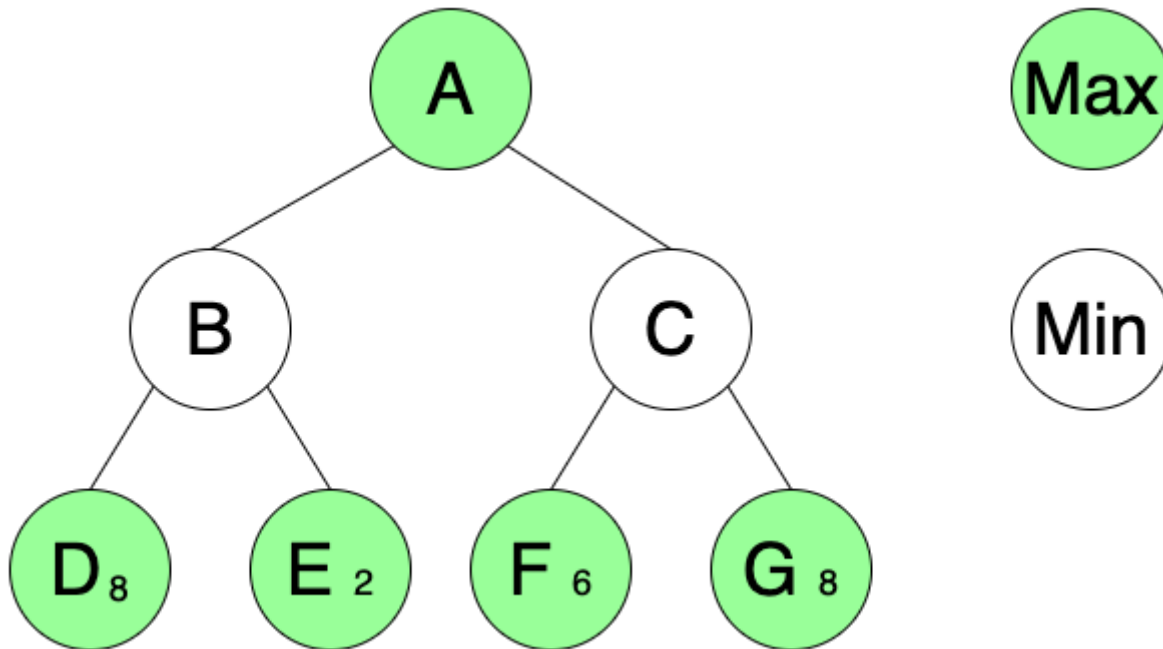


Adversarial Search Exercise

Use this game tree graph to answer the questions below. *Solve the problems by hand, not in code.* Use the videos posted in the Lecture section of the course on [Worldclass](#) for demonstrations of the techniques required and the algorithms below to clarify your thinking.



Questions

1. What value does alpha-beta pruning provide for adversarial search problems? How does it relate to minimax search? Please keep your answer to one paragraph.

Alpha-beta pruning is a layer on top of minimax search. It returns the same answer as minimax search, only more efficiently.

2. Using the MINIMAX algorithm, what value do you get for node A when searching *LEFT to RIGHT*? Does it change if you search *RIGHT to LEFT*?

6 and NO

3. Using the ALPHA-BETA algorithm, what value do you get for node A when searching *LEFT to RIGHT*? Does the value change if you search *RIGHT to LEFT*? If so, what is value do you find?

6 and NO

4. Using the ALPHA-BETA PRUNING algorithm which nodes if any are pruned from the tree when searching from *LEFT to RIGHT*? How about *RIGHT to LEFT*?

None and D

Adversarial Search Algorithms

Minimax Search

function MINIMAX-DECISION(*state*) **returns** *an action*
 return $\arg \max_{a \in \text{ACTIONS}(s)} \text{MIN-VALUE}(\text{RESULT}(\text{state}, a))$

function MAX-VALUE(*state*) **returns** *a utility value*
 if TERMINAL-TEST(*state*) **then return** UTILITY(*state*)
 $v \leftarrow -\infty$
 for each *a* **in** ACTIONS(*state*) **do**
 $v \leftarrow \text{MAX}(v, \text{MIN-VALUE}(\text{RESULT}(\text{state}, a)))$
 return *v*

function MIN-VALUE(*state*) **returns** *a utility value*
 if TERMINAL-TEST(*state*) **then return** UTILITY(*state*)
 $v \leftarrow \infty$
 for each *a* **in** ACTIONS(*state*) **do**
 $v \leftarrow \text{MIN}(v, \text{MAX-VALUE}(\text{RESULT}(\text{state}, a)))$
 return *v*

Alpha-Beta Search

function ALPHA-BETA-SEARCH(*state*) **returns** *an action*
 $v \leftarrow \text{MAX-VALUE}(\text{state}, -\infty, +\infty)$
 return the *action* in ACTIONS(*state*) with value *v*

function MAX-VALUE(*state*, α , β) **returns** *a utility value*
 if TERMINAL-TEST(*state*) **then return** UTILITY(*state*)
 $v \leftarrow -\infty$
 for each *a* **in** ACTIONS(*state*) **do**
 $v \leftarrow \text{MAX}(v, \text{MIN-VALUE}(\text{RESULT}(\text{state}, a), \alpha, \beta))$
 if $v \geq \beta$ **then return** *v*
 $\alpha \leftarrow \text{MAX}(\alpha, v)$
 return *v*

function MIN-VALUE(*state*, α , β) **returns** *a utility value*
 if TERMINAL-TEST(*state*) **then return** UTILITY(*state*)
 $v \leftarrow +\infty$
 for each *a* **in** ACTIONS(*state*) **do**
 $v \leftarrow \text{MIN}(v, \text{MAX-VALUE}(\text{RESULT}(\text{state}, a), \alpha, \beta))$
 if $v \leq \alpha$ **then return** *v*
 $\beta \leftarrow \text{MIN}(\beta, v)$
 return *v*