

---

# CS 232: Artificial Intelligence

## Fall 2026

---

Prof. Carolyn Anderson  
Wellesley College

# Reminders

---

- ◆ Next week's reading is pages 28-52 of *AI Needs You*
- ◆ I have help hours today (W423) and Monday from 3-4 (W323)
- ◆ Tutor hours:
  - Aeka: Thursdays 7:30-8:30 in H103
  - Amber: Fridays 1:30-2:30 in Whiteboard Alley
  - Coco: Mondays 7-8 in Whiteboard Alley
  - ◆ - Late Days



# Building in the Age of AI: A Founder's Panel

September 24, 2026 · 131 Oliver St 3rd floor



**Artem Lukoianov**  
PhD @ MIT CSAIL;  
co-founder of Sundai Club



**Rob Blaine**  
Founder & CEO @ TNT



**Brian Devaney**  
Partner at Underscore VC

Hosted in Partnership with

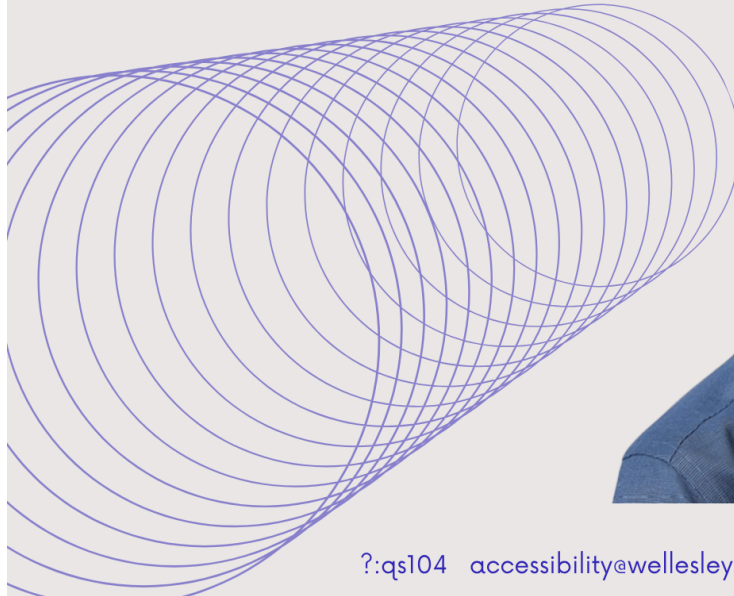
## Underscore VC

THURSDAY, OCTOBER 8  
12:45PM TO 2PM  
H105



<https://bit.ly/3V2EJLd>

CS Colloquium:  
**Tech Ethics and  
College Chaplaincy:  
Lessons from MIT  
Radius**

