Lecture 15:

Artificial Intelligence
Assignment 9 due tonight

- Submit through CMS
Second Prototype Due

- Thursday, October 23rd
- In-class playtesting
Logging release later today

- Will send out an email
- Needed for first release (Nov. 4th)
• Scripts
• Planning
• Pathfinding
Scripts

1. IF <condition is true> THEN <perform action>
2. IF <condition is true> THEN <perform action>
3. IF <condition is true> THEN <perform action>
First boss in Final Fantasy VII

Artificial Intelligence
First boss in Final Fantasy VII

• If (Count == 0 OR Count == 2) Then
  • SelectedTarget = random opponent
  • Use Search Scope on SelectedTarget
  • Count = Count + 1

• If (Count == 1 OR Count == 3) Then
  • With probability 2/3:
    • If Self HP < (Self Max HP / 2) Then
      • Use Scorpion Tail on SelectedTarget
    • Else
      • Use Rifle on SelectedTarget
  • With 1/3 Chance:
    • Use Scorpion Tail on SelectedTarget
  • Count = Count + 1
Example: Tic Tac Toe

1. If I can win, win

2. If opponent can win, prevent it

3. If center is available, take it

4. If corner is available, take it
Advantages/Disadvantages

- Easy to implement
- Can express complex behaviors
- “Smart” behavior can be very complex
- Not so scalable
- No natural way to vary difficulty
Adversarial Search

Now

I move

Opponent moves

I move

Artificial Intelligence
Importance of search depth in Chess

- Novice: ~4
- Master: ~8
- Grandmaster: ~12
- Deep Blue: 6 – 40
Advantages/Disadvantages

+ Potentially much smarter
+ Natural way to vary difficulty
- State-space explosion
- Game must have certain properties
- Unclear what to do if you can’t “see” the end
Heuristics: Chess

- Pawn: 1 point
- Knight & Bishop: 3 points
- Rook: 5 points
- Queen: 11 points
Heuristic Planning

Now

I move

Opponent moves
Heuristic Planning

Now

I move

Opponent moves

11 10 5 14 2 20 3 16
Now

I move

Opponent moves
Heuristic Planning

Now

I move

Opponent moves
Minimax

MAX

MIN
Advantages/Disadvantages

+ Scalable
+ Can be rational without “seeing” the endgame
- Strength depends a lot on the heuristic
- Still only works for some games
Simultaneous Actions

1/3 1/3 1/3
Artificial Intelligence
Artificial Intelligence
What Is an Optimal Policy?

Agent Policy

\[
\begin{bmatrix}
\frac{2}{3} \\
\frac{1}{3}
\end{bmatrix}
\]

right

left
What Is an Optimal Policy?

$$
\begin{bmatrix}
0 \\
1 \\
-1 \\
\end{bmatrix}
\begin{bmatrix}
1 \\
\frac{1}{3} \\
-\frac{1}{3} \\
1 \\
\end{bmatrix}
\begin{bmatrix}
-1 \\
1 \\
\frac{2}{3} \\
\frac{1}{3} \\
\end{bmatrix}
$$
What Is an Optimal Policy?

Agent Policy

\[
\begin{bmatrix}
\frac{2}{3} \\
\frac{1}{3}
\end{bmatrix}
\]

Opponent Response

\[
\begin{bmatrix}
0 & 1
\end{bmatrix}
\]

\[\frac{-1}{3}\]

Expected Reward
What Is an Optimal Policy?

\[
\begin{bmatrix}
\frac{2}{3} \\
\frac{1}{3}
\end{bmatrix}
\rightarrow
\begin{bmatrix}
0 & 1
\end{bmatrix}
\rightarrow -\frac{1}{3}
\]

\[
\begin{bmatrix}
0 \\
1
\end{bmatrix}
\rightarrow
\begin{bmatrix}
1 & 0
\end{bmatrix}
\rightarrow -1
\]

\[
\begin{bmatrix}
\frac{4}{5} \\
\frac{1}{5}
\end{bmatrix}
\rightarrow
\begin{bmatrix}
0 & 1
\end{bmatrix}
\rightarrow -\frac{3}{5}
\]

\[
\begin{bmatrix}
\frac{1}{4} \\
\frac{3}{4}
\end{bmatrix}
\rightarrow
\begin{bmatrix}
1 & 0
\end{bmatrix}
\rightarrow -\frac{1}{2}
\]

\[
\begin{bmatrix}
1 \\
0
\end{bmatrix}
\rightarrow
\begin{bmatrix}
0 & 1
\end{bmatrix}
\rightarrow -1
\]

\[
\begin{bmatrix}
\frac{7}{8} \\
\frac{1}{8}
\end{bmatrix}
\rightarrow
\begin{bmatrix}
0 & 1
\end{bmatrix}
\rightarrow -\frac{3}{4}
\]
What Is an Optimal Policy?

