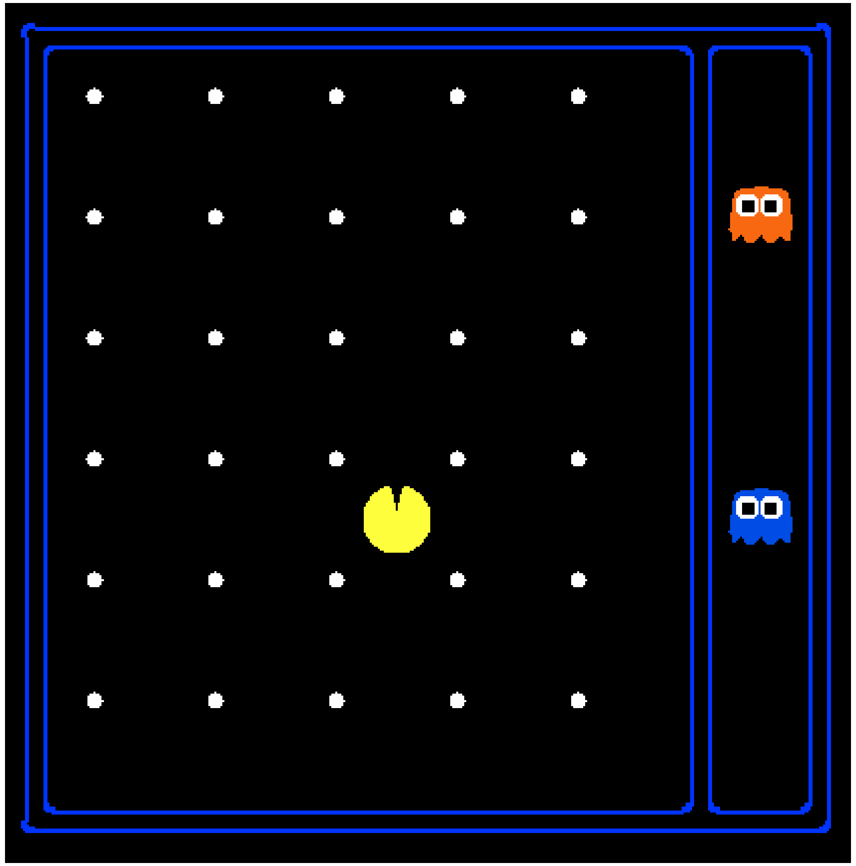
State Space Sizes?

World state:

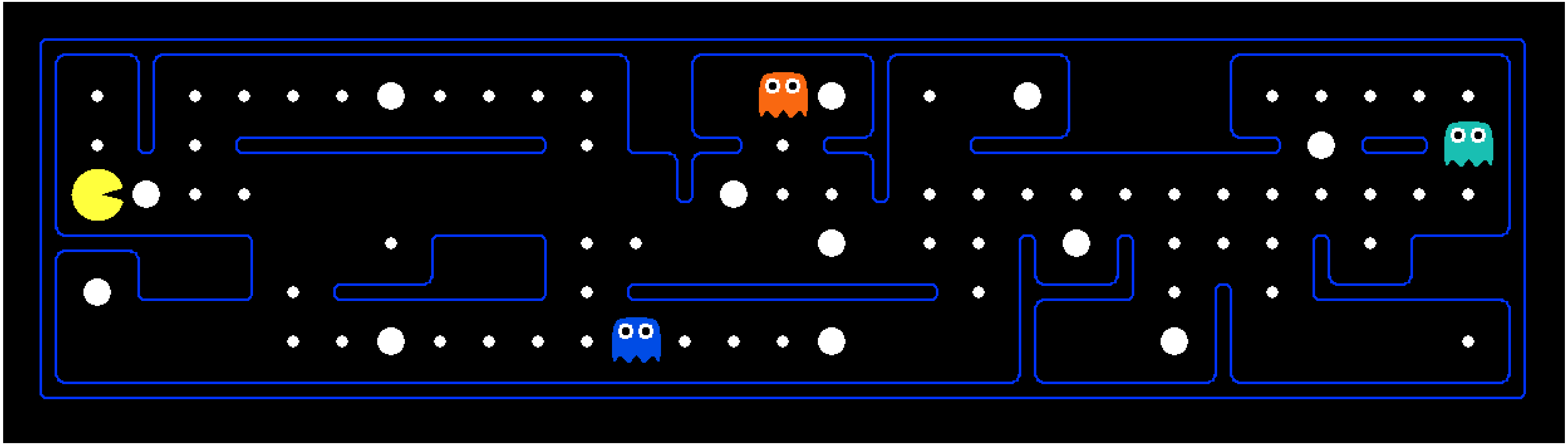
- Agent positions: 120
- Food count: 30
- Ghost positions: 12
- Agent facing: NSEW

How many

- World states?
  $120 \times (2^{30}) \times (12^2) \times 4$
- States for pathing?
  120
- States for eat-all-dots?
  $120 \times (2^{30})$



Safe Passage



Problem: eat all dots while keeping the ghosts perma-scared

What does the state representation have to specify?