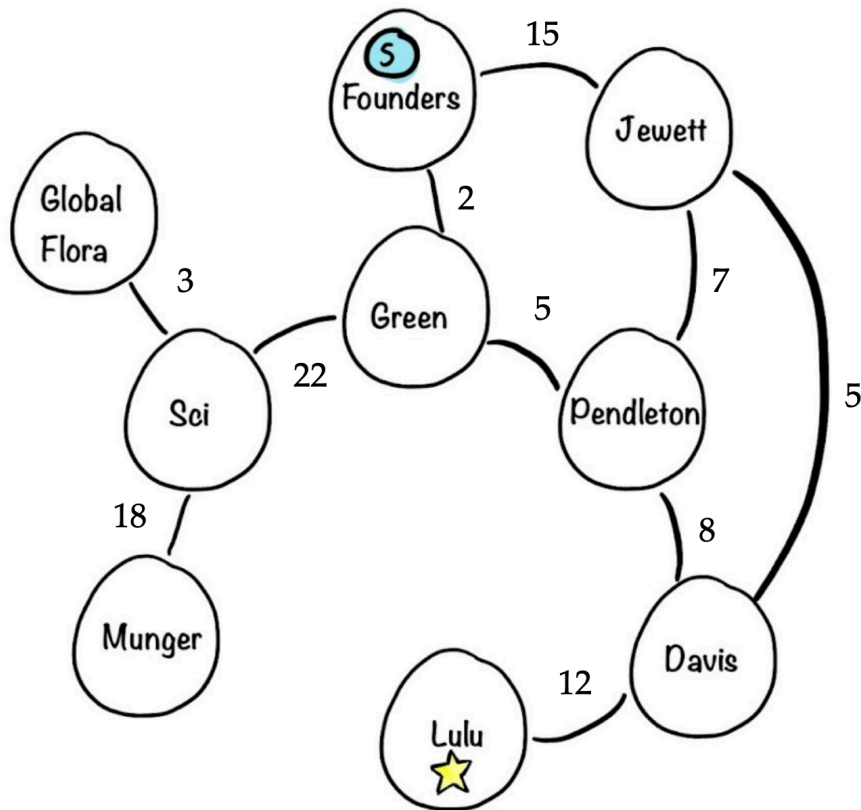


# Uniform-Cost Search

- ♦ Like breadth-first search, but with a twist:  
Expand the node with the **lowest path cost**
- ♦ Implementation:  
Frontier = priority queue ordered by  $g(n)$
- ♦ Key difference: when a new path to a node is discovered, if the cost is lower, we need to update the frontier!
- ♦ Also important: apply goal test only when a node is **visited**  
 $g(n)$ : cost to reach  $n$  from the start node

# Uniform-Cost Search

We only use information about the cost from start to current node.



| <u>Frontier</u>                  | <u>Visited</u> |
|----------------------------------|----------------|
| <del>Founders</del>              | Founders       |
| <del>Green - 2</del>             | Green          |
| <del>Jewett - 15</del> <b>14</b> | Pendleton      |
| <del>Pendleton - 7</del>         | Jewett         |
| <del>Sci - 24</del>              | Davis          |
| <del>Davis - 15</del>            | Sci            |
| <del>Lulu - 27</del>             | GF             |
| <del>GF - 27</del>               | Lulu*          |
| Munger - 42                      |                |

# A Better Idea

- ♦ Node expansion based on an **estimate** which includes distance to the goal.
- ♦ General approach of informed search:
  - ♦ Best-first search: node selected for expansion based on an **evaluation function  $f(n)$**
  - ♦  $f(n)$  includes an **estimate** of the distance to the goal

Add together: Cost from start to current node

make a good guess about this  Cost from current node to goal.

# A\* Search



# Admissible Heuristics

---

$A^*$  is **guaranteed** to find the shortest path if the heuristic **never over-estimates** the cost to the goal.



**Inadmissible (pessimistic)** heuristics break optimality by pushing good plans too far back on the frontier, where they may never be expanded.



**Admissible (optimistic)** heuristics may slow down bad plans but never outweigh true costs. That means that the true best plan will always be expanded.



# What is A\* Used For?

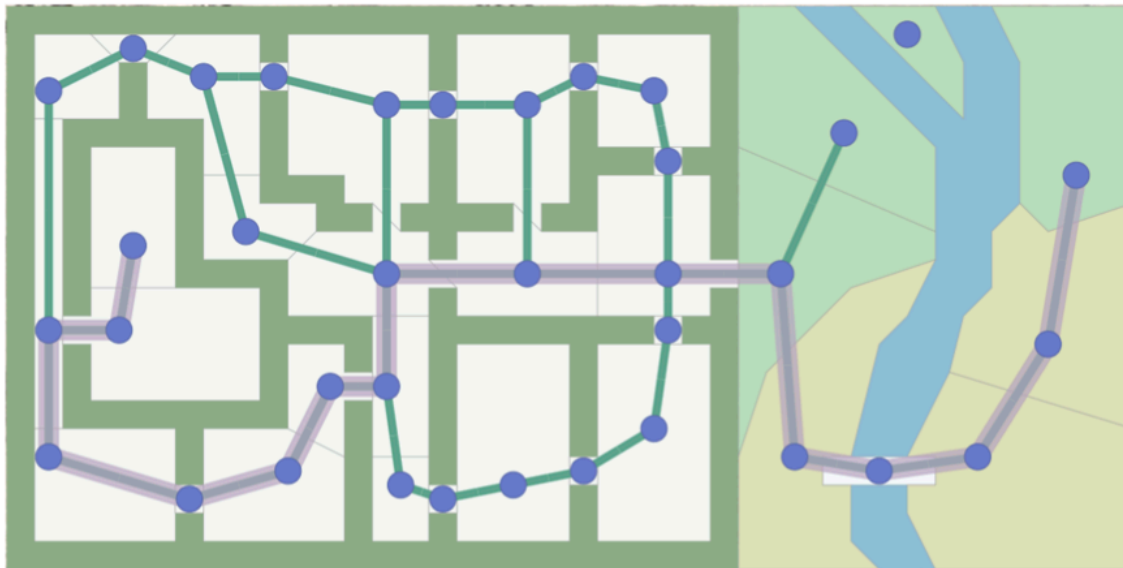
Pathing / routing problems (A\* is in your GPS!)