\[
\begin{bmatrix}
\frac{1}{2} \\
\frac{1}{2} \\
\end{bmatrix}
\rightarrow 0
\]
Agent’s Expected Reward = 0
Expected Reward

\[ \nu = \pi_o^T R \pi_a \]
Calculating Reward

\[
\max_{\nu, \pi} \nu \text{ such that } \\
\sum_{i} \pi_i = 1 \\
\pi \geq 0 \\
\nu \leq R \pi
\]
Pathfinding
Make a grid!

A

B

Artificial Intelligence
Pathfinding: Depth-First
Pathfinding: Breadth-First

Artificial Intelligence
Pathfinding: Breadth-First
Breadth-First is Slow!

Idea: use heuristics
A* Algorithm

- Score \( f = g + h \)
  - \( g \): distance on best path
  - \( h \): naïve distance to goal

Manhattan distance = 30 + 20 = 50
Pathfinding: $A^*$ Algorithm

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>f: 74</td>
<td>g: 14</td>
<td>h: 60</td>
<td></td>
</tr>
<tr>
<td>f: 60</td>
<td>g: 10</td>
<td>h: 50</td>
<td></td>
</tr>
<tr>
<td>f: 54</td>
<td>g: 14</td>
<td>h: 40</td>
<td></td>
</tr>
<tr>
<td>A</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

A* Algorithm

$\text{f} = \text{g} + \text{h}$

$\text{g}$: Cost from start to current node

$\text{h}$: Estimated cost to goal from current node

$\text{f}$: Total cost estimate from start to goal through current node
Pathfinding: A* Algorithm
Pathfinding: A* Algorithm

Artificial Intelligence
Pathfinding: A* Algorithm
## Pathfinding: A* Algorithm

<table>
<thead>
<tr>
<th>(f: 94, g: 24, h: 90)</th>
<th>(f: 74, g: 24, h: 70)</th>
<th>(f: 80, g: 20, h: 60)</th>
<th>(f: 74, g: 20, h: 70)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(f: 74, g: 14, h: 60)</td>
<td>(f: 54, g: 14, h: 40)</td>
<td>(f: 60, g: 10, h: 50)</td>
<td>(f: 40, g: 10, h: 30)</td>
</tr>
<tr>
<td>(f: 94, g: 24, h: 70)</td>
<td>(f: 74, g: 24, h: 70)</td>
<td>(f: 80, g: 24, h: 60)</td>
<td>(f: 74, g: 24, h: 70)</td>
</tr>
<tr>
<td>(f: 74, g: 14, h: 60)</td>
<td>(f: 54, g: 14, h: 40)</td>
<td>(f: 60, g: 10, h: 50)</td>
<td>(f: 40, g: 10, h: 30)</td>
</tr>
<tr>
<td>(f: 94, g: 24, h: 70)</td>
<td>(f: 74, g: 24, h: 70)</td>
<td>(f: 80, g: 24, h: 60)</td>
<td>(f: 74, g: 24, h: 70)</td>
</tr>
<tr>
<td>(f: 74, g: 14, h: 60)</td>
<td>(f: 54, g: 14, h: 40)</td>
<td>(f: 60, g: 10, h: 50)</td>
<td>(f: 40, g: 10, h: 30)</td>
</tr>
<tr>
<td>(f: 94, g: 24, h: 70)</td>
<td>(f: 74, g: 24, h: 70)</td>
<td>(f: 80, g: 24, h: 60)</td>
<td>(f: 74, g: 24, h: 70)</td>
</tr>
<tr>
<td>(f: 74, g: 14, h: 60)</td>
<td>(f: 54, g: 14, h: 40)</td>
<td>(f: 60, g: 10, h: 50)</td>
<td>(f: 40, g: 10, h: 30)</td>
</tr>
</tbody>
</table>

In case of tie, use most recently added.
Pathfinding: A* Algorithm
A* Mario

FPS: 24
Attempt: 1 of 1
AStarAgent
Selected Actions:

RIGHT
SPEED
Group Activity: AI++

- IF your game has no AI
  - That’s How We Roll
  - Box
  - Thermo
  - THEN write a script that solves a level of your game

- IF your game has minimal AI
  - Sleep Walker
  - Pyrokid
  - Epic’s Epic Epic
  - THEN improve the AI script

- IF your game has heavy AI
  - Zombify
  - Nameless
  - THEN discuss how you would add planning, heuristics, or intelligent randomness