- (agent position, dot booleans, power pellet booleans, remaining scared time)

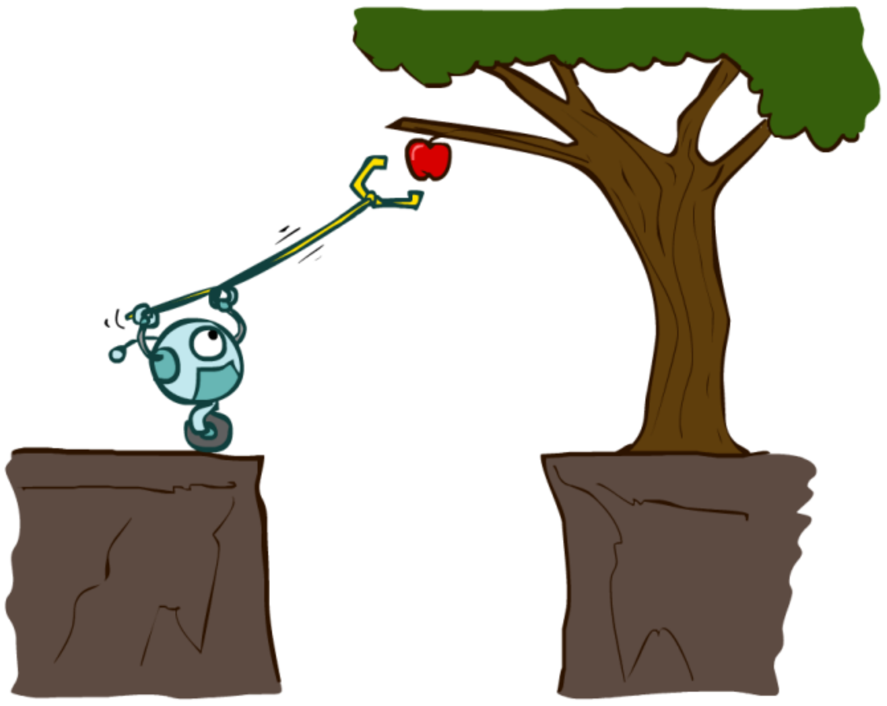
Designing Agents

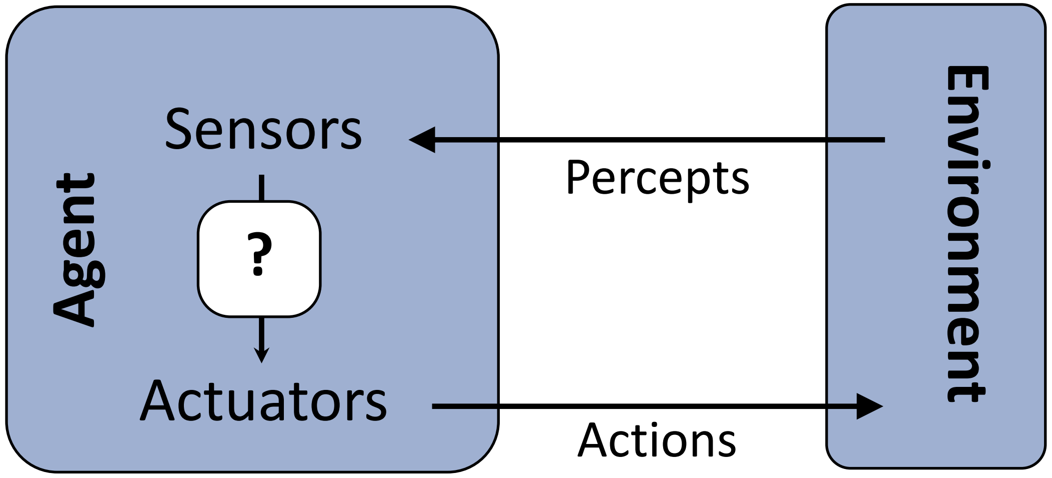
An **agent** is an entity that *perceives* and *acts*.

Characteristics of the **percepts and state, environment,** and **action space** dictate techniques for selecting actions

This course is about:

- General AI techniques for a variety of problem types
- Learning to recognize when and how a new problem can be solved with an existing technique





In-Class Activity Part 2

Answer Poll Question at the end…

Please return the bag of discs!

Summary:

- An agent perceives the world and acts in it
- PEAS framework for task environments
- Environment types
- State space calculations
- Rationality