Video games

Robot motion planning

Resource planning problems

...



# Heuristics

# Admissible Heuristics

---

$A^*$  is **guaranteed** to find the shortest path if the heuristic **never over-estimates** the cost to the goal.



**Inadmissible (pessimistic)** heuristics break optimality by pushing good plans too far back on the frontier, where they may never be expanded.



**Admissible (optimistic)** heuristics may slow down bad plans but never outweigh true costs. That means that the true best plan will always be expanded.

# Key Concept: Admissible Heuristics

A heuristic  $h(n)$  is admissible if it never overestimates the cost to reach the goal:

$\forall n$ ,  $n$  is a node:

$h(n) \leq h^*(n)$  where  $h^*$  is the true cost from  $n$  to goal

$h(n) \geq 0$        $h(G) = 0$  for any goal

If  $h(n)$  is admissible,  $A^*$  is optimal

# Heuristic Functions

What is an admissible heuristic for the 8 puzzle game?

$$h_{\text{op}}(n) = \# \text{ of out of place tiles}$$

$$h_{\text{nd}}(n) = \text{total Manhattan distance} \\ (\text{distance of each tile from its goal location})$$



(Average solution cost is 22 steps; a good heuristic can greatly reduce this!)

# Adversarial Search

# Search

---

So far, we have only considered one-player games.  
What happens when we add another player?

# Multiplayer Games

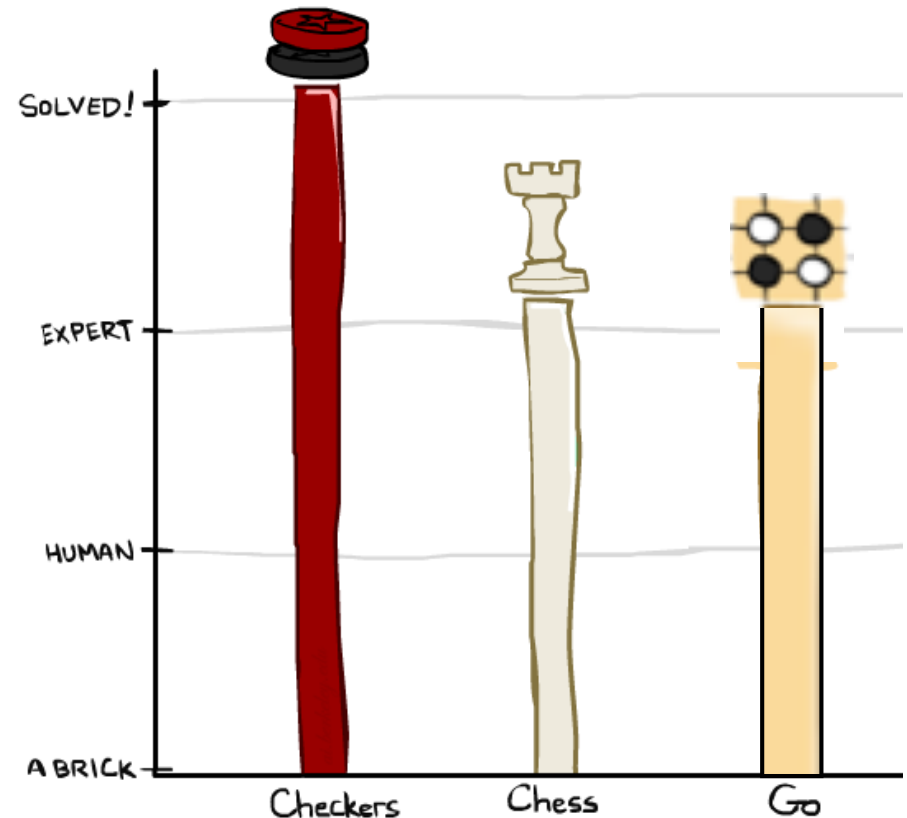
---

In competitive multiplayer games, we have to consider our opponent's possible actions, as well as our own.

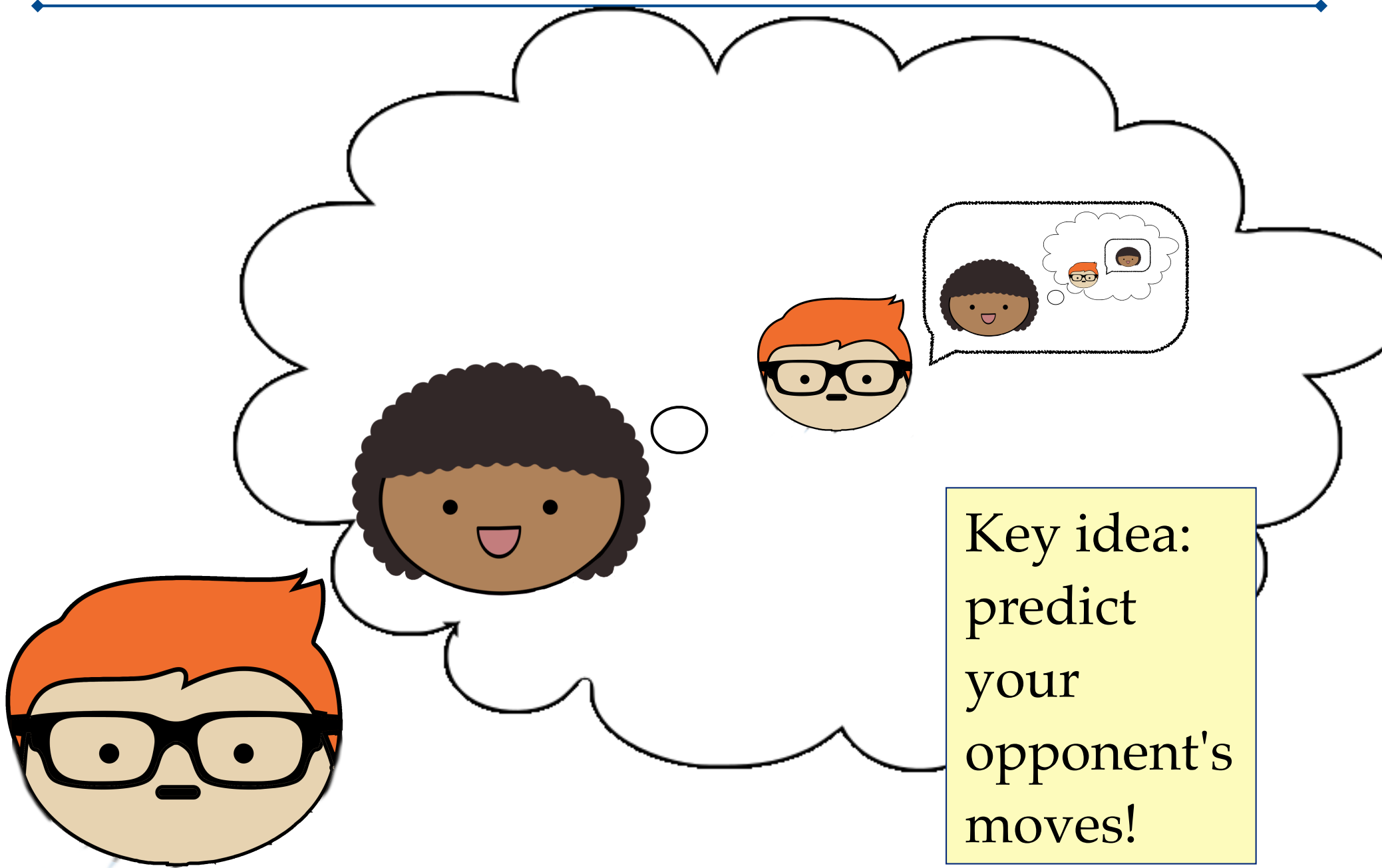
We call this **adversarial search**.

# Game Playing State-of-the-Art

- Checkers: 1950: First computer player. 1994: First computer champion: Chinook ended 40-year-reign of human champion Marion Tinsley using complete 8-piece endgame. 2007: Checkers solved!
- Chess: 1997: Deep Blue defeats human champion Gary Kasparov in a six-game match. Deep Blue examined 200M positions per second, used very sophisticated **evaluation** and undisclosed methods for extending some lines of search up to 40 ply. Current programs are even better, if less historic.
- Go: 2016: Alpha GO defeats human champion. Uses Monte Carlo Tree Search + neural network to learn evaluation function.
- Go + Chess + Shogi: 2017: Alpha Zero learns all 3 games using **reinforcement learning** to play against itself.



# Deterministic Games

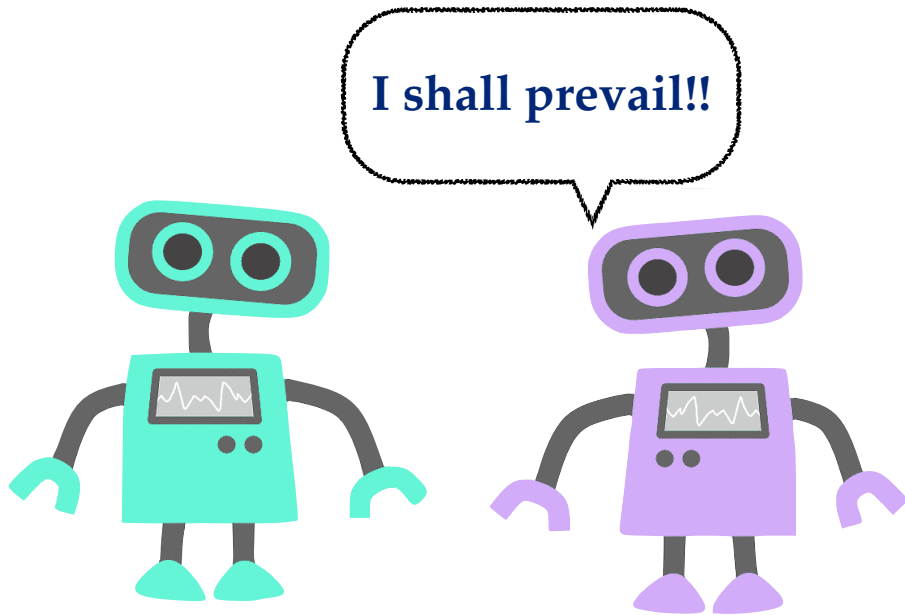


# Deterministic Games

---

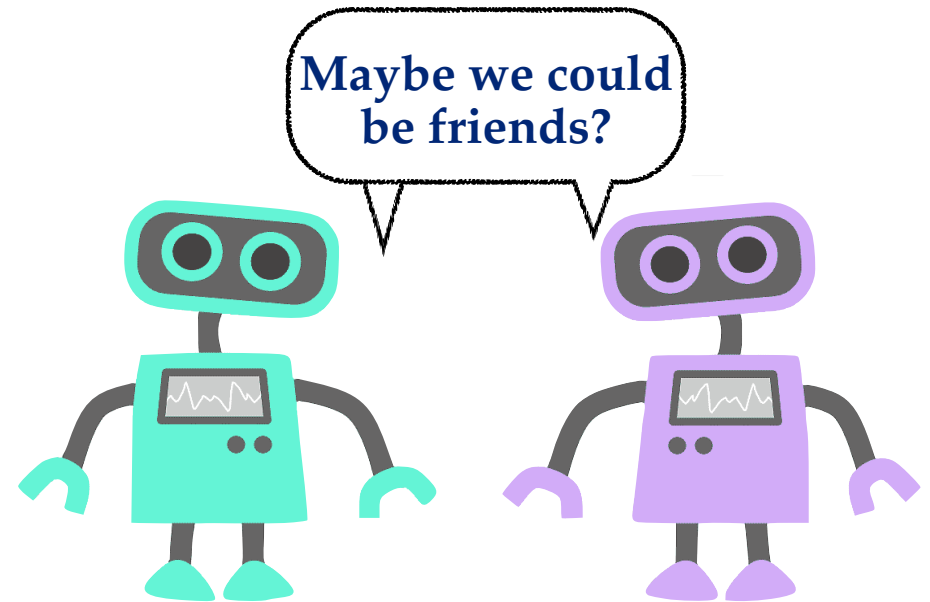
- States:  $S$
- Players:  $P = \{1 \dots N\}$
- Actions:  $A$
- Transition Function  $T(s, a) \rightarrow s'$
- Goal/ Test  $Goal(s) / T(s) \rightarrow True/False$   
Terminal

# Competing with Adversaries



## Zero-Sum Games

- Agents have opposite utilities (values on outcomes)
- A single score that one maximizes and the other minimizes
- Adversarial, pure competition

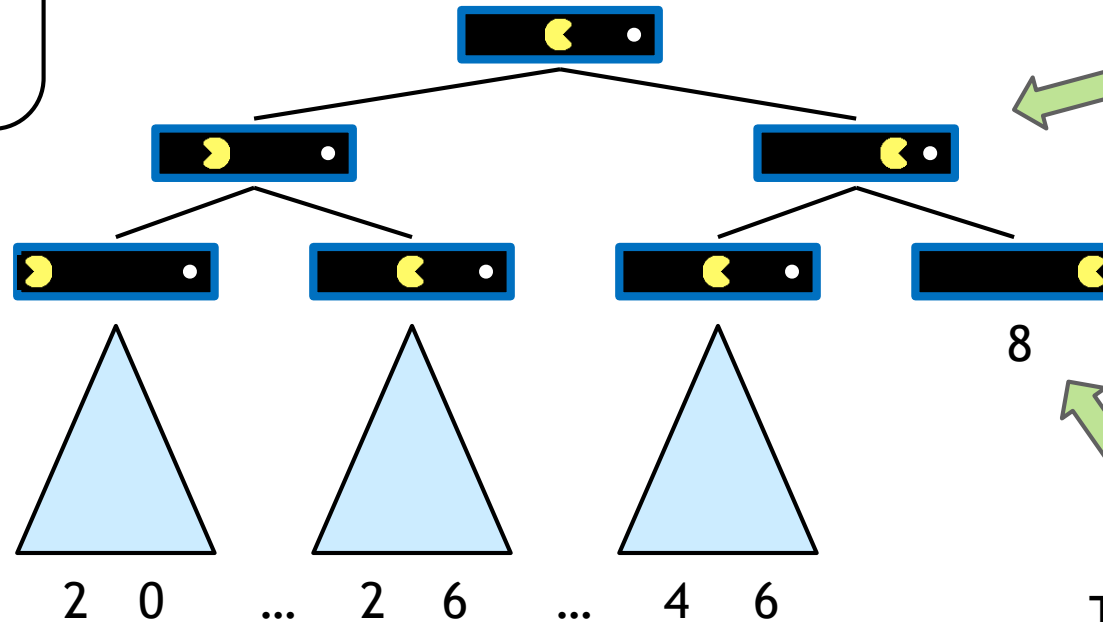


## General Games

- Agents have independent utilities (values on outcomes)
- Cooperation, indifference, competition, and more are all possible

# Value of a State

**Value of a state:**  
The best achievable  
outcome (utility)  
from that state



**Non-Terminal States:**

$$V(s) = \max_{s' \in \text{children}(s)} V(s')$$

**Terminal States:**

$$V(s) = \text{known}$$

# Adversarial Game Trees

